

# Das Kid3 Handbuch

Software Entwicklung: Urs Fleisch  
Übersetzung: Urs Fleisch



# Das Kid3 Handbuch

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>12</b>
<b>2</b>	<b>Benutzen von Kid3</b>	<b>13</b>
2.1	Funktionsmerkmale von Kid3 . . . . .	13
2.2	Arbeiten anhand eines Beispiels . . . . .	13
<b>3</b>	<b>Menüeinträge</b>	<b>15</b>
3.1	Die Elemente der Benutzeroberfläche . . . . .	15
3.1.1	Dateiliste . . . . .	15
3.1.2	Wiedergabeliste bearbeiten . . . . .	16
3.1.3	Ordnerliste . . . . .	17
3.1.4	Datei . . . . .	17
3.1.5	Tag 1 . . . . .	18
3.1.6	Tag 2 . . . . .	19
3.1.7	Tag 3 . . . . .	19
3.1.8	Frame-Liste . . . . .	20
3.1.9	Synchronisierter Text und Ereignis-Zeitcodes . . . . .	23
3.2	Das Menü Datei . . . . .	24
3.3	Das Menü Bearbeiten . . . . .	31
3.4	Das Menü Extras . . . . .	32
3.5	Das Menü Einstellungen . . . . .	35
3.6	Das Menü Hilfe . . . . .	41
<b>4</b>	<b>kid3-cli</b>	<b>42</b>
4.1	Befehle . . . . .	42
4.1.1	Hilfe . . . . .	42
4.1.2	Zeitüberschreitung . . . . .	42
4.1.3	Anwendung beenden . . . . .	42
4.1.4	Ordner wechseln . . . . .	43
4.1.5	Aktuellen Ordner ausgeben . . . . .	43
4.1.6	Ordnerliste . . . . .	43
4.1.7	Veränderte Dateien speichern . . . . .	43
4.1.8	Datei wählen . . . . .	43
4.1.9	Tag auswählen . . . . .	44

## Das Kid3 Handbuch

4.1.10	Tag-Element holen . . . . .	44
4.1.11	Tag-Element setzen . . . . .	45
4.1.12	Änderungen rückgängig machen . . . . .	46
4.1.13	Von Datei importieren . . . . .	46
4.1.14	Automatisch importieren . . . . .	46
4.1.15	Cover-Bilder herunterladen . . . . .	46
4.1.16	In Datei exportieren . . . . .	46
4.1.17	Stückliste erstellen . . . . .	46
4.1.18	Dateinamenformat anwenden . . . . .	46
4.1.19	Tag-Format anwenden . . . . .	47
4.1.20	Textkodierung anwenden . . . . .	47
4.1.21	Ordner umbenennen . . . . .	47
4.1.22	Stücke nummerieren . . . . .	47
4.1.23	Filter . . . . .	47
4.1.24	ID3v2.3 nach ID3v2.4 konvertieren . . . . .	48
4.1.25	ID3v2.4 nach ID3v2.3 konvertieren . . . . .	48
4.1.26	Dateiname von Tag . . . . .	48
4.1.27	Tag von Dateiname . . . . .	48
4.1.28	Tag zu anderem Tag . . . . .	48
4.1.29	Kopieren . . . . .	48
4.1.30	Einfügen . . . . .	49
4.1.31	Entfernen . . . . .	49
4.1.32	Kid3 einrichten . . . . .	49
4.2	Beispiele . . . . .	49
4.3	JSON Format . . . . .	50
<b>5</b>	<b>Danksagungen und Lizenz</b>	<b>52</b>
<b>A</b>	<b>Installation</b>	<b>53</b>
A.1	Bezugsquellen . . . . .	53
A.2	Anforderungen . . . . .	53
A.3	Kompilieren und Installation . . . . .	53
A.4	Konfiguration . . . . .	54
<b>B</b>	<b>D-Bus-Schnittstelle</b>	<b>55</b>
B.1	D-Bus-Beispiele . . . . .	55
B.2	D-Bus API . . . . .	56
B.2.1	Datei oder Ordner öffnen . . . . .	56
B.2.2	Tags aller nicht selektierten oder modifizierten Dateien entladen . . . . .	56
B.2.3	Alle veränderten Dateien speichern . . . . .	56
B.2.4	Detaillierte Fehlermeldung für einige Methoden . . . . .	56
B.2.5	Änderungen an Dateien rückgängig machen . . . . .	56

## Das Kid3 Handbuch

B.2.6	Automatischen Import starten . . . . .	56
B.2.7	Tags von Datei importieren . . . . .	57
B.2.8	Tags aus anderen Tags importieren . . . . .	57
B.2.9	Tags aus anderen Tags importieren in ausgewählten Dateien . . . . .	57
B.2.10	Cover-Bild herunterladen . . . . .	57
B.2.11	Tags in Datei exportieren . . . . .	58
B.2.12	Stückliste erstellen . . . . .	58
B.2.13	Dateien einer Stückliste abfragen . . . . .	58
B.2.14	Dateien einer Stückliste setzen . . . . .	58
B.2.15	Anwendung beenden . . . . .	58
B.2.16	Alles auswählen . . . . .	58
B.2.17	Gesamte Auswahl aufheben . . . . .	59
B.2.18	Gehe zur ersten Datei . . . . .	59
B.2.19	Gehe zur vorangehende Datei . . . . .	59
B.2.20	Gehe zur nächstes Datei . . . . .	59
B.2.21	Erste Datei auswählen . . . . .	59
B.2.22	Vorangehende Datei auswählen . . . . .	59
B.2.23	Nächste Datei auswählen . . . . .	59
B.2.24	Aktuelle Datei auswählen . . . . .	59
B.2.25	Aktuellen Ordner auf- oder zuklappen . . . . .	59
B.2.26	Dateinamenformat anwenden . . . . .	60
B.2.27	Tag Format anwenden . . . . .	60
B.2.28	Textkodierung anwenden . . . . .	60
B.2.29	Ordnername von Tags setzen . . . . .	60
B.2.30	Stücke in ausgewählten Dateien nummerieren . . . . .	60
B.2.31	Dateien filtern . . . . .	60
B.2.32	ID3v2.3 nach ID3v2.4 konvertieren . . . . .	60
B.2.33	ID3v2.4 nach ID3v2.3 konvertieren . . . . .	61
B.2.34	Ordnerpfad . . . . .	61
B.2.35	Name der aktuellen Datei . . . . .	61
B.2.36	Setze Dateinamen . . . . .	61
B.2.37	Setze Format für Dateiname aus Tags . . . . .	61
B.2.38	Generiere den Dateinamen aus den Tags . . . . .	61
B.2.39	Wert eines Elementes . . . . .	61
B.2.40	Setze Wert eines Elementes . . . . .	62
B.2.41	Alle Elemente eines Tags . . . . .	62
B.2.42	Technische Informationen über Datei . . . . .	62
B.2.43	Setze Tag aus Dateinamen . . . . .	62
B.2.44	Setze Tag aus anderem Tag . . . . .	62
B.2.45	Kopiere Tag . . . . .	63
B.2.46	Füge Tag ein . . . . .	63
B.2.47	Entferne Tag . . . . .	63
B.2.48	Lese die Konfiguration neu ein . . . . .	63
B.2.49	Spielt die ausgewählten Dateien ab . . . . .	63

<b>C</b>	<b>QML Schnittstelle</b>	<b>64</b>
C.1	QML Beispiele . . . . .	64
C.2	QML API . . . . .	66
C.2.1	Kid3Script . . . . .	66
C.2.2	Skript-Funktionen . . . . .	67
C.2.3	Applikations-Kontext . . . . .	67
C.2.4	Konfigurations-Objekte . . . . .	69

# Tabellenverzeichnis

3.1	Zuordnung der vereinheitlichten Frame-Namen in den unterschiedlichen Formaten	22
3.2	Eintrag in Bewertungs-Tabelle . . . . .	38

### **Zusammenfassung**

Kid3 ist eine Anwendung zum effizienten Editieren von ID3v1 und ID3v2 Tags in MP3 Dateien. Auch die Tags in Ogg/Vorbis, Opus, DSF, FLAC, MPC, APE, MP4/AAC, MP2, Speex, TrueAudio, WavPack, WMA, WAV, AIFF Dateien und Tracker Modulen (MOD, S3M, IT, XM) werden unterstützt. Es ist einfach, Tags in mehreren Dateien auf die gleichen Werte zu setzen (z.B. Album, Interpret, Jahr und Genre in allen Dateien eines Albums), Tags aus den Dateinamen zu generieren oder umgekehrt.



# Übersicht

```
kid3 [--help | --author | --version | --license | --desktopfile DATEI] [DATEI...]  
kid3-qt [--portable] [Qt-Optionen] [DATEI...]  
kid3-cli [--portable] [--dbus] [-h | --help] [-c BEFEHL1] [-c BEFEHL2...] [DATEI...]
```

# Optionen

## **--portable**

Speichert die Einstellungen in der Datei `kid3.ini` im Programmordner.

## **DATEI**

Wenn `DATEI` der Pfad auf einen Ordner ist, so wird er geöffnet. Wenn einer oder mehrere Datei-Pfade gegeben werden, so wird deren gemeinsamer Ordner geöffnet und die Dateien werden ausgewählt.

## **kid3**

### **--help**

Hilfe zu verfügbaren Optionen anzeigen.

### **--author**

Autoren-Information anzeigen.

### **--version**

Versionsinformation anzeigen.

### **--license**

Lizenz-Informationen anzeigen.

### **--desktopfile DATEI**

Der Basis-Dateiname des Desktop-Eintrags dieser Anwendung.

## **kid3-qt**

### **Qt-Optionen**

Siehe `qt5options(7)`.

## **kid3-cli**

**--dbus**

Aktiviert die D-Bus Schnittstelle.

**-c**

Führt einen Befehl aus. Mehrere `-c` Optionen sind möglich, die Befehle werden in der entsprechenden Reihenfolge ausgeführt. Siehe den Abschnitt über [kid3-cli](#) für eine Beschreibung der verfügbaren Befehle.

**-h | --help**

Hilfe zu verfügbaren Optionen und Befehlen anzeigen.

# Kapitel 1

## Einleitung

Kid3 ist eine Anwendung zum effizienten Editieren von ID3v1 und ID3v2 Tags in MP3 Dateien. Diese Tags können zwar mit den meisten MP3 Player Programmen editiert werden, jedoch nicht auf eine sehr komfortable und effiziente Art. Darüber hinaus werden auch die Tags in Ogg/Vorbis, Opus, DSF, FLAC, MPC, APE, MP4/AAC, MP2, Speex, TrueAudio, WavPack, WMA, WAV, AIFF Dateien und Tracker Modulen (MOD, S3M, IT, XM) unterstützt.

Kid3 kann weder MP3 grabben noch codieren, es ist darauf ausgelegt, die ID3 Tags von allen Dateien eines Albums auf eine möglichst effiziente Art zu editieren, d.h. mit so wenig Mausklicks und Tastendrücken wie möglich. Während die meisten anderen Programme entweder ID3v1 oder ID3v2 Tags editieren können, hat Kid3 volle Kontrolle über die Tags beider Versionen, es kann zwischen den beiden Formaten konvertieren und hat Zugang zu allen ID3v2 Frames. Tags mehrerer Dateien können gleichzeitig auf dieselben Werte gesetzt werden, so beispielsweise Interpret, Album, Jahr und Genre aller Dateien eines Albums. Wenn die Information für die Tags schon im Dateinamen enthalten ist, können die Tags automatisch aus dem Dateinamen generiert werden. Es ist auch möglich, den Dateinamen in einem beliebigen Format aus den Werten in den Tags zu setzen.

Das Editieren wird darüber hinaus durch automatische Ersetzung von Zeichenketten erleichtert, z.B. um ungültige Zeichen aus Dateinamen zu entfernen. Automatische Groß- und Kleinschreibung macht es einfach, eine konsistente Namensschreibweise in allen Tags zu verwenden.

Die Tag-Information für ganze Alben kann über [gnudb.org](http://gnudb.org), [MusicBrainz](http://MusicBrainz), [Discogs](http://Discogs), [Amazon](http://Amazon) oder von anderen Quellen mit Albuminformationen importiert werden. Das verwendete Format ist mit regulären Ausdrücken frei konfigurierbar.

Probleme und Vorschläge für neue Funktionen können dem Autor gemeldet werden.

## Kapitel 2

# Benutzen von Kid3

### 2.1 Funktionsmerkmale von Kid3

- Editieren von ID3v1.1 Tags
- Editieren von allen ID3v2.3 und ID3v2.4 Frames
- Editieren der Tags von mehreren Dateien
- Konvertieren zwischen ID3v1 und ID3v2 Tags
- Editieren von MP3, Ogg/Vorbis, Opus, DSF, FLAC, MPC, APE, MP4/AAC, MP2, Speex, TrueAudio, WavPack, WMA, WAV und AIFF Tags
- Generieren der Tags aus dem Dateinamen
- Generieren der Tags aus dem Inhalt der Tagfelder
- Generieren des Dateinamens aus den Tags
- Generieren und Ändern des Ordnernamens aus den Tags
- Generieren von Stücklisten
- Automatische Groß-/Kleinschreibung und Zeichenersetzung
- Import von [gnudb.org](http://gnudb.org), [MusicBrainz](http://MusicBrainz), [Discogs](http://Discogs), [Amazon](http://Amazon) und anderen Quellen
- Export als CSV, HTML, Stückliste, Kover XML oder in anderen Formaten. Exportierte CSV-Dateien können wieder importiert werden.

### 2.2 Arbeiten anhand eines Beispiels

Dieser Abschnitt beschreibt eine typische Sitzung mit Kid3. Nehmen wir an, wir haben einen Ordner mit MP3 Dateien der Stücke aus dem Album "Let's Tag" von der Gruppe "One Hit Wonder". Der Ordner ist im "Interpret - Album" Format benannt, in unserem Fall also `One Hit Wonder - Let's Tag`. Der Ordner enthält die Stücke im "Nummer Titel.mp3" Format, was auch Sinn macht, weil die Dateinamen kurz werden (wichtig bei den kleinen Anzeigen mobiler MP3 Player) und alphabetisch in der richtigen Reihenfolge geordnet werden (wichtig falls ein Hardware MP3 Player die Stücke in alphabetischer Reihenfolge abspielt oder in der Reihenfolge, wie sie auf CD gebrannt wurden - diese Reihenfolge ist bei **mkisofs** alphabetisch). Darüber hinaus sind Interpret und Album bereits im Ordnernamen enthalten und brauchen nicht im Dateinamen wiederholt zu werden. Doch zurück zu unserem Beispiel - der Ordner sieht folgendermaßen aus:

## Das Kid3 Handbuch

01 Intro.mp3

02 We Only Got This One.mp3

03 Outro.mp3

Diese Dateien haben noch keine Tags, wir werden diese mit Kid3 erzeugen. Wir benutzen **Öffnen** (**Datei** Menü oder Werkzeugleiste) und wählen eine der Dateien in diesem Ordner. Alle Dateien werden im Dateilistenfeld angezeigt. Aus Bequemlichkeit wollen wir die Tags aus den Dateinamen erzeugen. Daher wählen wir alle Dateien im Listenfeld aus und klicken auf **Nach: Tag 1** in der **Datei** Sektion. Dadurch werden Titel, Interpret, Album und Nummer in allen Dateien gesetzt. Um die Werte für Jahr und Genre in allen Dateien zu setzen, lassen wir alle Dateien selektiert und tippen "2002" in das Feld **Datum** und wählen "Pop" im **Genre** Auswahlfeld. Um einzig diese zwei Werte zu verändern, werden ihre Markierungsfelder automatisch aktiviert, und alle anderen Markierungsfelder bleiben leer. Wird nun durch Klick auf die erste Datei die Auswahl verändert, so sieht man, dass die Tags für die erste Datei die korrekten Werte enthalten. Die Tags der anderen Dateien können auch überprüft werden, indem eine nach der andern selektiert wird. Wenn wir mit den Tags zufrieden sind, können die Änderungen mit **Speichern** (**Datei** Menü oder Werkzeugleiste) übernommen werden. Wählen von **Stückliste erstellen** im **Datei** Menü erzeugt die Datei One Hit Wonder - Let's Tag.m3u im Ordner.

## Kapitel 3

# Menüeinträge

### 3.1 Die Elemente der Benutzeroberfläche

Die Kid3 Benutzeroberfläche ist in sechs Sektionen aufgeteilt: Links finden sich das Dateilistenfeld und das Ordnerlistenfeld, die rechte Seite enthält die **Datei**, **Tag 1**, **Tag 2** und **Tag 3** Sektionen.

Um mit der Tastatur zwischen den verschiedenen Sektionen zu navigieren, stehen verschiedene Tastaturkürzel zur Verfügung. In den Tag-Sektionen sind die Kürzel aktiv wenn nicht gerade Text bearbeitet wird oder wenn man sich in der ersten Spalte der Tabelle befindet.

- **Alt-Links:** Gehe zu vorangehender Sektion (**Command-]** auf macOS®)
- **Alt-Rechts:** Gehe zu nächster Sektion (**Command-[** auf macOS®)
- **Strg-Umschalt-V:** Von anderem Tag
- **Strg-C:** Kopieren
- **Strg-V:** Einfügen
- **Umschalt-Delete:** Entfernen
- **F2:** Bearbeiten
- **Insert:** Hinzufügen
- **Delete:** Löschen

#### 3.1.1 Dateiliste

Die Dateiliste enthält die Namen aller Dateien im geöffneten Ordner, welche den gewählten Dateinamenfilter (typischerweise \*.mp3 \*.ogg \*.opus \*.dsf \*.flac \*.mpc \*.aac \*.m4a \*.m4b \*.m4p \*.mp4 \*.mp2 \*.spx \*.tta \*.wv \*.wma \*.wav \*.aiff \*.ape) erfüllen. Eine einzelne oder mehrere Dateien können ausgewählt werden. Um keine Datei auszuwählen, kann in den leeren Bereich unterhalb der Dateieinträge geklickt werden. Die Auswahl bestimmt die Dateien, welche durch die unten beschriebenen Operationen verändert werden.

Nebst **Name** können noch andere Spalten **Größe**, **Typ** und **Änderungsdatum** mit Details zu den Dateien dargestellt werden. Eine Spalte wird angezeigt, wenn das Häkchen bei ihrem Namen im Kontextmenü der Kopfzeile gesetzt ist. Die Reihenfolge der Spalten kann mittels Drag'n'Drop verändert werden. Die Sortierreihenfolge kann durch Klicken auf den Spaltenkopf umgeschaltet werden.

Links von den Namen kann ein Symbol dargestellt werden: eine Diskette, um zu zeigen, dass die Datei verändert wurde, oder Information über die Tags welche vorhanden sind (V1, V2, V1V2 oder NO TAG, wenn nichts dargestellt wird, wurde die Datei noch nicht eingelesen).

Ordner werden mit Ordner-Symbolen gekennzeichnet. Wenn ein Ordner geöffnet wird, so werden seine Dateien in einem hierarchischen Baum dargestellt. Operationen können auf Dateien aus verschiedenen Ordnern angewendet werden, was nützlich ist, wenn die Musik-Dateien so organisiert werden, dass pro Interpret ein Ordner erstellt wird, der wiederum einen Ordner für jedes Album enthält.

Bei einem Rechtsklick in der Dateiliste öffnet sich ein Kontext-Menü mit folgenden Befehlen:

- **Alle ausklappen:** Klappt alle Ordnerbäume auf (nur der aktuelle Ordner, wenn die Umschalttaste gedrückt ist)
- **Alle einklappen:** Klappt alle Ordnerbäume ein
- **Umbenennen:** Ändert den Namen der Datei
- **In den Papierkorb verschieben:** Verschiebt die Datei in den Papierkorb
- **Wiedergabe:** Spielt die Datei ab, siehe [Wiedergabe](#). Wenn die ausgewählte Datei eine Wiedergabeliste ist, so werden deren Stücke abgespielt.
- **Bearbeiten:** Bearbeitet Wiedergabeliste, siehe [Wiedergabeliste bearbeiten](#).
- Bei den weiteren Menüeinträgen handelt es sich um Benutzerbefehle, welche in [Kid3 einrichten](#) bei **Benutzerbefehle** definiert werden können. Dort kann auch die Wiedergabe von Dateien per Doppelklick aktiviert werden.

### 3.1.2 Wiedergabeliste bearbeiten

Eine Wiedergabeliste kann entweder mit den Stücken eines Ordners oder leer erstellt werden, siehe hierzu [Stückliste erstellen](#). Die so erstellte Datei kann mittels Doppelklick oder Wahl von **Bearbeiten** aus dem Kontext-Menü der Dateiliste bearbeitet werden. Es wird ein Dialog mit den Einträgen der Stückliste geöffnet. Es können mehrere Stücklisten gleichzeitig bearbeitet werden.

Neue Einträge können mit "Drag'n'Drop" aus der Dateiliste, einem Dateimanager oder einer anderen Stückliste hinzugefügt werden. Wenn ein Eintrag aus einer anderen Stückliste hineingezogen wird, so wird der Eintrag abhängig vom System verschoben oder kopiert. Für die jeweils andere Operation muss gleichzeitig die Umschalt-Taste (**Umschalt**) bzw. die Steuerungstaste (**Strg**) gedrückt werden (auf macOS muss **Alt** gedrückt werden, um zu kopieren statt zu verschieben). Um Einträge innerhalb der Stückliste zu verschieben, kann ebenfalls "Drag'n'Drop" verwendet werden. Alternativ können Einträge mit den Tastenkombinationen **Strg-Umschalt-Up** und **Strg-Umschalt-Down** verschoben werden (auf macOS muss **Command** statt **Strg** gedrückt werden). Um einen Eintrag zu löschen, kann **Delete** gedrückt werden.

