

Brother Johns gesammeltes Encodingwissen

Version 2. Dezember 2004



<http://home.arcor.de/brotherjohn>

Erreichen könnt ihr mich übers gulli:board und
das deutsche oder englische Doom9-Forum.

Inhalt

1 Einleitung	1
1.1 Grundlagen des DVD-Formats	1
1.2 Liste der benötigten Software	4
2 Vorarbeiten	6
2.1 Ripping der DVD auf die Festplatte	6
2.2 Decodieren der VOBs mit DGIndex	7
2.3 Transcoding des Sounds	9
2.3.1 Transcoding mit der BeSweetGUI	11
2.3.2 Transcoding mit BeLight	18
2.4 Untertitel: Eine Einführung	24
2.4.1 Vorbereiten von festen Untertiteln	26
2.4.2 Erzeugen von dynamischen Text-Untertiteln	28
3 Encodieren mit Gordian Knot	32
3.1 Gordian Knot und Codecs konfigurieren	32
3.1.1 Auswahl von Container und Codec	33
3.1.2 Konfiguration des XviD-1.0.2-Codecs	35
3.1.3 Konfiguration des DivX-5.2.1-Codecs	41
3.2 Kalkulieren der Bitrate	44
3.3 Vorbereiten des Videos	46
3.4 Dynamische Untertitel und Kapitel einbinden	49
3.5 Abschließende Arbeiten und Encoding	50
4 Film fertigstellen	55
4.1 Codierten Film splitten	55
4.2 Muxen und splitten mit mkvmerge	57
4.3 AAC im OGM- oder AVI-Container	60
5 Changelog	61

1 Einleitung

1.1 Grundlagen des DVD-Formats

Die Eckdaten der DVD

Video	MPEG-2 720 × 576 Pixel PAL 720 × 480 Pixel NTSC
Audio	AC3, DTS, MPEG oder PCM maximal 8 Audiospuren
Untertitel	gespeichert als Pixelbilder Forced-Subs-Funktion maximal 32 Untertitelspuren
Datenrate	insgesamt maximal 10 Mbit/s

Bildformate

Die Welt ist sich uneins, das gilt auch für digitales Video. Europa setzt auf den PAL-Standard, Amerika auf NTSC. Entsprechend gibt es verschiedene DVDs. Wer einen Film aus Amerika bestellt und auf einem europäischen Player/Fernseher abspielen will, hat keine besonders große Freude daran, es sei denn, seine Geräte verstehen außer dem üblichen PAL auch ausdrücklich das NTSC-Format.

Wer nachrechnet wird feststellen, dass der heimische Fernseher ein Seitenverhältnis (AR: Aspect Ratio) von 4:3 (d. h. 1,33:1) hat, es sei denn, es ist ein 16:9-Modell (1,78:1). Das Bild einer PAL-DVD dagegen besitzt eine AR von $720 / 576 = 1,25$. Das passt doch nicht zusammen? Doch, tut es. Die DVD enthält das Bild nämlich in verzerrter Form. Erst beim Abspielen wird dafür gesorgt, dass es mit dem richtigen Seitenverhältnis am Monitor ankommt.

In die Tiefen der AR-Thematik einzusteigen, würde den Guide absolut sprengen. Nur soviel: Das menschliche Auge kann vertikal mehr Details erkennen als horizontal. Durch die Verzerrung lässt sich auf Kosten der Horizontalen in der wichtigen vertikalen Richtung eine höhere (detailreichere) Auflösung speichern. Insgesamt kommt dadurch beim Auge mehr Qualität an.

Dann existieren noch zwei Verzerrungsarten: anamorph für Widescreen-Filme (die mit den schwarzen Balken oben und unten) und non-anamorph für Vollbild-Filme. Grob gesagt liegt der Unterschied darin, dass Widescreen-Filme mit höherer Verzer-

nung gespeichert sind. Welche Art der Verzerrung vorliegt, wird später beim Encoding wichtig, wenn wir das Bild aufs richtige Seitenverhältnis umrechnen müssen.

Die MPEG-Videostandards

MPEG ist eine Abkürzung und steht für *Moving Picture Expert Group*, ein Gremium, das verschiedene Standards zur Codierung digitaler Videos erarbeitet.

MPEG-1 ist der älteste Standard. Verwendung findet er für die VideoCD und bei vielen im Internet angebotenen Videos, die größtmögliche Kompatibilität erreichen wollen. MPEG-1 ist nämlich erstens so weit verbreitet, dass ihn nahezu jeder Computer abspielen kann, und stellt zweitens so geringe Anforderungen an die Rechenleistung, dass die Videos auch auf veralteten Computern gut laufen. Nachteil ist, dass mit steigender Kompression die Qualität schnell deutlich abnimmt.

MPEG-2 dürfte MPEG-1 kaum nachstehen, was die Verbreitung angeht. Schließlich sind DVDs in diesem Format codiert. MPEG-2 erreicht eine ordentliche Kompression. Die mehr als 8 GB Speicherplatz der DVD tun dann ihr Übriges, um für erstklassige Qualität zu sorgen.

MPEG-4. Der Untertitel dieses Guides sagt es schon: MPEG-4 ist unser Zielformat. Gegenüber MPEG-2 lässt sich in der gleichen Dateigröße noch einmal deutlich mehr Qualität unterbringen. Deshalb können wir eine DVD auch ohne extreme Verluste auf zwei CDs schrumpfen.

Die MPEG-Standards allein bringen uns dem codierten Video allerdings noch nicht näher. Das MPEG-Gremium programmiert keine Codecs, sondern definiert nur, wie ein gültiger Videostream der entsprechenden MPEG-Version auszusehen hat. Daraus ergibt sich auch, welche Methoden beim Encoding angewendet werden können und welche nicht. Die Details der Codierung bleiben dann der Phantasie der Codec-Programmierer überlassen. Alle Tricks sind erlaubt, solange das Endergebnissen den Vorgaben der verwendeten MPEG-Version entspricht.

Diese Tatsache führt dazu, dass z. B. DivX, XviD und 3ivx drei unabhängige und verschiedene Codecs sind, die aber alle Videos nach dem MPEG-4-Standard erzeugen. Deshalb kann zumindest theoretisch XviD ein mit 3ivx erzeugtes Video abspielen oder 3ivx ein mit DivX erzeugtes usw. Dass das in der Praxis doch nicht so 100-prozentig funktioniert liegt daran, dass keiner der Codecs den MPEG-4-Standard absolut vollständig und korrekt implementiert.

Audioformate

Das wichtigste Audioformat der DVD nennt sich **AC3**, was für *Audio Codec 3* steht. Unter dem Begriff *Dolby Digital* wird er von der Firma Dolby entwickelt. AC3 ist ein verlustbehafteter Codec, der mit konstanter Bitrate arbeitet. Typischerweise werden für Stereo-Tonspuren 192 kbit/s verwendet, für Mehrkanalton 384 bzw. 448 kbit/s. AC3 unterstützt bis zu sechs Kanäle, wobei der sechste als zusätzlicher Basskanal ausgelegt ist. Soll heißen, die normalen Bässe stecken in den 5 vollständigen Kanälen, der sechste enthält die zusätzlichen Basseffekte, die die Wände wackeln lassen. Da er eigentlich kein vollständiger Kanal ist, hat es sich eingebürgert, ihn extra anzugeben. Daher kommt die Schreibweise **5.1**, d. h. 5 vollständige Kanäle (vorne links und rechts, hinten links und rechts, vorne Mitte) und ein zusätzlicher Basskanal (LFE: Low Frequency Effects).

Beliebt ist auch **DTS** (Digital Theater Sound), das genauso wie AC3 ein verlustbehaftetes Format ist, allerdings mit deutlich höheren Bitraten arbeitet und noch mehr Kanäle unterstützt (bis zu 6.1).

MPEG-Ton ist selten zu finden, in seiner Multikanalausprägung (bis zu 7.1) schon gar nicht. Deshalb möchte ich auch nicht intensiv darauf eingehen. Nur soviel: Es gibt zwei Varianten: MPEG-1 Layer II (kurz MP2, der Vorläufer von MP3), mit dem maximal Stereo und 384 kbit/s möglich sind, und MPEG-2, mit dem man 7.1 Kanäle bei maximal 912 kbit/s zur Verfügung hat.

