



Anleitung Ember Media Manager

Erstellt mit v.1.11.1

Scraper für Lokale & Netzwerk Daten

Stand März 2022

Inhaltsverzeichnis

Einleitung

1. Was macht Ember bzw. ein Scraper
2. Ember Menü auf Deutsch einstellen
3. Vorbereitungen

I. Optionen - Allgemein

1. Einstellungen – Optionen – Allgemein

II. Ember Einstellungen – Filme

1. Filme - Allgemein
2. Filme – Dateien und Quellen
3. Filme – Scraper Daten
4. Filme – Scraper Daten – TMDB
5. Filme – Scraper Daten – IMDB
6. Filme – Scraper Daten – OMDb & Trakt
7. Filme – Scraper Bilder
8. Filme – Scraper Bilder- FanartTV
9. Filme – Scraper Bilder- TMDB
10. Filme – Scraper Trailer

III. Ember Einstellungen - Filmsets

1. Filmsets - Allgemein
2. Filmsets - Allgemein – Dateien und Quellen
3. Filmsets – Scraper Daten
4. Filmsets – Scraper Daten – TMDB
5. Filmsets – Scraper Bilder
6. Filmsets – Scraper Bilder – FanartTV
7. Filmsets – Scraper Bilder – TMDB

IV. Ember Einstellungen – TV Serien

1. TV Serien - Allgemein
2. TV Serien – Dateien und Quellen
3. TV Serien – Scraper Daten
4. TV Serien – Scraper Daten- TMDB
5. TV Serien – Scraper Daten - TVDB
6. TV Serien – Scraper Daten - IMDB
7. TV Serien – Scraper Daten – Trakt.tv/OMDb
8. TV Serien – Scraper Bilder
9. TV Serien – Scraper Bilder - FanartTV

10.TV Serien – Scraper Bilder - TMDB

11.TV Serien – Scraper Bilder – TVDB

V. Ember Einstellungen- Module

1. Einstellungen - Module - Umbenennen
2. Einstellungen – Module – Kodi Interface
3. Einstellungen – Module – Trakt.tv

VI. Ember Einstellungen - Diverses

1. Einstellungen - Diverses - Kontextmenü
2. Einstellungen – Diverses – Medienlisten Editor
3. Einstellungen – Diverses – Videoquellen Zuordnung
4. Einstellungen – Diverses – Audio/Videocodec Zuordnung

VIII. Ember Datenbank füllen

VIV. Genre Manager

VV. Filmliste exportieren

VVI. Neue Inhalte (Filme/Serien) Scrapen

1. Neue Filme scrapen (hinzufügen)
- 2 .Neue TV Serien scrapen (hinzufügen)

Einleitung

Quelle der Anleitung: [Link](#)

Kodinerds Diskussion & Fragen Thema: [Link](#)

Dan Cooper Basic Tutorial (Grund Einweisung): [Link](#)

DanCooper (Ember Entwickler) auf Github: [Link](#)

Ember Media Manager ist ein Bibliothek Scraper für den PC und kann als alternative zum Kodi Scraper verwendet werden

Die Software nicht unter "C:\Programme" installieren, da in diesem Verzeichnis Adminrechte benötigt werden, damit Dateien geändert werden können

Ember ist portable, sprich Dateien wie Einstellungen, temporäre Dateien und Cache werden im Installationsordner abgelegt

Dadurch benötigt Ember Schreibrechte im eigenen Installationsordner

Da Ember Portable ist, kann man für ein Backup einfach den kompletten Ordner sichern oder nur den Unterordner *Profiles* (da der Cache ziemlich groß werden kann)

Bei der Installation von neuen Versionen (Update) muss immer der alte Installationsordner verwendet werden, da durch den Installer nicht mehr benötigte Dateien (Bibliotheken) gelöscht werden

Die Einstellungen wie auch Datenbank und Cache werden dabei behalten, die Datenbank wenn nötig beim ersten Start einer neuen Version upgegradet

Ein Backup vor der Installation einer neuen Version ist zu empfehlen!!!

Die hier genannten Einstellungen sind für deutschsprachige Benutzer in Verwendung mit Kodi optimiert Die ersten 4 Kapiteln & das letzte Kapitel sind, vor allem für Anfänger, die wichtigsten

[1. Was macht Ember bzw ein Scraper](#)

Mit einem Scraper, könnt Ihr z.B. Eure Videos welche Ihr auf einem NAS, am PC, einer Netzwerkfestplatte usw. liegen habt, einlesen und die dazugehörigen Bilder (Fanarts/Thumbs/Poster/Landscape usw.) und Informationen (Beschreibung, Laufzeit, Produktionsjahr usw.) aus dem Internet herunterladen

Ember macht eigentlich das gleiche wie der Kodi Scraper, jedoch sind die Möglichkeiten die Ember bietet sehr viel umfangreicher und die Dateien sind auch bei einem Verlust der Kodi Datenbank ohne erneutes Scrapen verfügbar

Außerdem werden mit Ember automatisch die .nfo Dateien (enthalten die Informationen des Films) erstellt

Es muss nicht die Bibliothek exportiert werden wie bei Kodi selbst

Ember legt für jeden Film selbständig einen eigenen Ordner an, in dem sich dann die Informationen befinden.

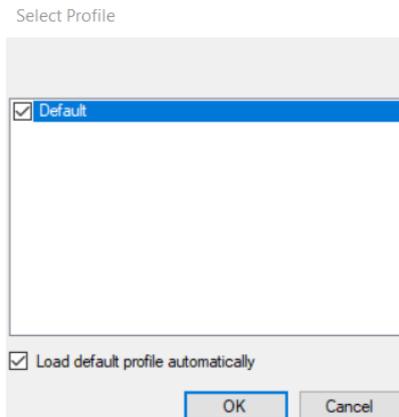
Wie lange der Scraper (Kodi oder Ember) zum Einlesen der Daten braucht, hängt von der Anzahl der Daten ab

Das alles kann Ember machen und noch viel mehr

Ihr müsst jedoch vorher einige Einstellungen machen, damit dass dann auch alles richtig funktioniert

Was und wie Ember Konfiguriert werden muss, ist in den obigen Links zu finden und hier in dieser Anleitung

Beim (ersten) Start wird ein Dialog zur Auswahl des Profils angezeigt. Setze hier einen Haken beim Eintrag *Default* (somit wird dieses Profil als Standardprofil markiert) und ebenfalls bei *Load default profile automatically*



2. Ember Menü auf Deutsch einstellen

Ember starten und im Menü, Edit – Settings wählen

Bei Interface – Interface Language, German auswählen

Im Anschluss rechts unten auf Apply und mit OK abschließen. Ember muss danach neu gestartet werden rechts unten auf Apply muss immer gemacht werden und zum Schluss mit OK abschließen

3. Vorbereitungen

Filme:

Es ist ein Ordner vorhanden, der als Quelle für Filme dient und in dem sich nur Filme und keine Serien befinden

Die Filme befinden sich in einzelnen Unterordnern, sprich es gibt in einem Ordner maximal einen Film

Wenn gemeinsame Unterordner für Filme in Filmsets verwendet werden sollen befindet sich in diesen Ordner jeweils ein Unterordner für jeden Film

Filmsets:

Für die Bilder der Filmsets steht ein gemeinsamer Ordner zur Verfügung

Serien:

Es ist ein Ordner vorhanden, der als Quelle für Serien dient und in dem sich nur Serien und keine Filme befinden

Für jede Serie besteht im Quellen-Ordner ein Ordner mit dem jeweiligen Serienname

Innerhalb der einzelnen Serien-Ordner können die Videodateien der Episoden an beliebigen Orten liegen (z.B. in Staffel-Verzeichnissen), müssen aber einen für [Kodi gültigen Dateinamen](#) mit Staffel- und Episodenummer-Angabe haben (Serien Ordner-Staffel Ordner-Episoden Benennung)

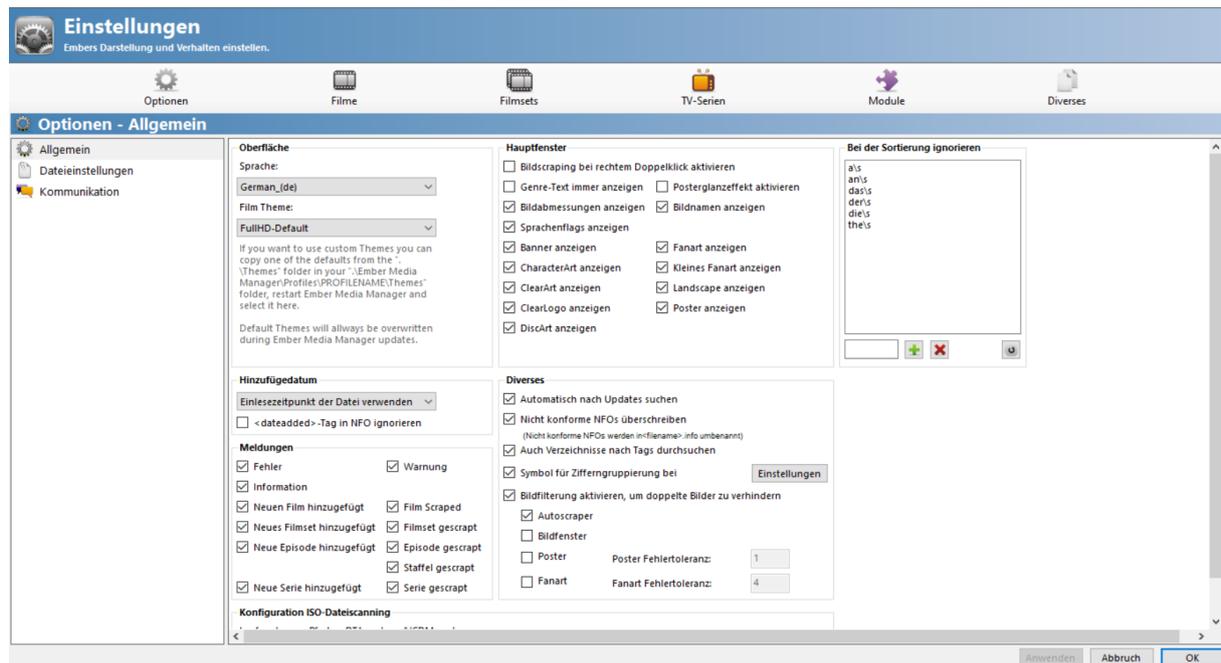
Solange an der Grundkonfiguration von Kodi und Ember nichts geändert wird werden Nummern evtl. vorhandener Staffeldordner ignoriert

Ember unterstützt bei Serien bzw. Episoden weder VIDEO_TS noch BDMV Strukturen

Diese müssen in ISO-Dateien gepackt oder gemuxt werden

I. Optionen – Allgemein (TOP)

1. Bearbeiten – Einstellungen – Optionen – Allgemein



Wie hier am Bild dargestellt einstellen

Im Hauptfenster kann *Fanart anzeigen* deaktiviert werden, falls es die Ansicht verwässert

Später zeigt ein Doppelklick auf eines der Bilder, in der Übersicht, das Bild vergrößert an

Diverses

Automatisch nach Updates suchen --> Sucht Ember bei jedem Start nach einer neuen Version

Nicht konforme NFOs überschreiben

Als "nicht konforme NFOs" werden NFOs bezeichnet, die über keine XML Struktur verfügen oder Fehler in der XML haben, z.B. Szene-Release-NFOs

Wenn diese Einstellungen deaktiviert ist werden fehlerhafte NFOs in ".info" umbenannt
Ember versucht immer, zumindest die IMDb-ID aus der Datei zu lesen, auch wenn sie nicht konform ist

Szene-Release-NFOs enthalten eigentlich immer eine IMDb-ID, welche das Video einem eindeutigen Film zuordnet

Wenn also eine solche NFO beim Einlesen vorhanden und die Einstellung aktiviert ist, liest Ember die IMDb-ID aus, und lässt die Datei bestehen

Die Datei wird dann beim Scrapen überschrieben

Wenn die Einstellung deaktiviert ist wird die IMDb-ID ebenfalls ausgelesen, die Datei aber sofort umbenannt

Auch Verzeichnisse nach Tags durchsuchen

Etwas schlecht übersetzt, aber mit "Tags" sind eigentlich "Flags" wie DVD, Bluray, TV usw. gemeint, welche die Videoquelle beschreiben und in Ember entsprechend angezeigt werden

Konfiguration ISO-Dateiscanning



Konfiguration ISO-Dateiscanning

Laufwerk Pfad zu DTAgent.exe/VCDMount.exe

Wer ISOs hat aus diesen die Informationen zu Audio- und Untertitelspuren auslesen will muss die Software "[Virtual CloneDrive](#)" installieren, dort vorzugsweise (nur) ein virtuelles Laufwerk einstellen und diesem einen fixen Laufwerksbuchstaben zuweisen

Dieser Laufwerksbuchstaben und der Pfad zu der Datei VCDMount.exe muss hier eingestellt werden

Daemon Tools funktioniert mittlerweile nicht mehr, da das CommandLine Interface von DT nicht mehr vernünftig funktioniert

Hinzufügedatum

Einlesezeitpunkt der Datei verwenden --> selbsterklärend

ctime (Fallback auf mtime) --> "Creation Time", also der Zeitpunkt, an dem diese Datei erstellt wurde

mtime (Fallback auf ctime) --> "Modification Time", also der Zeitpunkt, an dem diese bestehende Datei verändert wurde

Neuere von mtime und ctime --> selbsterklärend

<dateadded>-Tag in NFO ignorieren --> Film wurde schon einmal gescrapt und NFO ist vorhanden

Wenn dieser Tag in der vorhandenen NFO bereits eingetragen ist, dann nimmt Ember standardmäßig dieses Datum in seine DB auf und ignoriert die obigen Optionen

Also wenn der Haken gesetzt wird, nur dann nutzt Ember obige Optionen

Ist noch keine NFO vorhanden, dann ist dies ohne Wirkung

Bei der Sortierung ignorieren

Wer nicht möchte, dass Titel wie z.B "The Avengers" als "Avengers The" angezeigt und unter "A" gelistet werden, der löscht alle Einträge in dieser Spalte.

Dies wirkt sich nur auf die Anzeige in Ember aus, nicht auf den Titel, der in der NFO gespeichert wird

Meldungen

Es können, außer *Fehler*, alle Einträge deaktiviert werden wenn gewünscht

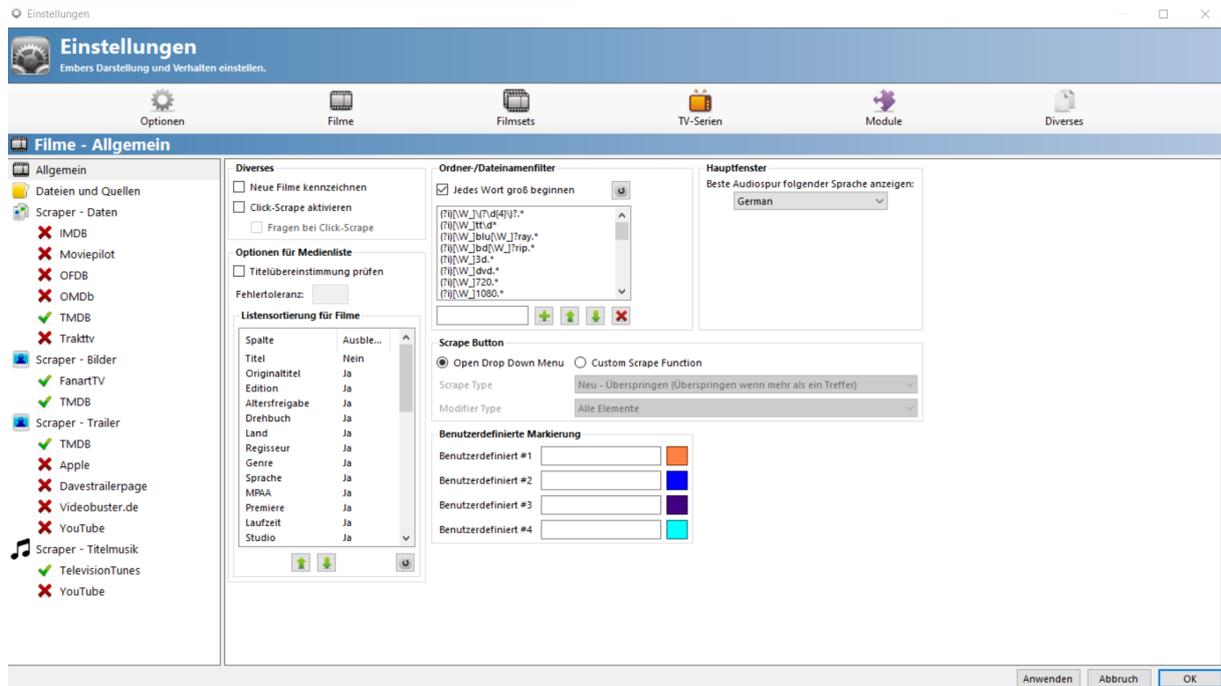
Bei *Dateieinstellungen* und *Kommunikation* muss nichts eingestellt werden

II. Ember Einstellungen – Filme (TOP)

Wird ein Film gescrapt, wird automatisch auch das Filmset erzeugt, wenn der Film auf TMDb einem Filmset zugewiesen ist

Das Filmset muss dann aber noch separat gescrapet werden, genauso wie man Filme scrapt

1. Bearbeiten – Einstellungen – Filme - Allgemein

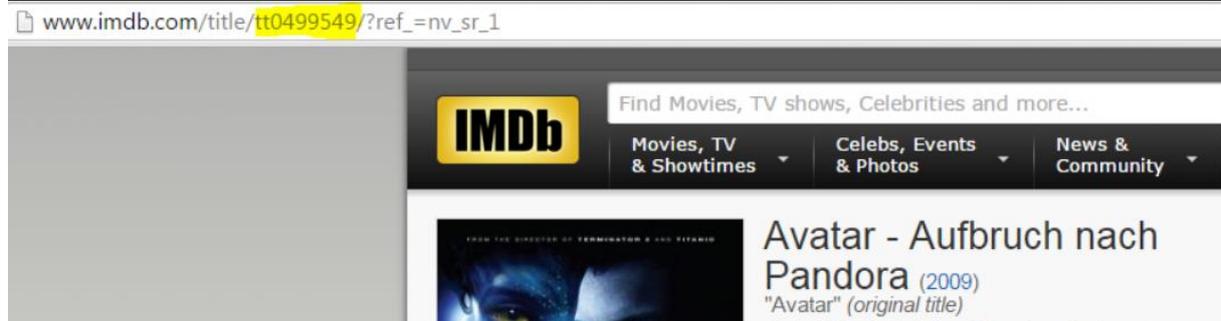


Diverses --> *Neue Filme kennzeichnen*: bedeutet, dass der Film markiert wird, wie STRG+M, Titel wird rot gefärbt

Ordner /Dateiname Filter --> *Jedes Wort groß beginnen*: macht aus z.B. "Adele und das Geheimnis des Pharaos.m2ts" -> "Adele Und Das Geheimnis Des Pharaos.m2ts"

Bei Ordner /Dateiname Filter lautet der 2. Eintrag `(?i)[\W_]tt\d*`: bedeutet, dass wenn im Dateinamen die IMDB-ID hinterlegt ist, diese ID, ohne weiteres Suchen, beim Scrapen verwendet wird

Beispiel: Avatar (2009) tt0499549.m2ts ergibt



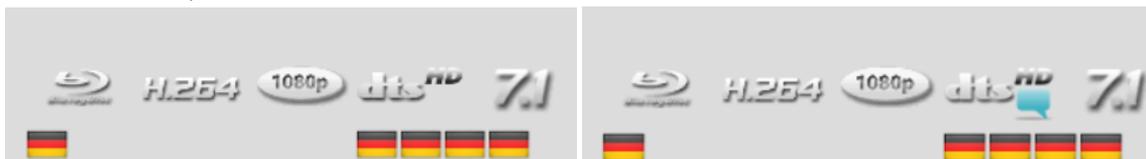
--> somit sind Verwechslungen für alle Zeit ausgeschlossen

Hauptfenster

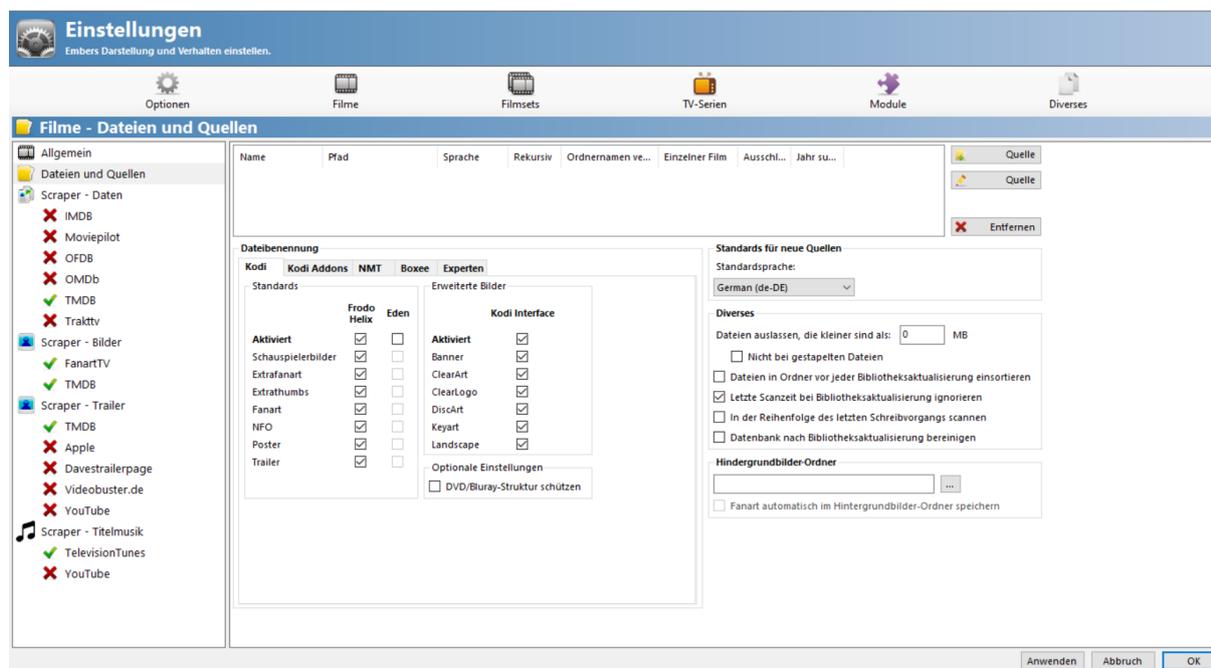
Wer hier German einstellt, dem wird dann in der Übersicht nicht einfach die beste Audiospur angezeigt, sondern die beste Deutsche, falls die Spuren mit Sprachinformationen versehen sind

Das hat den Vorteil, dass bei Dual- oder Multilanguage Filmen nicht die Info zur englischen Spur, welche meistens besser ist als die Deutsche, angezeigt wird

Dazu müssen dann natürlich alle Filme und Episoden mindestens als Dual-Language gespeichert werden
Einmal Deaktiviert, zweites Bild German



2. Bearbeiten – Einstellungen – Filme – Dateien und Quellen



Datei Benennung: hier wird nur der Reiter KODI benötigt, alles andere braucht man nicht zu beachten

Standards: alles aktivieren wie am Bild

Erweiterte Bilder: macht Sinn zu aktivieren und jeden Film immer gleich komplett mit allen Bildern zu scrapen

Zwar kann das Standard-Kodi-Skin nicht alle Bilder anzeigen, wer aber mal mit dem Gedanken spielt, ein anderes Skin zu testen, ist hier dann klar im Vorteil

Standards für neue Quellen

Datenbank nach Bibliotheksaktualisierung säubern:

Die Säuberung nach jedem Datenbank Update bringt nur wenig Vorteile, dafür verzögert es das Update, vor allem bei grossen Datenbanken

Im Normalfall kommen ja nur neue Filme und Serien hinzu, gelöscht wird eher selten was

Wer mit externen (USB-) Festplatten bzw. Speichermedien, welche nicht immer angehängt sind, arbeitet und diese gerade nicht aktiv sind, dann werden die Filme in der Ember-Datenbank gelöscht

Standardsprache:

Die Sprache German (de-DE) wählen und unten auf Anwenden drücken

Dies bewirkt, dass bei neuen Quellen die Sprache Deutsch voreingestellt ist

Dies wirkt sich nicht auf bereits vorhandene Quellen aus

Ausserdem kann die Sprache beim Erstellen einer Quelle immer noch geändert werden

Beim Ordner *Quelle*, jenene Ordner wählen, wo das zu scrapende Video Material liegt

Filmquelle

Quellenname:
Filme ✓

Quellpfad:
D:Ember Scraper\Filme ✗ ...

Quellenoptionen

Rekursiv scannen

Filme liegen in getrennten Ordnern *

Ordnernamen für Erstsuche verwenden

Jahr im Ordnernamen suchen

Pfad von Bibliotheksaktualisierungen ausschließen

Standardsprache: German (de-DE) ▾

* Dies MUSS aktiviert sein, um Extrathumbs und Dateien wie „movie.tbn“, „fanart.jpg“ usw. verwenden zu können.

OK Abbruch

Die Sprachzuordnung einzelner oder mehrerer Filme kann über das Kontextmenü geändert werden

Die Pfade können auch von Hand oder per Copy-Paste in das Feld eingetragen werden. Wichtig ist hierbei, dass die Gross-Kleinschreibung der Pfade korrekt ist.

Gültige Pfade sind:

- Ordner auf einer lokalen Festplatte, z.B. D:\Filme
- Netzlaufwerke, z.B: "\\Server\Filme" gemappt auf Laufwerk "Q:" (diese werden je nach Version von Windows im Dialog nicht angezeigt, da Microsoft das anscheinend nicht will)
- UNC-Adressen, z.B. "\\Server\Filme" oder "\\192.168.1.10\Filme"

Einschränkungen:

- der "Quellenname" muss eindeutig sein
- eine neue Quelle darf keinen Pfad einer bereits vorhandenen Quelle beinhalten (in Kodi möglich, hier nicht)
- eine neue Quelle darf kein Pfad innerhalb einer bereits vorhandenen Quelle sein (in Kodi möglich, hier nicht)

3. Bearbeiten – Einstellungen – Filme – Scraper Daten (TOP)

Diverses

- Deaktivierte Felder leeren
- Nur Schauspieler mit Bild berücksichtigen
- Nur Studios mit Bild berücksichtigen
- Handlung als Kurzbeschreibung verwenden Limit:
 - Nur wenn Kurzbeschreibung leer ist
- Handlung/Zusammenfassung bereinigen
- Erstelle <releasedate> Knoten in der NFO, mit den Werten von <premiered>
- Originaltitel als Titel verwenden
- YouTube-Trailer-Link in XBMC-kompatiblem Format speichern

Altersfreigaben

- Altersfreigabe für MPAA verwenden (XBMC)
 - Nur wenn keine MPAA gefunden wurde
 - Altersfreigabe: Zahl statt „Land:Zahl“ speichern (nur für YAMJ-Nutzer)
- MPAA-Wert wenn keine Altersfreigabe verfügbar ist:
-

Collection

- Filme automatisch den Filmsets hinzufügen
- Erweiterte Filmsetinformationen in der NFO speichern (Kodi 16.0 „Jarvis“ und neuer)
- YAMJ-kompatible Sets in NFO speichern

Ratings

- Default Rating e.g.: imdb, themoviedb, trakt, metacritic, tomatometerallcritics
- Create <rating> and <votes> nodes in NFO (default rating or first rating is used)

Unique IDs

- Default ID e.g.: imdb, tmdb
- Create <id> node in NFO (ID set as default or default ID is used)
 - Create <tmdb> node in NFO
 - Create <tmdbcolid> node in NFO

Laufzeitformat:

Spielzeit als Länge verwenden muss aktiviert sein, denn das Kodi-Interface überträgt die Spielzeit
Nur Eintrag <m> ist zulässig

Nur wenn Filmset ID und *Filme automatisch den Filmsets hinzufügen* (unter Collection) aktiviert sind, legt Ember gleich das Film-Set dazu an.