Um Einträge aus der Dateiliste mittels "Drag'n'Drop" zu ziehen, ist folgendes zu beachten: Die Einträge müssen links (bei den Icons) gezogen werden, eine entsprechende Geste im rechten Bereich führt zu einer Mehrfachselektion, eine solche ist somit immer noch einfach möglich.

Wenn eine Stückliste modifiziert wurde, so können die Änderungen mit **Speichern** übernommen oder mit **Abbrechen** verworfen werden. Beim Schließen des Fensters erfolgt eine Sicherheitsabfrage, wenn noch ungesicherte Änderungen vorhanden sind.

In einer Wiedergabeliste ausgewählte Stücke werden automatisch auch in der Dateiliste ausgewählt, womit auch ihre Tags bearbeitet werden können.

Um Aktionen auf einer Wiedergabeliste auszuführen, muss sie in der Dateiliste ausgewählt werden. In deren Kontext-Menü führt **Bearbeiten** zum hier beschriebenen Dialog und **Wiedergabe** spielt die Stückliste ab. Es lassen sich auch Benutzerbefehle für Wiedergabelisten definieren, zum [Beispiel Export Playlist Folder](#), welche die Stücke einer Wiedergabeliste in einen Ordner kopiert.



### 3.1.3 Ordnerliste

Die Ordnerliste enthält die Unterordner sowie den aktuellen (.) und den übergeordneten (..) Ordner. Sie erlaubt den raschen Wechsel zwischen Ordnern, ohne den Umweg über die **Öffnen** ... Funktion oder "Drag'n'Drop".

Die Anzeige, Reihenfolge und Sortierung der Spalten kann auf dieselbe Weise eingestellt werden wie bei der [Dateiliste](#) beschrieben.

### 3.1.4 Datei

Zeigt Informationen über Codierung (MP3, Ogg, Opus, DSF, FLAC, MPC, APE, MP2, MP4, AAC, Speex, TrueAudio, WavPack, WMA, WAV, AIFF), Bit- und Samplerate, Kanäle und die Länge der Datei an.

Das **Name** Editierfeld enthält den Dateiname wenn nur eine einzelne Datei ausgewählt ist. Wird dieses Feld verändert, so wird die Datei umbenannt, wenn **Speichern** ausgeführt wird.

Das **Format** Auswahl- und Editierfeld enthält das Format, welches verwendet wird, wenn der Dateiname aus dem ersten oder zweiten Tag erzeugt wird. Der Dateiname kann beliebige Zeichen enthalten, es kann sogar ein Ordneranteil abgetrennt durch einen Schrägstrich angegeben werden, doch dieser Ordner muss bereits existieren, damit das Umbenennen erfolgreich ist. Die folgenden Codes können verwendet werden, um Werte aus den Tags im Dateinamen einzusetzen:

- %s %{title} Titel (Song)
- %a %{artist} Interpret (Artist)
- %l %{album} Album
- %c %{comment} Kommentar (Comment)
- %y %{year} Jahr (Year)
- %t %{track} Nummer (Stück, z.B. 01)
- %t %{track.n} Nummer mit Feldlänge n (z.B. 001 für %{track.3})
- %T %{tracknumber} Nummer (Stück, ohne führende Null, z.B. 1)
- %g %{genre} Genre
- %{ignore} Wird ignoriert wenn Tags aus dem Dateinamen erzeugt werden

Die Formatcodes sind nicht auf die oben gegebenen Beispiele beschränkt. Alle Framenamen können benutzt werden, sowohl vereinheitlichte wie **%{albumartist}**, **%{discnumber.1}**, **%{bpm}** als auch formatspezifische wie **%{popm}**.

Es ist möglich, zusätzlichen Text vor und nach dem Wert, welcher für einen Formatcode ersetzt wird, einzufügen indem man diesen in Anführungszeichen innerhalb der geschwungenen Klammern hinzufügt. Dieser Text wird nur dann eingefügt, wenn der Wert für den Formatcode nicht leer ist. Als Beispiel könnte man für einen Dateinamen, welcher sowohl den Titel als auch den Untertitel enthalten soll, **%{title} [%{subtitle}]** als Format verwenden. Aber dies würde im Falle von nicht vorhandenem Untertitel dazu führen, dass der resultierende Dateiname mit [] endet. Um diese leeren Klammern am Ende zu verhindern, kann **%{title}%{ ' ' [ ' ' s ubtitle ' ' ] ' ' }** verwendet werden. Dieses Format lässt die eckigen Klammern, das führende Leerzeichen und den Untertitel weg, wenn kein entsprechendes Element vorhanden ist.

Die Liste der verfügbaren Formate kann in dem Dialog bearbeitet werden, welcher erscheint, wenn der Knopf **Dateiname von Tag** im **Dateien**-Reiter der [Einstellungen](#) gedrückt wird.

Ein zweites **Format** Auswahlfeld (mit Pfeil runter) wird verwendet, um die Tags aus dem Dateinamen zu generieren. Falls der Dateiname nicht diesem Format entspricht, werden noch einige andere, gängige Formate ausprobiert.

Einige häufig verwendete Dateinamen-Formate sind bereits vordefiniert und über das Auswahl-feld verfügbar, aber es ist auch möglich, ein eigenes Format in das Eingabefeld einzutragen.

Die Liste der verfügbaren Formate kann in dem Dialog bearbeitet werden, welcher erscheint, wenn der Knopf **Tag von Dateiname** im **Dateien**-Reiter der **Einstellungen** gedrückt wird.

Intern wird ein regulärer Ausdruck aus den Formatcodes gebildet. Wenn die volle Ausdruckskraft regulärer Ausdrücke gebraucht wird, ist es auch möglich, einen kompletten regulären Ausdruck mit Klammerkonstrukten, welche durch die Formatcodes eingeleitet werden, als Format für die Extraktion der Tags aus den Dateinamen anzugeben. Um beispielsweise die Stücknummern ohne das automatische Entfernen führender Nullen in die Tags zu kriegen, könnte ein Ausdruck wie `"/%{track}(\d+) %{title}(.*)"` verwendet werden.

**Von: Tag 1, Tag 2:** Setzt den Dateinamen gemäß dem ausgewählten Format und dem ersten Tag bzw. zweiten Tag.

**Nach: Tag 1, Tag 2:** Die Tags werden anhand des Dateinamens gesetzt. Dabei wird zuerst das in **Format** ausgewählte Dateinamenformat probiert. Falls der vorhandene Dateinamen nicht diesem Format entspricht, werden noch folgende Formate versucht:

- Interpret - Album/Nummer Titel
- Album/Nummer - Interpret - Titel
- /Interpret - Album - Nummer - Titel
- Album/Interpret - Nummer - Titel
- Album/Interpret - Titel
- Interpret/Album/Stück Song

Wenn eine einzelne Datei ausgewählt wurde, werden die Eingabefelder mit den aus dem Dateinamen extrahierten Werten gefüllt. Bei Mehrfachselektion werden die Werte direkt gemäß den Dateinamen gesetzt.

### 3.1.5 Tag 1

Die Eingabefelder für **Titel**, **Interpret**, **Album**, **Kommentar**, **Datum**, **Stücknummer** und **Genre** werden benutzt um die entsprechenden Werte im ersten Tag der ausgewählten Dateien zu editieren. Die Werte werden verändert, wenn die Dateiauswahl geändert wird oder vor Operationen wie **Speichern** und **Beenden** und wenn die entsprechenden Markierungsfelder links aktiviert sind. Dies ist nützlich, um nur bestimmte Werte zu verändern und die übrigen Werte nicht anzurühren.

Wenn eine einzelne Datei ausgewählt ist, werden alle Markierungsfelder aktiviert und die Eingabefelder enthalten die Werte aus den Tags in der Datei. Wenn ein Tag nicht vorhanden ist, werden die leeren Werte angezeigt, d.h. eine leere Zeichenkette für die Felder **Titel**, **Interpret**, **Album** und **Kommentar**, 0 für **Datum** und **Stücknummer** und ein leerer Auswahleintrag bei **Genre**. Die Werte können editiert werden und werden für die ausgewählte Datei beim Ändern der Auswahl gesetzt, sofern das entsprechende Markierungsfeld aktiviert ist. Die Datei wird dann durch ein Disketten-Symbol als modifiziert gekennzeichnet, doch die eigentliche Datei bleibt unverändert, bis **Speichern** ausgeführt wird.

Wenn mehrere Dateien ausgewählt werden, dann werden nur diejenigen Werte angezeigt, welche bei allen ausgewählten Dateien identisch sind. In allen übrigen Editierfeldern werden die leeren Werte (siehe oben) angezeigt. Alle Markierungsfelder sind inaktiv, um ungewolltes Verändern mehrerer Dateien zu verhindern. Soll ein Wert für alle selektierten Dateien verändert werden, so kann er editiert werden, und das entsprechende Markierungsfeld wird gesetzt. Der Wert wird dann für alle ausgewählten Dateien verändert wenn die Auswahl geändert wird und kann mit **Speichern** gesichert werden.

Die Markierungsfelder bestimmen auch, auf welche Felder die meisten anderen Befehle angewendet werden. So werden beim Kopieren, Einfügen und Transferieren zwischen Tags 1 und 2 nur markierte Felder verwendet. Um die Bedienung zu vereinfachen, wenn mehrere Dateien ausgewählt sind und daher alle Markierungsfelder inaktiv sind, verhalten sich diese Befehle wenn alle Markierungsfelder inaktiv sind gleich als wären alle markiert.

**Von Tag 2:** Die Tag 1 Felder werden auf die gleichen Werte gesetzt wie die entsprechenden Tag 2 Werte. Bei Mehrfachselektion werden die Tags der Dateien direkt gesetzt.

**Kopieren:** Die Kopierablage wird mit den Tag 1 Werten gefüllt. Bei folgenden Einfüge-Operationen werden nur solche Werte gesetzt, bei welchen hier das Markierungsfeld aktiviert war.

**Einfügen:** Fügt die Werte aus der Kopierablage in die Eingabefelder ein.

**Löschen:** Diese Funktion setzt alle Editierfelder auf die leeren Werte, entfernt also alle Werte. Die gespeicherte Datei enthält dann kein Tag 1 mehr.

### 3.1.6 Tag 2

Die Eingabefelder funktionieren gleich wie bei **Tag 1** beschrieben, die Länge der Zeichenketten ist jedoch nicht begrenzt.

Beim **Genre** können nebst den vorgegebenen auch eigene Namen verwendet werden - eine Eingabe ins Editierfeld genügt.

Das Tag 2 kann nicht bloß dieselben Werte enthalten wie das Tag 1, das Format ist vielmehr auf flexible Art aus mehreren Elementen aufgebaut, welche wiederum aus verschiedenen Feldern bestehen. Die Tag 2 Tabelle zeigt alle Elemente an, welche in der ausgewählten Datei vorhanden sind.

**Bearbeiten:** Ein Fenster wird geöffnet, in welchem alle Felder des gewählten Elementes editiert werden können. Wenn mehrere Dateien ausgewählt sind, so werden die editierten Felder bei allen ausgewählten Dateien angewendet, welche solch ein Element enthalten.

**Hinzufügen:** Ein Auswahlfenster mit allen unterstützten Elementtypen wird angezeigt und ein Element des gewählten Typs kann editiert und zu der Datei hinzugefügt werden. Dies funktioniert auch, wenn mehrere Dateien ausgewählt sind; das Element wird dann bei allen ausgewählten Dateien hinzugefügt.

**Löschen:** Löscht das ausgewählte Element in den ausgewählten Dateien.

**Albumcover hierhin ziehen** wird angezeigt wenn die Datei kein Album Cover Bild enthält. Bilder kann man mittels "Drag'n'Drop" vom Browser oder Dateimanager hinzufügen; sie werden dann hier angezeigt. Das Editieren oder Hinzufügen eines Bild-Elements kann mit einem Doppelklick ausgelöst werden.

### 3.1.7 Tag 3

Einige Dateien haben mehr als zwei Tags, bei diesen ist eine dritte Tag-Sektion sichtbar. Die folgenden Dateitypen haben solch eine **Tag 3** Sektion:

- MP3-Dateien können ein ID3v1.1 Tag, ein ID3v2 (2.3.0 oder 2.4.0) Tag und in der dritten Sektion ein APE-Tag haben. APE-Tags werden v.a. für Replay-Gain-Informationen verwendet. In der **Tag 3** Sektion ist diese Information sichtbar, und das APE-Tag kann mit dem **Entfernen**-Knopf gelöscht werden.
- Der RIFF-INFO-Chunk in WAV-Dateien ist in der **Tag 3** Sektion zugänglich, weil die **Tag 1** Sektion für ID3v1.1 Tags reserviert ist und die Einschränkungen dieser Tags umfasst. Die **Tag 2** Sektion wird wie bisher für ID3v2.4.0 Tags verwendet. RIFF-INFO-Chunks scheinen jedoch weiter verbreitet zu sein.

- FLAC-Dateien speichern ihre Meta-Daten normalerweise in einem Vorbis-Tag. Man trifft jedoch auch auf FLAC-Dateien mit IDv1 und ID3v2 Tags, welche in den Sektionen **Tag 1** und **Tag 3** sichtbar sind. ID3 Tags in FLAC-Dateien werden nur von TagLib unterstützt, daher muss das OggFlacMetadata-Plugin im **Plugins**-Reiter der **Einstellungen** abgeschaltet werden.

Die Eingabefelder funktionieren wie bei der **Tag 2** Sektion.

### 3.1.8 Frame-Liste

Kid3 kann die meisten Frames aller unterstützten Dateiformate editieren. Gebräuchliche Frames können über vereinheitlichte Namen angesprochen werden, so dass sie auch zwischen unterschiedlichen Dateiformaten ausgetauscht werden können. Frames ohne vereinheitlichte Namen können über die formatspezifische Bezeichnung erreicht werden.

Vereinheitlicht	ID3v2.3	ID3v2.4	MP4	ASF	Vorbis	RIFF
Titel	TIT2	TIT2	©nam	Title	TITLE	INAM
Interpret	TPE1	TPE1	©ART	Author	ARTIST	IART
Album	TALB	TALB	©alb	WM/AlbumTitle	ALBUM	IPRD
Kommentar	COMM	COMM	©cmt	Description	COMMENT	ICMT
Datum	TYER	TDRC	©day	WM/Year	DATE	ICRD
Stücknummer	TRCK	TRCK	trkn	WM/TrackNumber	TRACKNUMBER	IPRT oder ITRK
Genre	TCON	TCON	©gen	WM/Genre	GENRE	IGNR
Albuminterpret	TPE2	TPE2	aART	WM/AlbumArtist	ALBUMARTIST	
Arrangierung	IPLS	TIPL	ARRANGER	WM/Producer	ARRANGER	IENG
Interpret	TOLY	TOLY	AUTHOR		AUTHOR	
BPM	TBPM	TBPM	tempo	WM/BeatsPerMinute	BPM	IBPM
Katalognummer	TXXX:CATALOGNUMBER	TXXX:CATALOGNUMBER			CATALOGNUMBER	
Kompilation	TCMP	TCMP	cpil		COMPILATION	
Komponist	TCOM	TCOM	©wrt	WM/Composer	COMPOSER	IMUS
Leiter/Dirigent	TPE3	TPE3	CONDUCTOR	WM/Conductor	CONDUCTOR	
Copyright	TCOP	TCOP	cprt	Copyright	COPYRIGHT	ICOP
Beschreibung	TIT3	TIT3	desc	WM/SubTitleDescription	DESCRIPTION	
CD-Nummer	TPOS	TPOS	disk	WM/PartOfSet	DISCNUMBER	
Kodiert von	TENC	TENC	©enc	WM/EncodedBy	ENCODED-BY	ITCH
Kodierungseinstellungen	TSSE	TSSE	©too	WM/EncodingSettings	ENCODERSETTINGS	ISFT
Kodierdatum		TDEN		WM/EncodingTime	ENCODINGTIME	IDIT

## Das Kid3 Handbuch

Gruppierung	GRP1	GRP1	©grp		GROUPING	
Anfangst- onart	TKEY	TKEY		WM/InitialKey	INITIALKEY	
ISRC	TSRC	TSRC	ISRC	WM/ISRC	ISRC	ISRC
Sprache	TLAN	TLAN	LANGUAGE	WM/Language	LANGUAGE	ILNG
Texter	TEXT	TEXT	LYRICIST	WM/Writer	LYRICIST	IWRI
Text	USLT	USLT	©lyr	WM/Lyrics	LYRICS	
Medium	TMED	TMED	SOURCEMEDIA		SOURCEMEDIA	IMED
Stimmung		TMOO		WM/Mood	MOOD	
Originalalbum	TOAL	TOAL	ORIGINALALBUM	WM/OriginalAlbumTitle	ORIGINALALBUM	
Originalkünstler	TOPE	TOPE	ORIGINALARTIST	WM/OriginalArtist	ORIGINALARTIST	
Original Erscheinungsdatum	TORY	TDOR	ORIGINALDATE	WM/OriginalReleaseYear	ORIGINALDATE	
Musiker	IPLS	TMCL	PERFORMER		PERFORMER	ISTR
Bild	APIC	APIC	covr	WM/Picture	METADATA_BLOCK_PICTURE	
Herausgeber	TPUB	TPUB	PUBLISHER	WM/Publisher	PUBLISHER	IPUB
Bewertung	POPM	POPM	rate	WM/SharedUserRating	RATING	IRTD
Veröffentlichungsland	TXXX:RELEASECOUNTRY	TXXX:RELEASECOUNTRY			RELEASECOUNTRY	ICNT
Erscheinungsdatum		TDRL	RELEASEDATE		RELEASEDATE	
Remixer	TPE4	TPE4	REMIXER	WM/ModifiedBy	REMIXER	IEDT
Sortiername des Albums	TSOA	TSOA	soal	WM/AlbumSortOrder	ALBUMSORT	
Sortiername des Albuminterpreten	TSO2	TSO2	soaa		ALBUMARTISTSORT	
Sortiername des Interpreten	TSOP	TSOP	soar	WM/ArtistSortOrder	ARTISTSORT	
Sortiername des Komponisten	TSOC	TSOC	soco		COMPOSERSORT	

## Das Kid3 Handbuch

Sortiername des Titels	TSOT	TSOT	sonm	WM/TitleSortOrder	TITLESORT	
Untertitel		TSST	SUBTITLE	WM/SubTitle	SUBTITLE	PRT1
Webseite	WOAR	WOAR	WEBSITE	WM/AuthorURL	WEBSITE	IBSU
Werk	TIT1	TIT1	©wrk	WM/ContentGroupDescription	WORK	
WWW Audio-Datei	WOAF	WOAF		WM/AudioFileURL	WWWAUDIOFILE	
WWW Audio-Quelle	WOAS	WOAS		WM/AudioSourceURL	WWWAUDIOSOURCE	

Tabelle 3.1: Zuordnung der vereinheitlichten Frame-Namen in den unterschiedlichen Formaten

Bemerkungen zu den Zuordnungen der vereinheitlichten Frame-Namen:

- Die Zahl der vereinheitlichten Frame-Namen ist insofern beschränkt, als dass in allen unterstützten Dateiformaten eine sinnvolle Zuordnung möglich sein soll. Die meisten Tags unterstützen Frames mit beliebigen Namen; diese werden dann verwendet, wenn kein spezifisches Frame zur Verfügung steht (z.B. die Namen in Großbuchstaben in der Spalte MP4). Falls keine solche Möglichkeit besteht, so kann es vorkommen, dass für ein Format bestimmte Frame-Typen nicht unterstützt werden können, so Autor und Musiker für ASF (WMA).
- Die Zuordnungen sind nicht rein willkürlich, sondern orientieren sich daran, wie die Frames in anderen Programmen und Geräten verwendet werden. So legt das ID3v2-Frame "TPE2 - Band/Orchester/Begleitung" nicht unbedingt die Verwendung als Albuminterpret nahe, doch ist dies die gebräuchlichste Verwendung. Die eigentliche Bedeutung bei ID3v2 ist dann wiederum der Grund, wieso dieses Frame beim Importieren (z.B. von Discogs) für das Orchester verwendet wird, obwohl dies für andere Tag-Formate etwas sonderbar erscheinen mag.
- Die Zuordnung ist nicht immer eindeutig. So wird bei ID3v2.3 sowohl für Arrangierung als auch für Musiker ein IPLS-Frame verwendet. Beim Zurücklesen wird dann für beide Typen "Arrangierung" angezeigt.
- Die Frames Arrangierung und Musiker verwenden ein besonderes Format für ihren Inhalt: "Mitwirkung 1|Mitwirkender 1|Mitwirkung 2|Mitwirkender 2|...", beispielsweise "Chorus Master|Ernst Dunshirn|Soprano Vocals|Anna Netrebko". Dadurch wird für die IPLS bzw. TIPL/TMCL Frames in MP3-Dateien eine Zeichenkettenliste mit dem spezifizierten Inhalt erzeugt (das "|" dient als Trennzeichen zwischen den einzelnen Zeichenketten). Entsprechende Werte werden auch beim Import von Daten von Servern, welche diese Informationen bieten, gesetzt.
- Um einen spezifischen Frame-Namen zu verwenden, welcher mit einem vereinheitlichten Frame-Namen kollidiert, muss ein Ausrufezeichen vorangestellt werden. Wenn man beispielsweise einem Vorbis-Tag ein Frame vom Typ "Media" hinzufügt, so wird ein "SOURCEMEDIA"-Frame erzeugt, weil es so zugeordnet ist. Um ausdrücklich ein "MEDIA"-Frame und nicht ein "SOURCEMEDIA"-Frame zu erhalten, muss "!MEDIA" verwendet werden.