PCM schließlich und endlich kennt der Windowsbenutzer als Wave. Es handelt sich im Gegensatz zu allen anderen um ein unkomprimiertes Format, das entsprechend Platz belegt. Deshalb findet man es auch eher auf Konzert-DVDs und der DVD-Audio, wo das Hauptaugenmerk eher auf dem Ton als dem Bild liegt. PCM erlaubt bis zu 8 Kanäle.

Die typische DVD

Eine typische deutschsprachige DVD dürfte aktuell in etwa so ausgestattet sein:

- PAL, 720 × 576 Pixel MPEG-2-Video.
- Audiospuren in AC3 5.1, Deutsch und Originalsprache (also meistens Englisch).
- Oft eine deutsche 6.1 DTS-Audiospur.
- Evtl. zusätzliche Audiospuren mit Director's Comments o. ä.
- Eine Reihe von Untertiteln in verschiedenen Sprachen.

Es besteht also kein Grund, angesichts der Vielfalt der möglichen Formate in Panik auszubrechen, denn die Wirklichkeit sieht recht übersichtlich aus. Wer sich tiefer einlesen möchte, findet auf *www.dvd-tipps-ticks.de* massig Informationen.

1.2 Liste der benötigten Software

Gordian Knot RipPack und CodecPack

Bevor wir eine DVD auseinander nehmen können, brauchen wir natürlich die passende Softwareausstattung. Kein Problem, denn im Gordian Knot RipPack ist fast sämtliche nötige Software zusammengefasst. Die Video-Codecs finden sich im CodecPack. Natürlich hindert uns nichts daran, die nötigen Programme auch einzeln zusammenzusuchen, was gerade für XviD Sinn macht, der in heißen Phasen zu oft aktualisiert wird, um jedes Mal das Codec Pack zu erneuern. Einige der Tools (SubRip, mkvtoolnix) sind nicht im RipPack enthalten. Die müssen wir uns extra besorgen.

Links zur Software

- Gordian Knot
Die Kommandozentrale fürs Encoding.
<http://sourceforge.net/projects/gordianknot>
- DVD Decrypter
DVD Ripper. Entschlüsselt die DVD und kopiert sie auf die Festplatte.
Den Link habe ich aufgrund der verbraucherfeindlichen Rechtslage vorsichtshalber entfernt. Ich möchte nicht riskieren, von irgend einem Wegelagerer von Anwalt verklagt zu werden. Mit ein wenig suchen sollte trotzdem jeder die Homepage finden.
- DGIndex
Decoder für die VOB-Dateien der DVD. Ehemals DVD2AVI, jetzt Teil des DGMPGDec-Pakets.
<http://neuron2.net/fixd2v/decodefix.html>
- BeSweet und BeSweetGUI
Audio-Transcoding-Tool. BeSweet Beta-Versionen enthalten oft nicht alle nötigen Dateien. Deshalb auch die letzte stabile Version laden und zuerst installieren.
<http://besweet.notrace.dk>
- BeLight (nicht im RipPack enthalten)
Neue grafische Oberfläche für BeSweet. Klarer aufgebaut als BeSweetGUI, aber

auch nicht ganz so leistungsfähig.

<http://forum.doom9.org/showthread.php?s=&threadid=85566>

- VobSub

Tool für VobSub-Untertitel.

http://www.afterdawn.com/software/video_software/subtitle_tools/vobsub.cfm

- SubRip (nicht im RipPack enthalten)

Tool für die Erstellung dynamischer Untertitel.

<http://zuggy.wz.cz>

- AviSynth

Frameserver für das Video.

<http://www.avisynth.org>

- AviSynth-Plugins

Liste und Links für viele AviSynth-Plugins.

Convolution3D, Decomb, Deen, FluxSmooth, SimpleResize, TomsMoComp usw.

<http://www.avisynth.org/warpenterprises>

- VirtualDubMod

Video-Editing-Tool. Damit geschieht das eigentliche Encoding.

<http://sourceforge.net/projects/virtualdubmod>

- mkvtoolnix (nicht im RipPack enthalten)

Toolpaket rund um Matroska. Nötig für AAC in MKV.

<http://www.bunkus.org/videotools/mkvtoolnix/index.html>

- XviD Video-Codec

Binaries von Koepi. Den Quellcode gibt's auf xvid.org.

<http://koepi.org>

- DivX Video-Codec

Bis zur Version 5.1.1 verfügbar als kostenlose eingeschränkte Version, kostenlose Vollversion mit Werbung und kostenpflichtige Vollversion ohne Werbung. Die Adware der kostenlosen Vollversion lässt sich aber wieder rausschmeißen.

In der Registry unter *HKLM\Software\Microsoft\Windows\CurrentVersion\Run* den Eintrag *Trickler* löschen, außerdem den kompletten Schlüssel *HKLM\Software\Gator.com*, rebooten, fertig. Seit der Version 5.2 gibt es DivX mit vollem Funktionsumfang nur noch als 6-monatige Trial.

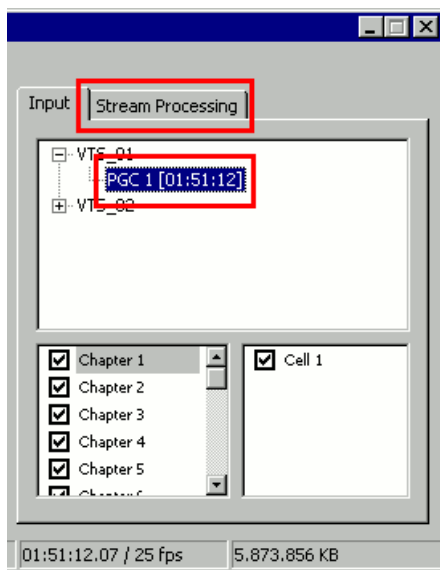
<http://www.divx.com/divx>

2 Vorarbeiten

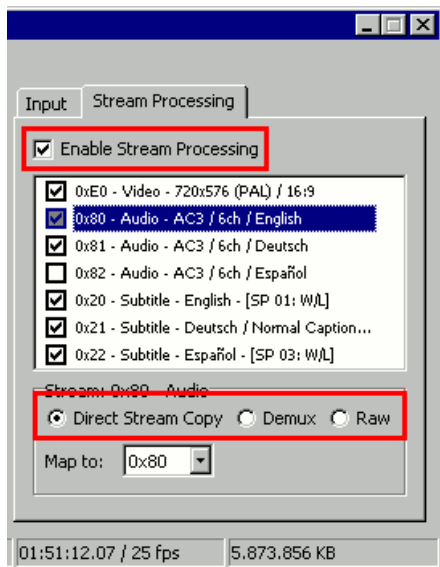
2.1 Ripping der DVD auf die Festplatte

Jetzt kann's losgehen. DVD einlegen und DVD Decrypter starten. Sollten wir die DVD schon auf der Platte liegen haben, müssen wir SmartRipper bzw. vStip verwenden oder ein virtuelles DVD-Laufwerk benutzen, denn DVD Decrypter kann nicht auf Dateien auf der Platte zugreifen.

Nach dem Einlesen der DVD erscheint ein Fenster, dessen rechte Seite etwa so wie unten aussieht (wenn nicht, über **Mode / IFO** umschalten).