Scraperfeld Global:

Altersfreigabe aktivieren und daneben auf *Germany* stellen

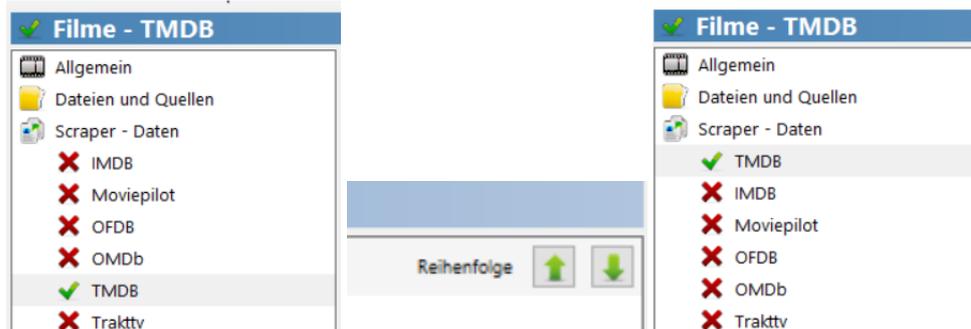
Altersfreigaben:

Benutze Altersfreigabe für MPAA (XBMC) aktivieren

Altersfreigaben und Scraperfeld Global Einstellung, gehören für deutsche Altersfreigaben zusammen

Wichtig ist hier noch, dass dies nur in der Scraper-Reihenfolge TMDB vor IMDB Sinn macht, denn ansonsten setzt IMDB immer einen MPAA Wert

Die Reihenfolge kann eingestellt werden, in dem z.B. TMDB links ausgewählt wird und dann mit den grünen Pfeilen rechts oben, TMDB nach oben verschoben wird



Wichtige Hinweis:

Viele User sind der Meinung dass Ember beim Scrapen viel zu lange braucht. Das liegt zum Einen daran, dass Ember sehr viele Informationen sich von den jeweiligen Anbietern holt, aber zum Anderen daran, dass ihr zu viele Scraper aktiviert habt und die Informationen dann mehrfach abgefragt werden
Die größte Einschränkung bildet aber der Anbieter selbst, denn der Eine oder Andere läßt nur eine bestimmte Anzahl an Anfragen in einer bestimmten Zeit zu

Hier mal eine kleine Übersicht:

TMDB lässt 30 API Calls in 10 Sekunden pro IP zu

Wird der Wert überschritten, pausiert die TMDB API automatisch für 30 Sekunden

Aus diesem Grund ist das Serien-Scrapen mit TMDB sehr langsam, da für jede Staffel und für jede Episode je ein API Call gemacht werden muss, damit man alle Daten bekommt

Aus dem Grund verzichtet Ember bei den Episoden-Infos auf gewisse Daten (externe IDs) und kommt somit mit einem API Call pro Serie und Staffel aus

Leider funktioniert das bei den TMDB Bildern für die Episoden nicht, da muss Ember für jede Episode einen API Call machen

Das Problem liegt aber am Aufbau der TMDB API, und so lange die nichts ändern (z.B. vollständigen Image Container gleich mit den Episoden-Infos mitliefern) wird sich auch betreffend Scrape-Dauer nichts verändern

Bei IMDB ist Ember noch nie an ein Limit gestossen und hier wurde auch nie was in der Richtung berichtet. Da IMDB aber keine offizielle API hat und Ember die Website auslesen muß, was vergleichsweise lange dauert, wird vermutlich Ember da auch nicht geblockt

Schlecht siehts bei OFDB aus, die blocken extrem schnell - schon nach ca. 80 Filmen in Folge

Dort ist man dann für 30-40 min gesperrt, danach gehts wieder

Die wollen aber auch nicht, dass man ihre Daten ausliest, deshalb wirds hier auch keine Verbesserung geben

Also merke - weniger ist manchmal mehr

4. Bearbeiten – Einstellungen – Filme – Scraper Daten – TMDB (TOP)

The screenshot shows the 'Einstellungen' (Settings) window for 'Filme - TMDB'. The sidebar on the left lists various scraper categories: 'Allgemein', 'Dateien und Quellen', 'Scraper - Daten' (with 'TMDB' selected), 'Scraper - Bilder' (with 'FanartTV' and 'TMDB'), and 'Scraper - Trailer' (with 'TMDB'). The main content area is titled 'Filme - TMDB' and contains several sections: 'Aktiviert' (checked), 'Scraperfelder - Scraper-spezifisch' (a grid of checkboxes for fields like 'Schauspieler', 'Originaltitel', 'Tagline', 'Altersfreigaben', 'Handlung', 'Titel', 'Filmset-ID', 'Premiere', 'Trailer', 'Länder', 'Bewertung', 'Credits (Autoren)', 'Regisseure', 'Laufzeit', 'Genres', and 'Studios'), and 'Scraper-Optionen' (with a 'TMDb-API-Schlüssel' label, a text input field, a button 'Eigenen API-Schlüssel', and checkboxes for 'Auf Englisch zurückgreifen', 'Inhalte für Erwachsene einschließen', and 'Suche um ± 1 Jahr erweitern, wenn kein Suchergebnis gefunden wurde').

Scraper Optionen:

Eigenen API Schlüssel: hat man sich einen eigenen TMDb API Key erstellt, so kann dieser hier eingetragen werden. Ansonst wird ein integrierter API Key verwendet

Dazu einfach auf den Text *Eigenen API Schlüssel* klicken und das Eingabefeld wird freigeschaltet

Auf Englisch zurückgreifen: bei fehlendem deutschen Inhalt wird englischer Inhalt verwendet

Benutzerkonto erstellen:

Auf <https://www.themoviedb.org/> Registrieren und ein Benutzerkonto anlegen, Captcha lösen, fertig

Es kann eine Wegwerf E-Mailadresse verwendet werden, ggf. mehrere Anbieter probieren

Es wird dann ein Bestätigungslink an die angegebene E-Mailadresse gesendet

Den Link in der E-Mail, in den Browser kopieren und Anmelden, Fertig

Das TMDb Konto ist nun erstellt

TMDB API-Key anfordern:

- Mit Eurem Konto anmelden und rechts oben auf das "Benutzerssymbol" klicken
- Einstellungen wählen
- unter Settings, am linken Rand, API wählen
- auf "Create" klicken
- Developer wählen
- Approve Terms of Use (AGB's), ganz unten Akzeptieren
- Es öffnet sich nun ein Fenster: API Schlüssel anfordern, wo die Eingabefelder mit erfundenen Daten ausgefüllt werden (somit gibt es keinen Bezug zu Eurer wahren Identität)
- im Anschluss auf "Erstellen" klicken
- nun wird unter Einstellungen - Settings - API - Details der *API Schlüssel (v3 auth)* angezeigt
Dieser wird nun in Ember eingefügt
- unter "Stats" ist ersichtlich, wie viele Zugriffe es mit diesem API-Key gegeben hat

5. Bearbeiten – Einstellungen – Filme – Scraper Daten – IMDB (TOP)

The screenshot shows the 'Einstellungen' (Settings) window for 'Filme - IMDB'. The sidebar on the left lists various scraper sources: 'Allgemein', 'Dateien und Quellen', 'Scraper - Daten' (with sub-items: TMDB, IMDB, Moviepilot, OFDB, OMDb, Traktv), 'Scraper - Bilder' (with sub-items: FanartTV, TMDB), and 'Scraper - Trailer' (with sub-items: TMDB, Apple, Davestrailerpage). The main content area is titled 'Filme - IMDB' and shows the following settings:

- Aktiviert
- Scraperfelder - Scraper-spezifisch**
 - Schauspieler
 - MPAA
 - Bewertung
 - Top 250
 - Altersfreigaben
 - inklusive Beschreibung *
 - Laufzeit
 - Länder
 - Originaltitel
 - Studios
 - Credits (Autoren)
 - Handlung
 - Distributoreninformation einschließen
 - Regisseure
 - Kurzbeschreibung
 - Tagline
 - Genres
 - Premiere
 - Titel
- * zusätzliche Seite(n) zu parsen, benötigt mehr Zeit zum Scrapen
- Scraper-Optionen**
 - Beliebte Titel
 - Nur Titel der Sprache:
 - Titelteiltreffer
 - Globaler Titel als Fallback
 - TV-Filmtitel
 - Videotitel
 - Kurzfilm Titel

Der Scraper ist sehr langsam, da die Webseite ausgelesen werden muss

Bearbeiten – Einstellungen – Filme – Scraper Daten – Moviepilot & OFDB

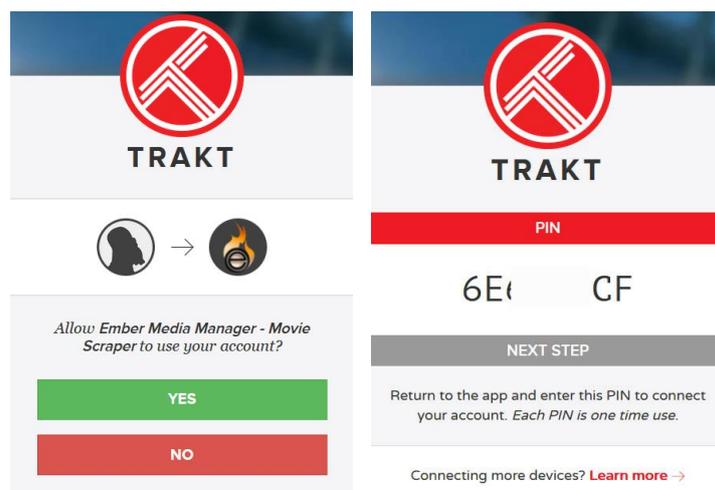
The screenshot shows two side-by-side views of the 'Einstellungen' (Settings) window for 'Filme - Moviepilot' and 'Filme - OFDB'. Both show the 'Scraper - Daten' section with 'Moviepilot' and 'OFDB' selected. The 'Scraperfelder - Scraper-spezifisch' section shows different settings for each:

- Filme - Moviepilot:**
 - Aktiviert
 - Altersfreigaben
 - Handlung
 - Kurzbeschreibung
- Filme - OFDB:**
 - Aktiviert
 - Altersfreigaben
 - Genres
 - Handlung
 - Titel

6. Bearbeiten – Einstellungen – Filme – Scraper Daten – OMDb & Trakt (TOP)



Trakt kann nur mit einem persönlichen Trakt.tv Benutzerkonto verwendet werden, hier [Trakt Konto erstellen](#). Wird Trakt aktiviert, so muss man sich vorher auf der [Trakt Homepage anmelden](#) und Ember mittels angezeigtem PIN zu Aktivieren



Trakt Konto erstellen:

Auf [Trakt.tv](#) ein Konto erstellen

In der Kategorie *Konto - Trakt* auf Berechtigung klicken

Es wird ein Infofenster angezeigt:

Webseite besuchen: <https://trakt.tv/activate>

Dort werdet Ihr dann aufgefordert den Code (der in Ember angezeigt wird) einzugeben, Continue Allow Ember to use Your Account (Erlaube Ember die Verwendung Deines Kontos), YES

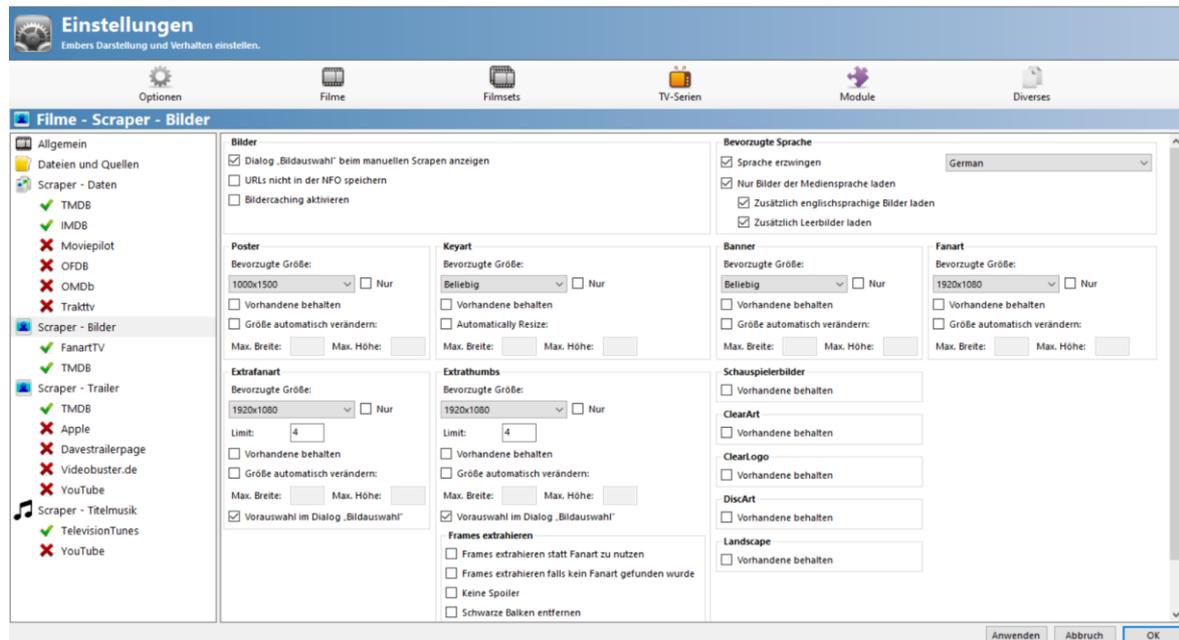
WooHoo! Your device is now connected and will automatically refresh in a few seconds (Dein Gerät ist jetzt verbunden und wird in wenigen Sekunden automatisch aktualisiert)

Der Takt Service kann ab jetzt genutzt werden

OMDB:

Für Bewertungen, falls eigener API Key vorhanden ist. Dieser kann kostenlos erstellt werden
Alle Felder aktivieren wenn OMDB verwendet wird

7.Bearbeiten – Einstellungen – Filme – Scraper Bilder (TOP)



Vorhandene behalten:

Nach dem Scraper werden, die Bild-Dateien beim Speichern immer überschrieben werden

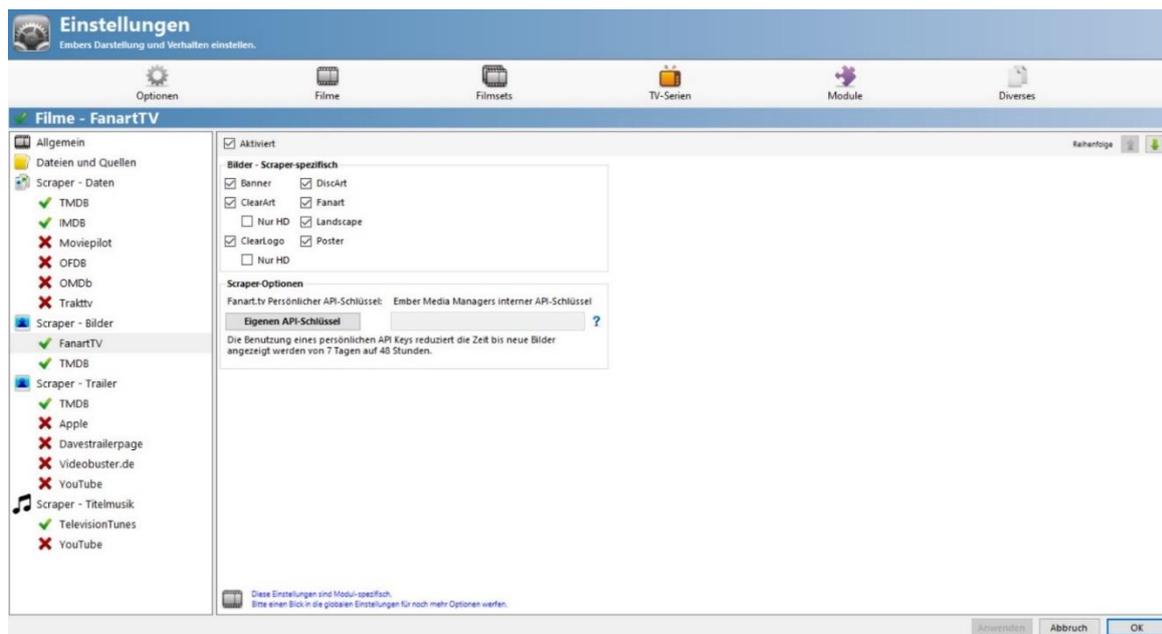
Wenn diese Funktion aktiviert ist, wird das vorhandene Bild beim Scraper oder Rescraper beibehalten und dann einfach neu unter dem aktivierten Dateinamen gespeichert wird

Bei *Extrafanarts* und *Extrathumbs* **muss** ein Limit gesetzt werden, da sonst Ember versucht alle Extras runterzuladen, was dann letzt endlich zu einem Absturz führen kann

Vorauswahl im Dialog Bildauswahl:

steht im Zusammenhang mit Optionen --> Allgemein --> Diverses --> Aktiviere Bildfilterung um doppelte Bilder zu verhindern

8. Bearbeiten – Einstellungen – Filme – Scraper Bilder- FanartTV (TOP)



Scraper Optionen:

Eigenen API Schlüssel: hier kann ein eigener [FanartTV](https://fanart.tv/) API Key eingetragen werden, wenn vorhanden

Um einen FanartTV API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden

Ansonst verwendet Ember den internen API Key

Benutzerkonto erstellen:

Auf der Homepage <https://fanart.tv/>, durch Klick auf Register, ein Benutzerkonto erstellen

Es kann auch eine Wegwerf/Einmal E-Mailadresse verwendet werden

An die angegebene E-Mailadresse wird ein Bestätigungslink geschickt

Nach dem Klick auf den Link, erfolgt eine Aufforderung, ein neues Passwort einzugeben

Durchführen und Klick auf *Reset Passwort*

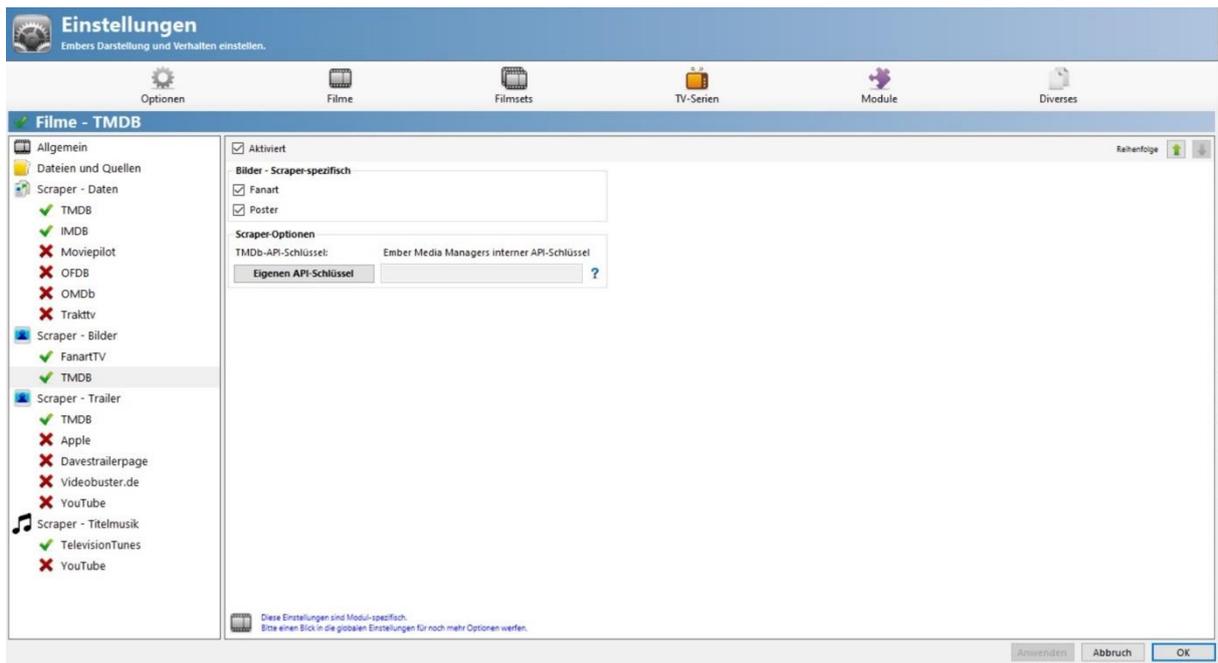
Zurück zur Homepage und Einloggen

Am unteren Ende der Homepage steht API und *Create API Key*, auswählen

Klick auf *Generate Personal API Key*

Es wird nun der Fanart API Key angezeigt, welcher dann in Ember eingetragen wird

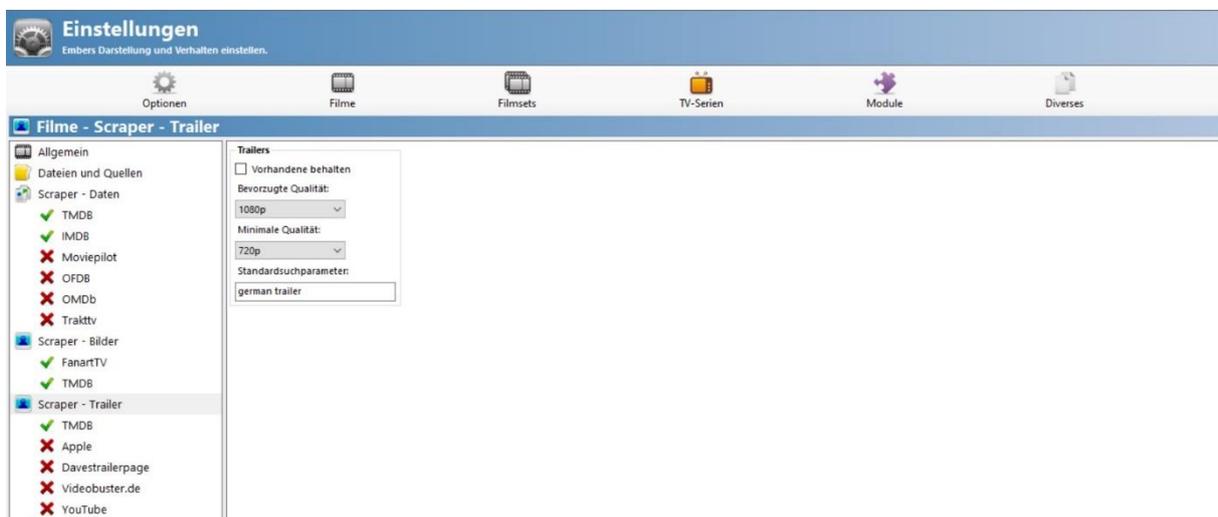
9. Bearbeiten – Einstellungen – Filme – Scraper Bilder - TMDB (TOP)



Scraper Optionen:

Eigene API Schlüssel: hier kann ein eigener [TMDB](#) API Key eingetragen werden, wenn vorhanden
Um einen TMDB API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden. Ansonst verwendet Ember den internen API Key

10. Bearbeiten – Einstellungen – Filme – Scraper Trailer (TOP)



Standardsuchparameter:

Einstellungen für Standardsuche um Trailer zu suchen. Der YouTube Scraper nutzt dann diese Einstellung

Scraper:

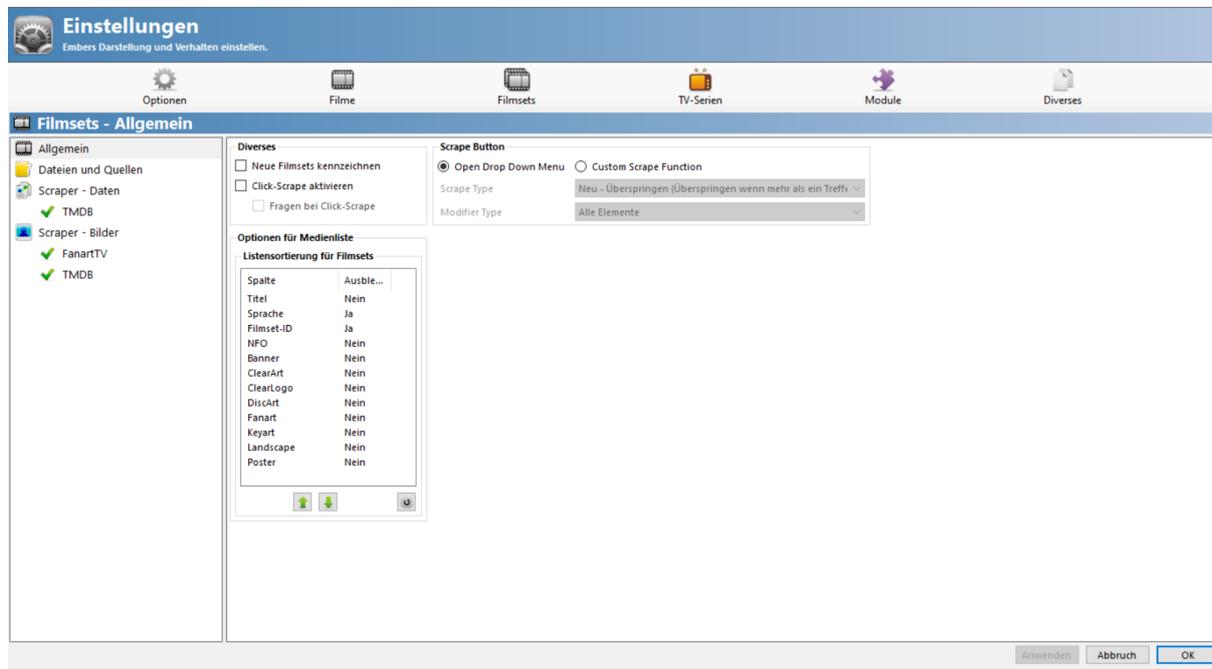
- **Videobuster.de:** liefert sehr viele Trailer in HD und deutscher Sprache, außerdem ist der Download extrem schnell
- **TMDb:** liefert Trailer von YouTube
- **YouTube:** sucht Trailer anhand des Filmtitels und Standardsuchparameters auf ... YouTube
- **Apple:** Trailer auf der Apple Webseite, meistens englisch