### 3.1.9 Synchronisierter Text und Ereignis-Zeitcodes

Für zeitlich mit den Audio-Daten synchronisierte Informationen steht ein spezieller Editor zur Verfügung. Diese Frames sind für ID3v2.3.0 und ID3v2.4.0 Tags unterstützt. Um ein solches Frame zu erstellen, muss in der Liste, welche nach Klick auf **Hinzufügen** erscheint, sein Name ausgewählt werden - **Synchronisierter Text** oder **Ereignis-Zeitcodes**. Beide Frame-Typen werden in einem gleichartigen Editor bearbeitet, bei den Ereignis-Zeitcodes kann aus einer Menge vordefinierter Codes ausgewählt werden, während beim synchronisierten Text der Liedtext eingegeben werden muss. Im folgenden wird die Eingabe von synchronisiertem Text erklärt.

Eine Datei, welche ein ID3v2 Tag enthält, wird selektiert und der Editor wird mit **Hinzufügen** und Auswahl von **Synchronisierter Text** geöffnet. Falls das Synchronisierter Text Frame schon existiert, wird es selektiert und auf **Bearbeiten** geklickt. Die Werkzeugleiste zum Abspielen von Audio-Dateien wird automatisch geöffnet, so dass das aktuelle Stück abgespielt und angehalten werden kann, um die Texte zu synchronisieren.

Die Einstellungen im oberen Teil des SYLT Editors müssen normalerweise nicht verändert werden. Wenn der Text Zeichen enthält, welche nicht im Latin 1 Zeichensatz enthalten sind, so kann die Textkodierung auf UTF16 (oder UTF8 für ID3v2.4.0) umgestellt werden. Für englischsprachige Texte und maximale Kompatibilität sollte ISO-8859-1 verwendet werden.

Der Abschnitt **Text** hat fünf Knöpfe oberhalb der Tabelle. **Hinzufügen** erstellt einen neuen Eintrageintrag in der Tabelle. Die Zeit wird von der momentanen Position der Wiedergabe genommen, so dass ein Eintrag für die aktuelle Stelle des abgespielten Stücks erzeugt wird. Die Einträge in der Tabelle müssen zeitlich geordnet sein, die neue Zeile wird daher entsprechend eingeordnet. Einträge mit ungültiger Zeit werden speziell behandelt: Wenn die aktuelle Zeile eine ungültige Zeit hat, so wird ihre Zeit durch die Wiedergabezeit ersetzt, und es wird keine neue Zeile hinzugefügt. Wenn die Zeit nicht ungültig ist, so wird die erste Zeile mit einer ungültigen Zeit genommen, falls eine solche vorhanden ist. Dieses Verhalten sollte es erleichtern, die Zeit-Spalte auszufüllen, wenn der Text bereits in der Tabelle vorhanden ist, aber die Zeiten noch fehlen (was der Fall ist, wenn unsynchronisierte Texte importiert werden). Man beachte, dass ungültige Zeiten mit 00:00.00 dargestellt werden, also gleich wie die Zeit ganz zu Beginn des Stücks, welche nicht ungültig ist. Um einen Zeiteintrag ungültig zu machen, kann die **Delete** Taste gedrückt werden oder **Löschen** aus dem Kontextmenü verwendet werden. Neue Zeilen, welche mit **Zeile einfügen** des Kontextmenüs oder durch **Importieren** bzw. **Von Zwischenablage** erstellt werden, enthalten ebenfalls ungültige Zeiten. Zeilen in der Tabelle können mit Klick auf **Löschen** oder dem **Zeilen entfernen** Kontextmenü gelöscht werden.

Synchronisierter Liedtext kann aus einer Datei importiert werden mit **Importieren**. Als Dateiformat wird LRC (einfach oder erweitert) erwartet. Wenn die ausgewählte Datei keine eckige Klammer in der ersten Zeile enthält, so wird angenommen, dass es sich um eine einfache Textdatei mit unsynchronisiertem Liedtext handelt. Die Zeilen einer solchen Datei werden dann in die Text-Spalte importiert und die Zeit-Spalte enthält ungültige Zeiten. Die Zeitinformatoren können mit Hilfe des **Hinzufügen**-Knopfs ergänzt oder manuell erfasst werden. Es ist auch möglich, Liedtexte mittels Kopieren und Einfügen zu importieren (**Von Zwischenablage**). **Exportieren** erlaubt es, synchronisierten Text in eine LRC-Datei zu schreiben. Man beachte, dass nur Einträge mit gültiger Zeit exportiert werden, und dass die Einträge zeitlich sortiert werden. Einträge mit ungültiger Zeit werden auch nicht in das SYLT-Frame geschrieben, man sollte daher alle Zeitinformatoren eintragen, bevor man den Editor verlässt.

Die **ID3 Spezifikation** legt es nahe, Zeitstempel für jede Silbe zu erfassen. Die meisten Player unterstützen jedoch bloß eine Granularität von einer Zeile bzw. einem Satz. Um beide Anwendungsfälle zu unterstützen, folgt Kid3 denselben Konventionen wie **SYLT Editor**. Es wird angenommen, dass Text, welcher in der Tabelle eingetragen wird, eine neue Zeile startet, es sei denn, er beginnt mit einem Leerzeichen oder Bindestrich. Diese Regel kann außer Kraft gesetzt werden, indem man den Text mit einem Unterstrich ('\_') beginnen lässt, um eine Fortsetzung der Zeile zu erzwingen, bzw. mit einem Doppelkreuz ('#') um eine neue Zeile auch dann zu erhalten, wenn diese mit einem Leerzeichen oder Bindestrich beginnt. Diese Umschaltzeichen werden nicht im SYLT-Frame gespeichert. Innerhalb des Frames beginnen neue Zeilen mit einem Zeilenumbruch (das Zeichen mit Hex-Code 0A), ansonsten handelt es sich um eine Fortsetzung der Zeile. Beim Lesen existierender SYLT-Frames prüft Kid3, ob der erste Text mit einem Zeilenumbruch beginnt.

Ist dies nicht der Fall, so wird angenommen, dass alle Texte separate Zeilen sind und keine Silbenfortsetzungen verwendet werden.

Während dem Abspielen einer Datei wird die aktuelle Zeile hervorgehoben, so dass die Korrektheit der Synchronisationsinformation überprüft werden kann. Existiert ein Versatz, so kann er mit dem Kontextmenü **Zeitabstand hinzufügen** korrigiert werden, wobei negative Werte die Zeit reduzieren. Die Funktion **Zur Position springen** ermöglicht es, die Position der Wiedergabe an die Zeit der ausgewählten Zeile zu setzen.

*Empfohlenes Vorgehen zum Erstellen eines neuen synchronisierten Liedtextes*

- Unsynchronisierten Liedtext besorgen, z.B. mit **Lyrics** → **Embed Lyrics** aus dem Kontextmenü der Dateiliste.
- Den unsynchronisierten Liedtext in die Zwischenablage kopieren, dazu einfach in der **Text-Zeile** der **Tag 2** Tabelle **Strg-C** drücken.
- Element mit synchronisiertem Liedtext hinzufügen (**Hinzufügen...**, **Synchronisierter Text**, **OK**), dann **Von Zwischenablage** klicken.
- Nun sind alle Zeilen des Liedtextes in der Tabelle, die Zeitstempel sind alle ungültig (0:0:0.00). Leere Zeilen können vorangehend gelöscht werden.
- Die Wiedergabe kann mit dem Abspiel-Knopf ► der Wiedergabe-Werkzeugliste, welche sich unten am Hauptfenster befindet, gestartet werden.
- Wenn die nächste Zeile mit ungültigem Zeitstempel wiedergegeben wird, kann **Hinzufügen** geklickt oder **Alt-H** gedrückt werden, der Zeitstempel wird gesetzt.
- Auf diese Weise fortfahren, bis alle Zeitstempel gesetzt sind. Falls man eine Zeile verpasst, so kann die Wiedergabe gestoppt und falsche Zeitstempel gelöscht werden (**Delete** Taste oder mit **Löschen** aus dem Kontextmenü). Um die Wiedergabe ab einem gewissen Zeitstempel zu starten, kann **Zur Position springen** aus dem Kontextmenü verwendet werden.

## 3.2 Das Menü Datei

### **Datei** → **Öffnen...** (Strg-O)

Ein Ordner wird geöffnet, und alle Dateien, welche den gewählten Dateinamenfilter erfüllen, werden im Dateilistenfeld angezeigt. Die gewählte Datei wird selektiert.

### **Datei** → **Zuletzt geöffnete Dateien**

Ein kürzlich geöffneter Ordner wird geöffnet.

### **Datei** → **Ordner öffnen ...** (Strg-D)

Ein Ordner wird geöffnet, und alle Dateien, welche den gewählten Dateinamenfilter erfüllen, werden im Dateilistenfeld angezeigt.

### **Datei** → **Neu laden** (F5)

Ordner neu einlesen. Veränderte Dateien müssen vorher gespeichert werden. Ausgeklappte Unterordner werden eingeklappt.

### **Datei** → **Speichern** (Strg-S)

Speichert alle veränderten Dateien im Ordner. Die veränderten Dateien sind durch ein Diskettensymbol gekennzeichnet. Wenn Dateinamen verändert wurden, so werden die entsprechenden Dateien umbenannt.



### Datei → Zuletzt gespeicherte Fassung

Macht die Änderungen an einer oder mehreren Dateien rückgängig. Wenn keine Dateien im Dateilistenfeld selektiert sind, so werden die Änderungen an allen Dateien rückgängig gemacht, ansonsten nur an den ausgewählten Dateien.

### Datei → Importieren...

Der Import Dialog wird verwendet, um Daten direkt von einem freedb.org Server, von einem MusicBrainz Server, von Discogs, Amazon oder anderen Quellen mit Albumtitellisten zu importieren.

Zum Importieren von einem freedb.org-Server wird **Von Server: gnudb.org** gewählt. Der zu suchende Interpret und der Name des Albums können in den zwei oberen Feldern eingegeben werden, und die gefundenen Alben werden nach Klick auf **Suchen** dargestellt, sobald sie von [www.gnudb.org](http://www.gnudb.org) empfangen werden. Die Stückdaten eines Albums werden nach Auswahl geladen, der verwendete freedb.org-Server kann gewählt werden, ebenso der CGI Pfad. Die importierten Daten werden in der Vorschau des Import-Dialogfensters dargestellt. Wenn man mit den Daten in der Vorschau zufrieden ist, können sie mit **OK** importiert werden.

Zum Importieren vom Discogs Server wird **Discogs** gewählt. Wie bei **gnudb.org** können auch hier Interpret und Album eingegeben werden und es wird ebenfalls eine Liste mit Alben zur Auswahl gestellt. Ein **Token** kann eingegeben werden, um das RESTful Discogs API zu verwenden statt der Web-Schnittstelle, welche oft geändert wird und dann funktioniert der Import-Parser nicht mehr. Dazu muss man sich bei [Discogs](http://Discogs) registrieren und dann dort ein Token generieren (Settings/Developers, Generate new token). Man sollte nicht vergessen, das Token mit **Einstellungen speichern** zu sichern, damit es auch für künftige Anfragen verwendet wird. Wenn **Standard-Tags** markiert ist, werden grundlegende Informationen importiert, so z.B. Interpret, Album und Titel. Wenn **Zusätzliche Tags** markiert ist, werden mehr verfügbare Informationen importiert, z.B. Musiker, Arrangierer oder der Herausgeber. Wenn **Cover-Bilder** markiert ist, werden Cover-Bilder heruntergeladen, falls vorhanden.

Zum Importieren von Amazon wird **Amazon** gewählt. Wie bei **Von gnudb.org** können auch hier Interpret und Album eingegeben werden und es wird ebenfalls eine Liste mit Alben zur Auswahl gestellt. Wenn **Zusätzliche Tags** markiert ist, werden mehr verfügbare Informationen importiert, z.B. Musiker, Arrangierer oder der Herausgeber. Wenn **Cover-Bilder** markiert ist, werden Cover-Bilder heruntergeladen, falls vorhanden.

In der gleichen Art kann mit **Von MusicBrainz Album** auch in der Album-Datenbank von MusicBrainz gesucht werden. Die Bedienung ist wie bei **gnudb.org** beschrieben.

Zum Importieren von einem MusicBrainz Server dient der Dialog, der nach Klicken auf **Von MusicBrainz Fingerprint** erscheint. Wie bei freedb kann auch hier der Server ausgewählt werden. Unten ist eine Tabelle, welche die importierten Stückdaten darstellt. Die rechte Spalte zeigt dabei den Zustand der MusicBrainz-Abfrage, welche gleich mit dem Öffnen des Dialogs mit "Suche läuft" beginnt. Es folgen dann ein Abfrage des Fingerprints und, falls diese kein Ergebnis liefert, eine Suche anhand der in der Datei vorhandenen Tags. Für eine erfolgreiche MusicBrainz-Suche kann es daher hilfreich sein, vorhandene Informationen (z.B. Interpret und Album) vorangehend in den Tags zu speichern. Wenn ein Resultat gefunden wurde, so endet die Suche im Zustand "Erkannt", andernfalls wurde nichts gefunden oder gleich mehrere mögliche Resultate, von denen der Benutzer eines auswählen kann. **OK** und **Anwenden** übernehmen die importierten Daten, **Abbrechen** schließt den Dialog. Das Schließen des Dialogs kann eine Weile dauern, da noch die ganze MusicBrainz-Maschinerie runtergefahren werden muss.

Zum Importieren von Albumdaten in Textform öffnet man einen Unterdiallog mit **Von Datei/Zwischenablage**. Dort stehen mehrere vorkonfigurierte Formate zur Verfügung. Die ersten zwei, "CSV unquoted" und "CSV quoted", dienen dazu, Daten welche über die Export Funktion exportiert wurden, wieder zu importieren. Diese Daten können natürlich von einer Tabellenkalkulation bearbeitet werden. CSV-Daten aus einer Tabellenkalkulation müssen mit Tabulator als Trennzeichen geschrieben werden und sollten sich über "CSV

quoted“ einlesen lassen. Dieses ist flexibler als “CSV unquoted“, dafür dürfen die enthaltenen Felder keine Anführungszeichen enthalten. Falls nur von Kid3 exportiert und wieder importiert wird, so sollte dies problemlos mit “CSV unquoted“ möglich sein. Im Kontextmenü der Dateiliste hat es darüber hinaus die Funktionen “Export CSV“ und “Import CSV“, welche Skripts benutzen, um CSV-Daten vollständiger und flexibler auszutauschen.

Das nächste Format, “freedb HTML text“, kann benutzt werden, um Informationen aus einer HTML-Seite von [freedb.org](http://freedb.org) zu kopieren. Dazu sucht man sich in freedb das Album, und wenn die gewünschte Information im Browser angezeigt wird, kann sie selektiert und in die Zwischenablage kopiert werden. Nach einem Klick auf **Von Zwischenablage** werden die importierten Daten in der Tabelle angezeigt und können mit **OK** übernommen werden. Das Ziel (**Tag 1**, **Tag 2** oder **Tag 1 und Tag 2**) kann mit einem Auswahlfeld festgelegt werden. Die Titel in der Dateiliste sollten in der korrekten Reihenfolge angezeigt sein um die richtigen Tags zu kriegen. Dies ist kein Problem, wenn sie nummeriert sind.

Das nächste vorkonfigurierte Format, “freedb HTML source“, kann dann benutzt werden, wenn die Import-Daten als HTML-Dokument vorliegen. Solch eine Datei kann im Datei Dialog, welcher nach Klick auf **Von Datei** erscheint, ausgewählt werden. Natürlich können die Daten auch in einem Editor kopiert und dann über die Zwischenablage übernommen werden. Dieses Format kann für Offline Import nützlich sein, obschon die HTML-Datei auch von einem Browser angezeigt und im ersten Format über die Zwischenablage importiert werden kann.

Weitere vorkonfigurierte Formate, z.B. “Track Title Time“ stehen zur Verfügung. Mit Klick auf **Hinzufügen** kann ein leeres benutzerdefiniertes Format erzeugt und vom Benutzer gesetzt werden. Die zwei Zeilen unterhalb des Formatnamens können hierzu mit regulären Ausdrücken gefüllt werden. Der erste Ausdruck wird einmal angewendet, um albumspezifische Daten wie Interpret, Album, Jahr und Genre zu extrahieren. Die zweite Zeile wird vom Dokumentanfang bis zum Schluss für jeden Titel probiert, um Titeldaten zu kriegen, also Nummer und Titel. In den regulären Ausdrücken können alle von Qt unterstützten Funktionen eingesetzt werden; also das meiste von dem, was Perl bietet. Klammerkonstrukte “(.)“ werden benützt, um Felddaten zu extrahieren und werden von einem Kid3-spezifischen Feldcode eingeleitet. Die Codes sind dieselben wie beim Dateinamenformat, auch hier sind nebst den unten aufgelisteten Codes beliebige Framenamen möglich:

- %s %{title} Titel (Song)
- %a %{artist} Interpret (Artist)
- %l %{album} Album
- %c %{comment} Kommentar (Comment)
- %y %{year} Jahr (Year)
- %t %{track} Nummer (Stück)
- %g %{genre} Genre
- %d %{duration} Länge (Dauer)

Beispiel: Ein regulärer Ausdruck für die zweite Zeile, um Titel aus einer .m3u Stückliste zu holen, könnte “%t{track}(\d+)\s+%t{title}(\S[^\r\n]\*)\.mp3[\r\n]“ sein. Alle Formate können geändert werden, indem der Name und die regulären Ausdrücke editiert werden und **Einstellungen speichern** gedrückt wird. Gespeichert werden sie in der Datei `kid3rc` im Konfigurationsordner. Diese Datei kann auch direkt verändert werden, um mehr Importformate zu haben, oder man kann sie löschen, um wieder zu den Standardeinstellungen zu kommen. Formate können mit **Entfernen** gelöscht werden.

**Genauigkeit** zeigt eine Abschätzung, wie gut die importierten Daten zu den Stücken passen. Dazu werden die Längen oder die Dateinamen verglichen um den Grad an Übereinstimmung in Prozent zu errechnen. **Cover-Bild** zeigt die URL des Cover-Bildes, welches heruntergeladen wird.

Um zu prüfen, ob die importierten Stücke den vorhandenen Dateien entsprechen, können die Längen verglichen werden. Diese Option kann mit dem Markierungsfeld **Teste maximal erlaubten Zeitunterschied** eingeschaltet werden und der maximal erlaubten Zeitunterschied kann in Sekunden eingestellt werden. Wird eine Überschreitung dieser Toleranzschwelle festgestellt, so wird die entsprechende Länge in der Vorschau rot dargestellt.

Falls nun die vorhandenen Dateien in einer anderen Reihenfolge als die importierten Stücke vorliegen, so sollte die Zuordnung korrigiert werden. Diese Arbeit wird durch die **Zuordnen mit** Knöpfe **Länge**, **Nummer** und **Titel** erleichtert, welche die Stücke dem jeweiligen Feld entsprechend zuordnen. Sollte die Zuordnung noch manuell korrigiert werden müssen, so kann ein Track nach Drücken der Steuerungs-Taste (**Strg/Strg**) mit der linken Maustaste gefasst und dann an die gewünschte Stelle gezogen werden.

Wenn der Import Dialog geöffnet wird, enthält die Tabelle die aktuell vorhandenen Werte der Tags. Der Tag Typ (Tag 1, Tag 2, Tag 1 und Tag 2) kann über das **Ziel** Auswahlfeld selektiert werden. Der Knopf rechts von diesem Auswahlfeld dient dazu, die Tabelle auf die in den Tags gespeicherten Werte zurückzusetzen. Über die Markierungsfelder in der ersten Tabellenspalte können die Stücke gewählt werden, deren Tags importiert werden sollen. Dies kann z.B. dann nützlich sein, wenn ein Ordner alle Stücke eines Doppelalbums enthält, aber bloß die Stücke der zweiten CD importiert werden sollen.

Um die Dateien, welche importiert werden, zu identifizieren, können die Dateinamen oder die vollständigen Pfade in der Tabelle dargestellt werden. Diese Einstellung ist über das Kontextmenü, welches bei Rechtsklick auf die Kopfzeile erscheint, möglich. Die Werte in der Import-Tabelle lassen sich editieren. Die momentan in den Tags vorhandenen Werte lassen sich mit dem Rückgängig-Knopf rechts vom **Ziel**-Listenfeld wiederherstellen. Dies ist auch dann nützlich, wenn man das **Ziel** verändert hat und die dort vorhandenen Werte in der Tabelle sehen möchte.

Wie in fast allen Dialogen dient der **Einstellungen speichern** Knopf dazu, die dialogspezifischen Optionen und die Fenstergröße persistent zu speichern.

**Von Tags** führt zu einem Unterdialog, mit welchem sich Tag Frames mit dem Inhalt aus anderen Tag Frames füllen lassen. Dies kann dazu benutzt werden, um einfach Informationen zwischen Tags zu kopieren oder einen Teil des Inhaltes eines Tag Frames zu extrahieren und in einem anderen Tag Frame einzufügen.