Automatisch ist schon der größte Titel auf der DVD ausgewählt (hier **VTS_01 / PGC 1**), der normalerweise dem Hauptfilm entspricht. Da wir den ganzen Film rippen wollen, können wir links den Zielordner auswählen und dann gleich auf das Register **Stream Processing** wechseln, das etwa so wie unten aussieht:

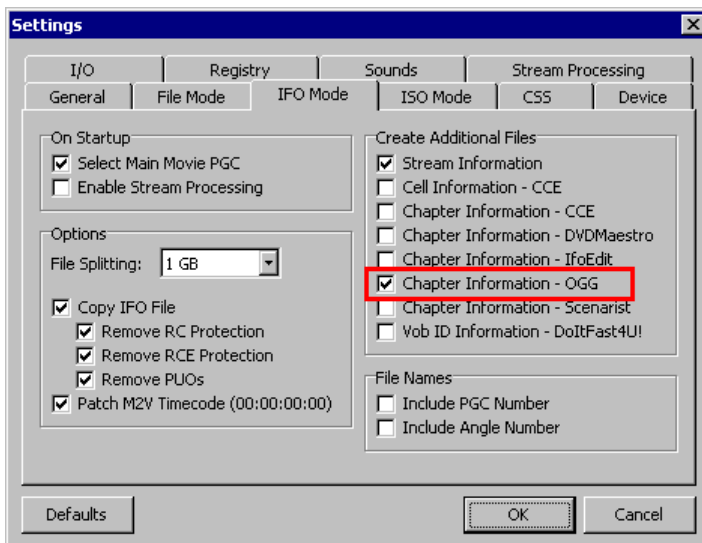


Den Haken bei **Enable Stream Processing** setzen wir und wählen dann aus, was wir alles rippen wollen. Dazu gehört natürlich das Video und mindestens eine Audiospur. Wir können den Sound in den VOB-Dateien lassen oder gleich als separate Audio-Dateien schreiben. Die beiden Methoden unterscheiden sich kaum, wer aber ganz exakt arbeiten will, sollte die Audios in Ruhe lassen und sie später mit DGIndex extrahieren. **Direct Stream Copy** lässt den Ton in den VOBs, **Demux** erzeugt extra Audio-Dateien.

Anmerkung: Manche Software-DVD-Player können bei gerippten VOBs nicht mehr zwischen den Audiospuren umschalten, sondern spielen ausschließlich die auf **0x80** gemappte Spur ab. Wer also den Film von den VOBs auf der Platte anschauen will, sollte darauf achten, dass die richtige Sprache auf **0x80** liegt. Dafür ist das **Map to**-Dropdown-Feld zuständig.

Untertitel lassen wir immer per **Direct stream copy** in den VOB-Dateien. Das Extrahieren übernimmt später VobSub bzw. SubRip.

Um die Kapitelinfos der DVD auszulesen, muss der richtige Haken gesetzt sein, und zwar unter **Tools / Settings / IFO Mode**.



Chapter Information - OGG schreibt die Kapitel im OGG-Format in eine Textdatei. Die anderen Formate sind für uns weniger interessant.

Links unten im Hauptfenster starten wir mit dem großen Button den Ripping-Prozess. Je nach DVD-Laufwerk, Verschlüsselung der DVD und Länge des Films dauert das Rippen zwischen einigen Minuten und einer halben Stunde. Dann haben wir im Normalfall zwischen 4 und 7 GByte auf der Platte: einige VOBs, eine ifo-Datei und eine txt-Datei mit den Kapitelinfos.

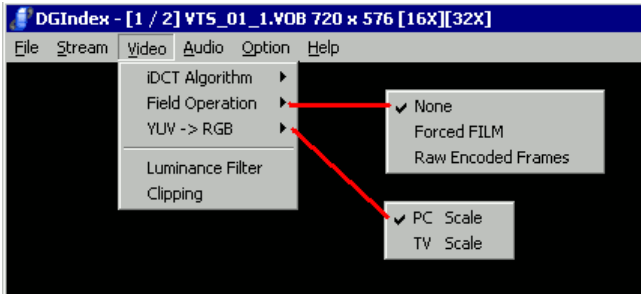
2.2 Decodieren der VOBs mit DGIndex

Ursprünglich hieß das Programm DVD2AVI, entwickelt von jackei. Nachdem jackei die Entwicklung eingestellt hatte, hat sich Donald »neuron2« Graft dem Programm angenommen und einige Bugs beseitigt. Seit kurzem heißt DVD2AVI nun DGIndex und ist Teil des DGMPGDec-Pakets.

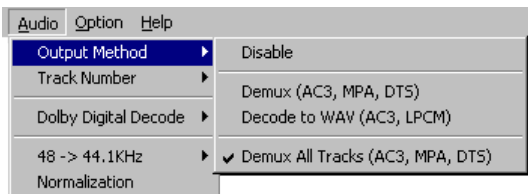
Wir starten DGIndex und gelangen über **F3** in den Öffnen-Dialog. Dort wählen wir die erste VOB-Datei aus, die sich **vt_s_01_1.vob** oder ähnlich nennt. Wichtig ist die **_1** ganz hinten. Beim Klick auf **öffnen** erscheint ein weiteres Fenster, in dem DGIn-

dex automatisch alle weiteren nötigen VOBs ausgewählt hat. Einfach mit **ok** bestätigen.

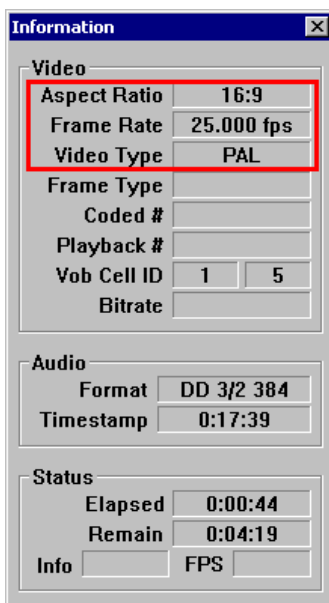
Die Einstellungen im **video**-Menü sollten wie unten aussehen. Den **idCT Algorithm** wählt DGIndex je nach Prozessor automatisch.



Im **Audio**-Menü regeln wir das Herausziehen der Audiospuren aus den VOBs.



Output Method bestimmt, auf welche Weise die Audiospuren extrahiert werden. **Disable** extrahiert gar nichts, **Demux** extrahiert nur die unter **Track Number** ausgewählte Spur und **Demux all Tracks** extrahiert alle vorhandenen Spuren. Mit **Decode to Wav** könnten wir den Sound gleich in Wave umwandeln. Das sollten wir aber besser BeSweet überlassen. Deshalb interessiert uns auch der Rest des **Audio**-Menüs nicht.



Über **F4** legen wir nun eine d2v-Projektdatei an.

Während DGIndex arbeitet, erscheint das Status-Fenster. Die rot umrandeten Werte sind wichtig, denn die brauchen wir später in Gordian Knot, um Auflösung und Seitenverhältnis des Bildes richtig einzustellen. Wir merken uns die Werte also gut.

Sobald unten im Fenster **FINISH** erscheint, ist der Speichervorgang abgeschlossen und wir haben die extrahierten Audiodateien und die d2v-Datei auf der Platte.

Wenn bei **Frame Type** zum Schluss **Interlaced** erscheint, ist das Video möglicherweise interlaced codiert. Dann sollten wir uns Szenen mit viel horizontaler Bewegung

suchen und prüfen, ob dort Kammartefakte wie im Bild unten auftreten. Manchmal wird der Effekt auch nur bei Szenenwechseln sichtbar, wenn sich altes und neues Bild ein Frame lang überlagern.



Hat das Video solche Effekte, müssen wir es später in Gordian Knot deinterlacen. DGIndex hat seine Arbeit jetzt auf jeden Fall erledigt und kann geschlossen werden.

Anmerkung: Interlacing taucht in einer unübersehbaren Vielzahl von Varianten auf. Ich bin auf dem Gebiet alles andere als ein Experte. Erwartet in diesem Guide also keine vollständigen und immer komplett richtigen Infos dazu.

2.3 Transcoding des Sounds

Die Audioformate

Für die Audiospuren gibt es viele Möglichkeiten; die einfache alte Formel »MP3 oder bei genügend Platz AC3« gilt nicht mehr. Deshalb sehen wir uns erst einmal die verschiedenen Audioformate an.

MP3. Der Klassiker. Mit MP3 brauchen wir uns um Abspielprobleme keine Sorgen zu machen, läuft überall. Der Platzbedarf liegt mit etwa 140 – 150 kbit/s im Rahmen, Raten unter 128 kbit/s sollten wir vermeiden. Größter Nachteil ist die Beschränkung auf Stereo. Den 6-Kanal-Ton der DVD können wir mit MP3 nicht beibehalten.