- *Davestrailerpage*: meistens Trailer auf Apple

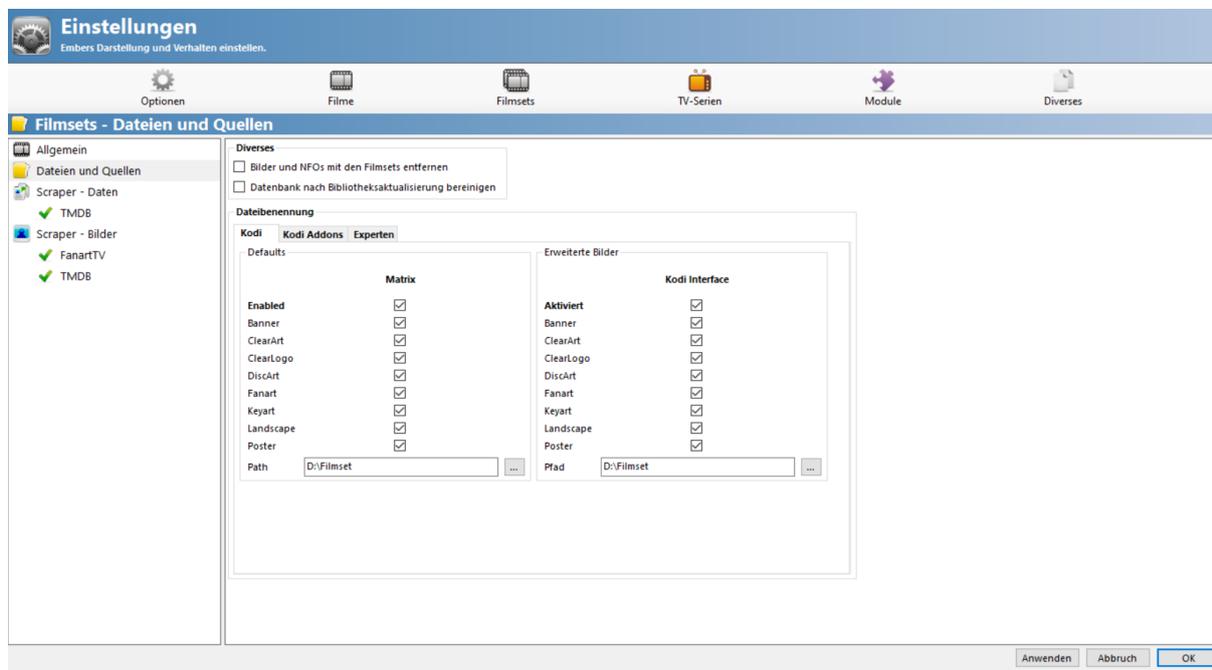
Es kann natürlich auch eine andere Qualität für die Trailer eingestellt werden, aber die obigen genügen meistens

III. Ember Einstellungen -Filmsets (TOP)

1. Bearbeiten –Einstellungen – Filmsets - Allgemein



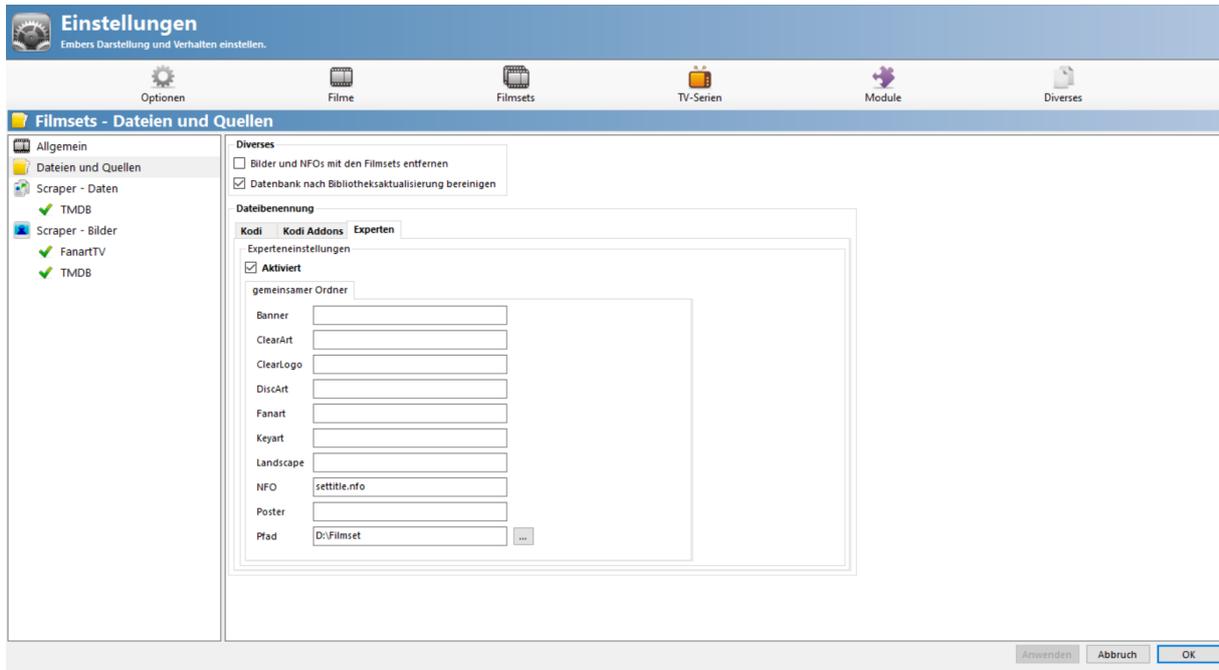
2. Bearbeiten –Einstellungen – Filmsets – Dateien und Quellen (TOP)



Dateibenennung --> Reiter Kodi --> Pfad:

hier muss ein Verzeichnis angegeben werden. In diesem werden dann alle Film-Set-Bilder, aller Sets, gespeichert (Beispiel D:\Filmset)

Im Reiter Kodi Addoons muss nichts eingestellt werden



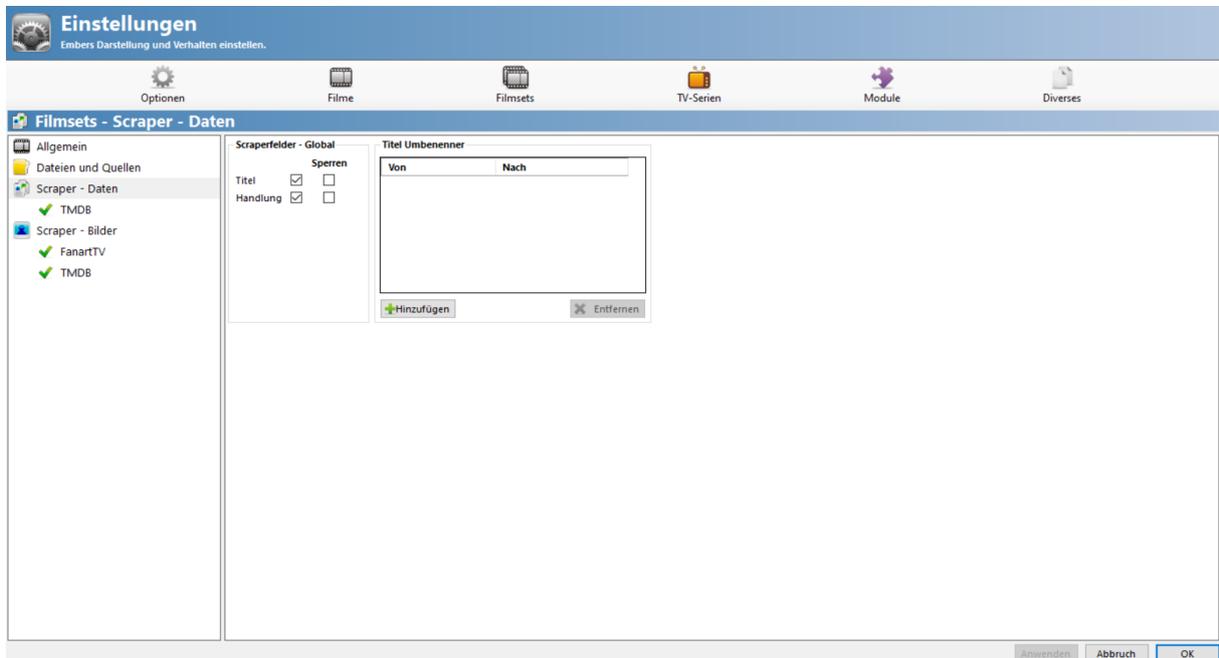
Dateibenennung --> Reiter Experten --> Pfad:

Pfad muss der selbe sein, wie bei Reiter Kodi

Kodi selbst verarbeitet keine Set.nfo, aber das eine oder andere Skin --> also gleich mitscrapen
Dafür sind obige Eingaben bei Reiter Experten zu machen

Achtung: es heißt jetzt *settitle.nfo* und nicht mehr *setname.nfo*

3. Bearbeiten –Einstellungen – Filmsets – Scraper Daten (TOP)



Titel Umbenener:

Ember benennt die Film-Sets: Filmtitel Filmreihe

Wer das anders haben will, trägt in die:

linke Spalte von Filmreihe und

rechte Spalte nach z.B. (Sets) oder Kollektion ein

4. Bearbeiten –Einstellungen – Filmsets – Scraper Daten – TMDb (TOP)

Scraperfelder – Scraper spezifisch

Eigene API Schlüssel: hier kann ein eigener [TMDb](#) API Key eingetragen werden, wenn vorhanden

Um einen TMDb API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden ([siehe Kap.II/4](#))

Ansonst verwendet Ember den internen API Key

5. Bearbeiten –Einstellungen – Filmsets – Scraper Bilder (TOP)

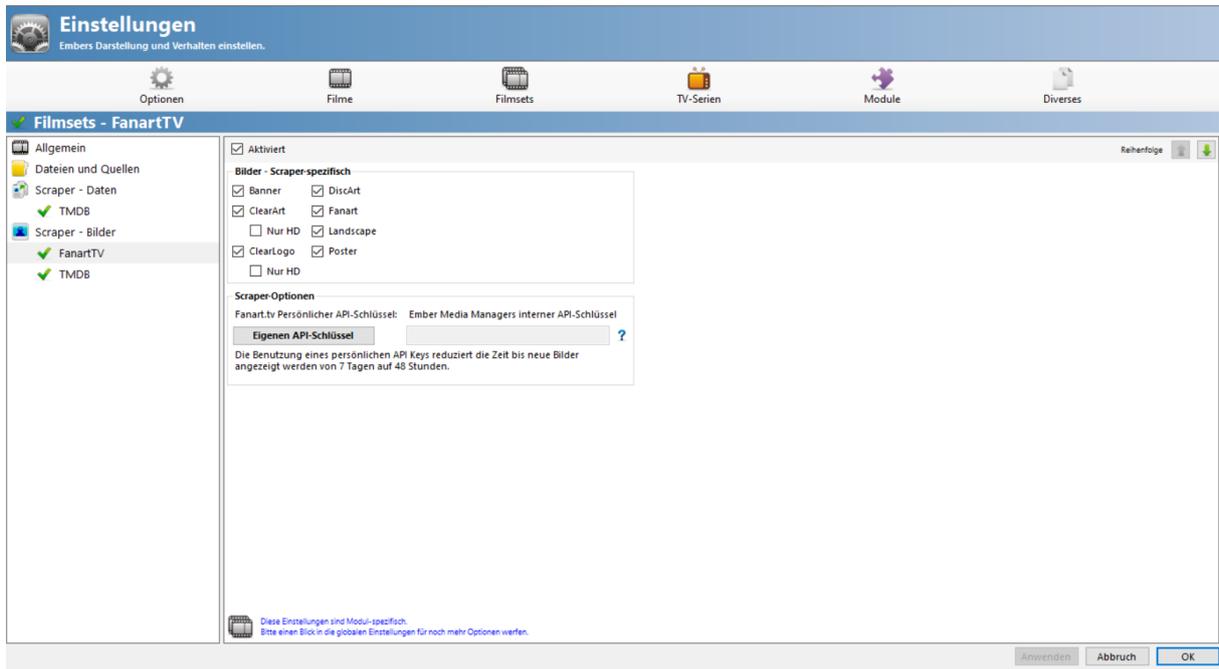
Achtung: *Vorhandene behalten* beachten!

Vorhandene behalten:

Nach dem Scrapen werden, die Bild-Dateien beim Speichern immer überschrieben

Wenn diese Funktion aktiviert ist, wird das vorhandene Bild beim Scrapen oder Rescrapen beibehalten und dann einfach neu unter dem aktivierten Dateinamen gespeichert wird

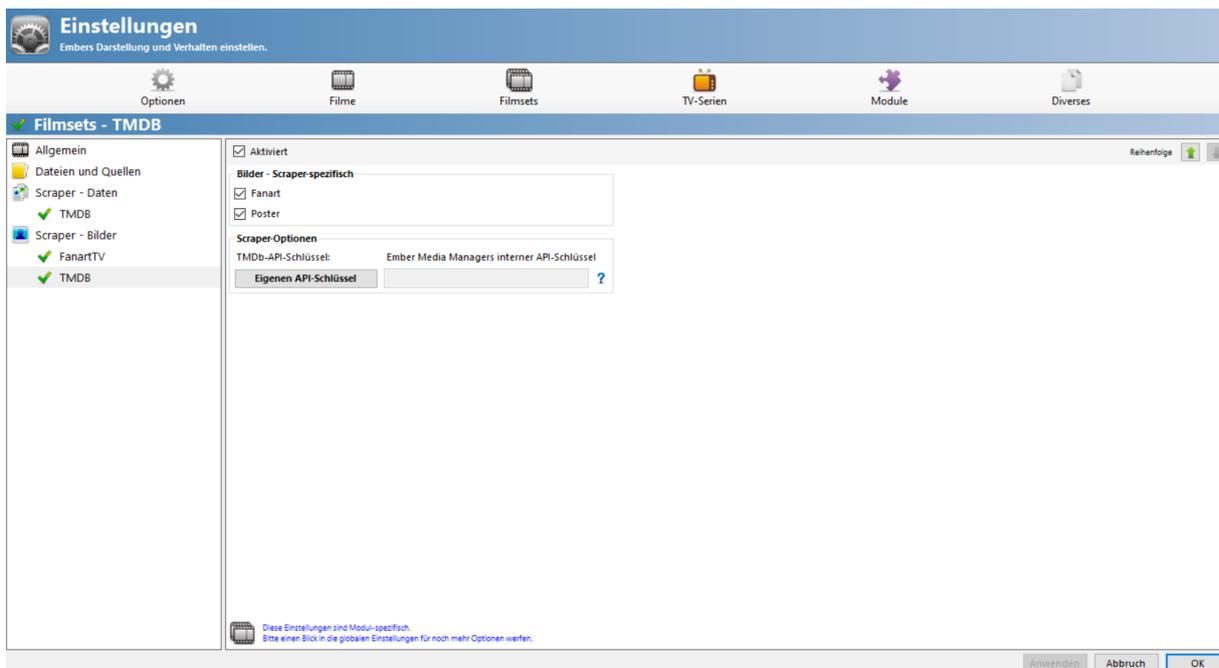
6. Bearbeiten –Einstellungen – Filmsets – Scraper Bilder – FanartTV (TOP)



Scraper Optionen:

Eigene API Schlüssel: hier kann ein eigener [FanartTV](#) API Key eingetragen werden, wenn vorhanden
Um einen FanartTV API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden ([siehe Kap.II/8](#)). Ansonst verwendet Ember den internen API Key

7. Bearbeiten –Einstellungen – Filmsets – Scraper Bilder – TMDB (TOP)



Scraperoptionen

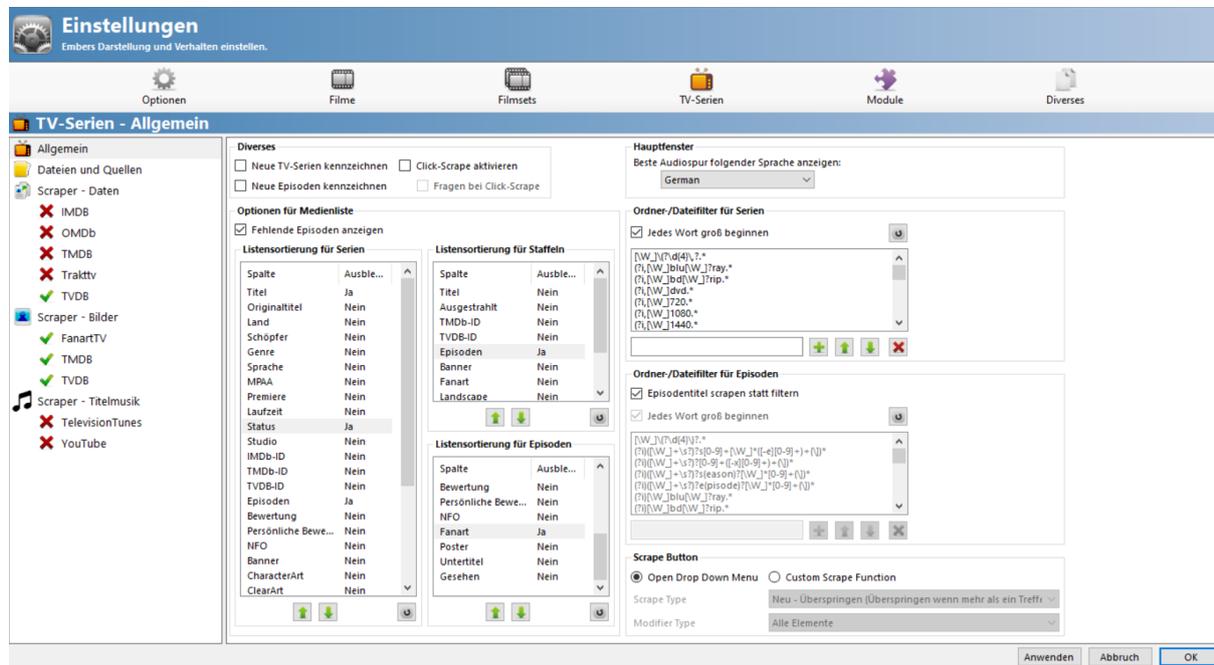
Eigene API Schlüssel: hier kann ein eigener [TMDB](#) API Key eingetragen werden, wenn vorhanden

Um einen TMDb API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden.
Ansonst verwendet Ember den internen API Key

IV. Ember Einstellungen – TV Serien (TOP)

1. Bearbeiten – Einstellungen – TV Serien - Allgemein

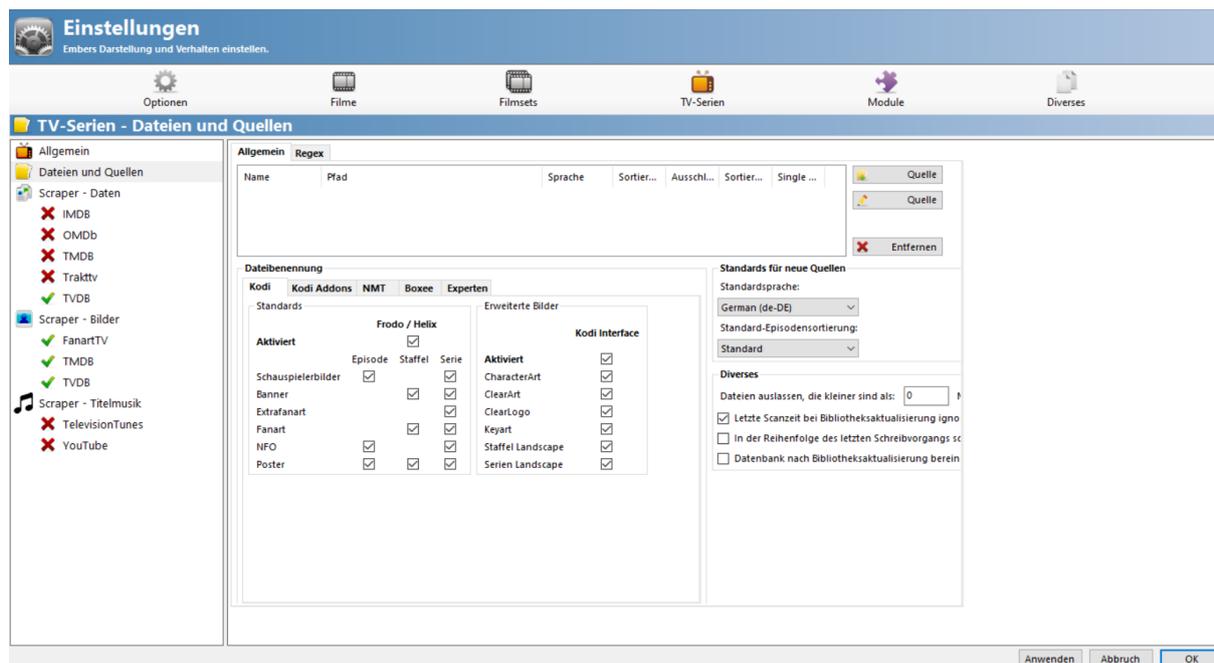
Vieles ist bereits bei [Filme](#) erklärt. Wie am Bild, wenige Einträge auf JA stellen



Ordner /Dateiname Filter --> *Jedes Wort groß beginnen*: macht aus z.B. "Adele und das Geheimnis des Pharaos.m2ts" -> "Adele Und Das Geheimnis Des Pharaos.m2ts"

Bei Ordner /Dateiname Filter lautet der 2. Eintrag `(?i)[W_]t\d*`: bedeutet, dass wenn im Dateinamen die IMDB-ID hinterlegt ist, diese ID, ohne weiteres Suchen, beim Scrapen verwendet wird

2. Bearbeiten – Einstellungen – TV Serien – Dateien und Quellen (TOP)

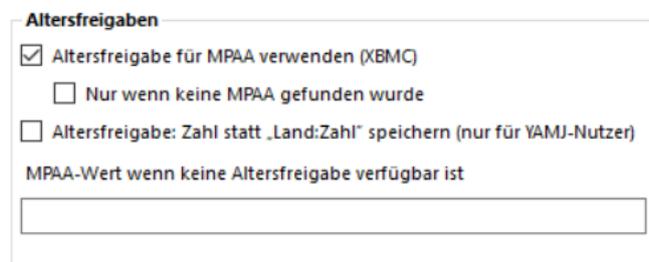
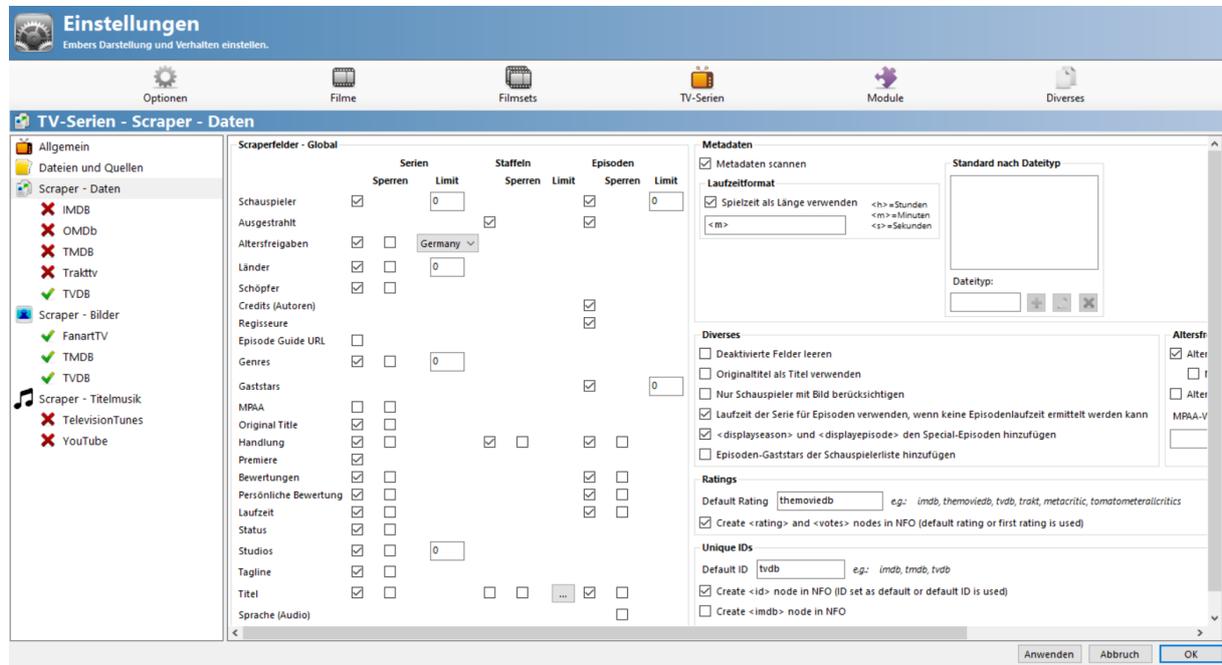


Datei Benennung: hier wird nur der Reiter *Kodi* benötigt, alles andere braucht man nicht zu beachten
Bei Reiter Kodi - Erweiterte Bilder - *Kodi Interface* aktivieren

Hinweis:

Hier gilt das Gleiche, wie es bereits oben bei [Filme](#), zu den Scrapern, geschrieben wurde
Also bitte um Beachtung

3. Bearbeiten – Einstellungen – TV Serien – Scraper Daten (TOP)



Staffel – Titel: ist standardgemäß nicht aktiviert und das sollte auch so sein

Nur dann kann Ember die Staffeln richtig numerisch sortieren. z.B. Staffel 01, Staffel 02 ... Staffel 10, Staffel 11, ansonsten wird daraus so eine Sortierung: Staffel 1, Staffel 10, Staffel 11, Staffel 2 usw.