Wie beim **Import von Datei/Zwischenablage** stehen frei konfigurierbare Formate zur Verfügung, um verschiedene Operationen durchzuführen. Vorkonfigurierte Formate erlauben es beispielsweise, den Wert des Interpreten zum Album Interpreten, Komponisten oder Leiter zu kopieren, bei Titeln, welche die Stücknummer enthalten, diese zu extrahieren und im Stücknummer-Frame zu speichern. Ebenso lässt sich ein Untertitel aus dem Titel extrahieren.

Das Erstellen eines eigenen Importformates wird im folgenden anhand eines Beispiels erklärt. Die Information im Untertitel soll auch im Kommentar gesetzt werden. Dazu wird ein neues Format mit **Hinzufügen** erstellt und ein neuer Name gesetzt, in diesem Beispiel also "Subtitle to Comment". Nun gibt man unter **Quelle** `"%{subtitle}"` und unter **Extraktion** `"%{comment}(.*)"` ein und klickt **Einstellungen speichern**.

Der Ausdruck in **Quelle** kann aus **Format Codes** für beliebige Tag Frames bestehen, wobei natürlich auch mehrere Codes verwendet werden können, um den Inhalt verschiedener Frames zu kombinieren. Für jedes Stück wird aus den Tags und dem **Quelle** Format ein Text erzeugt, und auf diesen wird dann der reguläre Ausdruck aus **Ermittlung** angewendet, um neue Tag Werte zu setzen. Dieser reguläre Ausdruck enthält Klammerkonstrukte `"(.)"`, um Felddaten zu extrahieren, und die Klammer-Konstrukte werden vom Format Code des Tag Frames eingeleitet. Es funktioniert also gleich wie bei den Formaten zum **Import von Datei/Zwischenablage**.

**Import von Tags...** ist auch direkt im Menü **Datei** verfügbar. Der Unterschied zwischen diesen zwei Funktionen ist, dass der Unterdialog des Import-Dialogs auf allen Dateien des aktuellen Ordners arbeitet, während die Menü-Funktion auf die selektierten Dateien angewendet wird (welche in verschiedenen Ordnern liegen können). Die Menü-Funktion verfügt über einen zusätzlichen Format Code `"%{__return}"` um den extrahierten Wert zurückzugeben, was im CLI und bei der QML-Schnittstelle nützlich sein kann.

#### **Datei** → **Importieren von gnudb.org...**

Importieren von einem freedb.org Server mit Album-Suche auf gnudb.org. Dieser Menüpunkt öffnet denselben Import Dialog wie **Importieren...**, öffnet jedoch direkt auch den **gnudb.org** Dialog.

**Datei → Importieren von Discogs...**

Importieren vom Discogs Server. Dieser Menüpunkt öffnet denselben Import Dialog wie **Importieren...**, öffnet jedoch direkt auch den **Discogs** Dialog.

**Datei → Importieren von Amazon...**

Importieren von Amazon. Dieser Menüpunkt öffnet denselben Import Dialog wie **Importieren...**, öffnet jedoch direkt auch den **Amazon** Dialog.

**Datei → Importieren von MusicBrainz Album...**

Importieren von der MusicBrainz Album Datenbank. Dieser Menüpunkt öffnet denselben Import Dialog wie **Importieren...**, öffnet jedoch direkt auch den **Von MusicBrainz Album** Dialog.

**Datei → Importieren von MusicBrainz Fingerprint...**

Importieren von einem MusicBrainz Server. Dieser Menüpunkt öffnet denselben Import Dialog wie **Importieren...**, öffnet jedoch direkt auch den **Von MusicBrainz Fingerprint** Dialog.

**Datei → Importieren von Tags...**

Wie [Import von Tags](#), jedoch wird der Import auf die selektierten Dateien angewendet.

**Datei → Automatisch importieren...**

Automatisch importieren lassen sich Informationen für mehrere Alben von verschiedenen Web-Diensten. Wenn Ordner in der Dateiliste selektiert sind, so werden Metadaten für die Stücke der gewählten Ordner importiert. Wenn kein Ordner ausgewählt ist, so werden Informationen für alle Ordner innerhalb der Dateiliste importiert.

Der Tag Typ (Tag 1, Tag 2, Tag 1 und Tag 2) kann über das **Ziel** Auswahlfeld selektiert werden.

Profile bestimmen, von welchen Servern die Album-Informationen geholt werden. Einige Profile sind bereits vordefiniert (All, MusicBrainz, Discogs, Cover Art), eigene Profile können mit dem **Hinzufügen** Knopf rechts von der **Profil** Auswahlliste erstellt werden.

Die Tabelle darunter zeigt die Server, welche benutzt werden, wenn mit dem ausgewählten Profil importiert wird. Der Import-Prozess für ein Album wird beendet, wenn alle geforderten Informationen gefunden wurden, daher ist die Reihenfolge der Server in der Tabelle wichtig. Sie kann mit den **Nach oben** und **Nach unten** Knöpfen verändert werden. Mit **Bearbeiten** kann ein bestehender Eintrag verändert werden. Die **Server** Auswahl stellt dieselben Server zur Verfügung wie die Import-Funktionen. **Standard Tags**, **Zusätzliche Tags** und **Cover-Bilder** bestimmen die Informationen, welche vom Server geholt werden sollen. Die **Genauigkeit** schließlich bestimmt die minimale Übereinstimmung, welche erreicht werden muss, damit die importierten Daten akzeptiert werden. Ist sie ungenügend, so wird ein Import mit dem nächsten Server versucht. Dieser Dialog mit den Server-Eigenschaften wird auch angezeigt wenn **Hinzufügen** gewählt wird, um einen neuen Eintrag zu erstellen. Bestehende Zeilen können mit **Entfernen** gelöscht werden.

Um einen automatischen Import mit dem ausgewählten Profil zu beginnen wird **Starten** gedrückt. Details über den laufenden Prozess werden im Dialog oben angezeigt. **Abbrechen** führt zu einer vorzeitigen Beendigung des Imports.

**Datei → Nach Cover-Bildern suchen...**

Der Dialog Nach Cover-Bildern suchen hilft bei der Suche nach Album Cover-Bildern. **Interpret/Album** werden wenn möglich aus den Tags übernommen. **Quelle** stellt eine Vielzahl von Websites mit Cover-Bildern zur Verfügung. Unter dem Namen ist die URL mit Interpret und Album als Parametern angegeben. Für URL-codierte Werte können dabei **%u{artist}** und **%u{album}** angegeben werden oder auch andere Werte aus den Tags, wie in [Kid3 einrichten](#) bei **Benutzerbefehle** beschrieben. Nach "Custom Source" können auch eigene Quellen hinzugefügt werden. Dazu wird "Custom Source" durch den Namen der Quelle ersetzt, dann Return gedrückt, die URL eingegeben und mit **Einstellungen speichern** gesichert. Das resultierende Browser-Kommando wird oben im Dialog dargestellt

und nach Drücken des **Browser**-Knopfes ausgeführt. Der Browser, welcher in den Einstellungen konfiguriert werden kann, wird mit der gewählten Quelle gestartet. Das gewünschte Cover-Bild kann dann aus dem Browser in das Kid3 Fenster gezogen werden und wird im Bild-Feld von Tag 2 der selektierten Dateien gesetzt.

Da nicht alle Browser Drag'n'Drop von Bildern unterstützen und häufig die Bilder auf Websites mit einer URL unterlegt sind, erhält Kid3 in solchen Fällen bloß die entsprechende URL und nicht das Bild. Wenn die URL auf ein Bild zeigt, wird dieses heruntergeladen. Falls die URL jedoch auf sonst eine Website hinweist, so muss von dieser auf das gewünschte Cover-Bild geschlossen werden. Dies ist die Aufgabe der Tabelle **URL Ersetzung**. Die linke Spalte **Vergleich** enthält einen regulären Ausdruck, welcher mit der URL verglichen wird. Wenn eine Übereinstimmung festgestellt wird, so werden die erhaltenen Klammer-Werte im Muster der Spalte **URL des Bildes** ersetzt (\1 usw.). Der ersetzte reguläre Ausdrucks enthält dann die URL des Bildes. Auf diese Weise können auch Bilder von Amazon, Google Images, usw. per Drag'n'Drop importiert werden. Natürlich können auch hier eigene Ersetzungen erfasst werden.

#### **Datei → Exportieren...**

Der Export Dialog wird verwendet, um die Daten aus den Tags in einer Datei oder der Zwischenablage zu speichern. Das oberste Editierfeld stellt einen einfachen Editor zur Verfügung, welcher eine Vorschau der zu exportierenden Daten bietet. Falls die Export-Daten Tabulatorzeichen enthalten, wird der Export in einer Tabelle dargestellt. Die Daten werden aus den Tags der Dateien im aktuellen Ordner generiert, und zwar gemäß dem unten eingestellten Format.

Diese Formateinstellung ist gleich aufgebaut wie im Import Dialog: Im obersten Feld ist der Titel (z.B. "CSV unquoted"), dann folgt der Header - dieser wird nur am Anfang der Datei generiert. Nun folgt das Stück-Format, es wird für jedes Stück angewendet. Schließlich gibt es noch einen Trailer, welcher die Datei noch abschließt.

Die Formatfelder enthalten keine regulären Ausdrücke wie im Import Dialog, sondern bloß Ausgabeformate, bei denen spezielle %-Ausdrücke durch Werte aus den Tags ersetzt werden. Das ganze verhält sich also gleich wie beim Dateinamenformat, es werden auch dieselben Codes verwendet, plus ein paar zusätzliche. Nicht nur die unten aufgelisteten Codes sind möglich, es können die Namen aller Tag Frames verwendet werden.

- %s %{title} Titel (Song)
- %a %{artist} Interpret (Artist)
- %l %{album} Album
- %c %{comment} Kommentar (Comment)
- %y %{year} Jahr (Year)
- %t %{track} Nummer (Stück, z.B. 01)
- %t %{track.n} Nummer mit Feldlänge n (z.B. 001 für %{track.3})
- %T %{tracknumber} Nummer (Stück, ohne führende Null, z.B. 1)
- %g %{genre} Genre
- %f %{file} Dateiname (File name)
- %p %{filepath} Pfadname
- %{modificationdate} Änderungsdatum
- %{creationdate} Erstellungsdatum
- %u %{url} URL
- %{dirname} Ordnername
- %d %{duration} Dauer in Minuten:Sekunden
- %D %{seconds} Dauer in Sekunden
- %n %{tracks} Anzahl Stücke im Album
- %e %{extension} Dateiendung



- %O %{tag1} Das Format von Tag 1 (ID3v1.1 oder leer falls nicht vorhanden)
- %o %{tag2} Das Format von Tag 2 (ID3v2.3.0, ID3v2.4.0, ID3v2.2.0, ID3v2.2.1, Vorbis, APE, MP4, ASF oder leer falls nicht vorhanden)
- %b %{bitrate} Bitrate in kbit/s
- %v %{vbr} VBR oder leer (nur für ID3v2.3 mit id3lib)
- %r %{samplerate} Samplerate in Hz
- %m %{mode} Kanalmodus (Stereo oder Joint Stereo)
- %h %{channels} Anzahl Kanäle (1 oder 2)
- %k %{codec} Codec (z.B. MPEG 1 Layer 3, MP4, Ogg Vorbis, FLAC, MPC, APE, ASF, AIFF, WAV)

Mit diesen Formatcodes sind bereits einige Formate vordefiniert. "CSV unquoted" trennt die einzelnen Felder durch Tabulatoren. Daten in diesem Format können einfach wieder über das gleichnamige Importformat in Kid3 importiert werden. "CSV quoted" setzt die Felder zusätzlich noch in Anführungszeichen, was den Import in Tabellenkalkulationen erleichtert. Dafür sollten in den Feldern keine Anführungszeichen vorkommen. "Extended M3U" und "Extended PLS" erzeugen Stücklisten mit Zusatzinformationen und absoluten Pfadnamen. "HTML" dient dazu, eine HTML-Seite mit verlinkten Stücken zu erstellen. "Kover XML" generiert eine XML-Datei, welche vom Cover-Druck-Programm Kover gelesen werden kann. "Technical Details" liefert Informationen über Bitrate, Samplerate, Kanäle, usw. "Custom Format" schließlich ist frei für ein eigenes Format. Natürlich können auch die anderen Formate beliebig angepasst werden, oder es können noch mehr Formate in der Datei kid3rc im Konfigurationsordner hinzugefügt werden.

Die Quelle der Daten (**Tag 1** oder **Tag 2**) kann mit einem Auswahlfeld festgelegt werden. Ein Klick auf **In Datei** bzw. **In Zwischenablage** speichert die Daten in einer Datei oder der Zwischenablage. **OK** und **Abbrechen** beenden den Dialog, wobei **OK** die aktuellen Einstellungen übernimmt.

#### **Datei** → **Stückliste erstellen...**

Erzeugt eine Stückliste. Das Format und der Inhalt der Stückliste können mit verschiedenen Optionen bestimmt werden.

**Wie Ordnername** generiert den Dateinamen der Wiedergabeliste aus dem Namen des Ordners. Es kann aber auch ein **Format** mit Werten aus den Tags benutzt werden, z.B. "%{artist} - %{album}", um die Namen von Interpret und Album im Dateinamen zu haben. Es können dieselben Format Codes wie bei [Exportieren](#) verwendet werden. **Erstelle neue leere Stückliste** erzeugt eine leere Stückliste mit dem angegebenen Namen. Die Dateierweiterung richtet sich nach dem Format der Stückliste.

Wo die Wiedergabeliste erstellt wird, kann bestimmt werden mit der Auswahlliste **Erstelle in**

#### **Aktueller Ordner**

Die Stückliste wird im aktuellen Ordner erstellt und enthält nur Dateien aus diesem Ordner. Der aktuelle Ordner ist der Ordner, in welchem sich die aktuelle Datei befindet. Wenn mehrere Dateien ausgewählt sind, ist wahrscheinlich die zuletzt ausgewählte Datei die aktuelle Datei.

#### **Jedem Ordner**

In jedem Ordner, welcher gelistete Dateien enthält, wird eine eigene Stückliste erstellt, welche die Dateien aus dem jeweiligen Ordner enthält.

#### **Basisordner**

Nur eine einzige Stückliste wird im Basisordner (d.h. dem Ordner der Dateiliste) erstellt, und sie enthält die gelisteten Dateien aus dem Basisordner und allen Unterordnern.

Das **Format** der Wiedergabeliste kann **M3U**, **PLS** oder **XSPF** sein.

**Nur ausgewählte Dateien einfügen** führt dazu, dass nur ausgewählte Dateien in der Stückliste erscheinen. Wenn ein Ordner ausgewählt ist, so gelten all seine Dateien ebenfalls als

selektiert. Wenn diese Option nicht aktiviert ist, gelangen alle Audio-Dateien in die Stückliste.

**Sortiere nach Dateiname** ordnet die Dateien wie üblich nach dem Dateinamen in der Wiedergabeliste an. Mit **Sortiere nach Tag-Feld** ist es möglich, eine Format-Zeichenkette mit Tag-Werten für die Sortierung zu verwenden. `"%{track}.3"` kann beispielsweise benutzt werden, um nach der Stücknummer zu sortieren (`".3"` wird verwendet, um 3 Stellen mit führenden Nullen zu kriegen, und somit korrekt nach der Nummer zu sortieren). Es ist auch möglich, mehrere Codes zu verwenden, z.B. `"%{genre}%{year}"`, um nach einer Zeichenkette aus Genre und Jahr zu sortieren.

Die Einträge in der Stückliste sind relative oder absolute Pfadangaben, je nachdem ob **Benutze relativen Pfad für Dateien in der Wiedergabeliste** oder **Benutze vollen Pfad für Dateien in der Wiedergabeliste** gesetzt ist.

**Nur Liste der Dateien schreiben** führt dazu, dass die Stückliste nur die Pfade zu den Dateien enthält. Wünscht man zusätzliche Informationen, so steht **Schreibe Informationen mit einer Format-Zeichenfolge** zur Verfügung, und es wird eine erweiterte Stückliste generiert.

#### **Datei → Beenden (Strg-Q)**

Beendet die Anwendung.

### 3.3 Das Menü Bearbeiten

#### **Bearbeiten → Alles auswählen (Alt-A)**

Selektiert alle Dateien.

#### **Bearbeiten → Abwählen (Strg-Umschalt-A)**

Deselektiert alle Dateien.

#### **Bearbeiten → Alles im Ordner auswählen**

Selektiert alle Dateien im aktuellen Ordner.

#### **Bearbeiten → Vorangehende Datei (Alt-Up)**

Selektiert die vorangehende Datei.

#### **Bearbeiten → Nächste Datei (Alt-Down)**

Selektiert die nächste Datei.

#### **Bearbeiten → Suchen... (Strg-F)**

Sucht Zeichenketten in den Dateinamen und Tags. Der **Suchen** Dialog ist eine Teilmenge des **Ersetzen**-Dialogs, welcher unten beschrieben ist.

#### **Bearbeiten → Ersetzen... (Strg-R)**

Ein Dialog wird geöffnet, mit welchem sich Zeichenketten in den Dateinamen und Tags suchen und ersetzen lassen. Die Suche kann auf bestimmte Elemente eingeschränkt werden, indem bei **Alles selektieren** das Häkchen entfernt wird und die gewünschten Elemente ausgewählt werden. Es sind auch Optionen vorhanden, um rückwärts zu suchen, Groß- und Kleinbuchstaben zu unterscheiden und für reguläre Ausdrücke.

Abhängig von der Anzahl Dateien kann die Suche etwas dauern, daher kann sie durch Schließen des Dialogs abgebrochen werden.

## 3.4 Das Menü Extras

### Extras → Dateinamenformat anwenden

Wenn **Format automatisch anwenden** im Konfigurationsdialog beim Dateinamenformat deaktiviert ist, kann man mit diesem Menüpunkt die Formatierung auf die Namen der ausgewählten Dateien anwenden. Diese Funktion kann auch angewendet werden, um zu testen, ob die Dateinamen dem eingestellten Format entsprechen. Hierzu sollten alle Dateien gespeichert werden; nach Anwendung des Formates sind dann nur diejenigen Dateien verändert (in der Dateiliste mit einem Diskettensymbol versehen), welche nicht dem eingestellten Format entsprechen. Ungewollte Änderungen können mit **Zuletzt gespeicherte Fassung** rückgängig gemacht werden.

### Extras → Tag-Format anwenden

Wenn **Format automatisch anwenden** im Konfigurationsdialog beim Tag Format deaktiviert ist, kann man mit diesem Menüpunkt die Formatierung auf die Tags der ausgewählten Dateien anwenden. Diese Funktion kann auch angewendet werden, um zu testen, ob die Tags dem eingestellten Format entsprechen. Hierzu sollten alle Dateien gespeichert werden; nach Anwendung des Formates sind dann nur diejenigen Dateien verändert (in der Dateiliste mit einem Diskettensymbol versehen), welche nicht dem eingestellten Format entsprechen. Ungewollte Änderungen können mit **Zuletzt gespeicherte Fassung** rückgängig gemacht werden.

### Extras → Textkodierung anwenden

Setzt die **Textkodierung**, welche in **Einstellungen/Kid3 einrichten.../Tags/Tag 2** eingestellt ist, für alle ausgewählten Dateien. Wenn UTF8 eingestellt ist, so wird für ID3v2.3.0 Tags UTF16 verwendet, denn dieses Format unterstützt UTF8 nicht.

### Extras → Ordner umbenennen...

Dieser Dialog bietet die Möglichkeit, automatisch den aktuellen Ordner gemäß den Tags umzubenennen. Verschiedene Formate sind vordefiniert, um Informationen über Interpret, Album und Jahr in den Ordnernamen zu übernehmen. Es ist auch möglich, ein eigenes Format zu definieren und die verfügbaren Formate zu bearbeiten (Knopf **Bearbeiten**). Die folgenden Codes können verwendet werden, um Werte aus den Tags im Ordnernamen einzusetzen:

- %s %{title} Titel (Song)
- %a %{artist} Interpret (Artist)
- %l %{album} Album
- %c %{comment} Kommentar (Comment)
- %y %{year} Jahr (Year)
- %t %{track} Nummer (Stück, z.B. 01)
- %t %{track.n} Nummer mit Feldlänge n (z.B. 001 für %{track.3})
- %T %{tracknumber} Nummer (Stück, ohne führende Null, z.B. 1)
- %g %{genre} Genre
- %{dirname} Ordnername (z.B. %{year}“ ”){dirname} fügt vorne am bestehenden Ordnernamen die Jahreszahl hinzu)
- %{max-year} Die maximale Jahreszahl, welche für diesen Ordner gefunden wurde, es können auch andere Codes als “year” verwendet werden
- %{min-year} Die minimale Jahreszahl, welche für diesen Ordner gefunden wurde
- %{unq-year} Die einzige Jahreszahl, welche für diesen Ordner gefunden wurde, leer falls nicht eindeutig



Wenn ein Ordner-Trennzeichen “/” im Format gefunden wird, werden mehrere Ordner erstellt. Will man einen neuen Ordner erstellen statt den aktuellen Ordner umbenennen, so wählt man **Ordner erstellen** statt **Ordner umbenennen**. Die Quelle der Tag-Information kann zwischen **Von Tag 1 und Tag 2**, **Von Tag 1** und **Von Tag 2** ausgewählt werden. Die voraussichtliche Ordnernamensänderung für die erste Datei kann in den Abschnitten **Von** und **Nach** des Dialogs gesehen werden.