AC3. Der Original-Ton von der DVD. Behalten wir ihn, entfällt die Konvertierung. Ohne die haben wir auch keinerlei Qualitätsverlust, und natürlich bleibt der Mehrkanal-Ton erhalten. Großer Nachteil: AC3 ist mit 384 kbit/s oder 448 kbit/s verdammt groß, so dass wir ihn für 1-CD-Rips oder mehrsprachige Videos fast immer vergessen können.

Allerdings muss AC3 nicht immer 6 Kanäle enthalten, auch 192-kbit-Stereo (z. B. die deutschen Tonspuren der Indiana-Jones-DVDs) oder sogar Mono (z. B. Audiokommentare) sind möglich.

DTS. Der Digital Theater Sound ist inzwischen auf DVDs sehr beliebt. Prinzipiell gilt das gleiche wie bei AC3, nur dass dts mit noch höheren Bitraten (768 kbit/s aufwärts) arbeitet. Persönlich finde ich es blödsinnig, für eine Audiospur so extrem viel Platz zu opfern.

Vorbis. Für Stereosound der Konkurrent zu MP3. 80 kbit/s klingen immer noch einwandfrei. Auch bei hoher Qualitätseinstellungen werden die 128 kbit/s kaum überschritten. Vorbis kann auch Mehrkanal-Ton, allerdings ist dieser Modus weder groß getestet noch wirklich ausgereift. Damit bleibt Vorbis praktisch auf Stereo beschränkt.

AAC ist das »MP3 der Zukunft« und offizielles Audioformat des MPEG-4-Standards. V. a. im HE-Modus ist die Komprimierungsleistung phantastisch: 70 kbit/s für Stereo sind kein Problem und 6-Kanal-Ton kriegen wir problemlos in 128 kbit/s unter. Zwei Nachteile: Gerade der HE-Modus braucht beim Abspielen einiges an Rechenleistung. Für ältere Rechner unter 500 MHz schrumpfen AACs Vorteile damit kräftig zusammen. FAAC, der einzige kostenlose AAC-Encoder, beherrscht auch nur den LC-Modus, nicht HE. Wer also die Vorteile von AAC vollständig nutzen will, kommt um einen kommerziellen Encoder nicht herum (Nero ist der beliebteste). Gerade für Besitzer einer Heimkino-Anlage lohnt sich die Investition aber auf alle Fälle.

Empfehlung: Wer seine Encodings auf dem Standalone-Player abspielen will, wird sich wohl oder übel auf MP3 (und evtl. AC3) beschränken müssen. In (mehr oder weniger) naher Zukunft könnte AAC dazustoßen. Davon abgesehen bietet sich MP3 nicht mehr an. Vorbis und AAC erreichen die gleiche Qualität bei geringeren Bitraten, zudem bietet MP3 keinen 6-Kanal-Ton. Als offizielles MPEG-4-Audioformat dürfte AAC langfristig die besten Chancen haben sich durchzusetzen. Nachteil im Moment ist das Fehlen eines freien Encoders, der alle Features unterstützt. Aber das kann sich jederzeit ändern. Und wer weiß was passiert, wenn jemand auf die Idee kommt, den Multichannel-Modus von Vorbis weiterzuentwickeln. Zusätzlich stellt sich im Zeitalter der DVD±R die Frage: Warum nicht gleich AC3/DTS behalten, wenn auf der Scheibe eh massig Platz ist? Das gilt natürlich nicht für das klassische Encoding auf 1 – 2 CDs.

Anmerkung: Für AC3 und DTS können wir den ganzen BeSweet-Prozess überspringen, da wir ja den Sound original von der DVD einfach übernehmen.

Die Programme

BeSweet-Kommandozeile. Das Tool der Wahl für Audio-Transcoding heißt BeSweet. Allerdings ist BeSweet ein Kommandozeilen-Programm, weshalb viele lieber eine GUI verwenden, um die kryptischen Optionen nicht selbst eingeben zu müssen – jedenfalls solange, bis die Kommandozeilenparameter in Fleisch und Blut übergegangen sind ;-).

Auch für die Kryptik habe ich ein paar Erklärungen geplant, weil es afaik keine deutschsprachige Anleitung zur BeSweet-Kommandozeile gibt. Wann die kommen könnte? Wenn ich einmal viel Zeit und Lust habe. Also – wie man auf dem gulli:board so schön unverbindlich sagt – bald™ *g*.

BeSweetGUI. Die traditionelle grafische Oberfläche für BeSweet, programmiert von Dannidin. Da BeSweetGUI wirklich für jede vorhandene Funktion einen Schalter anbietet, ist sie manchmal kaum einfacher zu verstehen als BeSweet selbst. Dafür bleibt kein Transcodingwunsch unerfüllt und zumindest spart man sich das Auswendiglernen der Kommandozeilenoptionen.

BeLight. Neue GUI für BeSweet von Kurtnoise13, bei der mehr die Einfachheit als die Vollständigkeit im Vordergrund steht. Und tatsächlich war BeSweet konfigurieren noch nie so einfach, ohne dass man auf wichtige Optionen verzichten müsste. Mich hat BeLight vollständig überzeugt.

Welche GUI man verwendet (oder gar die Kommandozeile selbst) ist Geschmackssache, das muss jeder für sich selbst entscheiden. Deswegen sind auch alle Erklärungen in beiden Kapiteln vorhanden.

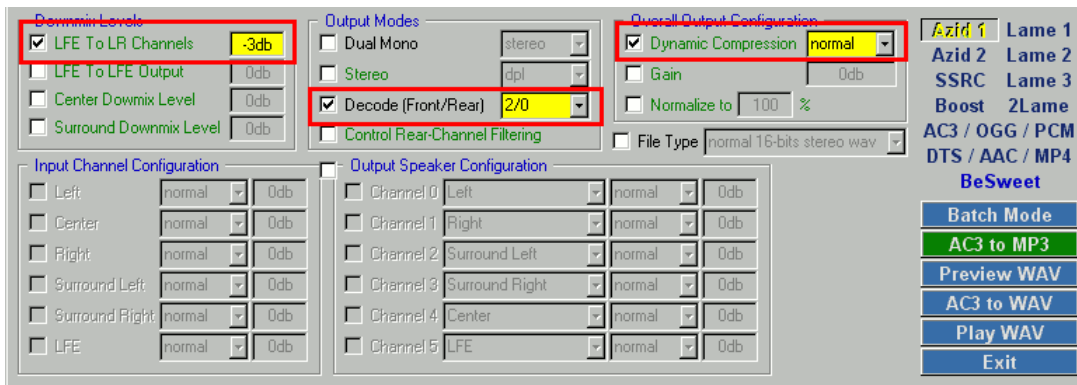
2.3.1 Transcoding mit der BeSweetGUI

Sound decodieren und bearbeiten

Wir starten die BeSweetGUI und wechseln mit den Buttons rechts auf die **Azid1**-Seite. Azid ist dafür zuständig, die Quell-AC3 zu decodieren und eventuell deren 6-Kanäle auf Stereo umzurechnen.

Anmerkung: Die Bezeichnungen der Tonformate sind etwas irreführend. 6-Kanal und 5.1 bezeichnet das selbe. Nur zählt man einmal alle Kanäle (vorne links und rechts, hinten links und rechts, Center und LFE), wogegen bei der 5.1-Schreibweise der Basskanal (LFE) extra dargestellt wird.