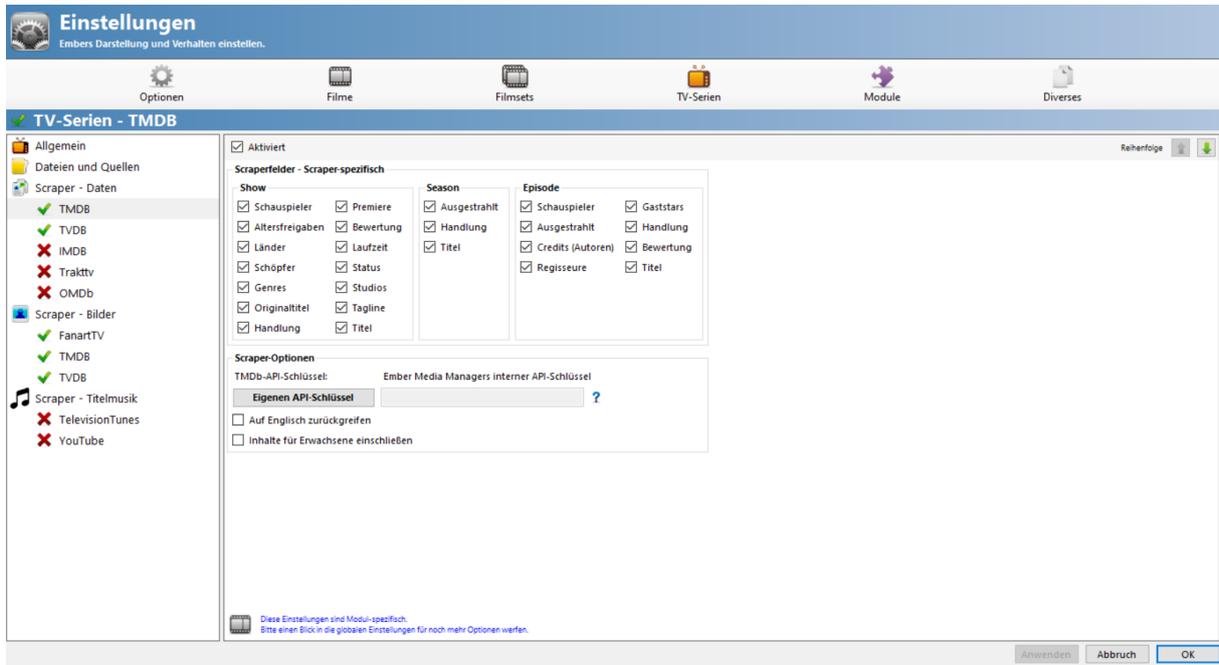
Laufzeit Format ist gleich wie bei Filme

Laufzeitformat:

Spielzeit als Länge verwenden muss aktiviert sein, denn das Kodi-Interface überträgt die Spielzeit
Nur Eintrag <m> ist zulässig

4. Bearbeiten – Einstellungen – TV Serien – Scraper Daten - TMDB (TOP)

Als erstes die Reihenfolge der verwendeten Scarper ändern mit den grünen Pfeilen auf der rechten Seite, so wie weiter oben schon erklärt

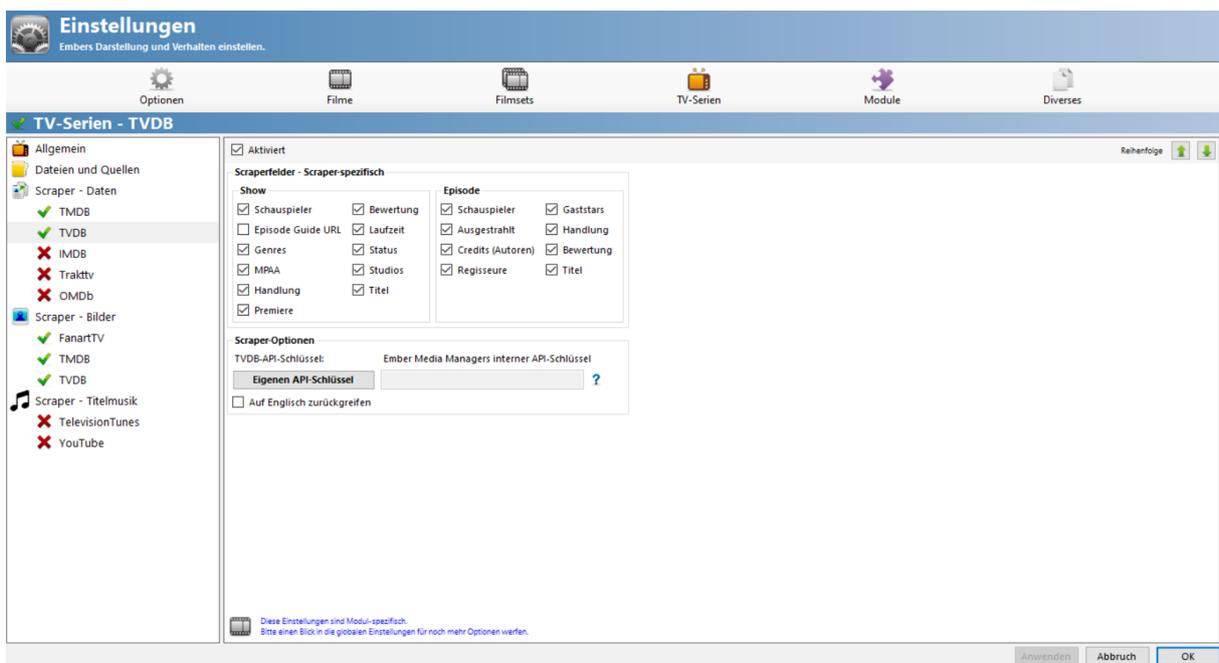


Scraperoptionen

Eigenen API Schlüssel: hier kann ein eigener [TMDB](#) API Key eingetragen werden, wenn vorhanden
 Um einen TMDB API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden.
 Ansonst verwendet Ember den internen API Key

5. Bearbeiten – Einstellungen – TV Serien – Scraper Daten - TVDB

(TOP)

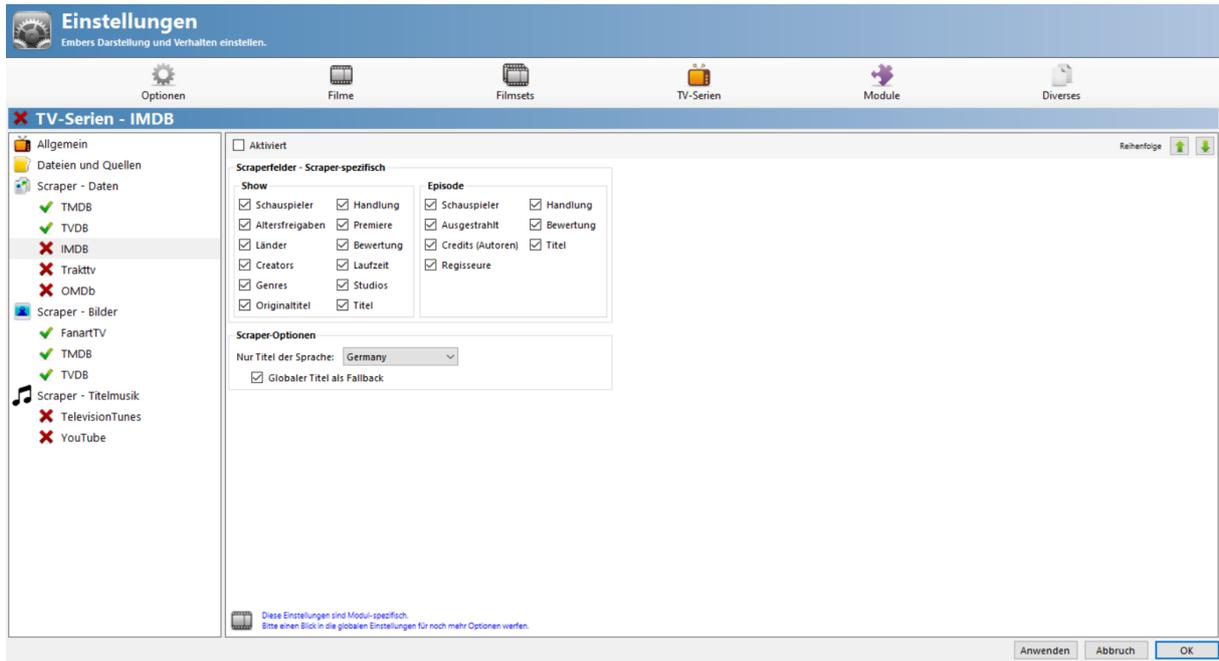


Scraperoptionen

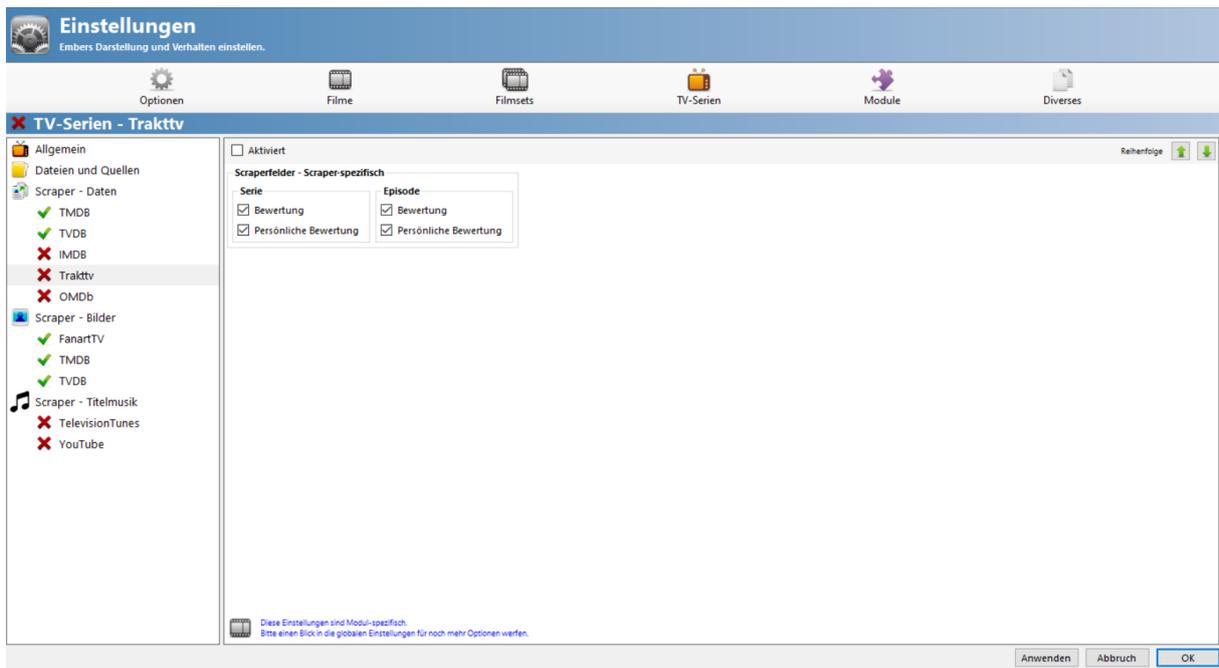
Eigenen API Schlüssel: hier kann ein eigener [TVDB](#) API Key eingetragen werden, wenn vorhanden
 Um einen TVDB API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden

Ansonst verwendet Ember den internen API Key. Im Jahr 2022 wird dieser Dienst kostenpflichtig werden

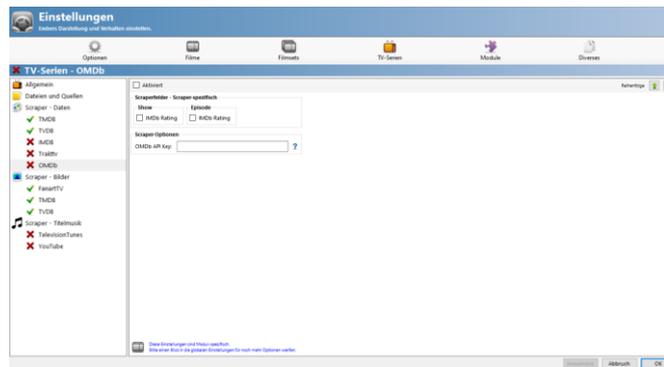
6. Bearbeiten – Einstellungen – TV Serien – Scraper Daten - IMDB (TOP)



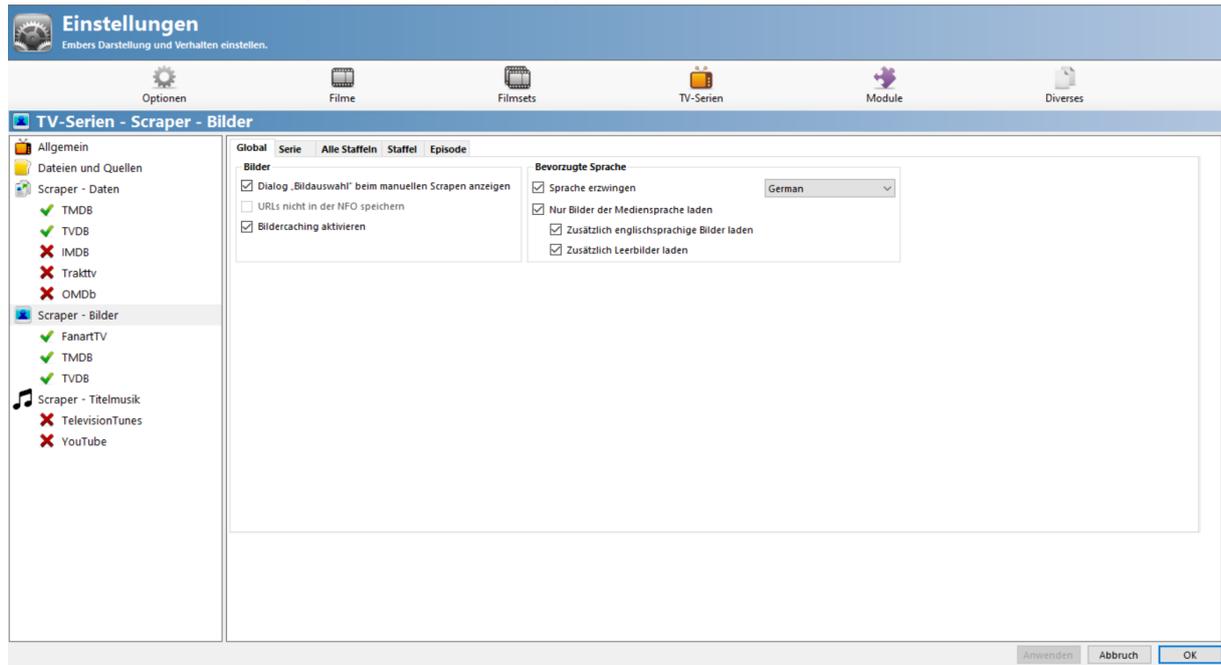
7. Bearbeiten – Einstellungen – TV Serien – Scraper Daten – Trakt.tv/OMDb (TOP)



Soll Trakt verwendet werden, so muss diese Einstellung aktiviert werden. Ein kostenloses Trakt.tv Benutzerkonto ist erforderlich. Anleitung für Trakt.tv ist im [Kap.II/6](#) zu finden



8. Bearbeiten – Einstellungen – TV Serien – Scraper Bilder (TOP)



Bildercaching aktivieren: unbedingt aktivieren, denn die Scraper lassen nur eine bestimmte Anzahl an Anfragen zu

TMDB z.B. limitiert die Calls auf 30 Calls pro 10 Sekunden

Wird dieses Limit überschritten, blockt TMDB für 30 Sekunden die Anfragen
Wenn jetzt eine TVSerien viele Episoden hat, dann dauert das Scrapen ewig

Einstellungen
Embers Darstellung und Verhalten einstellen.

Optionen Filme Filmsets TV-Serien Module Diverses

TV-Serien - Scraper - Bilder

Allgemein
Dateien und Quellen
Scraper - Daten
 TMDb
 TVDB
 IMDb
 Traktv
 OMDb
 Scraper - Bilder
 FanartTV
 TMDb
 TVDB
 Scraper - Titelmusik
 TelevisionTunes
 YouTube

Global Serie Alle Staffeln Staffel Episode

Poster
Bevorzugte Größe: 1000x1500 Nur
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Keyart
Bevorzugte Größe: 1000x1500 Nur
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Banner
Bevorzugte Größe: 1000x185 Nur
 Bevorzugter Typ: Beliebig
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Fanart
Bevorzugte Größe: 1920x1080 Nur
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Extrafanart
Bevorzugte Größe: 1920x1080 Nur
 Limit:
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:
 Vorauswahl im Dialog „Bildauswahl“

CharacterArt
 Vorhandene behalten

ClearArt
 Vorhandene behalten

ClearLogo
 Vorhandene behalten

Landscape
 Vorhandene behalten

Anwenden Abbruch OK

Einstellungen
Embers Darstellung und Verhalten einstellen.

Optionen Filme Filmsets TV-Serien Module Diverses

TV-Serien - Scraper - Bilder

Allgemein
Dateien und Quellen
Scraper - Daten
 TMDb
 TVDB
 IMDb
 Traktv
 OMDb
 Scraper - Bilder
 FanartTV
 TMDb
 TVDB
 Scraper - Titelmusik
 TelevisionTunes
 YouTube

Global Serie Alle Staffeln Staffel Episode

Poster
Bevorzugte Größe: 1000x1500 Nur
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Banner
Bevorzugte Größe: 1000x185 Nur
 Bevorzugter Typ: Beliebig
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Fanart
Bevorzugte Größe: 1920x1080 Nur
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Landscape
 Vorhandene behalten

Einstellungen
Embers Darstellung und Verhalten einstellen.

Optionen Filme Filmsets TV-Serien Module Diverses

TV-Serien - Scraper - Bilder

Allgemein
Dateien und Quellen
Scraper - Daten
 TMDb
 TVDB
 IMDb
 Traktv
 OMDb
 Scraper - Bilder
 FanartTV
 TMDb
 TVDB
 Scraper - Titelmusik
 TelevisionTunes
 YouTube

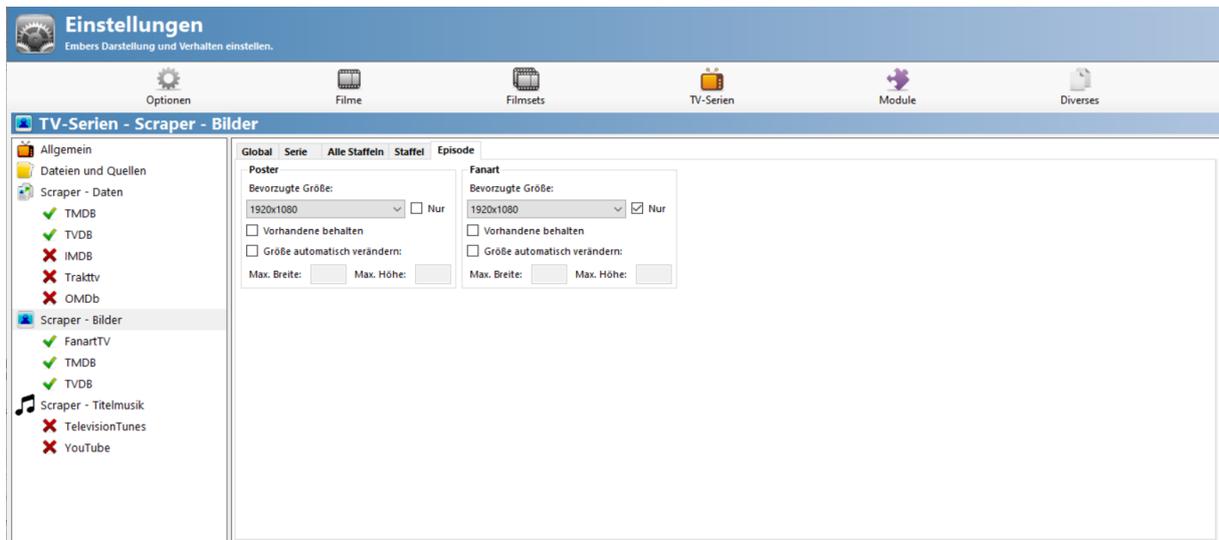
Global Serie Alle Staffeln Staffel Episode

Poster
Bevorzugte Größe: 1000x1500 Nur
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

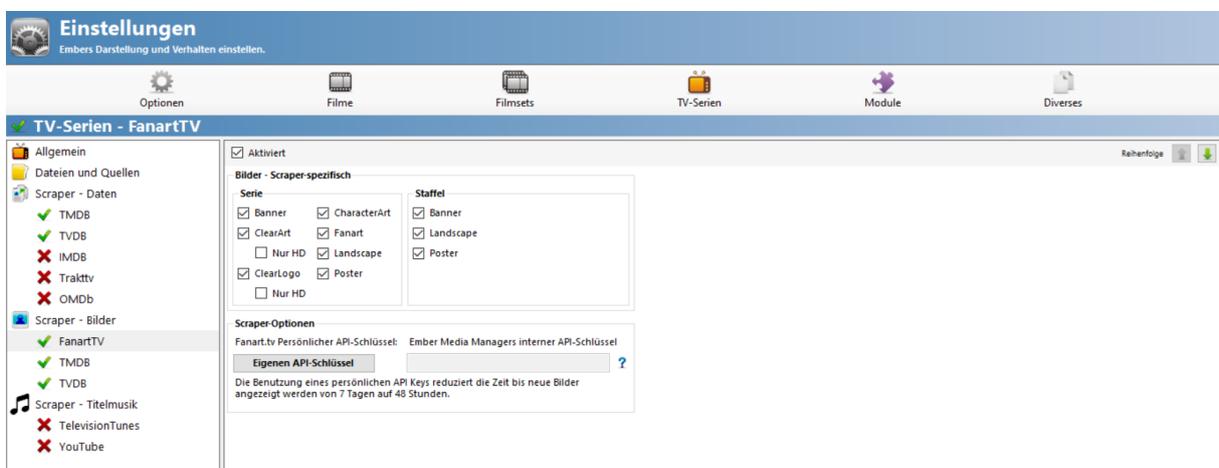
Banner
Bevorzugte Größe: 1000x185 Nur
 Bevorzugter Typ: Beliebig
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Fanart
Bevorzugte Größe: 1920x1080 Nur
 Vorhandene behalten
 Größe automatisch verändern:
 Max. Breite: Max. Höhe:

Landscape
 Vorhandene behalten



9. Bearbeiten – Einstellungen – TV Serien – Scraper Bilder - FanartTV (TOP)



Scraper Optionen:

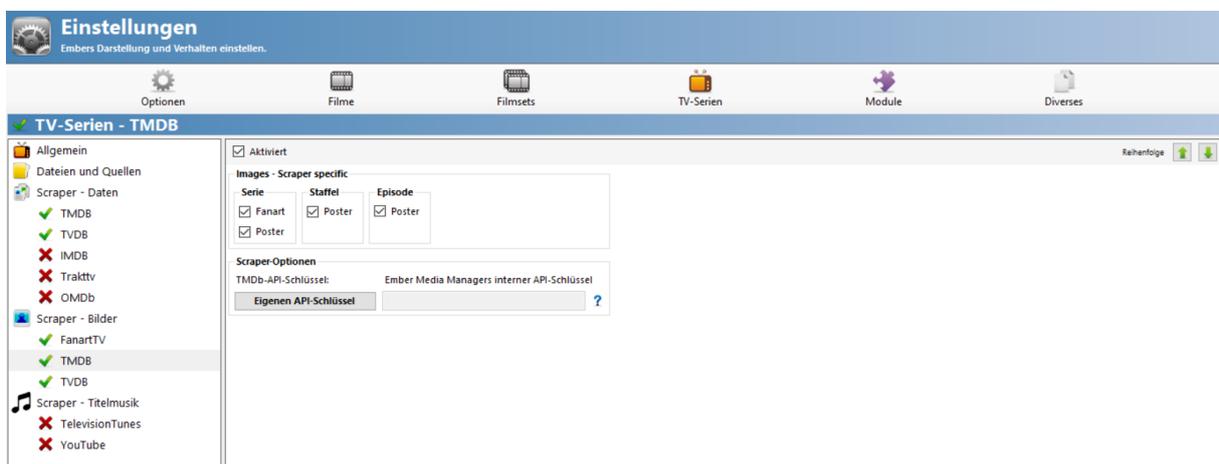
Eigene API Schlüssel: hier kann ein eigener [FanartTV](#) API Key eingetragen werden, wenn vorhanden

Um einen FanartTV API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden

Ansonst verwendet Ember den internen API Key

Anleitung zum Benutzerkonto erstellen ist im [Kap.II/8](#) zu finden

10. Bearbeiten – Einstellungen – TV Serien – Scraper Bilder - TMDB (TOP)

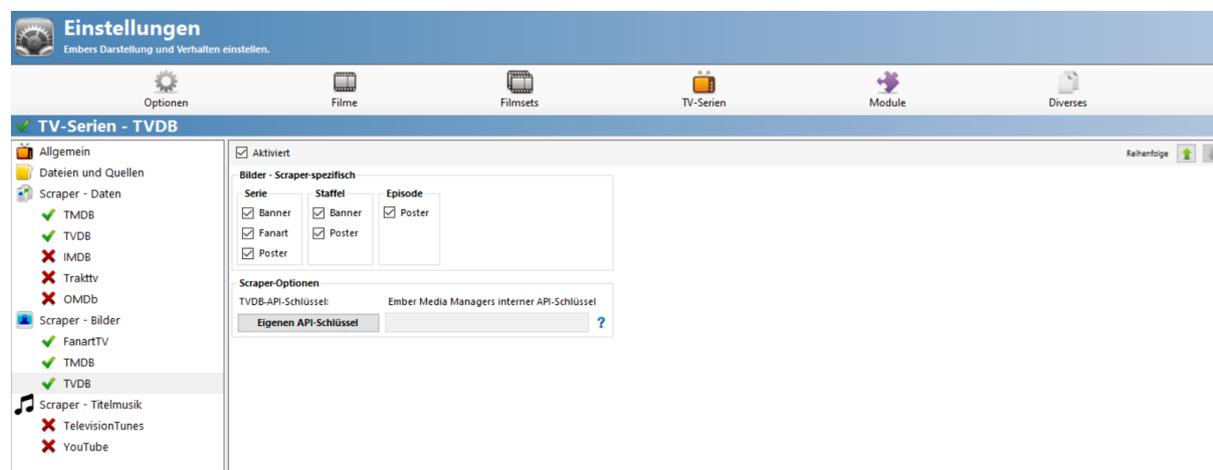


Scraper Optionen:

Eigenen API Schlüssel: hat man sich einen eigenen TMDB API Key erstellt, so kann dieser hier eingetragen werden. Ansonst wird ein integrierter API Key verwendet

Dazu einfach auf den Text *Eigenen API Schlüssel* klicken und das Eingabefeld wird freigeschaltet
Benutzerkonto erstellen siehe [Kap.II/4](#)

11. Bearbeiten – Einstellungen – TV Serien – Scraper Bilder – TVDB (TOP)



Scraperoptionen

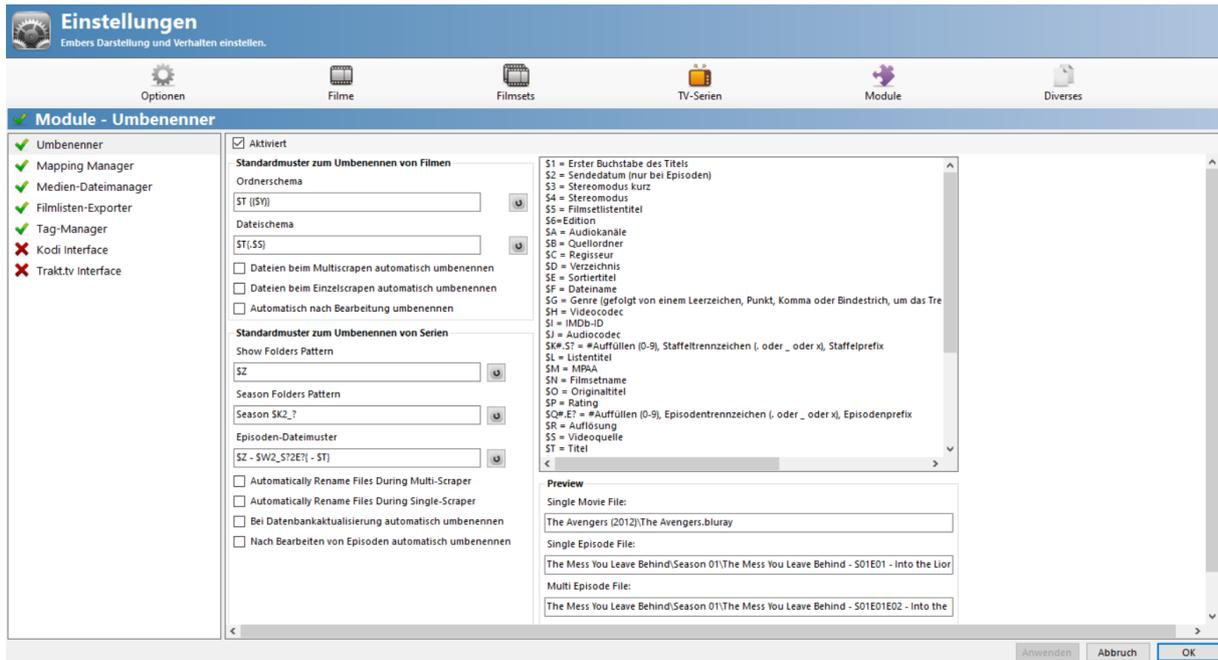
Eigenen API Schlüssel: hier kann ein eigener [TVDB](#) API Key eingetragen werden, wenn vorhanden