Es können auch mehrere Ordner umbenannt werden, indem sie selektiert werden.

#### Extras → Stücke nummerieren...

Falls die Stücknummern in den Tags nicht vorhanden oder falsch sind, so lassen sich mit dieser Funktion die Stücke automatisch aufsteigend nummerieren. Im Dialog kann die **Startnummer** angegeben werden. Falls nur ein Teil der Stücke nummeriert werden soll, so sind diese im Voraus auszuwählen.

Wenn **Anzahl Stücke** markiert ist, wird auch die Anzahl Stücke in den Tags gesetzt.

Es können auch die Stücke über mehrere Ordner hinweg nummeriert werden. Dazu müssen die Ordner aufgeklappt und selektiert werden.

Wenn **Zähler für jeden Ordner zurücksetzen** aktiviert ist, wird die Nummerierung bei jedem Ordner mit der **Startnummer** begonnen falls mehrere Ordner ausgewählt sind.

Der Dialog kann auch dazu benutzt werden, bestehende Stücknummern neu zu formatieren, ohne ihren Wert zu verändern. Dazu muss das Häkchen bei der **Startnummer** entfernt werden. Wenn **Anzahl Stücke** markiert ist, so wird das Total hinzugefügt. Falls es nicht erwünscht ist, die Anzahl Stücke in allen ausgewählten Dateien zu setzen, muss auch diese Option deaktiviert werden.

#### Extras → Filter..

Der Filter kann dazu verwendet werden, nur diejenigen Dateien anzeigen, welche gewisse Eigenschaften erfüllen. Dies kann dann nützlich sein, wenn man die Tags oder Dateinamen in einer größeren Sammlung bereinigen möchte, jedoch nur diejenigen Dateien bearbeiten möchte, welche nicht dem gewünschten Schema entsprechen. Der Ausdruck, der bestimmt, welche Dateien angezeigt werden, benutzt dieselben Formatcodes, welche auch beim Dateinamenformat, Importieren und Exportieren verwendet werden.

- %s %{title} Titel (Song)
- %a %{artist} Interpret (Artist)
- %l %{album} Album
- %c %{comment} Kommentar (Comment)
- %y %{year} Jahr (Year)
- %t %{track} Nummer (Stück, z.B. 01)
- %t %{track.n} Nummer mit Feldlänge n (z.B. 001 für %{track.3})
- %T %{tracknumber} Nummer (Stück, ohne führende Null, z.B. 1)
- %g %{genre} Genre
- %f %{file} Dateiname (File name)
- %p %{filepath} Absoluter Pfadname
- %e %{extension} Dateierdung
- %O %{tag1} Das Format von Tag 1 (ID3v1.1 oder leer falls nicht vorhanden)
- %o %{tag2} Das Format von Tag 2 (ID3v2.3.0, ID3v2.4.0, ID3v2.2.0, ID3v2.2.1, Vorbis, APE, MP4, ASF oder leer falls nicht vorhanden)
- %b %{bitrate} Bitrate in kbit/s
- %v %{vbr} VBR oder leer (nur für ID3v2.3 mit id3lib)
- %r %{samplerate} Samplerate in Hz
- %m %{mode} Kanalmodus (Stereo oder Joint Stereo)
- %h %{channels} Anzahl Kanäle (1 oder 2)

- `%k %{codec}` Codec (z.B. MPEG 1 Layer 3, MP4, Ogg Vorbis, FLAC, MPC, APE, ASF, AIFF, WAV)
- `%w %{marked}` Markiert, ist 1 wenn die Datei markiert ist (z.B. weil Felder abgeschnitten wurden oder wegen einer Verletzung des ID3v2 Standards), ansonsten leer
- `%1a %1{artist}`, ... Der Präfix 1 wird benutzt, um Werte von Tag 1 zu erhalten
- `%2a %2{artist}`, ... Der Präfix 2 wird benutzt, um Werte von Tag 2 zu erhalten

Diese Codes werden mit dem Wert für die Datei ersetzt und die resultierenden Zeichenketten können mit folgenden Operatoren verglichen werden:

- `s1 equals s2`: wahr wenn s1 und s2 gleich sind.
- `s1 contains s2`: wahr wenn s2 in s1 enthalten ist.
- `s matches re`: wahr wenn s den regulären Ausdruck re erfüllt.

Wahre Ausdrücke werden durch 1 ersetzt, falsche durch 0. Wahre Werte repräsentieren nebst 1 auch true, on und yes, falsche Werte entsprechend 0, false, off und no. Diese Wahrheitwerte können durch die logischen Operatoren not, and und or verknüpft werden (die Priorität ist in dieser Reihenfolge), und mit Klammern gruppiert werden.

Einige Filterregeln sind bereits vordefiniert und können als Beispiele für eigene Ausdrücke dienen:

#### All

Wenn die Dateiliste gefiltert ist - dies wird mit "[Gefiltert]" in der Titelzeile angezeigt - und man möchte wieder alle Dateien anzeigen, kann das Filtern mit diesem Filter rückgängig gemacht werden. Es wird ein leerer Ausdruck verwendet, ein Wahrheitswert wie true hätte aber denselben Effekt.

#### Filename Tag Mismatch

```
not (%{filepath} contains ` %{artist} - %{album}/%{track} %{title} `)
```

Testet, ob ein Dateipfad dem Dateinamenformat entspricht. Diese Regel wird automatisch angepasst, wenn sich das Dateinamenformat ändert.

#### No Tag 1

```
{tag1} equals ` `
```

Es werden nur Dateien ohne Tag 1 angezeigt.

#### No Tag 2

```
{tag2} equals ` `
```

Es werden nur Dateien ohne Tag 2 angezeigt.

#### ID3v2.3.0 Tag

```
{tag2} equals ` ID3v2.3.0 `
```

Es werden nur Dateien mit ID3v2.3.0 Tag angezeigt.

#### ID3v2.4.0 Tag

```
{tag2} equals ` ID3v2.4.0 `
```

Es werden nur Dateien mit ID3v2.4.0 Tag angezeigt.

#### Tag 1 != Tag 2

```
not (%1{title} equals %2{title} and %1{album} equals %2{album} and %1{artist} equals %2{artist} and %1{comment} equals %2{comment} and %1{year} equals %2{year} and %1{track} equals %2{track} and %1{genre} equals %2{genre})
```

Es werden Dateien mit Unterschieden zwischen Tag 1 und Tag 2 angezeigt.

#### Tag 1 == Tag 2

```
{1{title} equals %2{title} and %1{album} equals %2{album} and %1{artist} equals %2{artist} and %1{comment} equals %2{comment} and %1{year} equals %2{year} and %1{track} equals %2{track} and %1{genre} equals %2{genre}}
```

Es werden Dateien mit identischen Tag 1 und Tag 2 angezeigt.

#### Incomplete

```
%{title} equals '''' or %{artist} equals '''' or %{album} equals  
'''' or %{year} equals '''' or %{tracknumber} equals ''''  
or %{genre} equals ''''
```

Es werden nur Dateien angezeigt, welche leere Standard-Tag-Werte haben (Titel, Interpret, Album, Datum, Stücknummer, Genre).

#### No Picture

```
%{picture} equals ''''
```

Es werden nur Dateien ohne Bild angezeigt.

#### Marked

```
not (%{marked} equals '''' )
```

Es werden nur Dateien angezeigt, die markiert sind weil sie den ID3v2 Standard verletzen, weil Felder abgeschnitten wurden oder ein Bild zu groß ist.

#### Custom Filter

Dieser Eintrag kann verwendet werden, um einen eigenen Filter hinzuzufügen. Um beispielsweise nur Dateien anzuzeigen, deren Interpret mit "The" beginnt, ersetzt man "Custom Filter" durch einen Namen wie "The Bands" und drückt die **Eingabe** Taste. Dann fügt man folgenden Ausdruck in das Eingabefeld ein:

```
%{artist} matches '^The.*'
```

Gespeichert wird mit **Einstellungen speichern**. Die Dateien werden mit **Anwenden** gefiltert. Alle bearbeiteten Dateien werden im Textfeld angezeigt, mit einem "+" vor solchen, die den Filterausdruck erfüllen, und einem "-" vor den anderen. Wenn das Filtern fertig ist, werden nur noch diejenigen Dateien angezeigt, deren Interpret mit "The" beginnt, und in der Titelzeile wird "[Gefiltert]" angezeigt.

#### Extras → ID3v2.3 nach ID3v2.4 konvertieren

Wenn in den ausgewählten Dateien ID3v2.3 Tags vorhanden sind, so werden sie zu ID3v2.4 Tags konvertiert. Frames, welche von TagLib nicht unterstützt sind, werden verworfen. Nur Dateien ohne ungesicherte Änderungen werden konvertiert.

#### Extras → ID3v2.4 nach ID3v2.3 konvertieren

Wenn in den ausgewählten Dateien ID3v2.4 Tags vorhanden sind, so werden sie zu ID3v2.3 Tags konvertiert. Nur Dateien ohne ungesicherte Änderungen werden konvertiert.

#### Extras → Wiedergabe

Damit öffnet sich eine einfache Werkzeuggestreife zum Abspielen von Audio-Dateien. Sie enthält Knöpfe für die grundlegenden Operationen (**Wiedergabe/Pause**, **Wiedergabe stoppen**, **Zurück**, **Vorwärts**, **Schließen**), Regler für Position und Lautstärke und eine Anzeige der aktuellen Position. Abgespielt werden die selektierten Stücke bei Mehrfachselektion bzw. alle Stücke wenn eine oder keine Datei selektiert ist.

## 3.5 Das Menü Einstellungen

#### Einstellungen → Werkzeuggestreife anzeigen

Schaltet die Anzeige der Werkzeuggestreife ein und aus.

#### Einstellungen → Statusleiste anzeigen

Schaltet die Anzeige der Statusleiste, welche längere Aktionen wie Öffnen und Speichern eines Ordners anzeigt, ein und aus.

#### Einstellungen → Bild anzeigen

Schaltet die Anzeige des Cover-Bildes ein und aus.

### Einstellungen → Tags automatisch ausblenden

Wenn diese Option aktiv ist, werden leere Tags automatisch ausgeblendet. Manuell lassen sich die **Datei**, **Tag 1** und **Tag 2** Sektionen durch Druck auf die entsprechenden - bzw. + Schaltflächen zu- und aufklappen.

### Einstellungen → Kurzbefehle festlegen...

Öffnet einen Dialog, um den meisten Programm-Funktionen ein Tastatur-Kürzel zuzuordnen. Es gibt sogar einige Funktionen ohne entsprechenden Menüpunkt oder Knopf, z.B. nächste Datei, vorangehende Datei, alles auswählen.

### Einstellungen → Kid3 einrichten...

Öffnet den Konfigurationsdialog, der aus verschiedenen Seiten besteht, mit Optionen für Tags, Dateien, Benutzerbefehle und Netzwerk.

Optionen für Tags sind auf der Seite **Tags** zu finden, welche ihrerseits in vier Reiter unterteilt ist für **Tag 1**, **Tag 2**, **Tag 3** und **Alle Tags**.

Wenn **Markiere abgeschnittene Felder** aktiv ist, werden abgeschnittene ID3v1.1 Felder rot markiert. Die Textfelder in ID3v1.1 Tags haben bloß 30 Zeichen, der Kommentar nur 28 Zeichen. Auch Genre und Nummer sind Beschränkungen unterworfen, so dass Felder gekürzt werden können, wenn sie importiert oder von ID3v2 konvertiert werden. Abgeschnittene Felder und ihre Datei werden rot markiert; die Markierung wird entfernt, nachdem das Feld editiert worden ist.

Mit **Text Codierung für ID3v1** kann die Codierung, welche für ID3v1 Tags verwendet wird, eingestellt werden. Standardmäßig wird hier ISO-8859-1 erwartet, daher wird empfohlen, diese Einstellung so zu lassen. Da jedoch Tags mit anderer Codierung anzutreffen sind, kann hier eine andere Codierung gewählt werden, so dass dann die ID3v1 Tags nach ID3v2 transferiert werden können, welches Unicode Zeichen unterstützt.

Das Markierungsfeld **Format Nummer/Anzahl Stücke benutzen** bewirkt, dass im Stücknummer-Feld bei den ID3v2 Tags nicht bloß die Stücknummer gespeichert wird, sondern auch die Gesamtzahl der Stücke im Ordner.

Wenn **Genre als Text statt als Nummer** aktiviert ist, werden alle Genres als Text gespeichert, auch wenn ein ID3v1-Code für das Genre existiert. Ohne diese Einstellungen werden solche Genres als die Nummer des entsprechenden ID3v1-Codes gespeichert (in Klammern für ID3v2.3). Das Genre Metal wird also abhängig von dieser Option entweder als "Metal" oder als "(9)" gespeichert. Genres, für welche kein ID3v1-Code existiert, werden immer als Text gespeichert. Der Sinn dieser Option ist es, die Kompatibilität mit Geräten zu verbessern, welche die Codes nicht korrekt interpretieren.

Wenn **WAV-Dateien mit id3 Block in Kleinbuchstaben** aktiviert ist, wird "id3" statt "ID3" als Name für die RIFF Chunks verwendet, in welchen ID3v2 Tags für WAV-Dateien gespeichert werden. Normalerweise akzeptieren Kid3 und andere Programme, welche TagLib verwenden, beim Lesen von WAV-Dateien sowohl die Variante mit Kleinbuchstaben als auch die Variante mit Großbuchstaben, geschrieben wird aber immer "ID3". Weil Applikationen existieren, welche nur "id3" akzeptieren (z.B. JRiver Media Center und foobar2000), können mit dieser Einstellung Tags geschrieben werden, welche auch von solchen Applikationen gelesen werden können.

Wenn **Standardverletzungen markieren** aktiviert ist, werden ID3v2 Felder, welche den Standard verletzen, rot markiert. Details über die Verletzung werden in Tooltips angezeigt:

- Darf nur einmal vorkommen
- Neue Zeile ist verboten
- Wagenrücklauf ist verboten
- Besitzer darf nicht leer sein
- Muss numerisch sein
- Muss numerisch sein oder Nummer/Anzahl
- Format ist DDMM
- Format ist HHMM

- Format ist YYYY
- Muss mit einem Jahr und einem Leerzeichen beginnen
- Muss ISO-8601 Zeitstempel sein
- Muss eine Tonart sein, 3 Zeichen, A-G, b, #, m, o
- Muss ein ISO 639-2 Sprachcode sein, 3 Kleinbuchstaben
- Muss eine ISRC Kennung sein, 12 Zeichen
- Muss eine Liste von Zeichenketten sein, getrennt durch ' | '
- Hat überschüssige Leerzeichen

Die ID3-Standard-Dokumente sind online verfügbar:

- [ID3 tag version 2.3.0](#)
- [ID3 tag version 2.4.0 - Main Structure](#)
- [ID3 tag version 2.4.0 - Native Frames](#)

**Text Codierung** bestimmt die Codierung von ID3v2 Elementen und kann auf **ISO-8859-1**, **UTF16** oder **UTF8** gesetzt werden. **UTF8** ist für ID3v2.3.0 nicht gültig; wenn dieser Wert gesetzt ist, wird stattdessen **UTF16** verwendet. Für ID3v2.4.0 sind alle drei Codierungen möglich.

**Version für neue Tags** legt fest, ob neue ID3v2 Tags mit der Version 2.3.0 oder 2.4.0 erzeugt werden.

**Anzahl der Stellen in Stücknummer** bestimmt die Zahl der Dezimalstellen in den Stücknummer-Feldern. Die Nummern werden mit führenden Nullen auf die eingestellte Anzahl Stellen erweitert. Beispielsweise wird mit dem Wert 2 die Stücknummer 5 als "05" gesetzt.

Das Auswahlfeld **Name des Kommentar-Feldes** ist für Ogg/Vorbis und FLAC-Dateien relevant und bestimmt den Namen des Feldes, welches für Kommentare verwendet wird. Hier scheinen verschiedene Anwendungen unterschiedliche Namen zu verwenden, "COMMENT" wird beispielsweise von XMMS benutzt, während Amarok "DESCRIPTION" verwendet.

Das Format von Bildern in Ogg/Vorbis-Dateien wird bestimmt durch **Name des Bild-Feldes** und kann `METADATA_BLOCK_PICTURE` oder `COVERART` sein. Das erstere ist der offizielle Standard und benutzt dasselbe Format wie Bilder in FLAC-Dateien. Bei `COVERART` handelt es sich um eine frühere, inoffizielle Methode, um Bilder in Vorbis-Tags zu codieren. Manche Player benutzen dieses Feld.

Wenn das Häkchen bei **Markieren wenn größer als** gesetzt ist, so werden Dateien mit Bildern, welche die angegebene Größe überschreiten, rot markiert. Damit können Dateien mit übergroßen Bildern, welche von gewissen Applikationen und Playern nicht akzeptiert werden, aufgespürt werden. Der vorgegebene Limit ist 131072 Bytes (128 KB).

**Eigene Genres** kann zur Definition von Genres benutzt werden, welche nicht in der Standard-Liste enthalten sind, so z.B. "Gothic Metal". Solche Genres sind dann im **Genre** Auswahlfeld von **Tag 2** verfügbar. Für ID3v1.1 Tags können nur die vordefinierten Genres verwendet werden.

Die Liste mit den eigenen Genres kann auch dazu verwendet werden, die Anzahl der im **Genre** Auswahlfeld angebotenen Einträge auf diejenigen zu reduzieren, welche man braucht. Falls die Musiksammlung hauptsächlich aus Metal, Gothic Metal, Ancient und Hard Rock Nummern besteht, so kann man diese Genres eingeben und **Nur eigene Genres anzeigen** markieren. Das **Tag 2 Genre** Auswahlfeld enthält dann nur noch diese vier Genres, und man braucht nicht mehr die ganze lange Liste nach ihnen zu durchsuchen. Bei dieser Beispielkonfiguration sind im **Tag 1 Genre** Auswahlfeld nur noch Metal und Hard Rock verfügbar, denn diese zwei Einträge sind Standard Genres. Wenn **Nur eigene Genres anzeigen** nicht markiert ist, so werden die eigenen Genres am Ende der Auswahlliste aufgeführt.

**Schnellzugriff-Elemente** bestimmt, welche Frametypen immer bei **Tag 2** angezeigt werden. Solche Frames können dann hinzugefügt werden, ohne zuerst den **Hinzufügen** Knopf

zu drücken. Die Reihenfolge dieser Schnellzugriff-Elemente kann durch Ziehen und Lassen der Elemente verändert werden.

Das Auswahlfeld **Name des Stücknummer-Feldes** ist für RIFF INFO relevant und bestimmt den Namen des Feldes, welches für Stücknummern verwendet wird. Stücknummern sind im RIFF Standard nicht spezifiziert, einige Anwendungen verwenden "ITRK", andere "IPRT".

**Tag Format** enthält Einstellungen für das Format der Tags. Wenn **Format automatisch anwenden** aktiviert ist, wird die Eingabe während dem Editieren automatisch formatiert. **Überprüfung** aktiviert die Validation in Eingabefeldern mit Stücknummer/Total- und Datum/Zeit-Werten. **Groß/Kleinschreibung** kann auf **Keine Änderungen**, **Alles Kleinschreibung**, **Alles Großbuchstaben**, **Den ersten Buchstaben groß** oder **Alle ersten Buchstaben groß** gesetzt werden. Um die Eigenheiten einer Sprache bei der Konversion zwischen Klein- und Großbuchstaben zu berücksichtigen, kann eine Sprache in der darunterliegenden Auswahlliste gewählt werden. Die **Zeichen ersetzen** Liste kann mit beliebigen Zuordnungen von Zeichenketten gefüllt werden. Eine neue Zuordnung kann hinzugefügt werden, indem in der **Von** Zelle einer Zeile der zu ersetzende Text eingegeben wird und dann in der **Nach** Spalte der Ersetzungstext. Wenn der zu ersetzende Text mit einem Schrägstrich ("/") beginnt und endet, so wird ein regulärer Ausdruck verwendet. Enthält der reguläre Ausdruck Erfassungsgruppen, so werden \1, \2, ... in **Nach** mit der entsprechend erfassten Zeichenkette ersetzt. Zum Löschen einer Zuordnung wird die **Von** Zelle auf einen leeren Wert gesetzt (z.B. indem man zuerst einen Leerschlag eingibt und diesen dann wieder löscht). Einfügen und Löschen von Zeilen ist auch über das Kontext-Menü möglich, welches erscheint, wenn man die rechte Maustaste drückt. Ersetzungen werden nur durchgeführt, wenn das **Zeichen ersetzen** Markierungsfeld aktiviert ist.