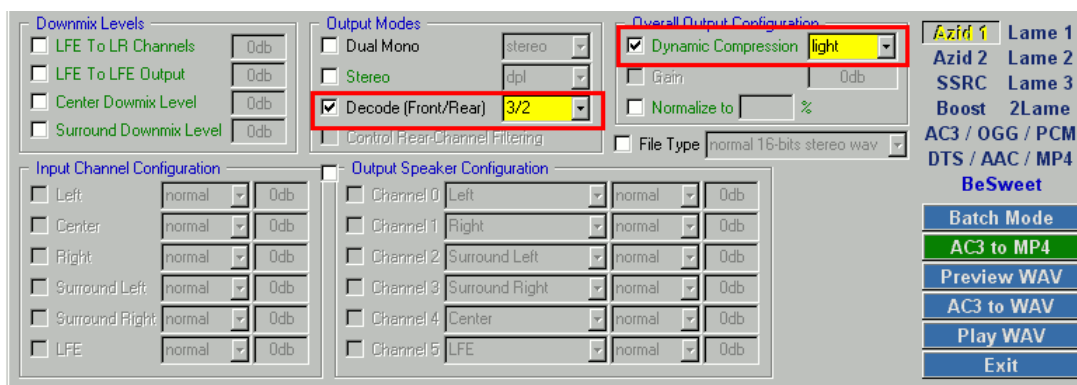
Für 5.1-nach-Stereo-Konvertierung sollte die **Azid1**-Seite so aussehen:



LFE to LR Channels bestimmt, mit welchem Pegel der Basskanal in die vorderen beiden Kanäle gemixt werden soll. Um ein Zuviel an Bass zu vermeiden, stellen wir hier **-3db** ein. Die **2/0** bei **Decode (Front/Rear)** stehen für »umrechnen auf Stereo«. Bleibt die Einstellung der **Dynamic Compression**. Als *Dynamik* des Sounds bezeichnet man die Unterschiede zwischen lauten und leisen Passagen. Je höher die Dynamik, desto höher ist der Lautstärken-Unterschied zwischen leisen und lauten Passagen. Die Audio-Spur eines Films hat von Natur aus eine recht hohe Dynamik. Das wird ganz klar, wenn wir uns den Showdown mit knatternden Maschinengewehren und die geflüsterte Liebesszene im Vergleich vorstellen.

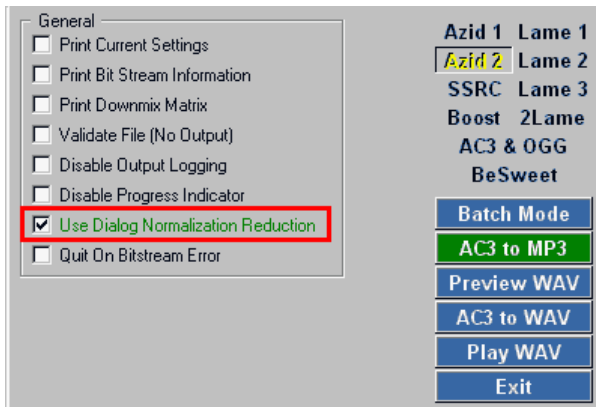
Um später keine nervig leise MP3 zu bekommen, gleichen wir mit der **Dynamic Compression** die Unterschiede in der Lautstärke an, d. h. im Extremfall hört sich das Flüstern genauso laut an wie die Maschinengewehr-Salve. Mit der Einstellung **normal** treiben wir es lange nicht so weit und erreichen eine für Stereo angemessene Dynamik-Kompression.

Behalten wir den 6-Kanal-Ton bei (nur bei AAC), sieht die **Azid1**-Seite so aus:

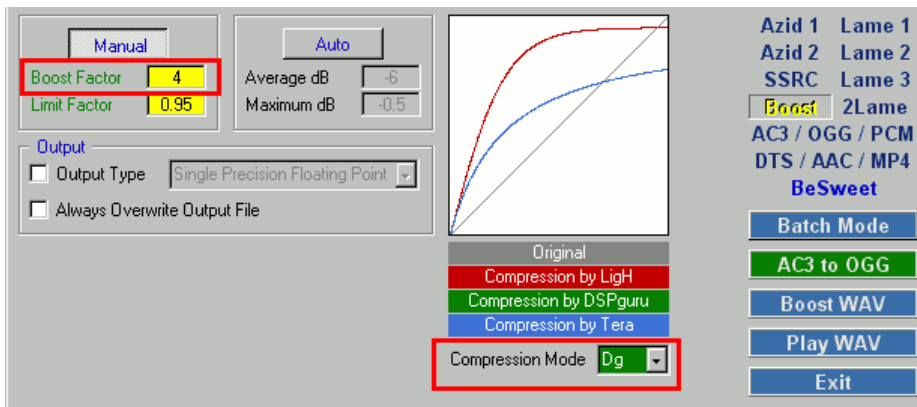


Die **3/2** bei **Decode (Front/Rear)** weist Azid an, alle 6 Kanäle beizubehalten. Um den Sound nicht unnötig zu verfälschen, sollten wir eine geringere **Dynamic Compression** als für Stereo nehmen. Da mir die Dynamik der AC3 manchmal einfach zu hoch ist, bevorzuge ich eine leichte Kompression. Man könnte sie auch ganz deaktivieren.

Die **Azid2**-Seite stellen wir immer so ein:



Wer die Dynamikkompression selbst konfigurieren möchte, nimmt anstatt Azids **Dynamic Compression Boost**. Achtung! Nicht beides zusammen verwenden!

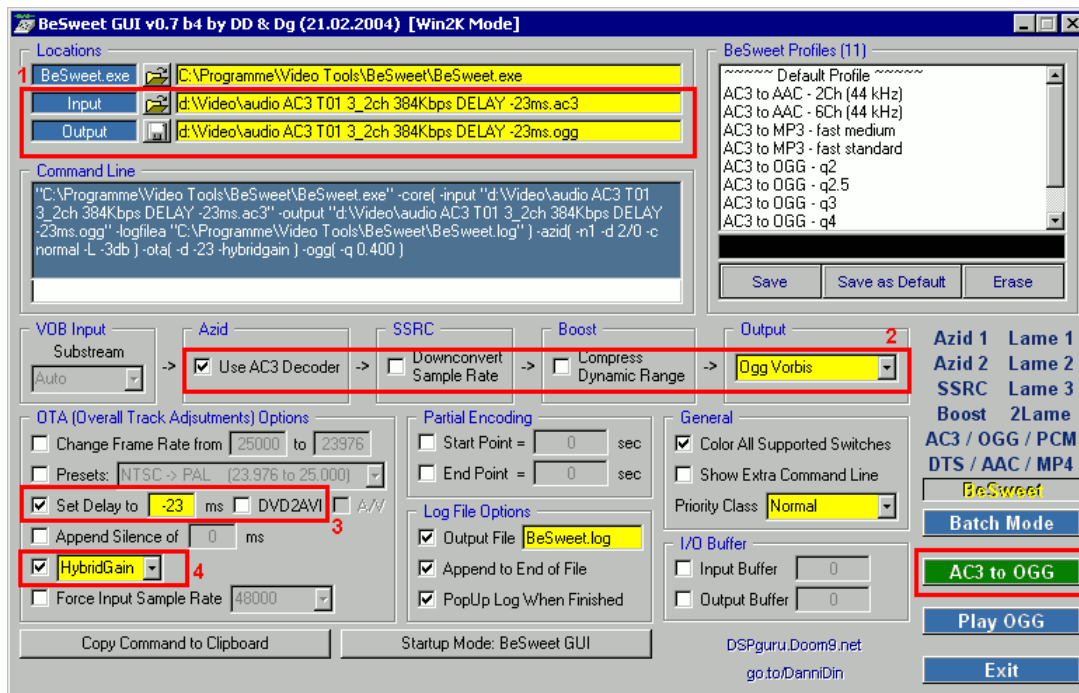


Boost hat es in sich. Wir können damit das letzte bisschen Dynamik aus der Tonspur herauskomprimieren, was in meinen Ohren schon nicht besonders gut klingt. Dazu kommt die Gefahr, durch zu viel Kompression Störgeräusche im Sound zu erzeugen. Deshalb sollten wir uns gut überlegen, ob der Einsatz von Boost wirklich nötig ist. Der Weg über Azids **Dynamic Compression** ist meistens der bessere. Ich habe Boost noch nie verwendet und habe das bis jetzt auch noch nicht bereut.

Wenn's denn sein soll, LigH schlägt im Doom9/Gleitz-Forum¹ folgendes vor: **Compression Mode LigH** mit einem **Boost Factor** um 3 macht Sinn, wenn der ursprüngliche Ton schon Stereo ist. Für 6-Kanal ist dagegen **Dg** mit **Boost Factor** um 4 sinnvoller. **Tera** eignet sich eher für Experimente.

Weiter geht's auf die **BeSweet**-Seite.