Um einen TVDB API Key zu erhalten, muss dort ein Benutzerkonto angelegt werden

Ansonst verwendet Ember den internen API Key. Im Jahr 2022 wird dieser Dienst kostenpflichtig werden

V. Ember Einstellungen- Module (TOP)

1. Bearbeiten – Einstellungen – Module – Umbenener



Der Umbenener ist ein sehr mächtiges und nützliches Werkzeug

Dieses bietet die Möglichkeit, Dateibenennungen wie z.B. "avatar.dl.x264.de.by_emule" verschwinden zu lassen

Ferner entnehmen Kodi und auch die Custom-Skins, die eine oder andere Information nicht aus der NFO, sondern aus dem Dateinamen, wie z.B. die Media Quelle (Source). Damit zeigt das Custom-Skin die richtige DVD oder BD-Hülle an

Also, der Umbenener bietet einfach die Möglichkeit, Filme, Serien, Staffeldorner und Episoden automatisch umzubenennen

Im Umbenener gibt's ein Unmenge an *Tags*, welche Werte aus den NFOs widerspiegeln
Die Aneinanderreihung von Tags nennen wir in Ember *Pattern* (zu deutsch "Schema")

Neben einfachen Tags wie $\$T$ für Titel, gibts noch komplexere wie $\$W2_S?2E?$ für die Staffel- und Episoden-Nummer

Daneben gibt noch *Tag-Attribute* wie { } (Optional), welche die Tags innerhalb der Klammern nur einsetzen, wenn Werte vorhanden oder spezielle Voraussetzungen erfüllt sind

Ein spezieller Tag ist ausserdem noch $\$B$, welcher das Quell-Verzeichnis widerspiegelt

Diesen Basis-Tag braucht man immer dann, wenn bei Film- oder Serien Verzeichnis Pattern, mehr als ein Ordner verwendet wird, also z.B. wenn es einen zusätzlichen Film-Set Ordner gibt

Ember setzt bei allen Ordner Pattern, das abschliessende \ selbst

Bei Filmen gibts es ein Pattern für den Filmordner und ein Pattern für den Dateinamen

Leere Dateinamen Pattern sind nicht erlaubt. Wenn der Dateiname also beibehalten werden soll, dann einfach den Tag $\$F$ einsetzen

Dieses Tag setzt den Wert ein, welcher die Datei VOR dem Umbenennen hat

Beispiele:

Stirb Langsam: (im Ordner) X:\Filme\S\Stirb Langsam 1\Stirb Langsam 1.szenerelase.x264.bl.mkv gehört zu einem Film-Set

Avatar: (im Ordner) X:\Filme\A\Avatar (2009)\Avatar.m2ts

Für das Ordner Schema wäre jetzt folgendes Pattern richtig:

```
$B\{$N\}$T {($Y)} $I $S
```

\$B\ = die in Ember definierte Quelle X:\Filme\S mit \ am Schluss

{ \$N\ } = optional (wenn vorhanden) der Filmset-Titel mit \ am Schluss --> hier Stirb Langsam (Set)\

\$T = Titel des Filmes Stirb Langsam 1

{ (\$Y) } = optional das Jahr (wenn vorhanden)

\$I = IMDB ID

\$S = Video Quelle

Ordner Ergebnis:

X:\Filme\S\Stirb Langsam (Set)\Stirb langsam 1 (1988) tt0095016 bluray\
X:\Filme\A\Avatar (2009) tt0499549 bluray\
X:\Filme\A\Avatar (2009) tt0499549 bluray.m2ts

Für das Dateischema wäre folgendes Pattern richtig:

```
$T {($Y)} $I $S
```

\$T = Titel des Filmes Stirb Langsam 1

{ (\$Y) } = optional (wenn vorhanden) das Jahr

\$I = IMDB ID

\$S = Video Quelle

Dateien-Ergebnis:

Stirb langsam 1 (1988) tt0095016 bluray.mkv

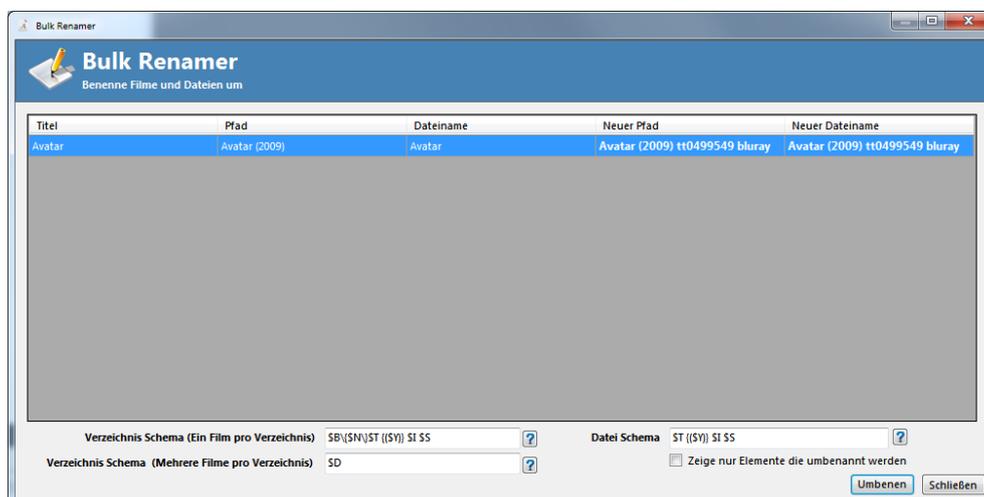
Avatar (2009) tt0499549 bluray.m2ts

Alles zusammen gesetzt ergibt:

X:\Filme\S\Stirb Langsam (Set)\Stirb langsam 1 (1988) tt0095016 bluray\
Stirb langsam 1 (1988) tt0095016 bluray.mkv

X:\Filme\A\Avatar (2009) tt0499549 bluray\
Avatar (2009) tt0499549 bluray.m2ts

Jetzt sind, Verzeichnis und Dateiname identisch, wie von Kodi gewünscht



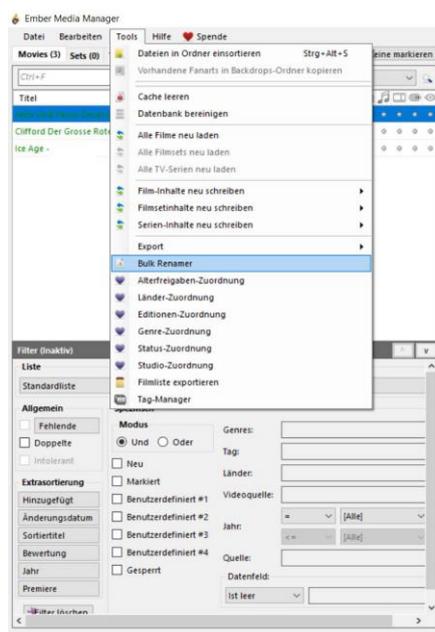
Der *Staffelordner* wird jedoch immer mit der Episode umbenannt
 Es gibt also keine Möglichkeit, nur einen Staffelordner umzubenennen
 Dies geschieht automatisch beim Umbenennen einer Episode

Bei den Serien gibt es ein getrennt Pattern für den Serienordner, für den Staffelordner und für die Episoden

Hier eine Aufstellung und Erklärung der zur Verfügung stehenden Variablen:

\$1	Erster Buchstabe des Titels
\$2	Aired Datum (nur bei Episoden)
\$3	ShortStereomode
\$4	Stereomode
\$A	Audio Kanäle
\$B	Quell-Verzeichnis
\$C	Regisseur
\$D	Verzeichnis
\$E	Sortier-Titel
\$F	Dateiname
\$G	Genre (Gefolgt von einem Leerzeichen Punkt oder Bindestrich um diesen Separator zu ändern)
\$H	Video Codec
\$I	IMDB ID
\$J	Audio Codec
\$K#.S?	# Padding (0-9) Staffel Separator (. oder _ oder x) Staffel Präfix
\$L	Listen-Titel
\$M	MPAA/FSK
\$N	Filmset Name
\$O	Original-Titel
\$P	Rating
\$Q#.E?	# Padding (0-9) Episoden Separator (. oder _ oder x) Episoden Präfix
\$R	Auflösung
\$S	Video Quelle
\$T	Titel
\$U	Land (Gefolgt von einem Leerzeichen, Punkt oder einem Bindestrich um den Separator zu ändern)
\$V	3D (wenn Multiview > 1)
\$W#.S?#.E?	# Padding (0-9) Staffel Separator (. oder _) Staffel Präfix # Padding (0-9) Episoden Separator (. oder _ oder x) Episoden Präfix
\$X.	Leerzeichen werden ersetzt durch .
\$Y	Jahr
\$Z	Serien Titel
{ }	Optional, wenn vorhanden dann wird Variable verwendet
\$?aaa?bbb?	Ersetze aaa mit bbb
\$!	Grosser Anfangsbuchstabe bei jedem Wort
\$.	konvertiere alles in Kleinbuchstaben
-\$	Vorheriges Zeichen entfernen, wenn das nächste Muster keinen Wert hat
-\$+	Nächstes Zeichen entfernen, wenn das vorheriges Muster keinen Wert hat
-\$^	Nächstes und vorheriges Zeichen entfernen, wenn das nächste Muster keinen Wert hat
Erklärung:	
#	Padding 1 – einstellig 2 – zweistellig usw
\$Z \$K2.S?	Serien Titel Buchstabe S Season zweistellig bei Multiseason: Buchstabe S von Season zweistellig Separator . Buchstabe S bis Season zweistellig
\$Z \$\$K2_?	Serien Titel Buchstabe S Season zweistellig bei Multiseason: Buchstabe S von Season zweistellig Separator _ bis Season zweistellig
\$Z \$K2_?	Serien Titel Season zweistellig, wenn Multiseason, dann mit _ anzeigen

Nach dem anpassen der Pattern, aktivieren wir den Reiter *Movies* und starten unter *Tools - Bulk Renamer*



Nachdem die Filmliste erstellt wurde, nochmal eine Kontrolle, ob die neuen Verzeichnisnamen und Dateinamen richtig sind, dann umbenennen. Die Film-Sets werden auch gleich synchronisiert

Sollte dies nicht der Fall sein einfach über *Tools - Alle Filmsets neu laden* synchronisieren

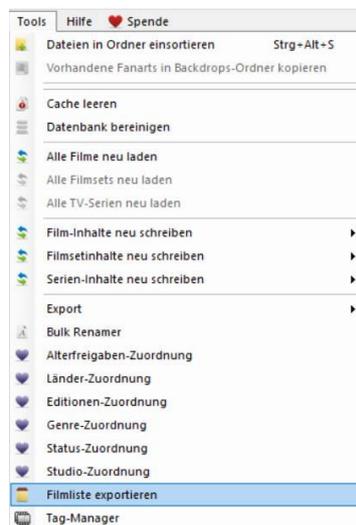
Der gleiche Ablauf mit dem Reiter TVShows

Es ist auch möglich, einzelne Filme/Serien zu markieren und zu filtern
Dann bearbeitet der *Bulk-Renamer* nur die Ausgewählten

Dabei werden die neuen Pfade automatisch in der Ember-DB gespeichert und natürlich auch mit dem Kodi-Interface nach Kodi synchronisiert



Hier ist ein vorhandenes Verzeichnis anzugeben bzw. eines anzulegen
Dieser Pfad wird unter *Tools - Filmliste exportieren* benötigt



2. Bearbeiten – Einstellungen – Module – Kodi Interface (TOP)

Jetzt erst widmen wir uns dem Kodi-Interface, da inzwischen alle Filme aufbereitet und auf aktuellem Stand sind

Dafür muss in Kodi der Webserver aktiviert werden

Kodi starten – Einstellungen (Zahnrad) – Dienste – Steuerung – **Webserver**

Steuerung über HTTP erlauben: *EIN*

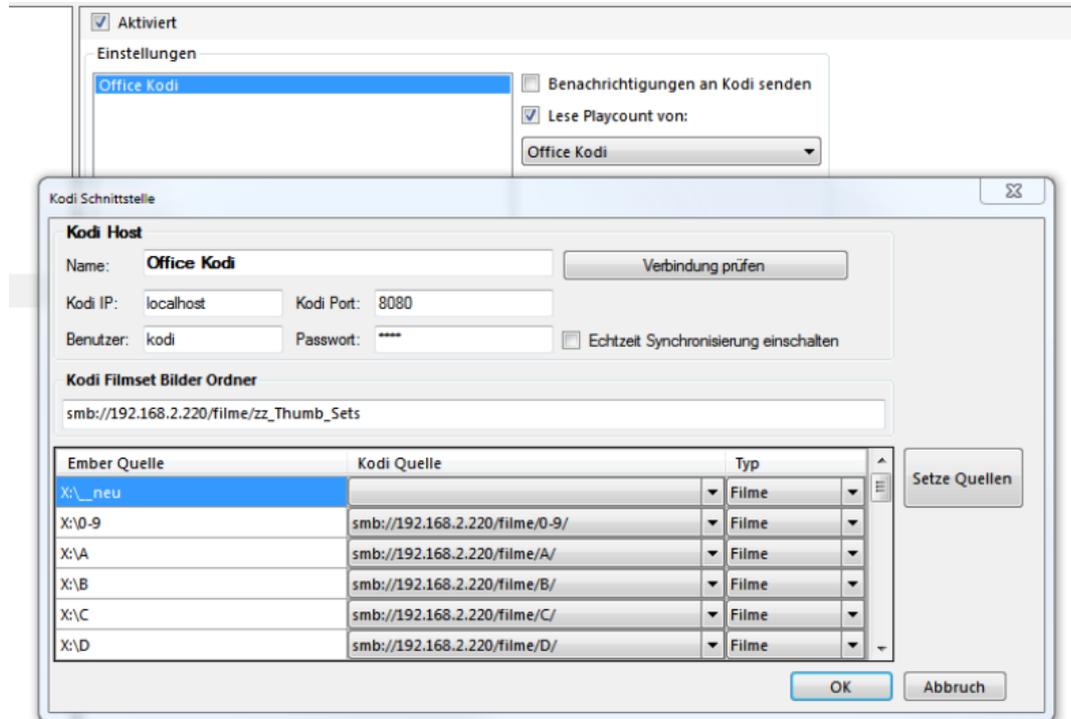
Port: *8080*

Benutzername: *kodi*

Passwort: *selber festlegen*

Kodi darf NICHT geschlossen werden!!

Nun in Ember Einstellungen --> Module --> Kodi Interface aktivieren und Hinzufügen anklicken



Kodi Host

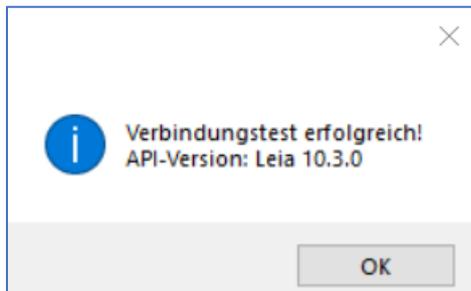
Name: selber festlegen

Kodi IP: nur localhost, wenn Ember und ein Kodi-Client auf dem selben PC installiert sind, ansonsten die IP des Kodi-Clients eingeben

Kodi Port, *Benutzer* und *Passwort*, so wie vorhin in Kodi eingegeben

Verbindung prüfen drücken

Ember meldet wenn eine Verbindung besteht



Kodi Filmset Bilder Ordner

Der Pfad zu den Filmsets für Kodi kann durchaus abweichend vom Pfad für Ember sein

Zwar das selbe Verzeichnis, aber zum Beispiel:

in Ember: X:\zz_Thumb_Sets (als gemappte Netzwerkfreigabe) und

in Kodi: smb://192.168.2.220/filme/zz_Thumb_Sets (als direkte SMB Freigabe)

Im Kodi Filmset Bilder Ordner muss der Pfad so eingetragen werden, wie er bei Kodi gesetzt ist

Setze Quellen

holt die Film-Quellen von diesem Kodi-Client

Anschliessend den Ember Quellen die richtige Kodi Quellen zuweisen, dann noch den Typ Filme oder TV Serien setzen

Wenn alle Quellen zugewiesen sind, nochmals den Button Setze Quellen drücken, damit speichert Ember das Setting (Einstellungen) im Verzeichnis Settings in der Datei Interface.Kodi.xml

OK drücken und fertig ist dieser Client

Achtung:

Der Ember Quelle X:__neu, ist hier keine Kodi Quelle zugewiesen (siehe Bild oben)

In solchen Fällen darf entweder *Echtzeitsynchronisierung einschalten* nicht aktiviert werden oder in diesem Verzeichnis darf kein gescrapter Film gespeichert sein bzw. gescrapt werden

Ansonsten crashed Ember, wenn der Scrapvorgang beendet ist

Wenn in der Zukunft eine weitere Ember-Quelle hinzukommt, MUSS diese auch in Kodi-Quell hinzugefügt werden und nicht vergessen, hier im Kodi-Interface, die neue Ember-Quelle der Kodi-Quelle zuordnen

Wer jetzt eine gemeinsame Kodi-Datenbank auf Basis von MySQL oder MariaSQL hat, ist hier fertig

Diejenigen, die mehrere Kodis nutzen mit jeweils eigener Datenbank, legen hier noch weitere Kodi-Hosts an

Prozedere ist das Gleiche

Wer jetzt noch die Playcounts mit dem Kodi-Interface gleich auslesen und in der NFO speichern möchte, aktiviert *Gesehen-Status abrufen von:* und wählt einen Kodi-Host aus (in unserem Beispiel Office Kodi)

Der Playcount wird in der Ember-db und in der NFO-Datei gespeichert

Funktionsweise des Kodi-Interface

Das Kodi-Interface kann die Aufträge nur abarbeiten, wenn **mindestens ein** Kodi-Host **gestartet** ist

Am besten natürlich alle, denn sonst sind die Daten nicht mehr übereinstimmend bei den einzelnen Hosts

Wir benötigen für den Sync immer die entsprechende ID von Kodi, damit wir die Daten syncen können

Dieses Beispiel bezieht sich auf die Synchronisation eines Filmes, funktioniert aber bei Filmsets, Serien, Staffeln und Episoden genauso

Alle Kodi Interface Aufträge (Tasks) laufen parallel, deshalb wird im Log auch mal was zwischen den Kodi Interface Logs geschrieben

Playcount kann aber nur an einen Host übermittelt werden

Der Syncvorgang wird immer automatisch gestartet:

wenn ein Objekt in die Ember-DB gespeichert (nach dem Scrapen, dem Editieren usw.) und für alle Kodi Interface Hosts ausgeführt wird und die Echtzeitsynchronisierung eingeschaltet ist

Wenn *Echtzeitsynchronisierung einschalten* nicht aktiviert ist, muss über das Kontextmenü --> Kodi Interface --> Synchronisieren, der Vorgang manuell angestoßen werden

Es wird Kodi sogar mitgeteilt, dass die gecachten Bilder ungültig sind und Kodi diese neu Zwischenspeichern (Cachen) soll

Das funktioniert zum Teil nicht sofort, da Kodi das im Background macht. Aber es funktioniert spätestens nach dem Neustart von Kodi

Es sollte aber auch genügen, wenn ihr in Kodi von Filme in z.B. TV wechselt und dann wieder zurück zu Filme

Im Beispiel dieser Film benützt:

X:\Filme\A\Avatar (2009) tt0499549 bluray\Avatar (2009) tt0499549 bluray.m2ts

Vor jedem Synchronisieren wird der Pfad anhand der Kodi Interface Host Einstellungen gemappt, z.B.

X:\Filme\A\Avatar (2009) tt0499549 bluray\Avatar (2009) tt0499549 bluray.m2ts =>

smb://192.168.2.220/filme\A\Avatar (2009) tt0499549 bluray\Avatar (2009) tt0499549 bluray.m2ts

Ermitteln der ID

1. SearchMovieByPath:

Path.startwith = smb://192.168.2.220/filme\A\Avatar (2009) tt0499549 bluray => wenn keine ID oder mehrere ID's, dann nächster Schritt, ansonsten Syncvorgang starten

2. SearchMovieByFilename:

Filename.endswith = Avatar (2009) tt0499549 bluray.m2ts => wenn keine ID oder mehrere ID's, dann nächster Schritt, ansonsten Syncvorgang starten

3. Film immer noch nicht in der Kodi-DB gefunden, evtl. ist er neu und noch nicht hinzugefügt worden

ScanVideoPath (in Kodi):

VideoLibrary.Scan(smb://192.168.2.220/filme\A\Avatar (2009) tt0499549 bluray)

4. Warte 1000ms, in der Hoffnung, dass Kodi bis dahin den Pfad durchsucht und den Film erstmalig in die eigene DB hinzugefügt hat

Somit sollte nun die ID auffindbar sein

Zu diesem Zeitpunkt hat Kodi bereits die Poster und Fanarts eingelesen, die zusätzlichen Bilder aber noch nicht (Kodi erkennt die ja nicht)

Es gibt in der JSON-API die Möglichkeit solange zu warten, bis Kodi effektiv fertig ist, aber das funktioniert irgendwie noch nicht ganz ohne Fehler

Deshalb warten wir einfach mal 1000ms

5. SearchMovieByPath:

Path.startwith = smb://192.168.2.220/filme\A\Avatar (2009) tt0499549 bluray => wenn keine ID oder mehrere ID's, dann Abbruch, ansonsten Syncvorgang starten

Syncvorgang

1. Laden der aktuellen Daten aus Kodi mittels ID
2. Wenn "Gesehen-Status abrufen von" aktiv ist für diesen Host und Ember Gesehen-Status weniger ist als der Kodi Gesehen-Status, dann wird der Gesehen-Status von Kodi übernommen und der Film in Ember muss erneut gespeichert werden
3. Eigentlicher Synchronisationsvorgang inkl. zusätzlicher Bilder wie ClearArt, DiscArt usw.

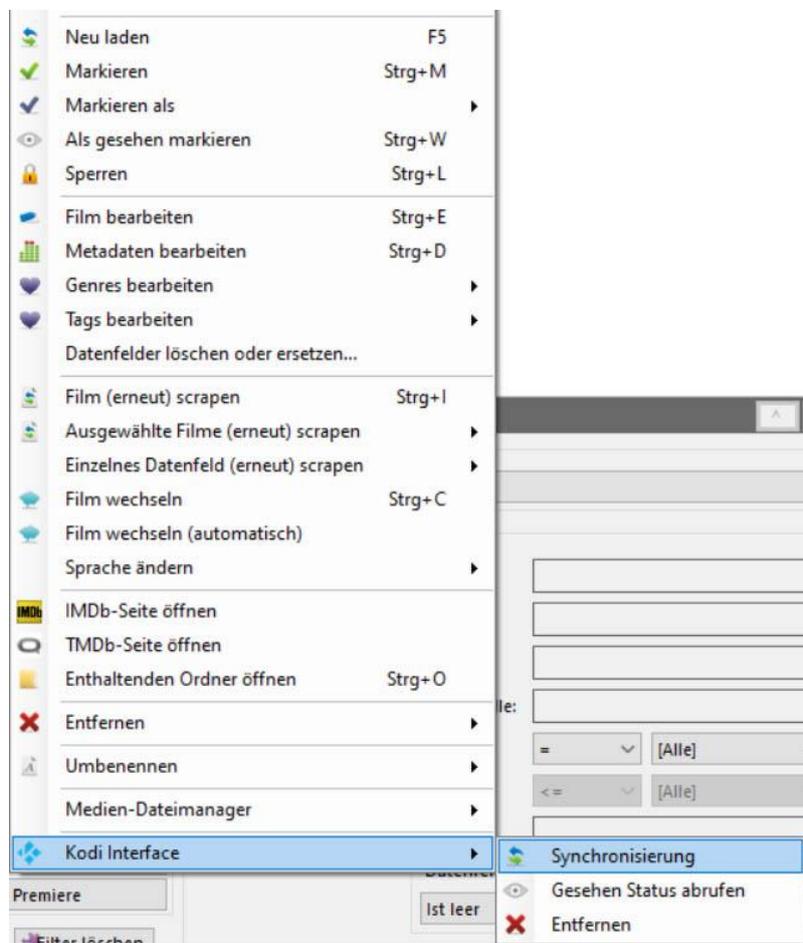
4. Ermitteln und löschen aller Textures (Bilder Cache) für diesen Film anhand des Pfades:
Path.startwith = smb://192.168.2.220/filme\A\Avatar (2009) tt0499549 bluray
5. Wenn *“Benachrichtigung an Kodi Senden“* aktiviert, dann wird eine Meldung an Kodi gesendet, dass das Objekt neu hinzugefügt oder geupdatet worden ist
6. Wenn muss erneut gespeichert werden = true, dann wird der Film mit *“Gesehen-Status“* und *“Zuletzt Gespielt“* aus Kodi erneut in die Ember-DB gespeichert, was wiederum einen neuen Sync-Task für alle RealTimeSync Hosts auslöst

Nach diesem erneuten Synchronisieren sind dann alle Kodi Hosts auf dem gleichen Gesehen-Status (PlayCount) und Zuletzt Gespielt (LastPlayed) Stand, auch wenn keine gemeinsame Kodi-DB verwendet wird

Verschiedene Möglichkeiten Ember und Kodi zu synchronisieren über das Kontextmenü

Am Ember Startbildschirm:

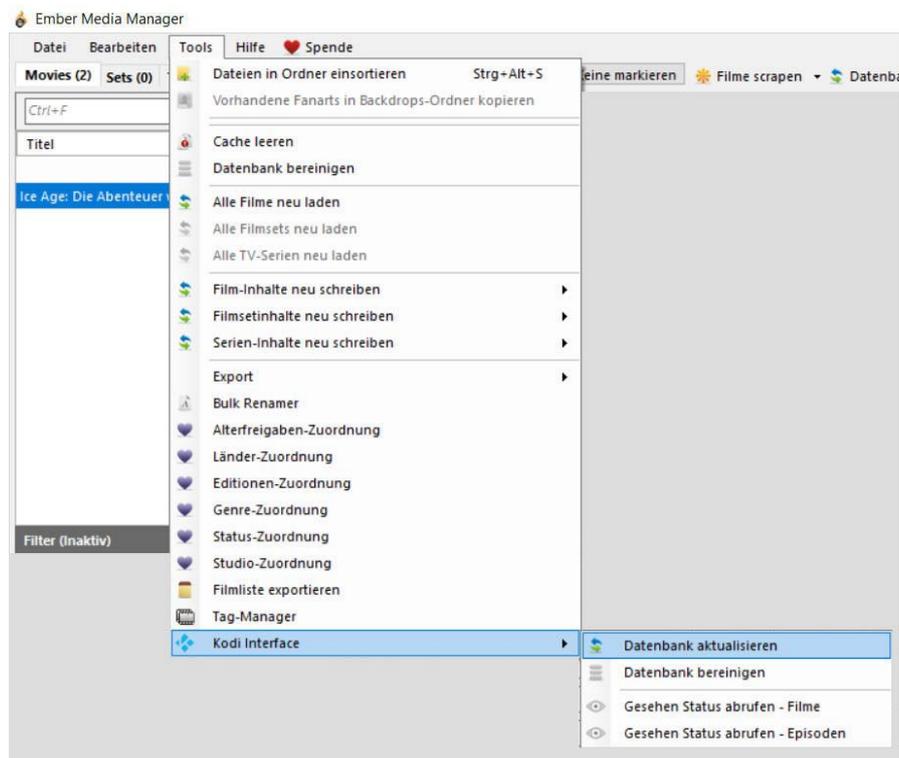
einen, mehrere oder auch alle Filme markieren und dann Kontextmenü mit der rechten Maustaste öffnen



Jetzt Synchronisierung wählen und starten

Hier werden ALLE Bilder an Kodi übertragen und in die Kodi-DB eingetragen

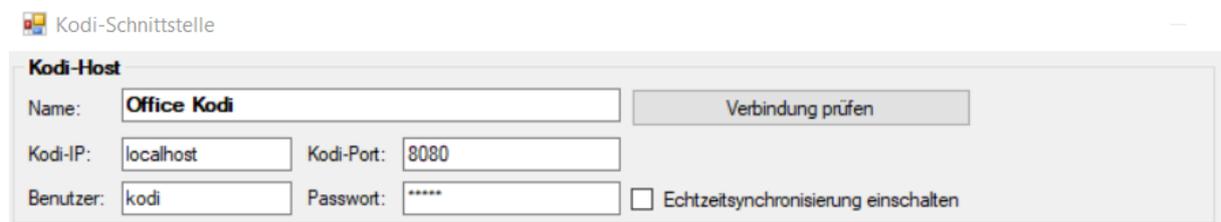
Weitere Möglichkeit besteht über **Tools**



Hier sendet Ember an Kodi den Befehl, mit der Kodi-eigene Funktion die Bibliothek zu aktualisieren bzw. zu säubern

Hier holt sich Kodi die Informationen selbst, so wie früher, direkt vom Datenträger - nicht und nichts über Ember!