Die Tabelle in **Bewertung** enthält die Zuordnungen zwischen der Anzahl Sterne und den effektiv in den Tags gespeicherten Werten. Die Frames mit Bewertungsinformationen sind in der Zeile Bewertung der **Frame-Liste** aufgeführt. In diesen Frames kann eine Bewertung mit 1 bis 5 Sternen gesetzt werden. Verschiedene Tag-Formate und unterschiedliche Applikationen benutzen unterschiedliche Werte um die Anzahl Sterne im Tag zu speichern. Um die korrekte Anzahl Sterne darzustellen schaut Kid3 in dieser Tabelle nach, und zwar nach einem Schlüssel aus dem Frame-Namen, z.B. "RATING" für Vorbis-Tags oder "IRTD" für RIFF INFO. Für ID3v2-Tags wird als Schlüssel zur Frame-ID "POPM" noch der Wert des "Email"-Feldes hinzugefügt, getrennt durch einen Punkt. Folglich können für ID3v2 unterschiedliche Zuordnungen verwendet werden, z.B. "POPM.Windows Media Player 9 Series" für die Werte, welche beim Windows Media Player und Explorer verwendet werden, oder bloß "POPM" für POPM-Frames mit leerem "Email"-Feld. Da mehrere Einträge für "POPM" existieren können, ist deren Reihenfolge wichtig. Wenn Kid3 ein neues POPM-Frame hinzufügt, wird der erste "POPM"-Eintrag benutzt, um den Wert des "Email"-Feldes zu bestimmen. Dieser Wert entscheidet dann über die Zuordnung der Werte zur Anzahl Sterne. Der erste Eintrag wird auch benutzt, wenn kein passender Schlüssel gefunden wurde, stellt also den Standardwert dar.

Nebst der Spalte **Name** für die Schlüssel gibt es noch die Spalten **1** bis **5** für die Werte, welche gespeichert werden, wenn die entsprechende Anzahl Sterne gesetzt wird. Andererseits werden diese Werte dazu benutzt um zu bestimmen, wie viele Sterne für einen im Tag gespeicherten Wert dargestellt werden sollen. Der Eintrag in der Tabelle unten enthält beispielsweise die Werte 1, 64, 128, 196 und 255. Die Schwellwerte für die anzuzeigende Anzahl Sterne liegen dazwischen und sind so gewählt, dass sie kompatibel mit dem Windows Explorer sind.

Name	1	2	3	4	5
POPM	1	64	128	196	255
Bereich	1-31	32-95	96-159	160-223	224-255

Tabelle 3.2: Eintrag in Bewertungs-Tabelle

Auf der Seite **Dateien** bestimmt das Markierungsfeld **Zuletzt geöffnete Dateien laden**, ob Kid3 beim nächsten Start wieder die zuletzt gewählte Datei öffnet und selektiert. Mit **Datei-Zeitstempel nicht verändern** kann eingestellt werden, dass die Modifikationszeit der Datei nicht verändert wird. **Dateiname für Cover** setzt den Namen, welcher vorgeschlagen wird, wenn ein eingebettetes Bild als Datei exportiert wird. Mit **Textkodierung (Export, Stückliste)** lässt sich die Codierung einstellen, welche beim Schreiben von Dateien verwendet wird. Die Voreinstellung **System** kann verändert werden wenn beispielsweise Stücklisten auf einem anderen Gerät verwendet werden sollen.

Wenn **Änderungen markieren** aktiviert ist, werden veränderte Felder markiert, indem der Hintergrund des Namens grau gesetzt wird.

Der Abschnitt **Dateiliste** bestimmt, welche Dateien in der Dateiliste angezeigt werden. Mit einem **Filter** können die in dieser Liste angezeigten Elemente auf Dateien mit unterstützter Dateierweiterung eingeschränkt werden. Um explizit anzugeben, welche Ordner in der Dateiliste erscheinen sollen oder bestimmte Ordner zu unterdrücken, können die Felder **Ordner einschließen** und **Ordner ausschließen** verwendet werden. Sie können Dateinamen-Platzhalter enthalten, beispielsweise `*/Musik/*` um nur den `Musik`-Ordner anzuzeigen, oder `*/iTunes/*` um den `iTunes`-Ordner auszublenden. Wenn mehrere solche Ausdrücke verwendet werden, so können sie durch Leerzeichen oder Strichpunkte getrennt werden.

Die Knöpfe **Dateiname von Tag** und **Tag von Dateiname** im Abschnitt **Format** öffnen Dialoge, um die Formate zu bearbeiten, welche in den **Format** Auswahlfeldern (mit Pfeilen rauf und runter) des Abschnittes **Datei** im Hauptfenster verfügbar sind.

**Dateinamen Format** enthält Einstellungen für das Format der Dateinamen. Dabei sind die selben Optionen wie bei **Tag Format** möglich.

Die Seite **Benutzerbefehle** enthält eine Tabelle mit Befehlen, welche über das Kontext Menü der Dateiliste ausgewählt werden können. Bei kritischen Operationen, beispielsweise dem Löschen von Dateien, empfiehlt es sich, **Nachfragen** zu markieren, damit vor dem Ausführen des Befehls eine Bestätigung verlangt wird. **Ausgabe** kann markiert werden, um die Ausgabe von Konsolen-Applikationen zu sehen (Standard und Fehler-Ausgabe). **Name** ist der Text, welcher im Kontext Menü dargestellt wird. **Befehl** schließlich ist die auszuführende Kommandozeile. Argumente können mit den folgenden Codes übergeben werden:

- `%F` `{files}` Dateipfad (eine Liste wenn mehrere Dateien selektiert sind)
- `%f` `{file}` Pfad einer einzelnen Datei
- `%uF` `{urls}` URLs (eine Liste wenn mehrere Dateien selektiert sind)
- `%uf` `{url}` URL zu einer einzelnen Datei
- `%d` `{directory}` Ordner
- `%s` `{title}` Titel (Song)
- `%a` `{artist}` Interpret (Artist)
- `%l` `{album}` Album
- `%c` `{comment}` Kommentar (Comment)
- `%y` `{year}` Jahr (Year)
- `%t` `{track}` Nummer (Stück, z.B. 01)
- `%t` `{track.n}` Nummer mit Feldlänge `n` (z.B. 001 für `{track.3}`)
- `%T` `{tracknumber}` Nummer (Stück, ohne führende Null, z.B. 1)
- `%g` `{genre}` Genre
- `%b` `{browser}` Kommando zum Starten des Web Browsers
- `%q` `{qmlpath}` Basisordner der mitgelieferten QML-Dateien

Der Code **@separator** kann verwendet werden, um eine Trennlinie in das Benutzerbefehle-Kontextmenü einzufügen. Menüpunkte können in Untermenüs gruppiert werden, indem man sie mit **@beginmenu** und **@endmenu** Befehlen umgibt. Der Name des Untermenüs wird durch die Spalte **Name** des **@beginmenu** Befehls definiert.



Zum Ausführen von QML Skripten wird als Befehl `@qml` verwendet. Danach wird der Pfad zum QML-Skript angegeben. Die mitgelieferten Skripte liegen im Ordner `%{qmlpath}/script/` (unter Linux typischerweise `/usr/share/kid3/qml/script/`, bei Windows in `qml/script/` innerhalb des Installationsordners, und bei macOS im App-Ordner `kid3.app/Contents/Resources/qml/script/`). Eigene Skripte können aber auch in einem beliebigen anderen Ordner gespeichert werden. Wenn der QML-Code grafische Elemente innerhalb eines Fensters öffnet, so muss `@qmlview` anstelle von `@qml` verwendet werden. Weitere Parameter werden dem QML-Skript übergeben und sind dort über die Funktion `getArguments()` verfügbar. Eine Übersicht einiger in QML verfügbarer Funktionen und Eigenschaften findet sich im Anhang [QML Schnittstelle](#).

Das Kommando, welches mit `%{browser}` eingefügt werden kann, wird oberhalb der Tabelle im Eingabefeld **Web Browser** definiert. Damit lassen sich leicht Webseiten mit Informationen zu den Audiodateien aufrufen. Zum Beispiel startet

```
%{browser} http://lyricwiki.org/%u(artist):%u(title)
```

eine Anfrage in [LyricWiki](#) mit Interpret und Titel. Das `“u“` in `%u(artist)` und `%u(title)` bewirkt, dass Interpret `%{artist}` und Titel `%{title}` URL-codiert werden. In gleicher Weise lassen sich auch leicht andere Abfragen definieren, z.B. eine Bildsuche bei [Google](#):

```
%{browser} http://images.google.com/images?q=%u(artist)%20%u(album)
```

Um dem Tag 2 Album Cover Bilder hinzuzufügen, kann man mit den oben beschriebenen Benutzerbefehlen über Google Images oder Amazon nach Bildern suchen. Die Bilder können mit `“Drag’n’Drop“` hinzugefügt werden. Bilder können auch mit **Hinzufügen** eines Bild Elementes und Importieren einer Datei oder Einfügen aus der Zwischenablage in ein Tag gebracht werden. Unterstützt sind Bilder für ID3v2, MP4, FLAC, Ogg und ASF Tags.

Zum Löschen und Hinzufügen von Einträgen steht in der Tabelle ein Kontext Menü zur Verfügung.

Die **Netzwerk** Seite enthält bloß ein Textfeld für die Adresse des Netzwerk Proxys und optional der Portnummer, abgetrennt durch einen Doppelpunkt. Dieser Proxy wird für den Import von einem Internet Server verwendet, wenn das Kästchen davor markiert ist.

Die verfügbaren Plugins können in der Seite **Erweiterungen** ein- bzw. ausgeschaltet werden. Die Plugins sind in zwei Listen unterteilt. **Metadaten Plugins & Priorität** enthält Plugins, welche Audio-Datei-Formate unterstützen. Die Reihenfolge der Einträge ist entscheidend, denn sie werden der Reihe nach durchprobiert. Gewisse Formate werden von mehreren Plugins unterstützt, daher werden Dateien durch das erste Plugin, welches sie unterstützt, geöffnet. **TaglibMetadaten** unterstützt die meisten Formate; wenn es zuoberst in der Liste ist, wird es die meisten Dateien öffnen. Um ein anderes Plugin zu verwenden, muss es vor **TaglibMetadaten** in der Liste stehen. Details über die Metadaten Plugins und Gründe, warum man sie statt TagLib verwenden möchte, folgen.

- **Id3libMetadaten**: Benutzt [id3lib](#) für ID3v1.1 und ID3v2.3 Tags in MP3, MP2 und AAC Dateien; unterstützt einige Frame-Typen mehr als TagLib.
- **OggFlacMetadaten**: Benutzt [libogg](#), [libvorbis](#) und [libvorbisfile](#) für Ogg Dateien und zusätzlich [libFLAC++](#) und [libFLAC](#) für FLAC Dateien. Dies sind die offiziellen Bibliotheken dieser Formate.
- **TaglibMetadaten**: Benutzt [TagLib](#), welche viele Datei-Formate unterstützt und für alle Audio-Dateien verwendet werden kann, welche Kid3 unterstützt.
- **Mp4v2Metadaten**: [mp4v2](#) wurde ursprünglich von Kid3 für M4A-Dateien verwendet. Kann bei Problemen mit der M4A-Unterstützung von TagLib verwendet werden.

**Verfügbare Erweiterungen** listet die verbleibenden Plugins auf. Ihre Reihenfolge ist unwichtig, sie können über die Kästchen links ein- oder ausgeschaltet werden.

- **AmazonImport**: Benutzt für die **Importieren von Amazon...** Funktion.
- **DiscogsImport**: Benutzt für die **Importieren von Discogs...** Funktion.



- **FreedbImport:** Benutzt für die Funktion **Importieren von gnudb.org**.
- **MusicBrainzImport:** Benutzt für die **Importieren von MusicBrainz-Album...** Funktion.
- **AcoustidImport:** Benutzt für die **Importieren von MusicBrainz-Fingerprint...** Funktion, welche von den Bibliotheken [Chromaprint](#) und [libav](#) abhängig ist.

Deaktivierte Erweiterungen werden nicht geladen, damit lässt sich also der Speicherverbrauch und die Ladezeit optimieren. Die Einstellungen auf dieser Seite werden aber erst nach einem Neustart von Kid3 aktiv.

## 3.6 Das Menü Hilfe

**Hilfe** → **Handbuch zu Kid3**

Öffnet dieses Handbuch.

**Hilfe** → **Über Kid3**

Zeigt eine kurze Information über Kid3 an.

## Kapitel 4

# kid3-cli

### 4.1 Befehle

**kid3-cli** bietet eine Kommandozeile für Kid3. Wenn ein Ordnerpfad als Parameter verwendet wird, so wird der Ordner geöffnet. Bei einem oder mehreren Dateipfaden wird der übergeordnete Ordner geöffnet, und die Dateien werden ausgewählt. Befehle werden dann auf diese Dateien angewendet. Befehle werden mit `-c` Optionen angegeben, wenn mehrere vorhanden sind, so werden sie der Reihe nach ausgeführt. Falls Dateien durch die Befehle verändert werden, so werden sie am Ende gespeichert. Wenn keine Befehlsoptionen vorhanden sind, so startet **kid3-cli** im interaktiven Modus. Dort können Befehle eingegeben werden, welche auf die ausgewählten Dateien angewendet werden. Die folgenden Abschnitte beschreiben alle verfügbaren Befehle.

#### 4.1.1 Hilfe

`help [BEFEHLSNAME]`

Zeigt Hilfe über die Parameter von `BEFEHLSNAME` oder über alle Befehle falls kein Befehlsname angegeben wird.

#### 4.1.2 Zeitüberschreitung

`timeout [default | off | ZEIT]`

Überschreibt den Wert für die maximal erlaubte Ausführungszeit der Befehle. Die Befehle werden nach einer spezifischen Zeit abgebrochen, nach 10 Sekunden bei `ls` und `albumart`, 60 Sekunden bei `autoimport` und `filter` und 3 Sekunden für alle anderen Befehle. Wenn eine große Anzahl von Dateien bearbeitet wird, so können diese Maximalzeiten zu restriktiv sein. Daher kann mit diesem Befehl der Timeout für alle Befehle auf `ZEIT` ms gesetzt werden, gänzlich abgeschaltet werden oder auf den Voreinstellungen belassen werden.

#### 4.1.3 Anwendung beenden

`exit [force]`

Beendet die Anwendung. Falls veränderte Dateien existieren, so muss der Parameter `force` mitgegeben werden.

#### 4.1.4 Ordner wechseln

```
cd [ORDNER]
```

Wenn kein *ordner* angegeben wird, so wird in den Benutzerordner gewechselt, sonst in den angegebenen Ordner. Bei einem oder mehreren Dateipfaden wird der übergeordnete Ordner geöffnet, und die Dateien werden ausgewählt.

#### 4.1.5 Aktuellen Ordner ausgeben

```
pwd
```

Gibt den Pfad des aktuellen Ordners aus.

#### 4.1.6 Ordnerliste

```
ls
```

Zeigt den Inhalt des aktuellen Ordners an. Dies entspricht der Dateiliste im Kid3 GUI. Fünf Zeichen auf der linken Seite des Dateinamens zeigen Zustandsinformationen an.

- > Datei ist ausgewählt.
- \* Datei ist verändert.
- 1 Datei hat ein Tag 1, sonst wird '-' angezeigt.
- 2 Datei hat ein Tag 2, sonst wird '-' angezeigt.
- 3 Datei hat ein Tag 3, sonst wird '-' angezeigt.

```
kid3-cli> ls
 1-- 01 Intro.mp3
> 12- 02 We Only Got This One.mp3
*1-- 03 Outro.mp3
```

In diesem Beispiel haben alle Dateien ein Tag 1, die zweite Datei hat auch ein Tag 2 und ist ausgewählt. Die dritte Datei ist verändert.

#### 4.1.7 Veränderte Dateien speichern

```
save
```

#### 4.1.8 Datei wählen

```
select [all | none | first | previous | next | DATEI]
```

Alle Dateien werden mit **select all** ausgewählt, mit **select none** wird die Selektion aufgehoben. Um die Dateien des aktuellen Ordners zu traversieren, wird mit **select first** gestartet, vorwärts geht es mit **select next** und rückwärts mit **select previous**. Dateien können über ihren Namen zur Auswahl hinzugefügt werden. Wildcards sind möglich, so werden mit **select \*.mp3** alle MP3-Dateien im aktuellen Ordner ausgewählt.

```
kid3-cli> select first
kid3-cli> ls
> 1-- 01 Intro.mp3
   12- 02 We Only Got This One.mp3
  *1-- 03 Outro.mp3
kid3-cli> select next
kid3-cli> ls
   1-- 01 Intro.mp3
> 12- 02 We Only Got This One.mp3
  *1-- 03 Outro.mp3
kid3-cli> select *.mp3
kid3-cli> ls
> 1-- 01 Intro.mp3
> 12- 02 We Only Got This One.mp3
>*1-- 03 Outro.mp3
```

### 4.1.9 Tag auswählen

tag [TAGNUMMERN]

Viele Befehle haben einen optionalen *TAGNUMMERN* Parameter, welcher festlegt, ob der Befehl auf Tag 1, 2 oder 3 wirkt. Wenn dieser Parameter weggelassen wird, so werden die standardmäßigen Tagnummern verwendet, welche mit dem Befehl **tag** festgelegt werden können. Voreingestellt ist ein Wert von 12, was bedeutet, dass Informationen vom Tag 2 geholt werden, falls es vorhanden ist, sonst vom Tag 1. Änderungen werden am Tag 2 durchgeführt. Die *TAGNUMMERN* können auf **1**, **2** oder **3** gesetzt werden, damit nur das entsprechende Tag verwendet wird. Wird kein Parameter angegeben, so wird die momentane Einstellung angezeigt.

### 4.1.10 Tag-Element holen

get [all | ELEMENTNAME] [TAGNUMMERN]

Dieser Befehl kann dazu benutzt werden, den Wert eines bestimmten Tag-Elements zu lesen oder um Informationen über alle Tag-Elemente zu holen (wenn kein Parameter vorhanden ist oder **all** verwendet wird). Veränderte Elemente werden mit einem '\*' markiert.

```
kid3-cli> get
Datei: MPEG 1 Layer 3 192 kbps 44100 Hz Joint Stereo
  Name: 01 Intro.mp3
Tag 1: ID3v1.1
  Titel      Intro
  Interpret  One Hit Wonder
  Album      Let's Tag
  Datum      2013
  Stücknummer 1
  Genre      Pop
kid3-cli> get title
Intro
```

Ein Bild kann in einer Datei gespeichert werden.

```
get picture:'/pfad/zu/folder.jpg'
```

Synchronisierter Liedtext wird im LRC-Format exportiert.

```
get SYLT:'/pfad/zu/lyrics.lrc'
```

Es können auch nur bestimmte Felder abgefragt werden, beispielsweise kann das Email-Feld eines Popularimeter-Frames mit `get POPM.Email` ausgegeben werden. Wenn eine Datei mehrere Frames des gleichen Typs enthält, so kann der Index in eckigen Klammern angegeben werden. So kriegt man beispielsweise den ersten Musiker aus einem Vorbis-Tag mit `get performer[0]`, den zweiten mit `get performer[1]`.

Der Pseudo-Feldname "selected" kann verwendet werden um zu prüfen, ob ein Frame selektiert ist, z.B. liefert `get artist.selected` dann 1 zurück, wenn der Interpret ausgewählt ist, sonst 0.

#### 4.1.11 Tag-Element setzen

```
set ELEMENTNAME ELEMENTWERT [TAGNUMMERN]
```

Dieser Befehl setzt den Wert eines bestimmten Tag-Elements. Wenn `ELEMENTWERT` leer ist, so wird das Element gelöscht.

```
kid3-cli> set remixer 'O.H. Wonder'
```

Um den Inhalt eines Bild-Elements aus einer Datei zu befüllen, verwendet man

```
set picture:'/pfad/zu/folder.jpg' 'Bildbeschreibung'
```

Synchronisierter Liedtext kann aus einer LRC-Datei importiert werden.

```
set SYLT:'/pfad/zu/lyrics.lrc' 'Beschreibung'
```

Um ein bestimmtes Feld eines Frames zu setzen, kann der Feldname nach einem Punkt angegeben werden, z.B. für das Zähler-Feld des Popularimeter-Frames.

```
set POPM.Counter 5
```

Eine Anwendung für das Setzen spezifischer Felder ist der Fall, dass man ein TXXX-Frame mit "rating" als Beschreibung möchte anstelle eines POPM Frames (dies scheint für die Anwendung gewisser Plugins nötig zu sein). Man kann ein solches TXXX-Rating-Frame mit `kid3-cli` erstellen, muss dazu jedoch zuerst das Frame mit der Beschreibung erstellen und dann den Wert setzen.

```
kid3-cli> set rating ""
kid3-cli> set TXXX.Description rating
kid3-cli> set rating 5
```

Der erste Befehl löscht ein allenfalls vorhandenes POPM frame, denn wenn ein solches existiert, würde `set rating 5` das POPM-Frame setzen und nicht das TXXX-Frame. Eine andere Möglichkeit wäre `set TXXX.Text 5`, aber dies funktioniert nur, wenn kein anderes TXXX-Frame vorhanden ist.

Um mehrere Frames desselben Typs zu setzen, kann der Index in eckigen Klammern angegeben werden, z.B. mehrere Musiker in einem Vorbis-Tag.

```
kid3-cli> set performer[0] 'Liza don Getti (soprano)'
kid3-cli> set performer[1] 'Joe Barr (piano)'
```

Um nur bestimmte Frames vor dem Kopieren, Einfügen oder Entfernen auszuwählen, kann der Pseudo-Feldname "selected" verwendet werden. Normalerweise sind alle Frames ausgewählt, um alle zu deselektieren, kann `set '*.selected' 0` verwendet werden, danach kann beispielsweise das Interpreten-Frame mit `set artist.selected 1` ausgewählt werden.

### 4.1.12 Änderungen rückgängig machen

revert

Macht alle Änderungen an den ausgewählten Dateien rückgängig (oder an allen Dateien, falls keine selektiert sind).

### 4.1.13 Von Datei importieren

import *DATEI* *FORMATNAME* [*TAGNUMMERN*]

Die Tags werden aus *DATEI* importiert im Format *FORMATNAME* (z.B. `''CSV unquoted''`, siehe [Importieren](#)).