1) <http://forum.gleitz.info/showpost.php?p=41321&postcount=2>



Hier stellen wir bei (1) Quell- und Zieldatei ein. Vorsicht dabei, die GUI ist etwas eigenwillig programmiert. Eine zusätzliche Kontrolle, ob die Dateinamen auch wirklich passen, kann nicht schaden. Bei der ersten Verwendung der GUI müssen wir evtl. auch den Pfad zur BeSweet.exe angeben.

Bei (2) stellen wir rechts das gewünschte Sound-Ausgabeformat ein, also **MP3**, **Ogg Vorbis** oder **MP4 (AAC)**. Ein Haken bei **Compress Dynamic Range** aktiviert die **Boost**-Einstellungen, **Downconvert Sample Rate** sollte abgeschaltet bleiben. Wenn eine solche Konvertierung nötig wird (bei AAC), erkennt das BeSweet automatisch. Um nach dem Transcoding feststellen zu können, ob Fehler aufgetreten sind, sollten wir die **Log File Options** aktivieren.

Damit kommen wir zu (3). **Delay** ist der Wert, um den die Audiospur zum Video verschoben sein muss, um exakt synchron zu werden. Diese Angabe steht im Dateinamen, »DELAY -23ms« in unserem Fall. Diesen Wert (incl. Vorzeichen!) übernehmen wir in das gelbe Feld.

Das Delay an dieser Stelle schon zu berücksichtigen ist nicht zwingend. Wenn wir später Audio und Video zusammenmuxen, können wir es auch dort angeben. Wichtig ist: Nur eine der beiden Methoden verwenden! Wer das Delay mit BeSweet abhandelt, darf es später nicht noch ein zweites Mal berücksichtigen. Anders herum genauso: Wer es nicht in BeSweet abhandelt, muss es dann später beim Muxen tun.

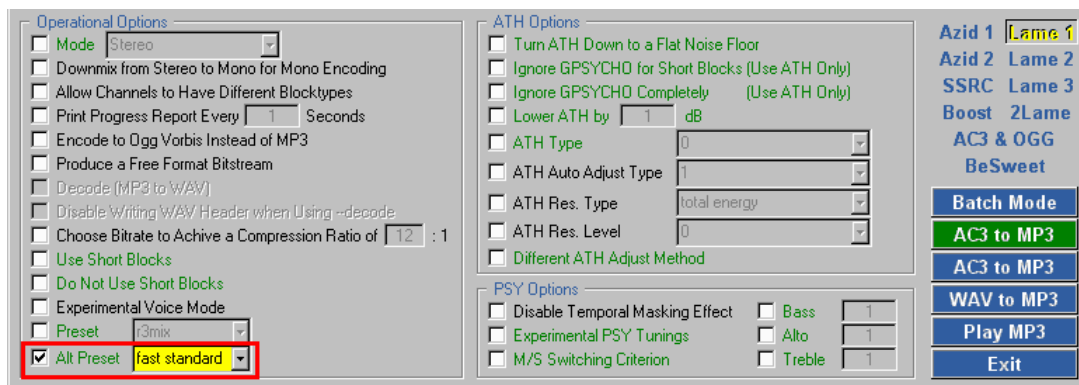
Abschließend müssen wir den Sound noch normalisieren, d. h. die Lautstärke auf 100% anheben. Durch die Dynamic Compression haben wir zwar schon die Lautstärke-Unterschiede innerhalb der Tonspur angeglichen, insgesamt ist sie aber immer

noch viel zu leise. Deshalb stellen wir **HybridGain** (4) ein. Die Ausnahme bildet AAC, für das **PreGain** die richtige Wahl ist. Der Unterschied kommt daher, dass HybridGain gleich am Anfang des Transcodings je nach Quelldatei einen festen Wert auf die Lautstärke aufschlägt und die Differenz zu 100 % in einem PostGain-Tag in der Datei speichert. Der Audiodecoder muss dann später beim Abspielen des Films diesen Tag auslesen und die Lautstärke noch entsprechend erhöhen. Leider gibt es noch keinen AAC-Decoder, der mit PostGain-Tags umgehen kann. Deshalb muss für dieses Format BeSweet die komplette Arbeit erledigen.

Über den grünen Button rechts (**AC3 to OGG**, die Beschriftung passt sich jeweils den Audioformaten an) können wir die Konvertierung starten – halt! Später. Erst müssen wir noch den passenden Encoder konfigurieren.

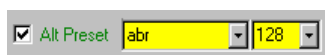
MP3 mit dem Lame-Encoder

Lame encodiert den Sound in MP3. Wer sich (nicht nur über MP3) ein bisschen einlesen möchte, sollte einmal bei www.HydrogenAudio.org vorbeischaun.



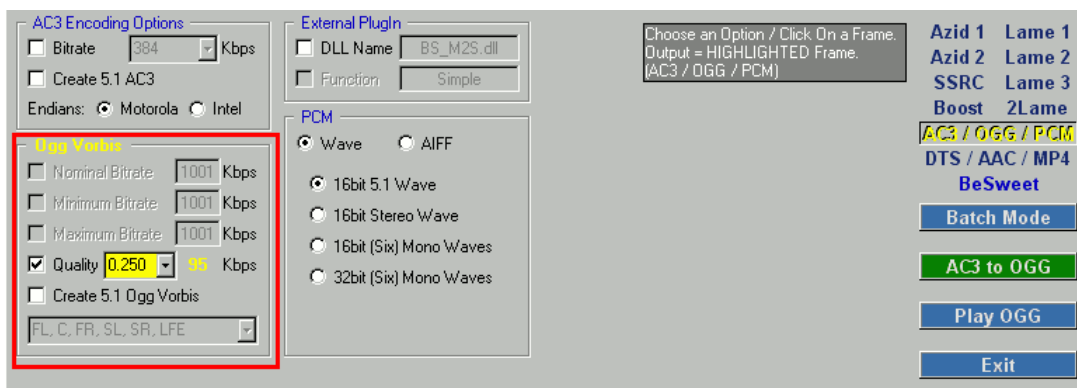
Mehr als die Alt Presets brauchen wir nicht. Die bieten für jeden Film eine passende Einstellungen und sind sehr intensiv getestet. Für einen normalen 2-CD-Rip eignet sich die **alt-preset fast standard**-Einstellung gut. Wer mit einem schnellen Rechner ausgestattet ist, kann das **fast** weglassen, um noch ein bisschen mehr Qualität heraus zu kitzeln. Der Unterschied ist aber marginal, dafür läuft **fast standard** mehr als doppelt so schnell. Das Preset erzeugt eine MP3-Datei mit variabler Bitrate und einwandfreier Qualität, deren Bitrate bei etwa 140 – 160 kbit liegt. Mehr ist wirklich nicht nötig.

Gerade für 1-CD-Rips kann das Standard-Preset aber schon einmal zu viel sein, dann müssen wir die Bitrate weiter drücken. **Alt Preset abr 128** dürfte sich dafür gut eignen.



Abr steht für *average bitrate*, d. h. die Bitrate ist variabel, ihr Durchschnitt wird vom Encoder aber auf den angegebenen kbit-Wert getrimmt wird. Schließlich können wir noch **cb**r auswählen. Das steht für *constant bitrate*, d. h. die Bitrate ist für die komplette Datei festgelegt und ändert sich nicht. Variable Bitrate erzeugt bei gleich großen Dateien bessere Qualität, deswegen sollten wir an CBR nicht zu viele Gedanken verschwenden.