Erklärung zu Echtzeitsynchronisierung einschalten

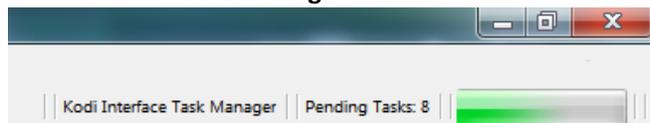


Wenn diese Option aktiviert ist, dann überträgt Ember umgehend bei folgenden Aktionen:

Wenn bei einem Kodi-Host *Echtzeitsynchronisierung einschalten* aktiviert ist, dann werden alle Daten und Bilder nach jedem Scrapen oder Editieren direkt an Kodi übermittelt

Diese Funktionen erst aktivieren, wenn Ihr alles fertig eingerichtet habt in Ember

Kodi-Interface Task Manager



Während eines Synchronisationsvorgang ist der Task Manager aktiv

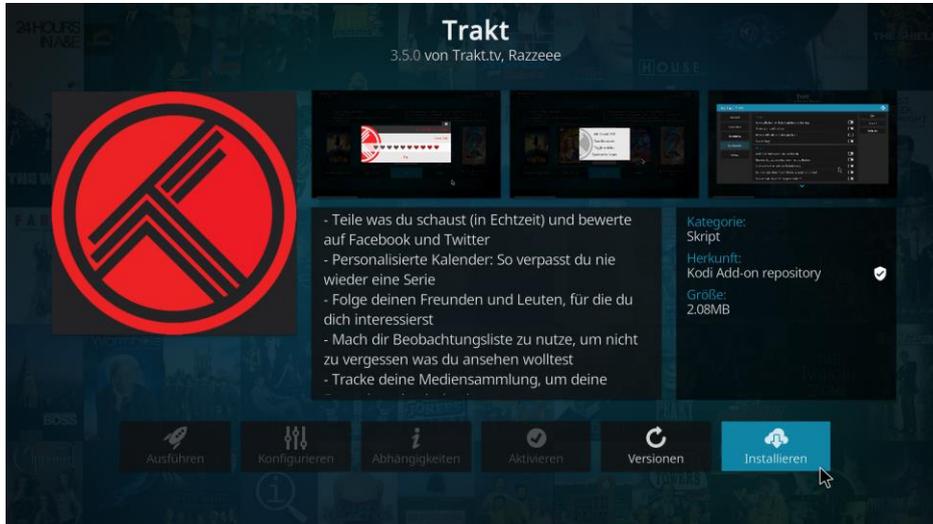
Während der Task Manager einen Task nach dem anderen abarbeitet, kann mit Ember normal weiter gearbeitet werden

Ember aber erst beenden, wenn alle Tasks fertig sind, ansonsten sind nicht alle Aufträge übertragen, da Ember Task für Task einzeln abarbeitet und nicht in einem Stück übertragen werden kann

3. Bearbeiten – Einstellungen – Module – Trakt.tv (TOP)

Das Original Trakt Addon (Scrobbler) lässt sich über das Kodi-Repository (Standard in Kodi) installieren

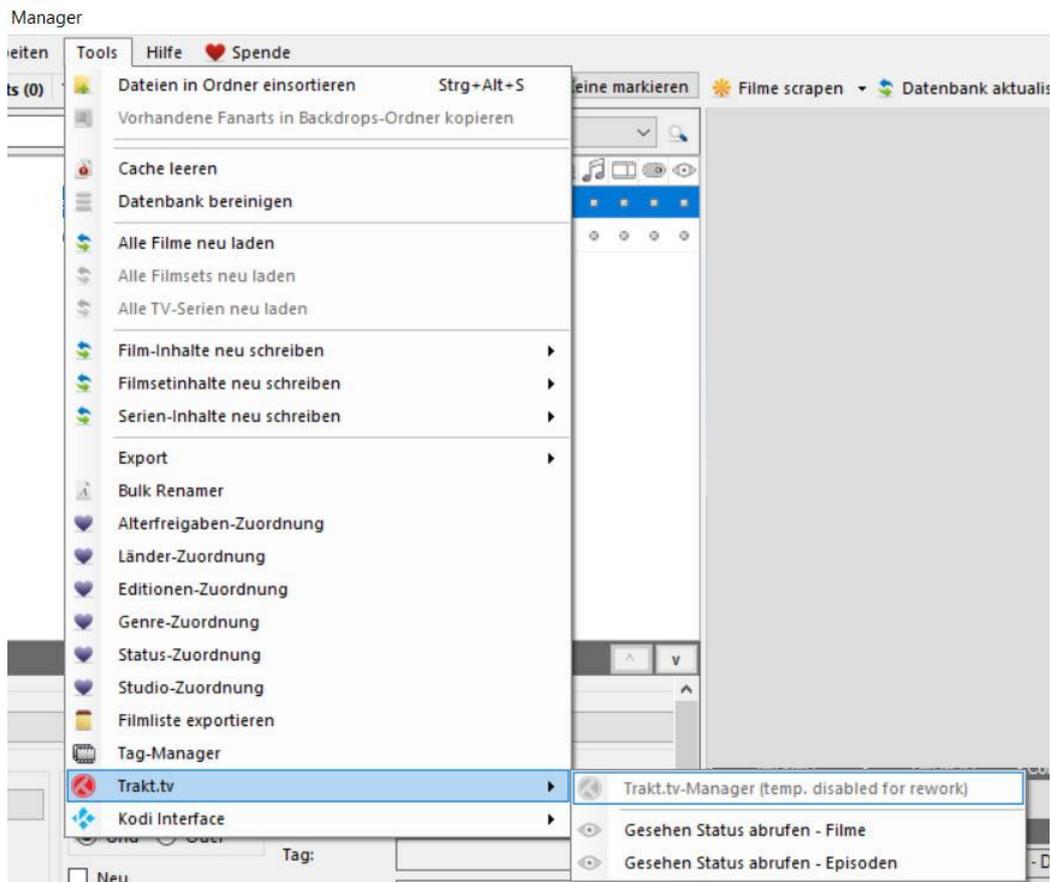
Mehr Informationen dazu gibts [hier](#) oder auf [Github](#)



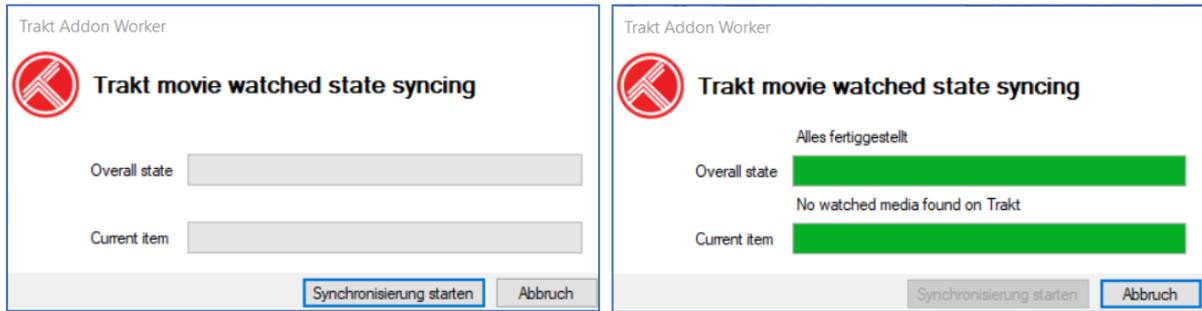
Trakt eignet sich hervorragend um *Gesehen-Status/Fortsetzungspunkt* und *Zuletzt Gespielt* Informationen mit Ember synchron zu halten

Trakt Konto Erstellen und mit Ember verknüpfen siehe [Kap.II/6](#)

Im Anschluss in Ember am Startbildschirm auf *Tools – Trakt.tv*



Gesehen Status abrufen ist nun für Filme/Episoden möglich, nach Auswahl auf *Synchronisierung starten*



Filme und Episoden sind in zwei getrennten Schritten zu aktualisieren

Wollt Ihr in Kodi nun eine komplette Serie oder Staffel zu einer Trakt Liste hinzufügen, geht das wie folgt:

Listen, können nach Anmeldung auf der Trakt Homepage, erstellt werden

Komplette Serie oder Staffel suchen, Kontextmenü öffnen- Trakt Manager

In dem sich nun öffnenden Fenster, könnt Ihr eine Liste wählen, wo die Serie/ Staffel gespeichert werden soll

Wollt Ihr in Kodi nun eine Staffel oder Episode in Trakt als gesehen markieren, geht das wie folgt:

Serie wählen - Staffel wählen - Episode wählen - Kontextmenü öffnen - Watched in Trakt (In Trakt angesehen) klicken

Auf die gleiche Weise, könnt Ihr auch die komplette Serie als "In Trakt gesehen" wählen

Wollt Ihr in Kodi nun einen Film zu einer Trakt Liste hinzufügen, geht das wie folgt:

Das Vorgehen ist dabei gleich wie bei den Serien

Film wählen - Kontextmenü öffnen - Trakt manager

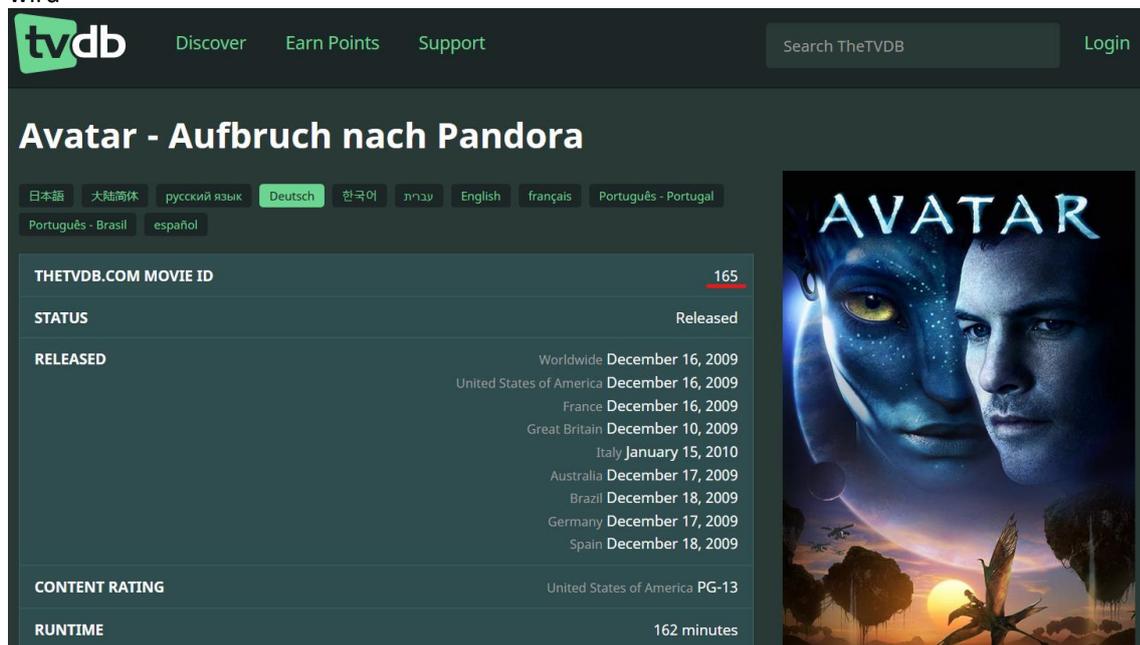
In dem sich nun öffnenden Fenster, könnt Ihr eine Liste wählen, wo der Film gespeichert werden soll

Wollt Ihr in Kodi nun einen Film in Trakt als gesehen markieren, geht das wie folgt:

Film wählen - Kontext Menü öffnen - watched in Trakt (In Trakt angesehen) wählen

Es können auf der Homepage von Trakt.tv, Filme&Serien selbst zur Trakt Datenbank hinzugefügt/nachgetragen werden, wenn diese nicht vorhanden sind

Dazu wird die Film oder Serien ID Nummer benötigt, welche Euch auf <https://thetvdb.com/> angezeigt wird



THETVDB.COM MOVIE ID	165
STATUS	Released
RELEASED	Worldwide December 16, 2009 United States of America December 16, 2009 France December 16, 2009 Great Britain December 10, 2009 Italy January 15, 2010 Australia December 17, 2009 Brazil December 18, 2009 Germany December 17, 2009 Spain December 18, 2009
CONTENT RATING	United States of America PG-13
RUNTIME	162 minutes

Am unteren Ende der Trakt Homepage, findet Ihr das Import Menü für Filme (Import Movie) und Serien (Import TV Show), hier muß die ID Nummer eingeben

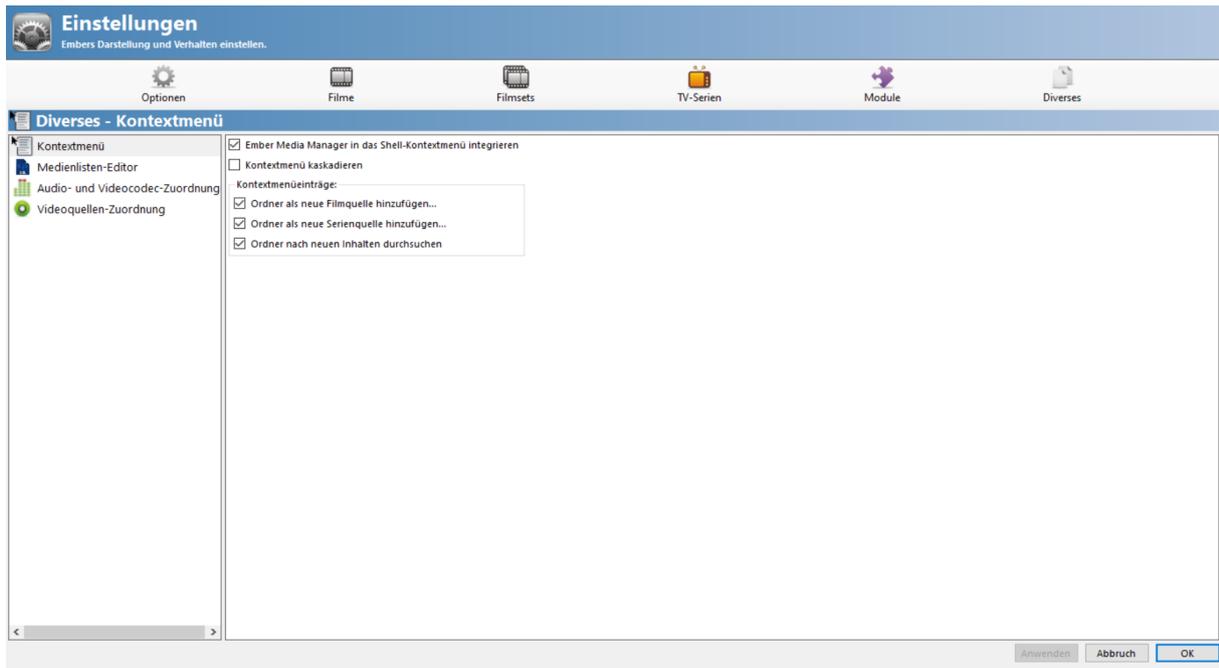


Um diese Menü Punkte zu sehen, muß man sich vorher Anmelden

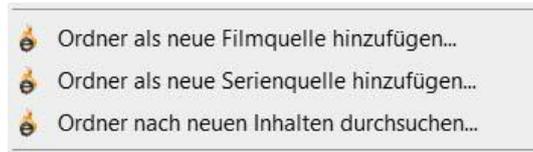
VI. Ember Einstellungen – Diverses (TOP)

1. Bearbeiten – Einstellungen – Diverses - Kontextmenü

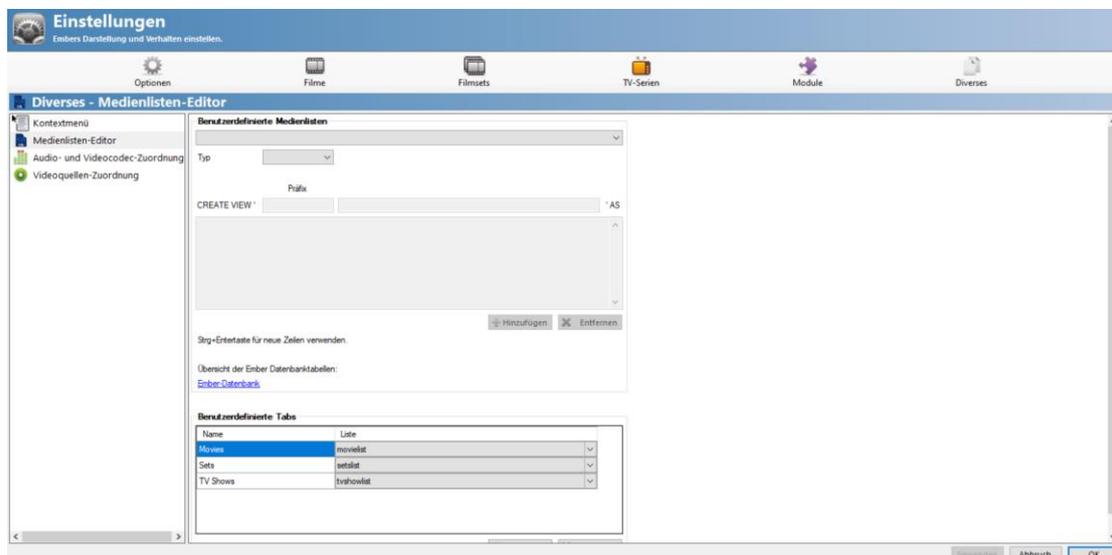
Ember als Administartor (*Als Administrator ausführen...*) starten, damit sich die Einträge in das Shell Kontextmenü integrieren können



Auswählen, was später im Kontext Menü direkt Anwählbar sein soll



2. Bearbeiten – Einstellungen – Diverses – Medienlisten Editor (TOP)



Einfach mal ein Beispiel, was hier so alles geht

Wir Erstellen einen weiteren *Benutzerdefinierte Tabs*, der zusätzlich die noch nicht gesehene Filme anzeigt

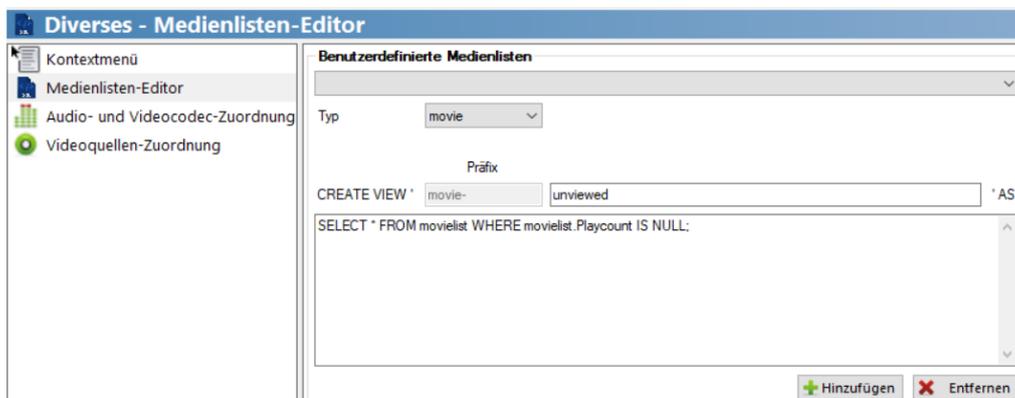
Benutzerdefinierte Medien Listen

Unter Typ 'movie' auswählen --> jetzt kann eine eigene Datenbank Ansicht angelegt werden

Bei CREATE VIEW wird in dem Feld ein einmaliger Name vergeben, hier z.B: unviewed

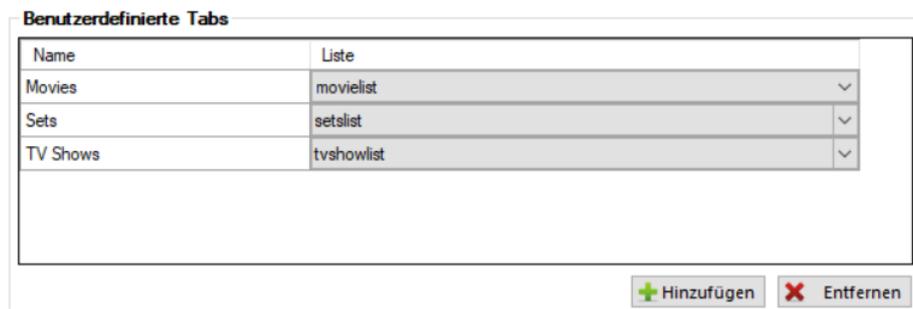
Dann noch diesen SQL-Code eingefügt:

```
SELECT *  
FROM  
  movielist  
WHERE  
  movielist.Playcount IS NULL;
```

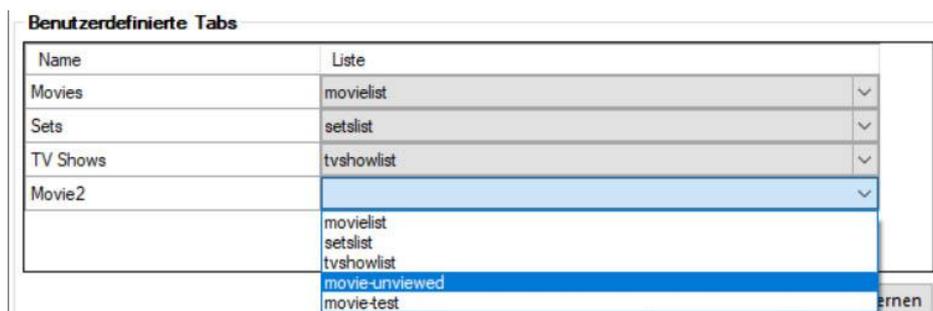


Button Hinzufügen und schon ist ein DB-View erstellt

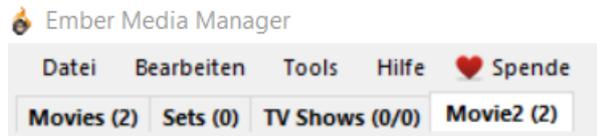
Jetzt wird ein *Benutzerdefinierter Tab* angelegt und dabei diese SQL-Abfrage verwendet



Dazu Hinzufügen klicken und in Spalte *Name* einen entsprechenden Namen vergeben z.B.Movie2
Dann Spalte *Liste* --> movie-unviewed auswählen, auf Anwenden und fertig



Diese Aktion erfordert einen Ember-Neustart



Die hier erstellten Ansichten sind an vielen Stellen später verwendbar z.B. bei Tools, Filmliste Exportieren usw.