Falls für *DATEI* **tags** angegeben wird, so wird aus anderen Tags importiert. Anstelle von *FORMATNAME* werden dann *QUELLE* und *EXTRAKTION* als Parameter verlangt, siehe [Importieren von Tags](#). Um das Importieren aus Tags auf die selektierten Dateien anzuwenden, kann **tagsel** statt **tags** verwendet werden. Hiermit kann auch mit einer *EXTRAKTION* von `%{__return}(.+)` der extrahierte Wert ausgegeben werden.

### 4.1.14 Automatisch importieren

autoimport [*PROFILNAME*] [*TAGNUMMERN*]

Es wird automatisch importiert unter Verwendung des Profils *PROFILNAME* (siehe [Automatisch importieren](#), `''All''` wird benutzt, falls kein Profil angegeben wird).

### 4.1.15 Cover-Bilder herunterladen

albumart *URL* [all]

Setzt das Albumcover mit einem von *URL* heruntergeladenen Bild. Die Regeln zur Umwandlung einer allgemeinen URL (z.B. von Amazon) zu einer Bild-URL werden verwendet, diese können im [Nach Cover-Bildern suchen](#) Dialog definiert werden. Um das Albumcover mit einer lokalen Datei zu setzen kann der `set` Befehl verwendet werden.

```
kid3-cli> albumart
http://www.amazon.com/Versus-World-Amon-Amarth/dp/B000078DOC
```

### 4.1.16 In Datei exportieren

export *DATEI* *FORMATNAME* [*TAGNUMMERN*]

Tags werden in die Datei *DATEI* exportiert im Format *FORMATNAME* (z.B. `''CSV unquoted''`, siehe [Exportieren](#)).

### 4.1.17 Stückliste erstellen

playlist

Erstellt eine Stückliste in dem in der Konfiguration gesetzten Format, siehe [Stückliste erstellen](#).

### 4.1.18 Dateinamenformat anwenden

filenameformat

Wendet das Dateinamenformat an, welches in der Konfiguration gesetzt ist, siehe [Dateinamenformat anwenden](#).

### 4.1.19 Tag-Format anwenden

tagformat

Wendet das Tag-Format an, welches in der Konfiguration gesetzt ist, siehe [Tag-Format anwenden](#).

### 4.1.20 Textkodierung anwenden

textencoding

Wendet die Textkodierung an, welche in der Konfiguration gesetzt ist, siehe [Textkodierung anwenden](#).

### 4.1.21 Ordner umbenennen

renamedir [*FORMAT*] [create | rename | dryrun] [*TAGNUMMERN*]

Mit diesem Befehl werden Ordner umbenannt oder neu erstellt anhand der Werte in den Tags nach einem bestimmten *FORMAT* (z.B. `%{artist} - %{album}`), siehe [Ordner umbenennen](#), wenn kein Format angegeben wird, so wird dasjenige verwendet, welches im **Ordner umbenennen** Dialog gesetzt ist. Normalerweise wird der Modus `rename` verwendet; um neue Ordner zu erstellen, muss `create` explizit angegeben werden. Die zum Umbenennen nötigen Aktionen werden sofort ausgeführt, um bloß eine Vorschau über diese Aktionen zu sehen, kann die `dryrun` Option verwendet werden.

### 4.1.22 Stücke nummerieren

numbertracks [*TRACKNUMMER*] [*TAGNUMMERN*]

Nummeriert die ausgewählten Dateien beginnend mit *TRACKNUMMER* (1 wenn nichts angegeben wird).

### 4.1.23 Filter

filter [*FILTERNAME* | *FILTERFORMAT*]

Filtert die Dateien, so dass nur Dateien angezeigt werden, welche dem *FILTERFORMAT* entsprechen. Der Name eines vordefinierten Filterausdrucks (z.B. `''Filename Tag Mismatch''`) kann anstelle eines Filterausdrucks verwendet werden, siehe [Filter](#).

```
kid3-cli> filter '%{title} contains "tro"'
Begonnen
 /home/urs/One Hit Wonder - Let's Tag
+ 01 Intro.mp3
- 02 We Only Got This One.mp3
+ 03 Outro.mp3
Fertig
kid3-cli> ls
 1-- 01 Intro.mp3
 1-- 03 Outro.mp3
kid3-cli> filter All
Begonnen
 /home/urs/One Hit Wonder - Let's Tag
+ 01 Intro.mp3
+ 02 We Only Got This One.mp3
+ 03 Outro.mp3
Fertig
```

```
kid3-cli> ls
1-- 01 Intro.mp3
12- 02 We Only Got This One.mp3
1-- 03 Outro.mp3
```

#### 4.1.24 ID3v2.3 nach ID3v2.4 konvertieren

to24

#### 4.1.25 ID3v2.4 nach ID3v2.3 konvertieren

to23

#### 4.1.26 Dateiname von Tag

fromtag [FORMAT] [TAGNUMMERN]

Setzt die Dateinamen der ausgewählten Dateien anhand der Werte in den Tags, z.B. **fromtag** '%{track} - %{title}' 1. Wenn kein Format angegeben wird, wird dasjenige verwendet, welches im GUI gesetzt ist.

#### 4.1.27 Tag von Dateiname

totag [FORMAT] [TAGNUMMERN]

Setzt die Tag-Elemente anhand der Dateinamen, z.B. **totag** '%{albumartist} - %{album }/%{track} %{title}' 2. Wenn kein Format angegeben wird, wird dasjenige verwendet, welches im GUI gesetzt ist. Falls der Dateiname nicht diesem Format entspricht, werden noch einige andere, gängige Formate ausprobiert.

#### 4.1.28 Tag zu anderem Tag

syncto TAGNUMMER

Kopiert die Tag-Elemente von einem Tag zum anderen Tag, um beispielsweise das ID3v2 Tag anhand der Werte im ID3v1 Tag zu setzen, wird **syncto** 2 verwendet.

#### 4.1.29 Kopieren

copy [TAGNUMMER]

Kopiert die Tag-Elemente der selektierten Datei in die interne Ablage. Sie können dann in einer anderen Datei mit dem **paste** Befehl gesetzt werden.

Um nur ein paar ausgewählte Frames zu kopieren, kann der **set** Befehl mit dem Pseudo-Feldnamen "selected" verwendet werden. Um beispielsweise nur die CD-Nummer- und Copyright-Frames zu kopieren, verwende man

```
set '*.selected' 0
set discnumber.selected 1
set copyright.selected 1
copy
```



### 4.1.30 Einfügen

paste [*TAGNUMMER*]

Setzt die Tag-Elemente gemäß dem Inhalt der **copy** Ablage in den ausgewählten Dateien.

### 4.1.31 Entfernen

remove [*TAGNUMMER*]

Entfernt ein Tag.

Es ist möglich, nur bestimmte Frames zu entfernen, indem man sie auswählt wie beim **copy** Befehl beschrieben.

### 4.1.32 Kid3 einrichten

config [*OPTION*] [*WERT*]

Konfigurationseinstellung lesen oder setzen

Die *OPTION* besteht aus einem Gruppennamen und einem Eigenschaftsnamen, getrennt durch einen Punkt. Wenn keine *OPTION* mitgegeben wird, so werden alle verfügbaren Gruppennamen aufgelistet. Wenn nur ein Gruppenname angegeben wird, so werden die Eigenschaften der Gruppe angezeigt. Für eine Gruppe mit Eigenschaft wird der aktuelle Wert ausgegeben. Um einen neuen Wert zu setzen, kann dieser als zweites Argument angegeben werden.

Wenn der Wert für eine Einstellung aus einer Liste besteht, so müssen alle Listenelemente als Argumente angegeben werden. Um also ein neues Element zu einer bestehenden Liste hinzuzufügen, müssen alle existierenden Elemente angegeben werden, gefolgt vom neuen Element. In solch einem Fall ist es einfacher, den JSON-Modus zu verwenden, denn dort kann die bestehende Liste einfach kopiert und durch das neue Element ergänzt werden.

## 4.2 Beispiele

Setzt Titel mit Apostroph. Befehle, welche **kid3-cli** mit `-c` übergeben werden, müssen in Anführungszeichen stehen, falls sie nicht bloß aus einem einzelnen Wort bestehen. Wenn solch ein Befehl selbst ein Argument mit Leerzeichen hat, muss dieses Argument ebenfalls in Anführungszeichen gesetzt werden. In UNIX<sup>®</sup> Shells können einfache oder doppelte Anführungszeichen verwendet werden, aber in der Windows Eingabeaufforderung ist es wichtig, dass außen doppelte und innen einfache Anführungszeichen verwendet werden. Wenn im Text innerhalb der einfachen Anführungszeichen ein einfaches Anführungszeichen verwendet wird, so muss diesem ein umgekehrter Schrägstrich vorangestellt werden, wie im folgenden Beispiel gezeigt wird:

```
kid3-cli -c "set title 'I\'ll be there for you'" /path/to/dir
```

Setzt das Albumcover in allen Dateien eines Ordners unter Verwendung der automatischen Import-Funktion:

```
kid3-cli -c "autoimport 'Cover Art'" /pfad/zum/ordner
```

Kommentar entfernen und Tag-Format für beide Tags in allen MP3-Dateien eines Ordners anwenden:

```
kid3-cli -c "set comment '' 1" -c "set comment '' 2" \  
-c "tagformat 1" -c "tagformat 2" /pfad/zum/ordner/*.mp3
```

Automatischer Import in Tag 2, auch ins Tag 1 kopieren, die Dateinamen aus Tag 2 setzen und schließlich eine Stückliste erstellen:

```
kid3-cli -c autoimport -c "syncto 1" -c fromtag -c playlist \  
/pfad/zum/ordner/*.mp3
```

Für alle Dateien mit einem ID3v2.4.0 Tag, dieses nach ID3v2.3.0 konvertieren und das Arrangierung Element entfernen:

```
kid3-cli -c "filter 'ID3v2.4.0 Tag'" -c "select all" -c to23 \  
-c "set arranger ''" /pfad/zum/ordner
```

Dieses Python-Skript erzeugt mit **kid3-cli** iTunes Sound Check iTunNORM Frames aus Replay Gain Information.

```
#!/usr/bin/env python3  
# Generate iTunes Sound Check from ReplayGain.  
import os, sys, subprocess  
  
def rg2sc(dirpath):  
    for root, dirs, files in os.walk(dirpath):  
        for name in files:  
            if name.endswith(('.mp3', '.m4a', '.aiff', '.aif')):  
                fn = os.path.join(root, name)  
                rg = subprocess.check_output([  
                    'kid3-cli', '-c', 'get "replaygain_track_gain"',  
                    fn]).strip()  
                if rg.endswith(b' dB'):  
                    rg = rg[:-3]  
                try:  
                    rg = float(rg)  
                except ValueError:  
                    print('Value %s of %s in not a float' % (rg, fn))  
                    continue  
                sc = (' ' + ('%08X' % int((10 ** (-rg / 10)) * 1000)) * 10  
                    subprocess.call([  
                        'kid3-cli', '-c', 'set iTunNORM "%s"' % sc, fn])  
  
if __name__ == '__main__':  
    rg2sc(sys.argv[1])
```

### 4.3 JSON Format

Um die Antworten von **kid3-cli** einfacher von Programmen zu verarbeiten, können sie im JSON-Format ausgegeben werden. Wenn das Kommando im JSON-Format daherkommt, so wird auch die Antwort in JSON geliefert. Wird im Kommando eine kompakte JSON-Repräsentation verwendet, so wird auch das Resultat kompakt. Wenn das Kommando ein `id`-Feld enthält, so wird es als JSON-RPC interpretiert und die Antwort enthält ein `jsonrpc`-Feld und die `id` der Anfrage. Dieselben Kommandos wie im Standard-CLI werden verwendet, der Befehl ist im `method`-Feld und die Parameter (sofern es solche hat) in der `params`-Liste. Die Antwort enthält ein `result`-Objekt, welches auch null sein kann, falls der entsprechende **kid3-cli**-Befehl kein Resultat ausgibt. Im Falle eines Fehlers wird ein `error`-Objekt zurückgegeben, welches `code` und `message` wie in JSON-RPC enthält.

```
kid3-cli> {"method":"set","params":["artist","An Artist"]}  
{"result":null}
```

## Das Kid3 Handbuch

```
kid3-cli> {"method":"get","params":["artist",2]}
{"result":"An Artist"}
kid3-cli> {"method": "get", "params": ["artist"]}
{
  "result": "An Artist"
}

kid3-cli> {"jsonrpc":"2.0","id":"123","method":"get","params":["artist"]}
{"id":"123","jsonrpc":"2.0","result":"An Artist"}
```

## Kapitel 5

# Danksagungen und Lizenz

Kid3

Programmiert von Urs Fleisch [ufleisch@users.sourceforge.net](mailto:ufleisch@users.sourceforge.net)

Übersetzung von Urs Fleisch [ufleisch@users.sourceforge.net](mailto:ufleisch@users.sourceforge.net)

Diese Dokumentation ist unter den Bedingungen der [GNU Free Documentation License](#) veröffentlicht.

Dieses Programm ist unter den Bedingungen der [GNU General Public License](#) veröffentlicht.

## Anhang A

# Installation

### A.1 Bezugsquellen

Kid3 ist zu finden unter <https://kid3.kde.org>.

### A.2 Anforderungen

Kid3 braucht [Qt](#). [KDE](#) ist empfohlen aber nicht notwendig, denn Kid3 kann auch als reine Qt Applikation kompiliert werden. Kid3 läuft auf Systemen, wo diese Programmbibliotheken vorhanden sind, z.B. auf GNU/Linux, Windows und macOS. Um Ogg/Vorbis Dateien zu taggen werden [libogg](#), [libvorbis](#) und [libvorbisfile](#) benötigt, für FLAC Dateien [libFLAC++](#) und [libFLAC](#). Für MP3 Dateien wird die [id3lib](#) verwendet. Für diese vier Formate kann auch die [TagLib](#) verwendet werden, welche zudem auch Opus, MPC, APE, MP2, Speex, TrueAudio, WavPack, WMA, WAV, AIFF Dateien und Tracker Module unterstützt. Für das Importieren über akustische Fingerabdrücke werden [Chromaprint](#) und [libav](#) verwendet.

Kid3 ist für die meisten Linux-Distributionen, Windows und macOS erhältlich. Links finden sich auf <https://kid3.kde.org>.

### A.3 Kompilieren und Installation

Man kann Kid3 mit oder ohne KDE kompilieren. Ohne KDE ist Kid3 eine reine Qt Applikation und es fehlen einige Konfigurations- und Session-Funktionen.

Für die KDE Version wird im obersten Ordner getippt:

```
% cmake .  
% make  
% make install
```

Um für andere Versionen von Qt oder KDE zu kompilieren, setzt man die entsprechenden **cmake** Optionen.

Falls nicht alle Programmbibliotheken vorhanden sind, so wird Kid3 mit reduzierter Funktionalität kompiliert. Es sollte also darauf geachtet werden, dass die Entwicklungspakete aller gewünschten Bibliotheken auch installiert sind. Andererseits kann mit **cmake**-Optionen bestimmt werden, was verwendet werden soll. Die Voreinstellung ist **-DWITH\_TAGLIB:BOOL=ON -DWITH\_MP4V2:BOOL=OFF -DWITH\_ID3LIB:BOOL=ON -DWITH\_CHROMAPRINT:BOOL=ON**

`-DWITH_VORBIS:BOOL=ON -DWITH_FLAC:BOOL=ON` . Entsprechend kann auch `OFF` gesetzt werden.

Um Kid3 als Qt Applikation ohne KDE zu kompilieren, wird die `cmake` Option `-DWITH_APPS=Qt` benutzt. Um sowohl eine KDE als auch eine Qt Applikation zu kompilieren, wird `-DWITH_APPS=Qt;KDE` gesetzt.

Um eine bestimmte Qt-Installation zu nutzen, kann `-DQT_QMAKE_EXECUTABLE=/Pfad/zu/qmake` gesetzt werden.

Für die Erzeugung von RPM-Paketen ist die Datei `kid3.spec` vorhanden, für Debian<sup>®</sup>-Pakete steht das Skript `build-deb.sh` zur Verfügung.

Die Qt Anwendung kann auch für Windows und macOS kompiliert werden. Das Skript `buildlibs.sh` kann verwendet werden, um alle benötigten Programmbibliotheken herunterzuladen und zu kompilieren und ein Kid3-Paket zu erstellen.

## A.4 Konfiguration

Unter KDE werden die Einstellungen in `.config/kid3rc` gespeichert. Als Qt-Anwendung liegt diese Datei in `.config/Kid3/Kid3.conf`. Unter Windows wird die Konfiguration in der Registry gespeichert, unter macOS in einer `plist`-Datei.

Die Umgebungsvariable `KID3_CONFIG_FILE` kann gesetzt werden, um den Pfad der Konfigurationsdatei zu bestimmen.

## Anhang B

# D-Bus-Schnittstelle

### B.1 D-Bus-Beispiele

Unter Linux steht eine D-Bus Schnittstelle zur Verfügung, mit welcher Kid3 über Skripte gesteuert werden kann. Skripte können in jeder Sprache, für die D-Bus-Anbindungen existieren, geschrieben werden (z.B. in Python) und können über die **Benutzerbefehle** die Funktionalität von Kid3 erweitern.

Mit folgenden Befehlen kann der Interpret im Tag 2 auf den Wert "One Hit Wonder" gesetzt werden:

#### Shell

```
dbus-send --dest=org.kde.kid3 --print-reply=literal \  
/Kid3 org.kde.Kid3.setFrame int32:2 string:'Artist' \  
string:'One Hit Wonder'
```

oder einfacher mit **qdbus** von Qt (**qdbusviewer** kann benutzt werden, um die D-Bus Schnittstelle über eine grafische Benutzeroberfläche zu erkunden):

```
qdbus org.kde.kid3 /Kid3 setFrame 2 Artist \  
'One Hit Wonder'
```

#### Python

```
import dbus  
kid3 = dbus.SessionBus().get_object(  
    'org.kde.kid3', '/Kid3')  
kid3.setFrame(2, 'Artist', 'One Hit Wonder')
```

#### Perl

```
use Net::DBus;  
$kid3 = Net::DBus->session->get_service(  
    "org.kde.kid3")->get_object(  
    "/Kid3", "org.kde.Kid3");  
$kid3->setFrame(2, "Artist", "One Hit Wonder");
```

## B.2 D-Bus API

Das D-Bus API ist in `org.kde.Kid3.xml` spezifiziert. Die Kid3 Schnittstelle hat die folgenden Methoden:

### B.2.1 Datei oder Ordner öffnen

boolean **openDirectory**(string path);

*path*

Pfad zu Datei oder Ordner

Wenn OK wird true zurückgegeben.

### B.2.2 Tags aller nicht selektierten oder modifizierten Dateien entladen

**unloadAllTags**(void);

### B.2.3 Alle veränderten Dateien speichern

boolean **save**(void);

Wenn OK wird true zurückgegeben.

### B.2.4 Detaillierte Fehlermeldung für einige Methoden

string **getErrorMessage**(void);

Gibt eine detaillierte Fehlermeldung zurück.

### B.2.5 Änderungen an Dateien rückgängig machen

**revert**(void);

### B.2.6 Automatischen Import starten

boolean **batchImport**(int32 tagMask, string profileName);

*tagMask*

Tag Maske (Bit 0 für Tag 1, Bit 1 für Tag 2)

*profileName*

Name des zu verwendenden Profils



## B.2.7 Tags von Datei importieren

boolean **importFromFile**(int32 tagMask, string path, int32 fmtIdx);

**tagMask**

Tag Bit (1 für Tag 1, 2 für Tag 2)

**path**

Pfad zu Datei

**fmtIdx**

Format Index

Wenn OK wird true zurückgegeben.

## B.2.8 Tags aus anderen Tags importieren

**importFromTags**(int32 tagMask, string source, string extraction);

**tagMask**

Tag Bit (1 für Tag 1, 2 für Tag 2)

**source**

Format-Spezifikation, um Ausgangstext aus den vorhandenen Tags zu erhalten

**extraction**

Regulärer Ausdruck mit Elementnamen und Klammergruppen, um Treffer aus dem Ausgangstext zu extrahieren

## B.2.9 Tags aus anderen Tags importieren in ausgewählten Dateien

array **importFromTagsToSelection**(int32 tagMask, string source, string extraction);

**tagMask**

Tag Bit (1 für Tag 1, 2 für Tag 2)

**source**

Format-Spezifikation, um Ausgangstext aus den vorhandenen Tags zu erhalten

**extraction**

Regulärer Ausdruck mit Elementnamen und Klammergruppen, um Treffer aus dem Ausgangstext zu extrahieren

**returnValues**

Extrahierter Wert für `“%{__return}(,+)”`

## B.2.10 Cover-Bild herunterladen

**downloadAlbumArt**(string url, boolean allFilesInDir);

**url**

URL des Bildes oder einer dazugehörigen Website

**allFilesInDir**

falls true, wird das Bild in allen Dateien des Ordners hinzugefügt

### B.2.11 Tags in Datei exportieren

boolean **exportToFile**(int32 tagMask, string path, int32 fmtIdx);

**tagMask**

Tag Bit (1 für Tag 1, 2 für Tag 2)

**path**

Pfad zu Datei

**fmtIdx**

Format Index

Wenn OK wird true zurückgegeben.