Vorbis

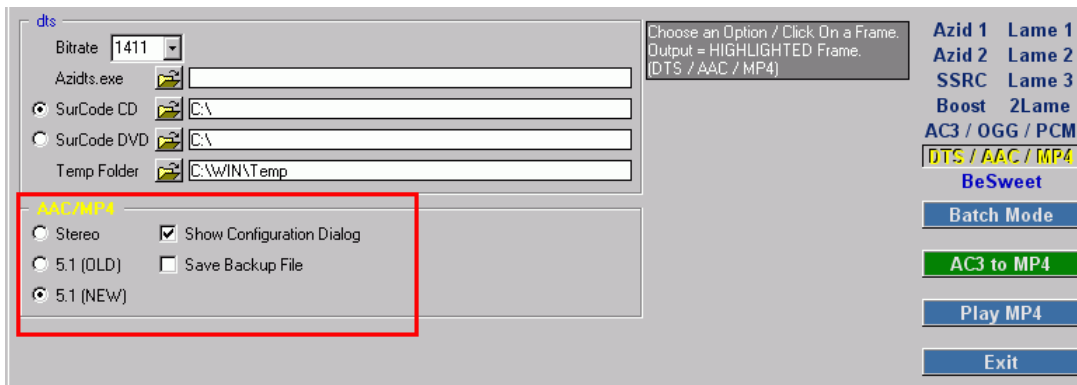


Ähnlich simpel wie bei Lame sind die Vorbis-Einstellungen. Wir müssen uns nur für den gewünschten **Quality**-Level entscheiden. **0.200** ergibt etwa 80 kbit/s und ist als sichere Untergrenze recht brauchbar. **0.500** liefert uns etwa 135 – 140 kbit/s und taugt gut als Obergrenze.

AAC mit dem Nero-Encoder

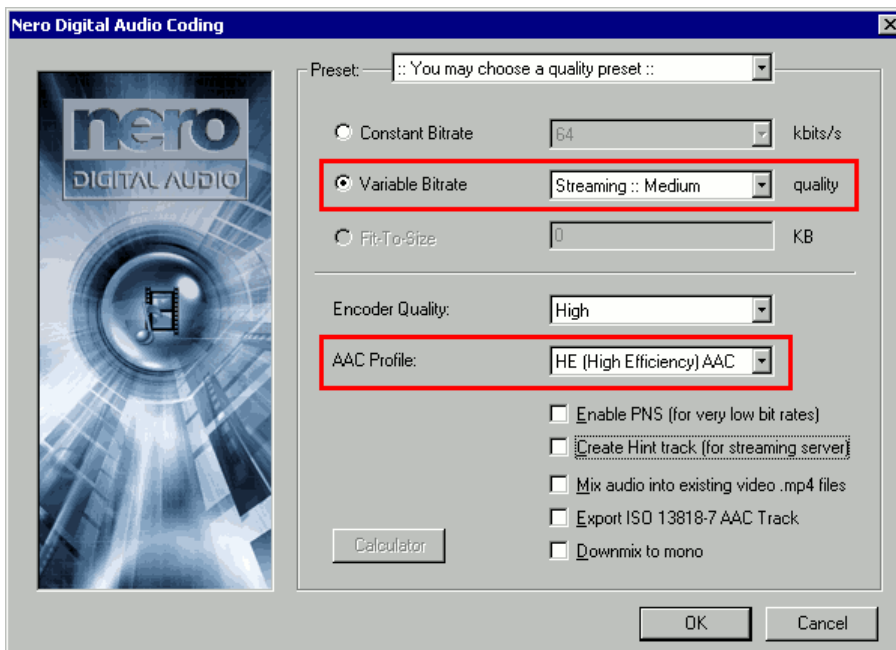
Um den Nero-Encoder verwenden zu können, kopieren wir erst die *Aac.dll* und *aacenc32.dll* aus *C:\Programme\Gemeinsame Dateien\Ahead\AudioPlugins* in den BeSweet-Ordner, bei älteren Nero-Versionen auch die *NeroIPP.dll* aus *C:\Programme\Gemeinsame Dateien\Ahead\Lib*.

Die Konfiguration erledigen wir auf der **DTS / AAC / MP4**-Seite.



Stereo dürfte sich selbst erklären. Wollen wir 6-Kanal-Ton und arbeiten mit einer älteren Nero-Version bis 6.0.0.11, wählen wir **5.1 (OLD)**, bei neueren Neros nehmen wir **5.1 (NEW)**.

Bei ausgeschaltetem Show Configuration Dialog übernimmt Nero ohne Nachfrage die Einstellungen vom letzten Encoding. Ist die Option aktiviert, öffnet sich der Einstellen-Dialog des Codecs. Das passiert allerdings erst einige Minuten nach dem Encoding-Start, wenn der Normalisierungsdurchlauf abgeschlossen ist.



Als **AAC Profile** stellen wir **HE (High Efficiency)** ein. **Variable Bitrate** regelt die Qualität/Dateigröße. Für Stereo-Ton liefert uns das **Streaming**-Profil kompakte 75 kbit/s, das ist wenig genug. Für 6 Kanäle können wir circa diese Bitraten erwarten:

Tape	115
Radio	128
Internet	160
Streaming	210

Dabei klingt **Tape** immer noch gut. Meistens verwende ich **Internet**. HE funktioniert übrigens nicht mit höheren Einstellungen als **Streaming**. Nero schaltet dann automatisch auf LC um.

Transcoding starten

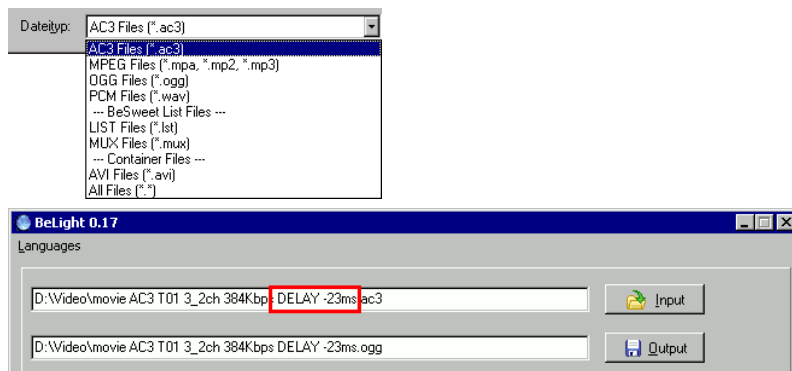
Damit sind alle Einstellungen abgeschlossen. Wir wechseln zurück auf die **BeSweet**-Seite und starten den kompletten Transcoding-Vorgang mit dem grünen Button. (Bei AAC daran denken, dass der Codec-Dialog erst nach der Normalisierung auftaucht.) Je nach Rechenleistung kann das eine Zeit lang dauern. Den ganzen Vorgang wiederholen wir für evtl. weitere Audiospuren.

Zwar hätten wir das Audio-Transcoding auch mit Gordian Knot erledigen können, allerdings stehen uns dort lange nicht alle Optionen von BeSweet zur Verfügung. Außerdem lässt sich das Video schöner kalkulieren, wenn die Größen der Audiospuren schon exakt feststehen.

2.3.2 Transcoding mit BeLight

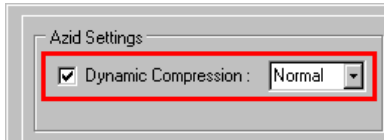
Sound decodieren und bearbeiten

Wir starten BeLight und öffnen über den **Input**-Button oben rechts die Quelldatei. Im Öffnen-Fenster müssen wir aufpassen, dass auch der korrekte **Dateityp** eingestellt ist, sonst erscheint die Datei nicht. Für DVDs dürfte das voreingestellte **AC3** meistens passen.



BeLight öffnet die Quelldatei und trägt automatisch einen Namen für die Zieldatei ein, den wir natürlich manuell anpassen können. Die Dateierweiterung verändert sich je nach gewähltem Zielformat.

Nun betrachten wir die linke Seite des BeLight-Fensters.



Der erste Eintrag gehört zu Azid. Azid ist dafür zuständig, die AC3-Tonspur zu decodieren und eventuell 6-Kanal auf 2-Kanal (Stereo) umzurechnen.

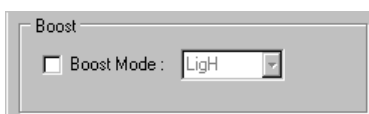
Anmerkung: Die Bezeichnungen der Tonformate sind etwas irreführend. 6-Kanal und 5.1 bezeichnet das selbe. Nur zählt man einmal alle Kanäle (vorne links und rechts, hinten links und rechts, Center und LFE), wogegen bei der 5.1-Schreibweise der Basskanal (LFE) extra dargestellt wird.