3. Bearbeiten – Einstellungen – Diverses – Videoquellen Zuordnung (TOP)



Quellen

Zusätze, welche im Dateinamen auf das Quellmaterial hinweisen

Bevor man hier etwas einträgt, erstmal auf regex101.com testen, ob es funktioniert

Videoquelle nach Dateierweiterung zuordnen

Dateierweiterungen woraus das Quellmaterial ersichtlich ist, muss mit führendem Punkt eingegeben werden

Achtung: beide unterscheiden nach Groß- und Kleinschreibung

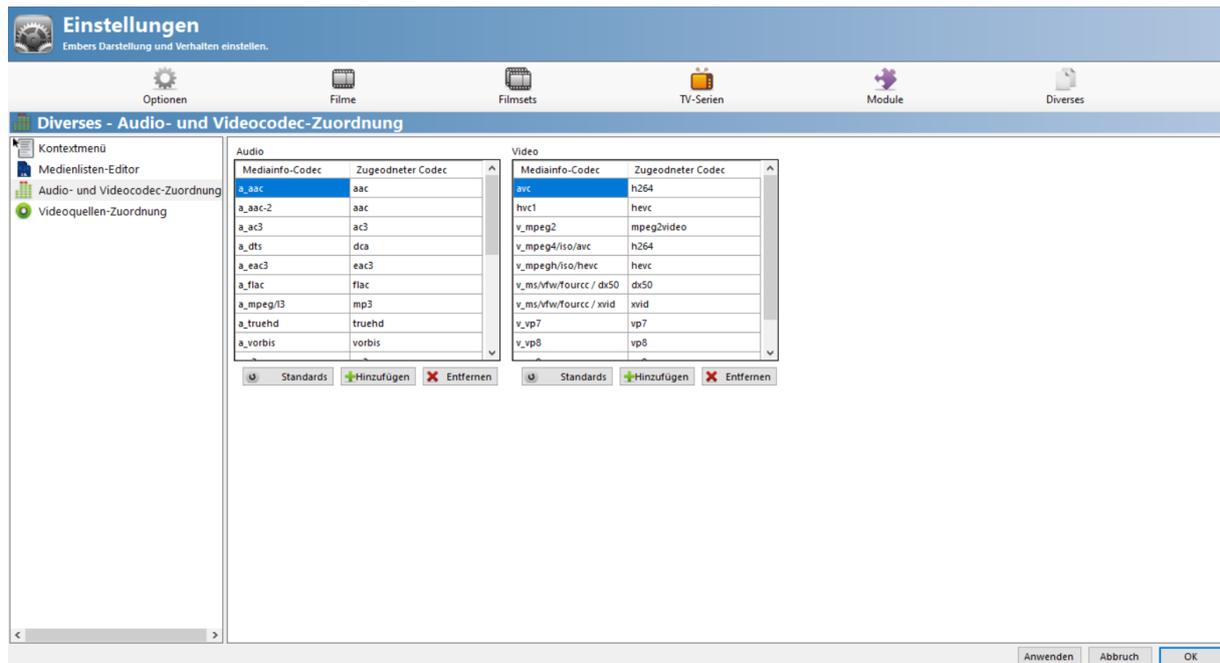
Beispiele:

Datei Erweiterung	Quellenname
.m2ts	bluray
.M2TS	bluray
.ts	bluray
.iso	dvd
.ISO	dvd
.mpg	dvd
.MPG	dvd
.avi	dvd

Eigentlich sollte nicht anhand der Dateierweiterungen die Qualität des Filmes bestimmen werden, sondern anhand des Dateinamens

z.B. MKV ist ein Containerformat und kann vom SD-Material bis zum UHD-Material alles aufnehmen
Also nur im "Notfall" dies einsetzen

4. Bearbeiten – Einstellungen – Diverses – Audio/Videocodec Zuordnung (TOP)



Gleich wie bei Media Source, auch hier ist auf Groß- und Kleinschreibung zu achten

VIII. Ember Datenbank füllen (TOP)

Jetzt wird es Zeit die Ember-Datenbank zu füllen

Zum Beispiel Reiter *Movies* auswählen und *Datenbank aktualisieren* drücken



Es wird nun jener Ordner gescannt, welcher am Anfang dieser Anleitung angelegt wurde und wo sich die Filme bzw. Serien befinden

Jetzt warten bis Ember alle Daten in seiner Datenbank gespeichert hat

Das kann schon einige Zeit in Anspruch nehmen, je nachdem wie groß das Film/Serien Angebot (Datenbestand/Datenbank) ist

Ember erkennt zwar die Film-Sets und legt diese auch an, liest aber den Pfad, worin die Set-Informationen und Bilder abgespeichert sind nicht aus

Deshalb ist eine weitere Aktion einmalig notwendig

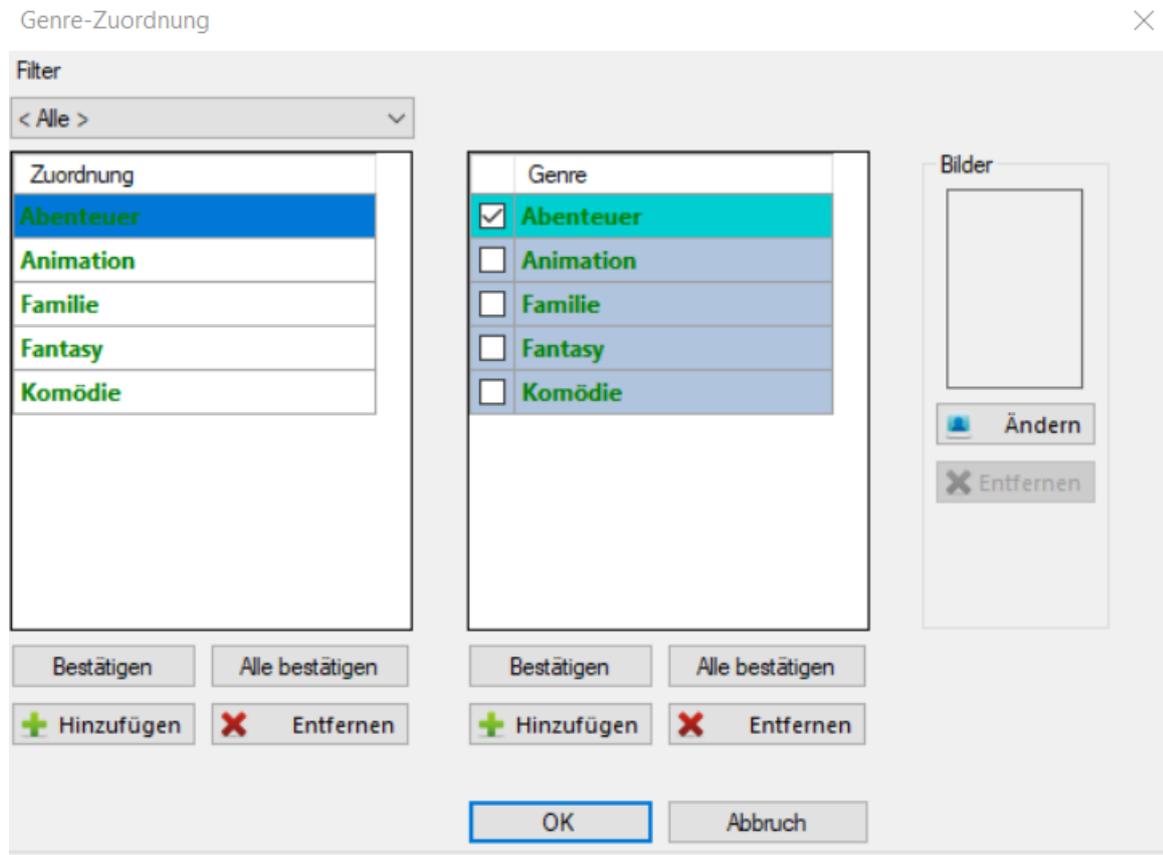
Tools- Alle Filmsets neu laden holt sich diese Infos in die Datenbank

Jetzt sollten überall grüne Haken sein und die gescrapten Bilder angezeigt werden

VIV. Genre Manager (TOP)

Im vorigen Schritt wurde die Ember-Datenbank erstmalig gefüllt und erst jetzt ist der Genre Manager nutzbar

Tools --> Genre Zuordnung



Bei Genre Zuordnung können einfach Genres entfernt wurden

Dazu gibts zwei Spalten: Link die Spalte "Zuordnung", rechts die Spalte "Genre".

Das Ganze funktioniert so:

Linke Spalte ist das Ergebnis aus der Ember-Datenbank, rechts was dann später tatsächlich gespeichert werden soll

Wird die Genre Zuordnung das erste Mal geöffnet wird, stehen links die selben Namen wie rechts

Beispiel (siehe Bild oben):

wenn links z.B. Abenteuer gewählt wird, sollte rechts Abenteuer ausgewählt sein und einen Haken haben

Das heisst nun, wenn der Scraper Abenteuer geliefert hat, wird auch Abenteuer gespeichert

Aus diesem Grund müssen/sollen erstmal beide Spalten die selben Werte haben

Nach und nach wird beim Scrapen die linke **und** rechte Tabelle erweitert

Wie diese Genre Zuordnungs Tabelle funktioniert wird nun erklärt

Wird z.B. das Genre "Splatter" (als Beispiel zum ersten Mal) gescrapt, macht Ember folgendes:

- Suche unter *Zuordnung* nach dem Namen "Splatter" --> kein Eintrag gefunden
- erstelle neue *Zuordnung* "Splatter" und verweise auf Genre "Splatter"
- prüfe, ob *Genre* "Splatter" bereits existiert --> falls nicht, erstelle *Genre* "Splatter"

Danach könnt ihr wieder in die Einstellungen gehen, wo nun links die *Zuordnung* und rechts das *Genre* "Splatter" neu aufgelistet ist

Wenn das für euch passt dann müsst ihr nichts weiter machen

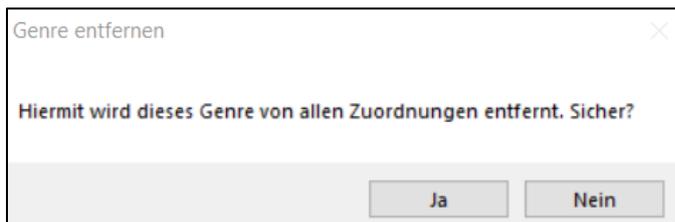
Ihr könnt entweder einzeln oder auch alle neuen Einträge übernehmen

Beim Übernehmen passiert nichts anderes, als dass die Farbe geändert wird
Die Farbe dient wirklich nur als Indikator für neue Einträge

Wenn ihr mit dem Ergebnis nicht einverstanden seid, dann gibts zwei Möglichkeiten:

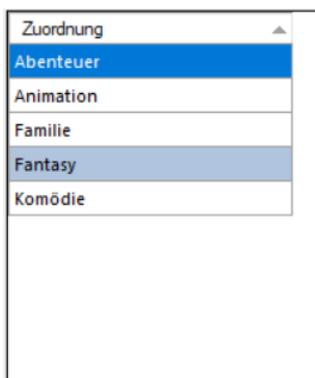
- *Genre* "Splatter" soll ignoriert werden
- *Genre* "Splatter" soll geändert bzw. umgewandelt werden

Wenn ihr das *Genre* "Splatter" ignorieren wollt, dann könnt ihr in der rechten Spalte einfach das *Genre* löschen (Entfernen)

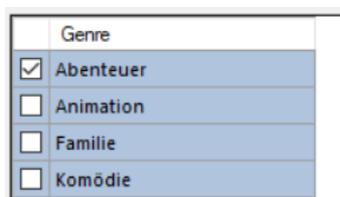


Damit hat *Genre* "Splatter" in der linken Spalte keine *Zuordnung* mehr und wird nach dem Scrapen entfernt

Bei *Zuordnungen* links, die keine *Zuordnung* haben, wird diese Zeile grau hinterlegt



Wenn ein *Genre* in der rechten Spalte *Genre* grau hinterlegt ist, dann bedeutet das, dass diesem *Genre* noch kein Bild zugeordnet ist



Wenn ihr nun wollt, dass anstelle von "Splatter" einfach "Horror" und "Blutorgie" verwendet wird, dann klickt links (Zuordnung) auf den Eintrag "Splatter" und wählt rechts (Genre) aus, was anstelle von "Splatter" verwendet werden soll

Hierbei könnt ihr beliebig viele Genres per Haken setzen auswählen oder auch eine neues Genre anlegen Diese Genre, die nicht eins zu eins über einstimmen und eine neues Genre zugeordnet bekommen haben, sind in der linken Spalte an der schwarzen Schrift zu erkennen

Links gibt es ebenso den Button Entfernen

Dieser Button macht eigentlich keinen Sinn, denn Ember wird links immer wieder den Genreeintrag hinzufügen, wenn er beim Scrapen auftaucht

Es macht also nur Sinn, dem Eintrag links die entsprechende Zuweisungen rechts zu entfernen

Wie oben beschrieben wird das Genre dann zukünftig ignoriert. Zum Schluss mit OK bestätigen

Wird im Genre Manager etwas anpasst, dann wird das erstmal nicht auf die bestehenden Filmen und Serien in der Datenbank übernommen!

Dazu muss nach dem Ändern, unter *Tools - Datenbank bereinigen* gedrückt werden

Hierbei wird die Genre-DB aufgeräumt, die Filme und Serien geändert und die NFOs neu erstellt

Übrigens, *Datenbank bereinigen* kann je nach Datenbestand dauern, da ja jede NFO neu geschrieben wird Also nicht unruhig werden

Wenn es fertig ist, sollte ein Meldungsfenster erscheinen mit: *Alles fertiggestellt*

Zum Abschluss fehlen jetzt noch die Bilder zu den Genres

Dazu in der rechten Spalte Genre jedes einzelne Genre auswählen und diesem dann bei *Bilder über Ändern* ein passendes Bild dazu auswählen

Dieses Bild wird nur in Ember verwendet und hat nichts mit den Genre-Bildern in Kodi zu tun

VV. Filmliste exportieren (TOP)

Nachdem die Ember Datenbank nun wohl gefüllt ist, können die Filme und TV Shows exportiert werden

Dazu gibt es jetzt ein neues Template, *template modernWallCard*

Einen kleinen Vorgeschmack gibts [hier](#) beim Entwickler

Wer diese Liste veröffentlichen möchte, will vielleicht doch nicht alle Information preisgeben

Im Ember-Verzeichnis ist das Template hier abgespeichert:

EmberMediaManager\Modules\generic.embercore.movieexporter\template modernWallCard\template modernWallCard

Jetzt von der Datei *German_(de_DE).html* ein Backup anlegen

Mit einem Editor (z.B. Note++) die html Datei öffnen und jetzt z.B. folgende Änderungen vornehmen:

Soll die Anzahl der Filme und TV Shows nicht angezeigt werden, löscht man den rot markierten Text:

```
<ul class="nav navbar-nav">
<li class=""><a href="#">Filme (<span id="total_movies"><$TOTAL_MOVIES</span></a></li>
<li class=""><a href="#tvshow">Serien (<span id="total_tv_shows"><$TOTAL_TVSHOWS</span> /
```

```
<span id="total_tv_episode"><$TOTAL_TVEPISODES></span></a></li>
</ul>
```

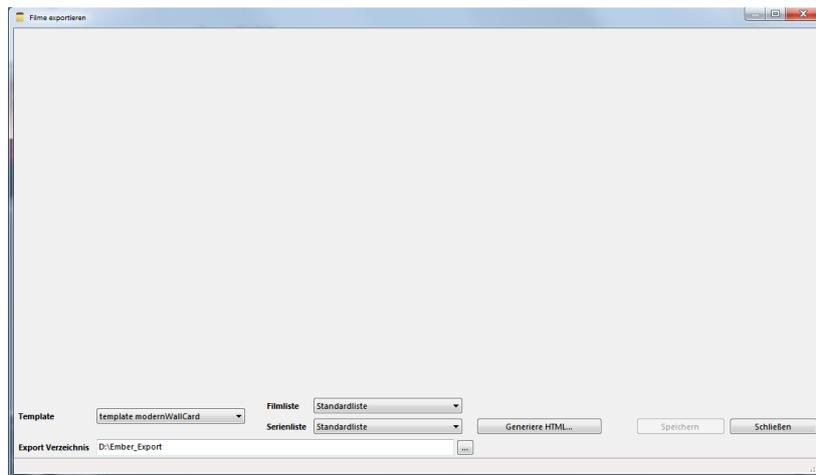
Wer der Meinung ist, dass die Dateigröße auch niemanden etwas angeht (Datenherkunft), löscht in diesem Block, den rot markierten Text:

```
<tr>
<td>Video</td>
<td><$VIDEO></td>
</tr>
<tr>
<td>Audio</td>
<td><$AUDIO></td>
</tr>
<tr>
<td>Dateigröße</td>
<td><$FILESIZE></td>
</tr>
```

Tools-Filmliste exportieren

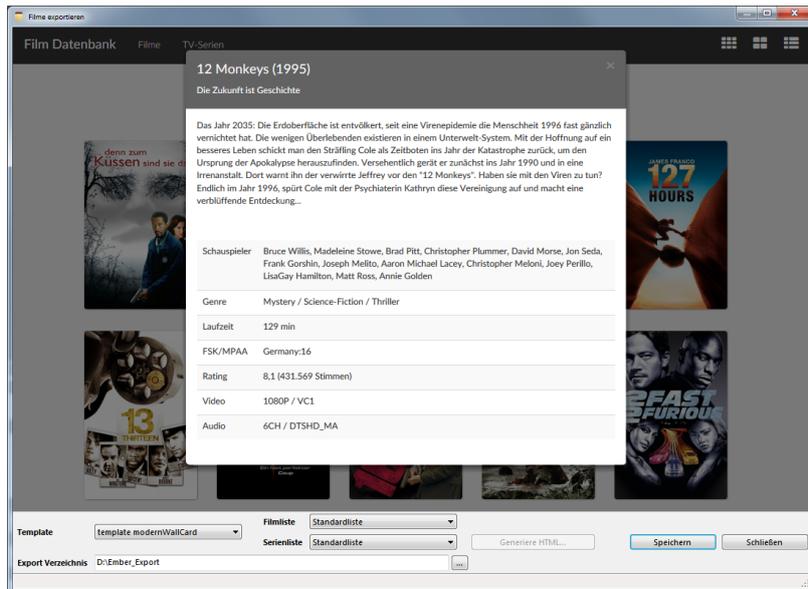
Exportpfad: den gewünschten Exportpfad festlegen

Bei Template, *template modernWallCard* wählen und dann auf *HTML erzeugen* klicken



Ember stellt jetzt die gesamte Liste zusammen und generiert anschliessend das Template

Je nach Umfang der Datenbank dauert das schon ein bisschen



Jetzt noch den Button Speichern drücken und Ember speichert alles in dem vorgegebenen Verzeichnis

VVI. Neue Inhalte (Filme/Serien) Scrapen

1. Neue Filme scrapen (hinzufügen) (TOP)

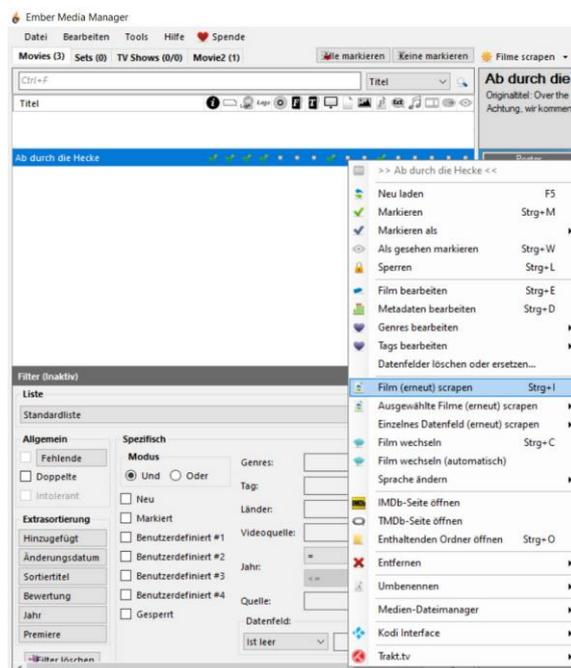
Wird ein Film gescrap, wird automatisch auch das Filmset erzeugt, wenn der Film auf TMDb einem Filmset zugewiesen ist

Das Filmset muss dann aber noch separat gescrapet werden, genauso wie man Filme scrap

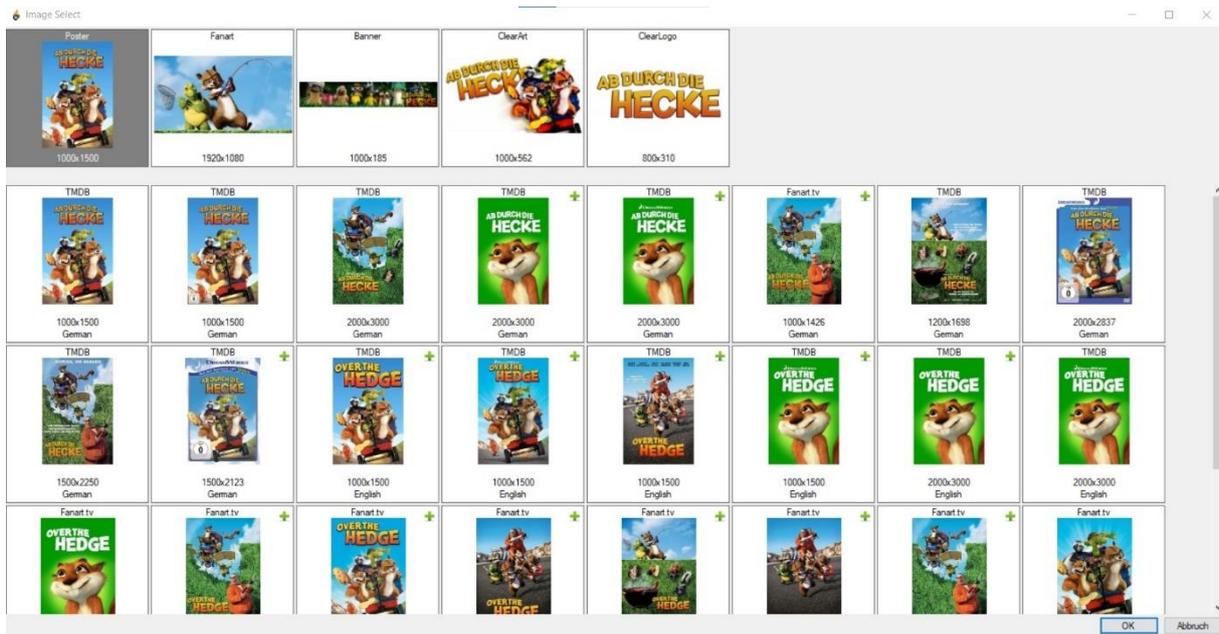
In der Ember-Quelle wurde ein neuer Film abgespeichert

Über den *Datenbank aktualisieren* wird dieser Filme nun in die Ember-DB geholt

Dann Film auswählen, Kontextmenü öffnen und *Film (erneut) scrapen* wählen (oder STRG+I startet den Scrapvorgang)



Bei diesem manuellen Scrapvorgang, hat man maximalen Einfluß was gescrap wird

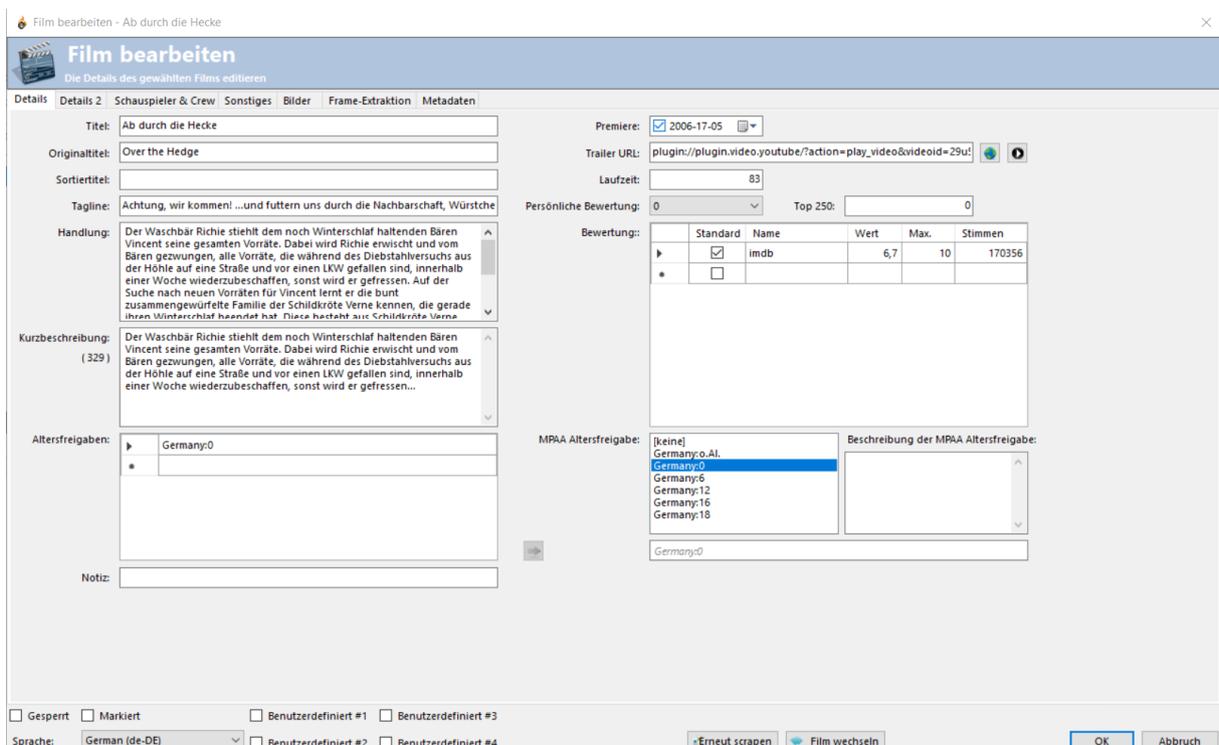


Die erste Zeile sind die Film-Arts. Wird hier etwas ausgewählt, so ändert sich der Inhalt ab Zeile 2. Ab der zweiten Zeile steht eine Auswahl an Bildern zur Verfügung

Immer zuerst in der ersten Zeile ein Film-Art vorauswählen, dann ab der 2. Zeile per Rechtsklick, mit *Select Image*, das gewünschte Bild auswählen

Übrigens, ein Doppelklick auf ein Bild zeigt dieses Bild vergrößert an

Zum Schluss mit Ok bestätigen



Hier wird das Scrap-Ergebnisse angezeigt. Nun kann ein Wunscreiter ausgewählt und Einträge verändert werden

Trailer URL:

Durch Klick auf die Kugel öffnet sich ein neues Fenster wo ein Trailer zum Herunterladen festgelegt wird
Im neuen Fenster *Scrapen* klicken

Trailer wählen

Wähle Trailer zum Herunterladen

Titel	Sprache	Dauer	Variante	Typ	Quelle	Addon
Ab durch die Hecke - Trailer	German	0:55	1080p (H264) 384 kbit/s (AAC 6Ch) .mkv	Clip	YouTube	TMDb
Ab durch die Hecke - Trailer	German	2:09	360p (H264) 128 kbit/s (AAC) .mkv	Trailer	YouTube	TMDb
Ab durch die Hecke-Trailer	German	1:18	360p (H264) 128 kbit/s (AAC) .mkv	Trailer	YouTube	TMDb

Scrapen

Auf YouTube suchen
Ab durch die Hecke german trailer Suchen

Benutzerdefiniert
Direkter Link oder YouTube URL:

Lokale Datei:

Im Browser öffnen In NFO speichern Abbruch

Trailer wählen und *In NFO speichern* bestätigen

Ember Media Manager

Ab durch die Hecke (2006)

USPRE? AVCI 720p DGBDRBY 5.1

Poster: Fanart:

Informationen

Regisseure: Karey Kirkpatrick / Tim Johnson
Credits: Karey Kirkpatrick / Len Blum / Lorne Cameron / David
Premiere: 2006-05-17
Schauspieler: Bruce Wills als RJ (voice), Gary Shandling als Veme (voice), Steve Caroll als Henry (voice), Wanda Sykes als Stella (voice), William Shatner als Ozze (voice), Nick Nolte als Vincent (voice), Thomas Haden Church als Dwayne (voice), Alison Janney als Gladys (voice), Eugene Levy als Lou (voice), Catherine O'Hara als Penny (voice), Avi Lavigne als Heather (voice), Omid Djalil als Tiger (voice), Sami Kirkpatrick als Bucky (voice), Shane Baumel als Spike (voice), Madison Davenport als Quillo (voice), Ariel Winter als (voice)

Kurzbeschreibung:
Der Waschbär Richie stellt dem noch Winterschlaf haltenden Bären Vincent seine gesamten Vorräte. Dabei wird Richie erwischt und vom Bären gezwungen, alle Vorräte, die während des Diebstahlsversuchs aus der Höhle auf eine Straße und vor einen LKW gefallen sind, innerhalb einer Woche wiederzubeschaffen, sonst wird er gefressen...