### B.2.12 Stückliste erstellen

boolean **createPlaylist**(void);

Wenn OK wird true zurückgegeben.

### B.2.13 Dateien einer Stückliste abfragen

array **getPlaylistItems**(string path);

**path**

Pfad zu Stücklistendatei

Gibt eine Liste mit absoluten Pfaden zu den Stücken der Wiedergabeliste zurück.

### B.2.14 Dateien einer Stückliste setzen

boolean **setPlaylistItems**(string path, array items);

**path**

Pfad zu Stücklistendatei

**items**

Liste mit absoluten Pfaden zu den Stücken der Wiedergabeliste

Wenn OK wird true zurückgegeben, false wenn nicht alle Stücke gefunden wurden oder das Speichern fehlgeschlagen ist.

### B.2.15 Anwendung beenden

**quit**(void);

### B.2.16 Alles auswählen

**selectAll**(void);

### **B.2.17 Gesamte Auswahl aufheben**

`deselectAll(void);`

### **B.2.18 Gehe zur ersten Datei**

boolean `firstFile(void);`

Gibt true zurück, wenn eine Datei existiert.

### **B.2.19 Gehe zur vorangehende Datei**

boolean `previousFile(void);`

Gibt true zurück, wenn eine vorangehende Datei existiert.

### **B.2.20 Gehe zur nächstes Datei**

boolean `nextFile(void);`

Gibt true zurück, wenn eine nächste Datei existiert.

### **B.2.21 Erste Datei auswählen**

boolean `selectFirstFile(void);`

Gibt true zurück, wenn eine Datei existiert.

### **B.2.22 Vorangehende Datei auswählen**

boolean `selectPreviousFile(void);`

Gibt true zurück, wenn eine vorangehende Datei existiert.

### **B.2.23 Nächste Datei auswählen**

boolean `selectNextFile(void);`

Gibt true zurück, wenn eine nächste Datei existiert.

### **B.2.24 Aktuelle Datei auswählen**

boolean `selectCurrentFile(void);`

Gibt true zurück, wenn eine aktuelle Datei existiert.

### **B.2.25 Aktuellen Ordner auf- oder zuklappen**

boolean `expandDirectory(void);`

Ein Dateilistenelement ist ein Ordner, wenn `getFileName()` einen Namen mit '/' als letztem Zeichen zurückgibt.

Gibt true zurück, wenn das aktuelle Dateilistenelement ein Ordner ist.

### B.2.26 Dateinamenformat anwenden

`applyFilenameFormat(void);`

### B.2.27 Tag Format anwenden

`applyTagFormat(void);`

### B.2.28 Textkodierung anwenden

`applyTextEncoding(void);`

### B.2.29 Ordnername von Tags setzen

boolean `setDirNameFromTag(int32 tagMask, string format, boolean create);`

***tagMask***

Tag Maske (Bit 0 für Tag 1, Bit 1 für Tag 2)

***format***

Ordnerformat

***create***

true zum Erstellen, false zum Umbenennen

Wenn OK wird true zurückgegeben, sonst ist eine Fehlermeldung mit `getErrorMessage()` abrufbar.

### B.2.30 Stücke in ausgewählten Dateien nummerieren

`numberTracks(int32 tagMask, int32 firstTrackNr);`

***tagMask***

Tag Maske (Bit 0 für Tag 1, Bit 1 für Tag 2)

***firstTrackNr***

Nummer für erste Datei

### B.2.31 Dateien filtern

`filter(string expression);`

***expression***

Filter Ausdruck

### B.2.32 ID3v2.3 nach ID3v2.4 konvertieren

`convertToId3v24(void);`

### B.2.33 ID3v2.4 nach ID3v2.3 konvertieren

`convertToId3v23(void);`

Wenn OK wird true zurückgegeben.

### B.2.34 Ordnerpfad

`string getDirectoryName(void);`

Gibt den absoluten Ordnerpfad zurück.

### B.2.35 Name der aktuellen Datei

`string getFileName(void);`

Gibt den absoluten Dateinamen zurück, wenn es ein Ordner ist, endet er mit “/”.

### B.2.36 Setze Dateinamen

`setFileName(string name);`

*name*

Dateiname

Die Datei wird umbenannt, wenn der Ordner gespeichert wird.

### B.2.37 Setze Format für Dateiname aus Tags

`setFileNameFormat(string format);`

*format*

Dateinamenformat

### B.2.38 Generiere den Dateinamen aus den Tags

`setFileNameFromTag(int32 tagMask);`

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

### B.2.39 Wert eines Elementes

`string getFrame(int32 tagMask, string name);`

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

*name*

Elementname (z.B. “artist”)

Um binäre Daten wie ein Bild von einem Element in einer Datei zu speichern, kann der Dateiname dem Elementnamen hinzugefügt werden, z.B. “Picture:/Pfad/zu/Datei”. Auf dieselbe Weise kann synchronisierter Liedtext exportiert werden, z.B. “SYLT:/Pfad/zu/Datei”.

Gibt Wert des Elementes zurück.

### B.2.40 Setze Wert eines Elementes

boolean **setFrame**(int32 tagMask, string name, string value);

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

*name*

Elementname (z.B. "artist")

*value*

Wert des Elementes

Für Tag 2 (*tagMask* 2) gilt: Wenn kein Element mit *name* existiert, so wird ein neues Element hinzugefügt, wenn *value* leer ist, so wird das Element gelöscht. Um binäre Daten wie ein Bild einem Element hinzuzufügen, kann der Dateiname dem Elementnamen hinzugefügt werden, z.B. "Picture:/Pfad/zu/Datei". "SYLT:/Pfad/zu/Datei" kann verwendet werden, um synchronisierten Liedtext im LRC-Format zu importieren.

Wenn OK wird true zurückgegeben.

### B.2.41 Alle Elemente eines Tags

array of string **getTag**(int32 tagMask);

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

Gibt eine Liste mit abwechselnd Elementnamen und -werten zurück.

### B.2.42 Technische Informationen über Datei

array of string **getInformation**(void);

Eigenschaften sind Format, Bitrate, Samplerate, Channels, Duration, Channel Mode, VBR, Tag 1, Tag 2. Eigenschaften, welche nicht vorhanden sind, werden ausgelassen.

Gibt eine Liste mit abwechselnd Eigenschaftsnamen und -werten zurück.

### B.2.43 Setze Tag aus Dateinamen

**setTagFromFileName**(int32 tagMask);

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

### B.2.44 Setze Tag aus anderem Tag

**setTagFromOtherTag**(int32 tagMask);

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

### **B.2.45 Kopiere Tag**

`copyTag(int32 tagMask);`

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

### **B.2.46 Füge Tag ein**

`pasteTag(int32 tagMask);`

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

### **B.2.47 Entferne Tag**

`removeTag(int32 tagMask);`

*tagMask*

Tag Bit (1 für Tag 1, 2 für Tag 2)

### **B.2.48 Lese die Konfiguration neu ein**

`reparseConfiguration(void);`

Automatisierte Konfigurationsänderungen sind durch direktes Modifizieren der Konfigurationsdatei und anschließendes Einlesen möglich.

### **B.2.49 Spielt die ausgewählten Dateien ab**

`playAudio(void);`

## Anhang C

# QML Schnittstelle

### C.1 QML Beispiele

QML Skripte lassen sich über das Kontextmenü der Dateiliste aufrufen und in den Einstellungen auf der Seite [Benutzerbefehle](#) setzen. Die dortigen Beispielskripte bieten eine Vorlage für die Programmierung eigener Skripte. Codiert wird in JavaScript, hier das obligate "Hello World":

```
import Kid3 1.0

Kid3Script {
  onRun: {
    console.log("Hello world, directory is", app.dirName)
    Qt.quit()
  }
}
```

Wird dieses Skript unter `/path/to/Example.qml` abgespeichert, so kann als Benutzerbefehl `@qml /path/to/Example.qml` definiert werden. Wenn als Name **QML Test** gesetzt und **Ausgabe** aktiviert wird, so lässt sich das Skript über den Kontextmenüpunkt **QML Test** der Dateiliste starten, und die Ausgabe ist im Fenster sichtbar.

Alternativ lässt sich das Skript auch unabhängig von Kid3 mit den QML-Werkzeugen starten.

```
qml -apptype widget -I /usr/lib/kid3/plugins/imports /path/to/Example.qml
```

oder

```
qmlscene -I /usr/lib/kid3/plugins/imports /path/to/Example.qml
```

Bei Windows<sup>®</sup> und macOS<sup>®</sup> muss der Importpfad angepasst werden, damit er auf `imports` bei den Plugins im Installationsordner verweist. Außerhalb von Kid3 gestartete Skripte verwenden den aktuellen Ordner, es sollte also vorangehend in den gewünschte Ordner gewechselt werden.

Um die Titel im Tag 2 aller Dateien des aktuellen Ordners aufzulisten, könnte folgendes Skript verwendet werden:

```
import Kid3 1.0

Kid3Script {
  onRun: {
    app.firstFile()
    do {
```



## Das Kid3 Handbuch

```
        if (app.selectionInfo.tag(Frame.Tag_2).tagFormat) {
            console.log(app.getFrame(tagv2, "title"))
        }
    } while (app.nextFile())
}
}
```

Wenn sich viele Dateien im Ordner befinden, blockiert ein solches Skript die Benutzeroberfläche für einige Zeit. Für längere Aktionen sollte man daher kurze Pausen einlegen. Die unten stehende Alternative lagert die Arbeit an einer einzelnen Datei in eine Funktion aus. Diese ruft sich am Ende mit einem Timeout von 1 ms wieder selbst auf, sofern noch weitere Dateien zu bearbeiten sind. Dadurch bleibt das GUI auch während dem Ablauf des Skripts benutzbar.

```
import Kid3 1.0

Kid3Script {
  onRun: {
    function doWork() {
      if (app.selectionInfo.tag(Frame.Tag_2).tagFormat) {
        console.log(app.getFrame(tagv2, "title"))
      }
      if (!app.nextFile()) {
        Qt.quit()
      } else {
        setTimeout(doWork, 1)
      }
    }

    app.firstFile()
    doWork()
  }
}
```

Mit **app.firstFile()** und **app.nextFile()** werden alle Dateien des aktuellen Ordners bearbeitet. Falls es nur die ausgewählten Dateien sein sollen, so sollten stattdessen **firstFile()** und **nextFile()** verwendet werden, diese Funktionen sind in der Komponente `Kid3Script` enthalten. Das folgende Beispiel kopiert bloß die CD-Nummer- und Copyright-Frames der ausgewählten Datei.

```
import Kid3 1.1

Kid3Script {
  onRun: {
    function doWork() {
      if (app.selectionInfo.tag(Frame.Tag_2).tagFormat) {
        app.setFrame(tagv2, "*.selected", false)
        app.setFrame(tagv2, "discnumber.selected", true)
        app.setFrame(tagv2, "copyright.selected", true)
        app.copyTags(tagv2)
      }
      if (!nextFile()) {
        Qt.quit()
      } else {
        setTimeout(doWork, 1)
      }
    }

    firstFile()
    doWork()
  }
}
```

```
}
```

Weitere Beispiele sind bei Kid3 dabei und bereits bei den Benutzerbefehlen eingetragen.

- **ReplayGain to SoundCheck** (`ReplayGain2SoundCheck.qml`): Erstellt iTunesNORM SoundCheck Informationen aus Replay Gain Werten.
- **Resize Album Art** (`ResizeAlbumArt.qml`): Skaliert eingebettete Bilder runter auf 500x500 Pixel.
- **Extract Album Art** (`ExtractAlbumArt.qml`): Speichert alle eingebetteten Bilder als Dateien ab, ohne Duplikate zu erzeugen.
- **Embed Album Art** (`EmbedAlbumArt.qml`): Bettet Bilder aus Dateien in die Audio-Dateien ein, welche sich im selben Ordner befinden.
- **Embed Lyrics** (`EmbedLyrics.qml`): Holt unsynchronisierte Liedtexte von einem Webserver.
- **Text Encoding ID3v1** (`ShowTextEncodingV1.qml`): Hilft dabei, die Codierung zu finden, indem es die ID3v1 Tags in allen verfügbaren Zeichencodierungen anzeigt.
- **ID3v1 to ASCII** (`Tag1ToAscii.qml`): Konvertiere Zeichen aus den lateinischen Unicodeblöcken im ID3v1 Tag nach ASCII.
- **English Title Case** (`TitleCase.qml`): Formatiere den Text in den Tags nach den Regeln englischer Groß- und Kleinschreibung in Titeln.
- **Rewrite Tags** (`RewriteTags.qml`): Schreibt die Tags in allen ausgewählten Dateien neu.
- **Export CSV** (`ExportCsv.qml`): Exportiert rekursiv alle Tags aller Dateien in eine CSV-Datei.
- **Export Playlist Folder** (`ExportPlaylist.qml`): Kopiert alle Dateien einer Playlist in ein Ordner und benennt sie gemäss ihrer Position um.
- **QML Console** (`QmlConsole.qml`): Einfache Konsole, um die Funktionen von Kid3's QML API auszuprobieren.

## C.2 QML API

Das API lässt sich leicht über die QML Console erkunden, welche als Beispiel für ein Skript mit Benutzeroberfläche verfügbar ist.

### C.2.1 Kid3Script

Kid3Script ist eine normale QML-Komponente im Plugin-Ordner. Man kann ebenso gut andere QML-Komponenten als Basis brauchen. Die Verwendung von Kid3Script sorgt dafür, dass sich eine Skript-Funktion einfach im `onRun` Signal Handler aufrufen lässt. Darüber hinaus werden noch einige Funktionen geboten:

```
onRun: Signal Handler, welcher beim Start des Skripts aufgerufen wird
tagv1, tagv2, tagv2v1: Konstanten für tag Parameter
script: Zugriff auf Skript-Funktionen
configs: Zugriff auf Konfigurations-Objekte
getArguments(): Liste mit Skript-Argumenten
isStandalone(): true wenn Skript nicht aus Kid3 heraus gestartet wird
setTimeout(callback, delay): Startet callback nach delay ms
firstFile(): Erste ausgewählte Datei
nextFile(): Nächste ausgewählte Datei
```

## C.2.2 Skript-Funktionen

Da JavaScript und somit auch QML nur einen eingeschränkten Funktionsumfang für Skripte mitbringt, stellt das `script` Objekt einige Methoden zur Verfügung, welche den Funktionsumfang erweitern, unter anderem:

```

script.properties(obj): Qt Eigenschaften des Objekts
script.writeFile(filePath, data): Schreibe in Datei, true wenn OK
script.readFile(filePath): Lese Daten von Datei
script.removeFile(filePath): Lösche Datei, true wenn OK
script.fileExists(filePath): true wenn Datei existiert
script.fileIsWritable(filePath): true wenn Datei schreibbar ist
script.getFilePermissions(filePath): Lese Dateiberechtigungen (Mode Bits)
script.setFilePermissions(filePath, modeBits): Setze Dateiberechtigungen
script.classifyFile(filePath): Klasse der Datei (Ordner "/", Symlink "@",
  Exe "**", Datei " ")
script.renameFile(oldName, newName): Datei umbenennen, true wenn OK
script.copyFile(source, dest): Kopiere Datei, true wenn OK
script.mkdir(path): Erstelle Ordner, true wenn OK
script.removeDir(path): Lösche Ordner, true wenn OK
script.tempPath(): Ordner temporärer Dateien
script.musicPath(): Musik-Ordner
script.listdir(path, [nameFilters], [classify]): Liste der Ordneinträge
script.system(program, [args], [msecs]): Systemkommando synchron ausführen,
  [exit code, standard output, standard error] wenn kein Timeout
script.systemAsync(program, [args], [callback]): Systemkommando asynchron
ausführen, callback wird mit [exit code, standard output, standard error] ←
  aufgerufen
script.getenv(varName): Wert einer Umgebungsvariablen
script.setenv(varName, value): Setze Wert einer Umgebungsvariablen
script.getQtVersion(): Qt Version, z.B. "5.4.1"
script.getDataMd5(data): MD5 Hash (hex) der Daten
script.getDataSize(data): Anzahl Datenbytes
script.dataToImage(data, [format]): Erzeuge Bild aus Daten
script.dataFromImage(img, [format]): Daten aus einem Bild
script.loadImage(filePath): Lade Bild aus einer Datei
script.saveImage(img, filePath, [format]): Speichere Bild in Datei, true ←
  wenn OK
script.imageProperties(img): Eigenschaften eines Bildes, enthält
  "width", "height", "depth" und "colorCount", leer wenn Bild ungültig
script.scaleImage(img, width, [height]): Liefert skaliertes Bild
    
```

## C.2.3 Applikations-Kontext

Über QML ist ein großer Teil der Kid3 Funktionalität verfügbar. Das API ist ähnlich wie bei [D-Bus](#). Für Details wird auf die dortige Beschreibung verwiesen.

```

app.openDirectory(path): Ordner öffnen
app.unloadAllTags(): Entlade alle Tags
app.saveDirectory(): Speichern
app.revertFileModifications(): Zuletzt gespeicherte Fassung
app.importTags(tag, path, fmtIdx): Datei importieren
app.importFromTags(tag, source, extraction): Aus Tags importieren
app.importFromTagsToSelection(tag, source, extraction): Aus Tags der ←
  ausgewählten Dateien importieren
app.downloadImage(url, allFilesInDir): Bild herunterladen
app.exportTags(tag, path, fmtIdx): Datei exportieren
    
```

## Das Kid3 Handbuch

```
app.writePlaylist(): Stückliste erstellen
app.getPlaylistItems(path): Stücke einer Wiedergabeliste
app.setPlaylistItems(path, items): Setze Stücke einer Wiedergabeliste
app.selectAllFiles(): Alles auswählen
app.deselectAllFiles(): Gesamte Auswahl aufheben
app.firstFile([select], [onlyTaggedFiles]): Erste Datei
app.nextFile([select], [onlyTaggedFiles]): Nächste Datei
app.previousFile([select], [onlyTaggedFiles]): Vorangehende Datei
app.selectCurrentFile([select]): Aktuelle Datei auswählen
app.selectFile(path, [select]): Spezifische Datei auswählen
app.getSelectedFilePaths([onlyTaggedFiles]): Alle ausgewählten Dateien
app.requestExpandFileList(): Alle ausklappen
app.applyFilenameFormat(): Dateinamenformat anwenden
app.applyTagFormat(): Tag-Format anwenden
app.applyTextEncoding(): Textkodierung anwenden
app.numberTracks(nr, total, tag, [options]): Stücke nummerieren
app.applyFilter(expr): Filter
app.convertToId3v23(): ID3v2.4 nach ID3v2.3 konvertieren
app.convertToId3v24(): ID3v2.3 nach ID3v2.4 konvertieren
app.getFilenameFromTags(tag): Dateiname von Tag
app.getTagsFromFilename(tag): Tag von Dateiname
app.getAllFrames(tag): Objekt mit allen Elementen
app.getFrame(tag, name): Hole Element
app.setFrame(tag, name, value): Setze Element
app.getPictureData(): Daten aus Bild-Element
app.setPictureData(data): Setze Daten in Bild-Element
app.copyToOtherTag(tag): Tag zu anderem Tag
app.copyTags(tag): Kopieren
app.pasteTags(tag): Einfügen
app.removeTags(tag): Entfernen
app.playAudio(): Wiedergabe
app.readConfig(): Konfiguration lesen
app.applyChangedConfiguration(): Konfiguration anwenden
app.dirName: Ordnername
app.selectionInfo.fileName: Dateiname
app.selectionInfo.filePath: Absoluter Dateiname
app.selectionInfo.detailInfo: Format Details
app.selectionInfo.tag(Frame.Tag_1).tagFormat: Tag 1 Format
app.selectionInfo.tag(Frame.Tag_2).tagFormat: Tag 2 Format
app.selectionInfo.formatString(tag, format): Ersetze Format-Codes
app.selectFileName(caption, dir, filter, saveFile): Öffnet Dateidialog, um
eine Datei auszuwählen
app.selectDirName(caption, dir): Öffnet Dateidialog, um Ordner auszuwählen
```

Für asynchrone Operationen können Callbacks mit Signalen verbunden werden.

```
function automaticImport(profile) {
  function onAutomaticImportFinished() {
    app.batchImporter.finished.disconnect(onAutomaticImportFinished)
  }
  app.batchImporter.finished.connect(onAutomaticImportFinished)
  app.batchImport(profile, tagv2)
}

function renameDirectory(format) {
  function onRenameActionsScheduled() {
    app.renameActionsScheduled.disconnect(onRenameActionsScheduled)
    app.performRenameActions()
  }
}
```

```
app.renameActionsScheduled.connect (onRenameActionsScheduled)
app.renameDirectory(tagv2v1, format, false)
}
```

## C.2.4 Konfigurations-Objekte

Die einzelnen Konfigurationen sind über Methoden von `configs` verfügbar. Ihre Eigenschaften können in der QML Console aufgelistet werden.

```
script.properties (configs.networkConfig())
```

Eigenschaften können gesetzt werden:

```
configs.networkConfig().useProxy = false
```

```
configs.batchImportConfig()
configs.exportConfig()
configs.fileConfig()
configs.filenameFormatConfig()
configs.filterConfig()
configs.findReplaceConfig()
configs.guiConfig()
configs.importConfig()
configs.mainWindowConfig()
configs.networkConfig()
configs.numberTracksConfig()
configs.playlistConfig()
configs.renDirConfig()
configs.tagConfig()
configs.tagFormatConfig()
configs.userActionsConfig()
```