ID	TMDB ID	Collections	Altersfreigaben
m0327084	7518		Germany 0

Plat: C:\Users\Name_Laptop\Downloads\Ab durch die Hecke\Ab d...
Trailer Plat: plugin://plugin.video.youtube/?action=play_video&videoid=23U...

Countries: United States of America
Tags:

Handlung:
Der Waschbär Richie stiehlt dem noch Winterschlaf haltenden Bären Vincent seine gesamten Vorräte. Dabei wird Richie erwischt und vom Bären gezwungen, alle Vorräte, die während des Diebstahlsversuchs aus der Höhle auf eine Straße und vor einen LKW gefallen sind, innerhalb einer Woche wiederzubeschaffen, sonst wird er gefressen. Auf der Suche nach neuen Vorräten für Vincent lernt er die bunt zusammengewürfelte Familie der Schilfkroie Veme kennen, die gerade ihren Winterschlaf beendet hat. Diese besteht aus Schilfkroie Veme, Eichhörnchen Hammy, Stinktier Stella sowie zwei Opossums und fünf Stachelchweinen. Mit Hilfe dieser Gruppe will er in die neugebaute Vorstadt hinter der großen Hecke eindringen, die während des Winterschlafs der Gruppe geplant wurde und nun deren Wald begrenzt. Richie will in der Siedlung die Vorräte für Vincent stehlen.

Metadaten
Videospuren: 1
Audiospuren: 1
Videospur 1
- Größe: 300x448
- Seitenverhältnis: 1.785
- Screenshot Progressive
- Codec: avc1
- Bitrate: 1084
- Dauer: 4950
- Dateigröße (MB): 913
Audiospur 1

Das wars, der erste neue Film ist gescrapt

Wird ein neuer Film gescrapt, von dem es ein Film-Set gibt, legt Ember dieses Film-Set automatisch an

Bei Filmsets gibts keine Datenauswahl, sondern gleich die Bilder, die Auswahl ist so wie bei Filmen

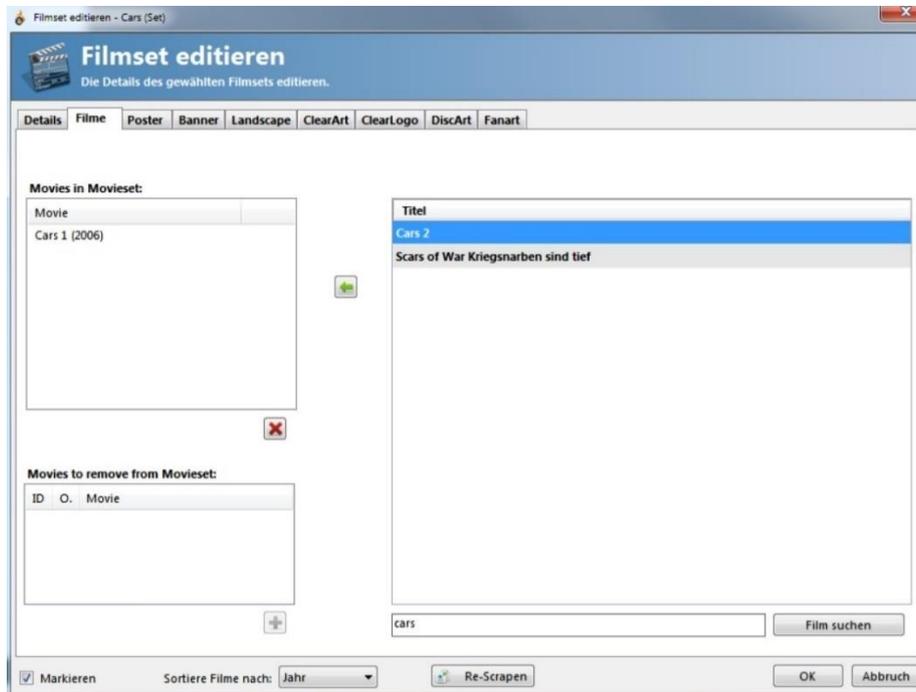
Handlung, gibt es bei vielen Filmsets keine Inhalte

Beispiel die Filmreihe Cars

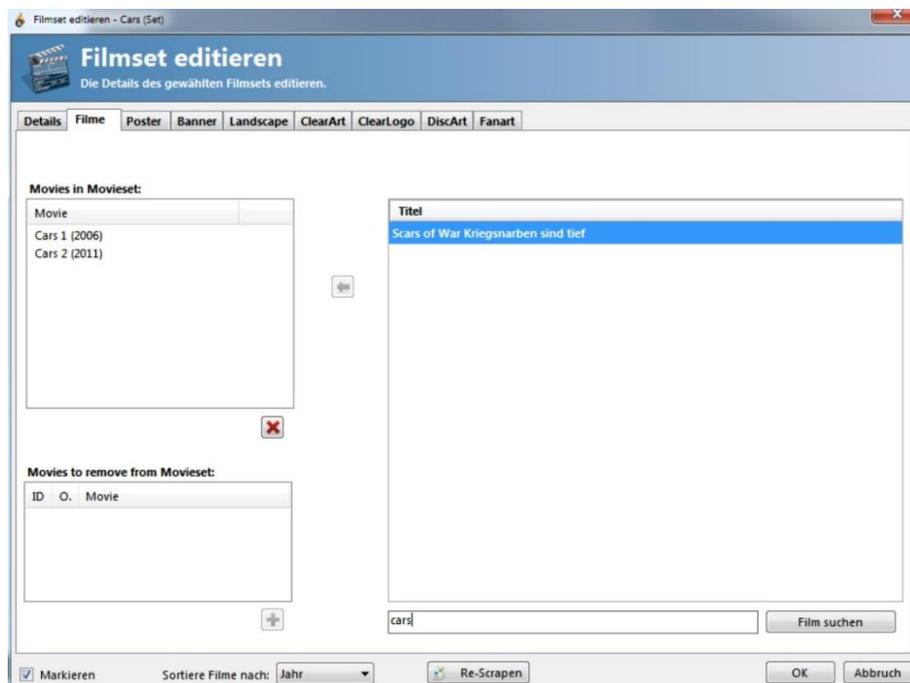
Cars1 wurde als erster Film gescraped

Dieser Film hat das Filmset ausgelöst

Es gibt aber zu dieser Film-Sammlung noch einen weiteren Film - nämlich Cars2, der auch in der Ember-Datenbank vorhanden ist



Über *Film suchen* einfach eine Suche nach Cars starten



Den gesuchter Film markieren und mit dem grünen Pfeil nach links in *Movies in Movieset* verschieben

2 .Neue TV Serien scrapen (hinzufügen) (TOP)

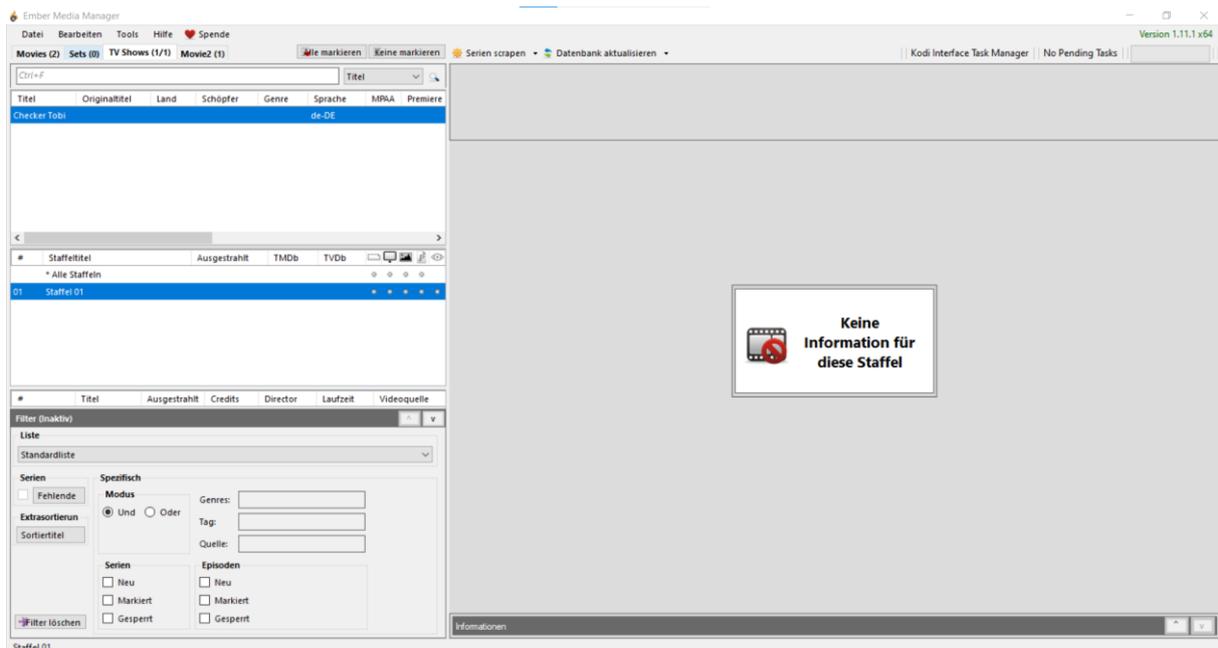
In der Ember-Quelle wurde eine neue Serie abgespeichert

Wie in Kodi, ist es auch hier wichtig, dass die Ordnerbezeichnung Ember/Kodi konform ist

Serien Name – Staffel Ordner (Staffel 1, Staffel 2 usw.) – Episoden Benennung (S01E01 oder so S01E02 Eis Check usw.)

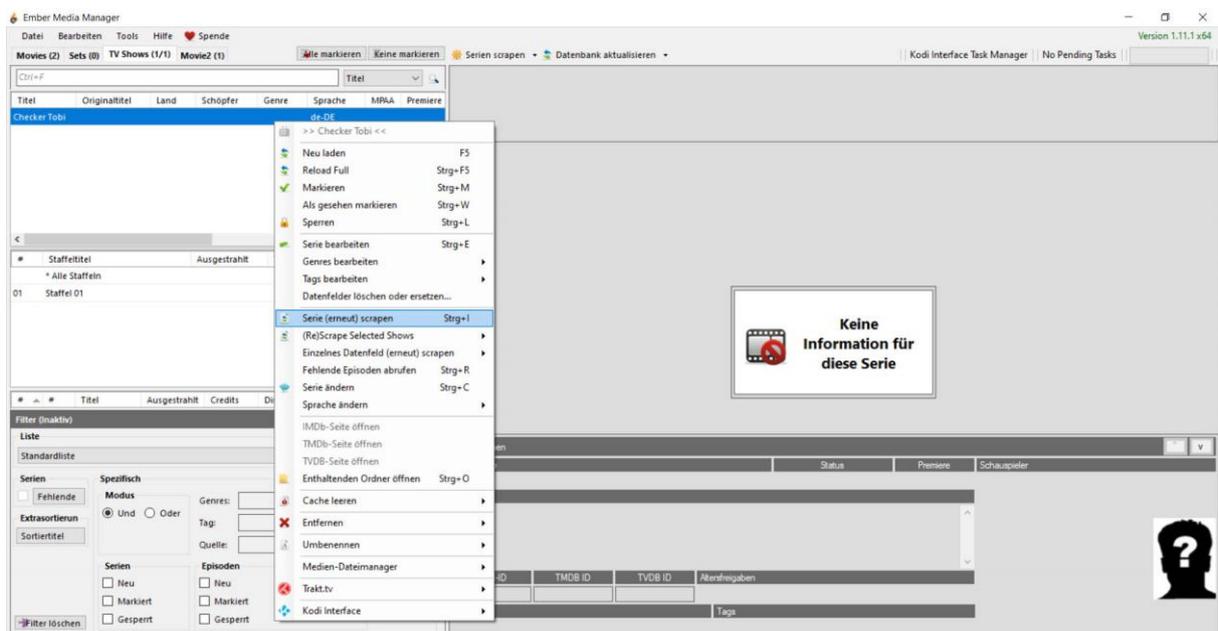
Übersicht von Serien ist z.B. auf www.fernsehserien.de zu finden

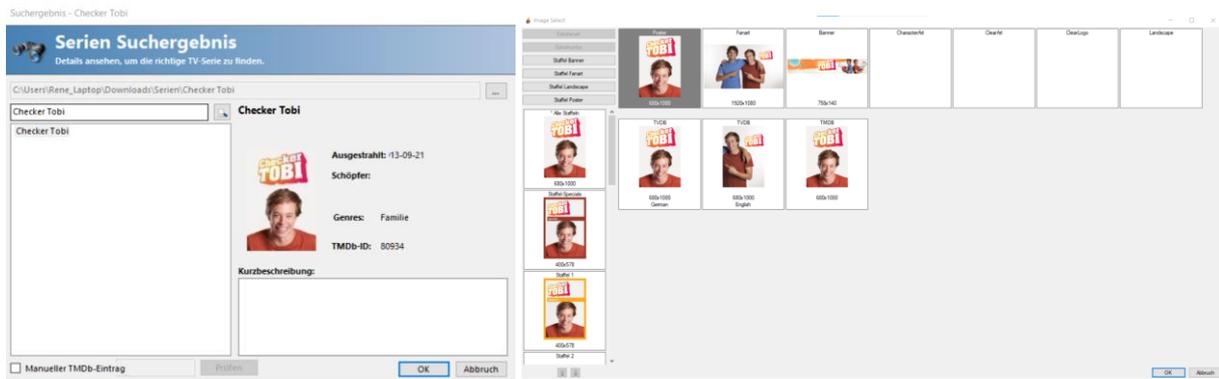
Über den *Datenbank aktualisieren* wird diese Serie nun in die Ember-DB geholt



Bei den Reitern, *TV Shows* auswählen

Dann Serie auswählen, Kontextmenü öffnen und *Serie (erneut) scrapen* wählen (oder STRG+I startet den Scrapvorgang)





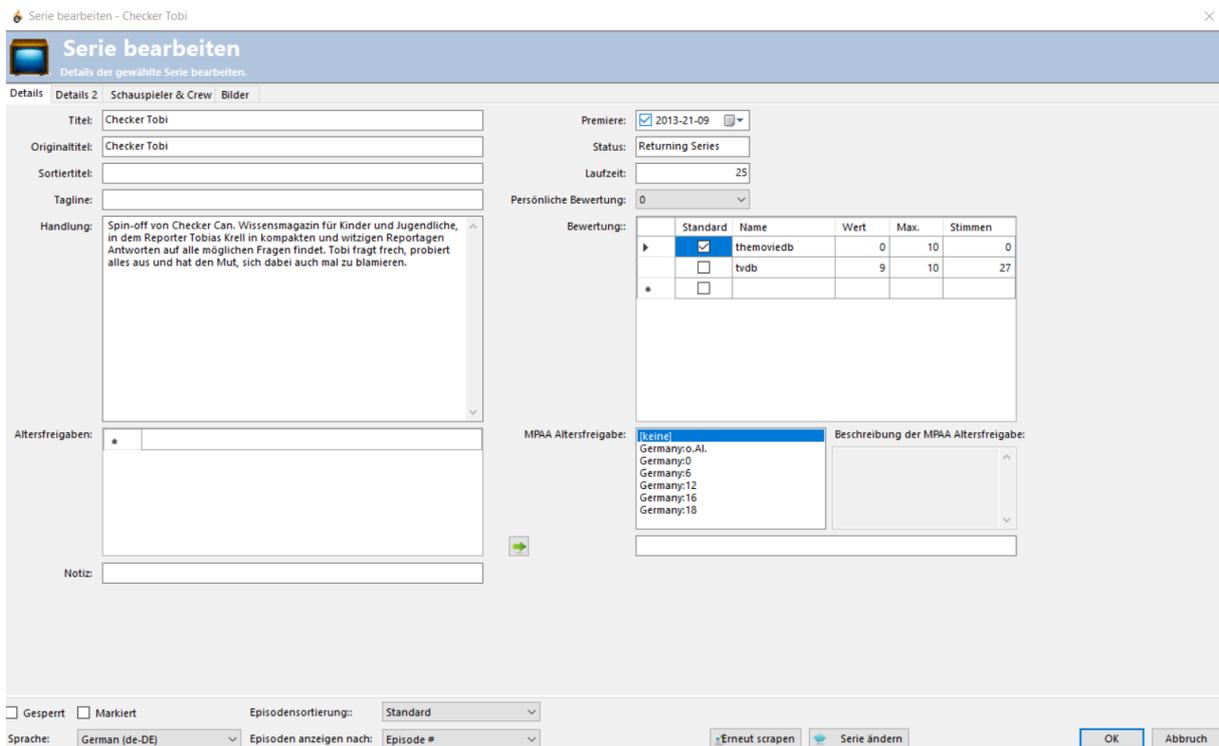
Die erste Zeile sind die TVSerien-Arts. Wird hier etwas ausgewählt, so ändert sich der Inhalt ab Zeile 2. Ab der zweiten Zeile steht eine Auswahl an Bildern zur Verfügung.

Immer zuerst in der ersten Zeile ein TVSerien-Art vorauswählen, dann ab der 2. Zeile per Rechtsklick, mit *Select Image*, das gewünschte Bild auswählen.

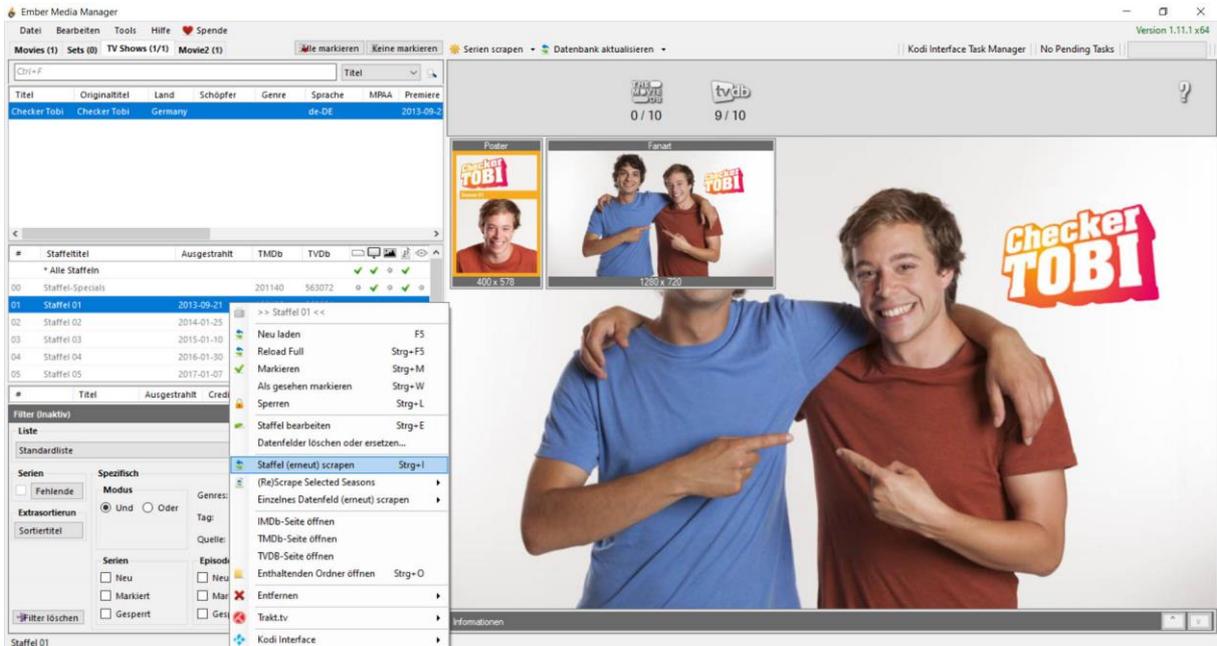
Zusätzlich ist es am linken Rand noch möglich, weitere Bilder Kategorien (z.B. Staffel Fanart) auszuwählen und auch darin wieder Bilder festzulegen.

Übrigens, ein Doppelklick auf ein Bild zeigt dieses Bild vergrößert an.

Zum Schluss mit Ok bestätigen.



Hier wird das Scrap-Ergebnisse angezeigt. Nun kann ein Wunscreiter ausgewählt und Einträge verändert werden, dann OK.



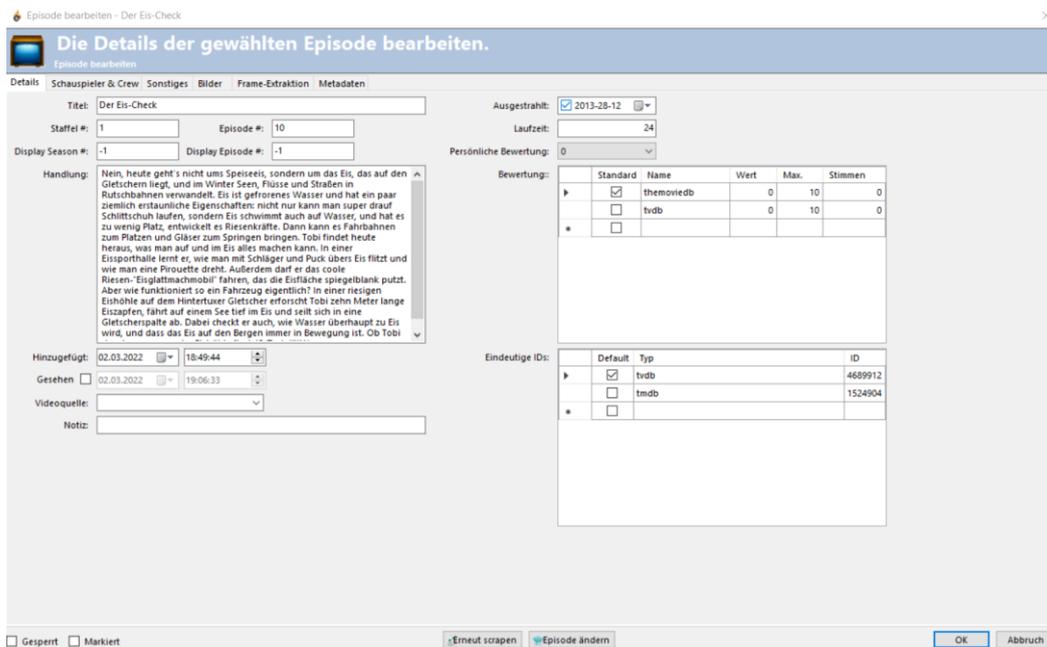
Ebenso kann eine Staffel neu eingelsen werden.
Staffel wählen, Kontext Menü öffnen und *Staffel (erneut) scrapen* wählen

Titel	Originaltitel	Land	Schöpfer	Genre	Sprache	MPAA	Premiere
Checker Tobi	Checker Tobi	Germany			de-DE		2013-09-2

#	Staffeltitel	Ausgestrahlt	TMDb	TVDb			
	* Alle Staffeln						
00	Staffel-Specials		201140	563072			
01	Staffel 01	2013-09-21	106495	563084			
02	Staffel 02	2014-01-25	106496				
03	Staffel 03	2015-01-10	106497				

#	Titel	Ausgestrahlt	Credits	Director	Laufzeit	Videoquelle
08	Der Wind-Ch...	2013-11-16			25	
09	Der Christba...	2013-12-21			25	
10	Der Eis-Check	2013-12-28			24	
11	Der Cowboy-...	2014-01-25			25	
12	Der Schwim...	2014-02-15			25	

Wie auf dem Bild zu sehen ist, werden erkannte Episoden einer Staffel "in schwarzer Farbe" angezeigt



Ein Doppelklick auf die Episode, macht ein bearbeiten der Inhalte möglich

Das wars, die neue Serie ist gescrap

Wenn eine Serie komplett gescrappt worden ist, dann legt Ember alle bekannten Episoden in der Ember-Datenbank ab

Je nach Einstellung (*fehlende Episoden anzeigen*), werden diese Fehlende (graue Farbe) in der Liste angezeigt oder nicht

Vorhanden sind sie in der Ember-DB aber auf jeden Fall nach dem Scrapen

Wird nun eine Serie/Staffel oder Episode ausgewählt und das Kontextmenü geöffnet, so gibt es beim Entfernen, zwei Möglichkeiten:

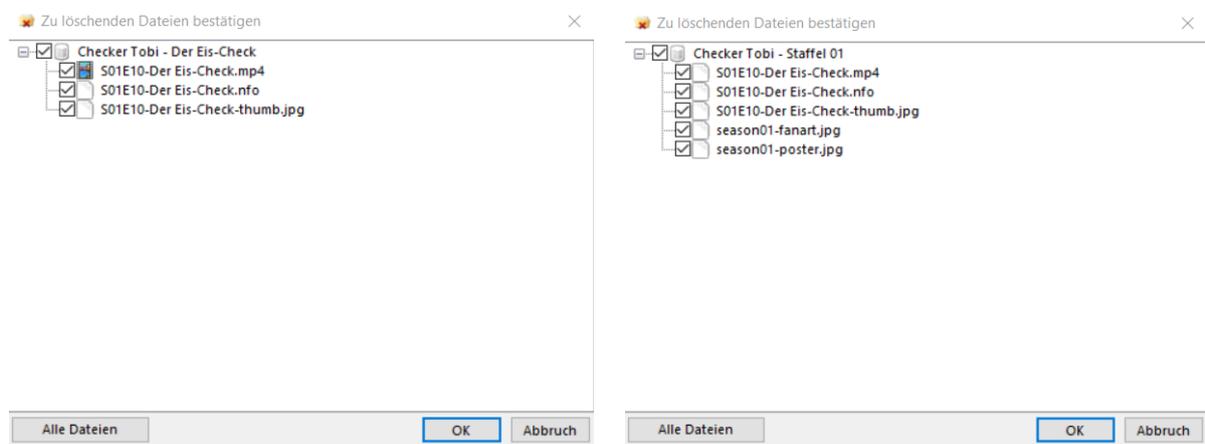
ENTF (aus Datenbank entfernen)

STRG+ENTF (Serie löschen/Staffel löschen/Episode löschen)

Bei **ENTF** wird nur die Datei-Verlinkung in der Ember-DB (inkl. aller Bilder und sonstigem lokalem Zeugs) gelöscht, welche zur Serie/Staffel/Episode gehören

Bei **STRG+ENTF** wird ein Dialog angezeigt, der alle lokalen Dateien dieser Serie/Staffel/Episode versucht zu ermitteln

Die ausgewählten Daten werden dann beim Bestätigen **von der Festplatte gelöscht!!**



Wird nun eine lokal vorhandene Episode gelöscht, dann wird diese Episode in Ember zurück auf Missing gestellt

Damit ist der Eintrag trotzdem noch in der Episodenliste drin. Es könnte ja sein, dass die lokale Datei nur falsch angeschrieben oder zugeordnet worden ist

Vor dem Zurücksetzen auf Missing wird überprüft, ob es noch einen weiteren Eintrag mit der selben Staffel- und Episoden-Nummer gibt

Falls ja, dann wird der zu löschende Eintrag komplett aus der Datenbank entfernt, damit es keine doppelten Einträge gibt

Ansonsten wird auf Missing gesetzt

Wird jetzt ein Eintrag mit ENTf gelöscht, der bereits auf Missing steht, dann ist davon auszugehen, dass man den Eintrag wirklich nicht mehr in der Liste haben will und der Eintrag wird aus der Ember-DB entfernt