Mit **Dynamic Compression** stellen wir die gewünschte Dynamikkompression für die AC3 ein. Als Dynamik des Sounds bezeichnet man die Unterschiede zwischen lauten und leisen Passagen. Je höher die Dynamik, desto höher ist der Lautstärken-Unterschied zwischen leisen und lauten Passagen. Die Audio-Spur eines Films hat von Natur aus eine recht hohe Dynamik. Das wird ganz klar, wenn wir uns den Show-down mit knatternden Maschinengewehren und die geflüsterte Liebesszene im Vergleich vorstellen.

Beim Downmix auf Stereo würden wir ohne Kompression eine nervig leise Zieldatei bekommen, deshalb gleichen wir mit der **Dynamic Compression** die Unterschiede in der Lautstärke an. D. h. im Extremfall hört sich das Flüstern genauso laut an wie die Maschinengewehr-Salve. Mit der Einstellung **normal** treiben wir es lange nicht so weit und erreichen eine für Stereo angemessene Dynamik-Kompression.

Wollen wir den 6-Kanal-Ton beibehalten, sollten wir eine geringere **Dynamic Compression** als für Stereo nehmen, um den Sound nicht unnötig zu verfälschen. Da mir die Dynamik der AC3 manchmal einfach zu hoch ist, bevorzuge ich eine leichte (**light**) Kompression. Man könnte sie auch ganz deaktivieren (Haken wegklicken).

Wer die Dynamikkompression selbst konfigurieren möchte, nimmt anstatt Azids Dynamic Compression Boost. Achtung! Nicht beides zusammen verwenden!

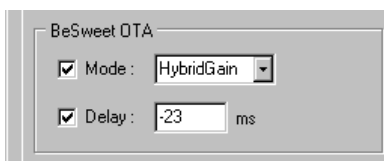


Boost hat es in sich. Wir können damit das letzte bisschen Dynamik aus der Tonspur herauskomprimieren, was in meinen Ohren schon nicht besonders gut klingt. Dazu kommt die Gefahr, durch zu viel Kompression Störgeräusche im Sound zu erzeugen. Deshalb sollten wir uns gut überlegen, ob der Einsatz von Boost wirklich nötig ist. Der Weg über Azids **Dynamic Compression** ist meistens der bessere. Ich habe Boost noch nie verwendet und habe das bis jetzt auch noch nicht bereut.

Wenn's denn sein soll, LigH schlägt im Doom9/Gleitz-Forum² folgendes vor: **Boost Mode LigH** macht Sinn, wenn der ursprüngliche Ton schon Stereo ist. Für 6-Kanal-Quellen ist dagegen **Dg** sinnvoller. **Tera** eignet sich eher für Experimente.



Mit SSRC können wir die Abtastfrequenz des Tons von den üblichen 48 kHz in einen anderen Wert umrechnen. Da die Soundkarten inzwischen ausgestorben sind, die mit 48 kHz nicht umgehen können, dürfen wir den Punkt ignorieren. Zwar braucht HE-AAC eine Umrechnung auf 44,1 kHz, die erledigt BeSweet aber automatisch.



Nun müssen wir den Sound noch normalisieren, d. h. die Lautstärke auf 100 % anheben. Durch die Dynamic Compression von oben haben wir zwar schon die Lautstärken-Unterschiede innerhalb der Tonspur angeglichen, insgesamt ist sie aber immer noch viel zu leise. Das Anheben erledigt **Mode** im **OTA**-Abschnitt.

Haben wir uns für AAC als Zielformat entschieden, heißt die richtige Einstellung **PreGain**, für alle anderen Formate **HybridGain**. Der Unterschied kommt daher, dass HybridGain gleich am Anfang des Transcodings je nach Quelldatei einen festen Wert auf die Lautstärke aufschlägt und die Differenz zu 100 % in einem PostGain-Tag in der Datei speichert. Der Audiodecoder muss dann später beim Abspielen des Films diesen Tag auslesen und die Lautstärke noch entsprechend erhöhen. Leider gibt es noch keinen AAC-Decoder, der mit PostGain-Tags umgehen kann. Deshalb muss für dieses Format BeSweet die komplette Arbeit erledigen.

Damit kommen wir zum **Delay**. Das ist der Wert, um den die Audiospur zum Video verschoben sein muss, um exakt synchron zu werden. Diese Angabe steht im Dateinamen, »DELAY -23ms« in unserem Fall.³ Diesen Wert übernimmt BeLight automatisch.

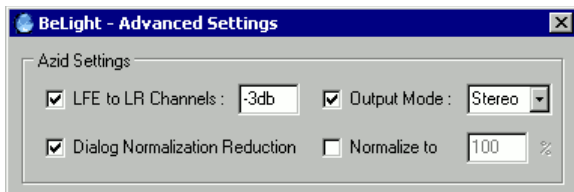
2) <http://forum.gleitz.info/showpost.php?p=41321&postcount=2>

3) siehe Screenshot S. 18

Eine Kontrolle, ob der Wert im Dateinamen auch mit dem eingetragenen übereinstimmt, kann trotzdem nicht schaden.

Das Delay an dieser Stelle schon zu berücksichtigen ist nicht zwingend. Wenn wir später Audio und Video zusammenmuxen, können wir es auch dort angeben. Wichtig ist: Nur eine der beiden Methoden verwenden! Wer das Delay mit BeSweet abhandelt, darf es später nicht noch ein zweites Mal berücksichtigen. Anders herum genauso: Wer es nicht in BeSweet abhandelt, muss es dann später beim Muxen tun.

Damit kommen wir zu den **Advanced Settings**.



Für den Downmix auf Stereo sollten die Optionen so aussehen wie oben. **LFE to LR Channels** bestimmt, mit welchem Pegel der Basskanal in die vorderen beiden Kanäle gemixt werden soll. Um ein Zuviel an Bass zu vermeiden, stellen wir hier **-3db** ein.

Mit dem **Output Mode** definieren wir, welche Art von Stereo erzeugt wird: **Mono**, normales **Stereo**, Dolby Pro Logic (**DPL**) oder Dolby Pro Logic II (**DPL2**). Pro Logic codiert Surroundinformationen in die beiden Stereokanäle, so dass – einen passenden Decoder vorausgesetzt – beim Abspielen zumindest ein Teil der ursprünglichen Surroundinformationen wieder hergestellt werden kann.

Behalten wir die ursprünglichen 6 Kanäle bei, entfernen wir sowohl bei **LFE to LR Channels** als auch bei **Output Mode** die Haken. Dann haken wir noch ganz unten rechts **Output Log file** an und können das Optionenfenster wieder schließen.

AC3-, MP2-, Wave-Encoding

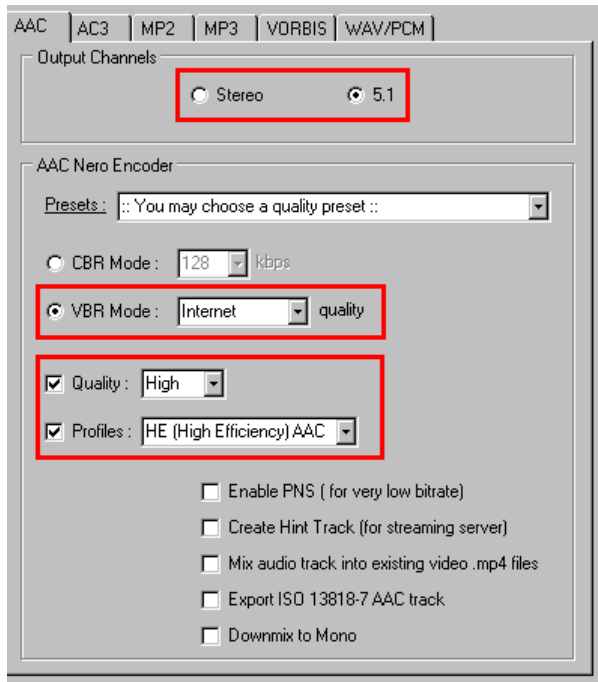
Alle dieser drei Möglichkeiten empfehlen sich nicht. BeSweets AC3-Encoder erzeugt nicht gerade beste Qualität und dazu viel zu leise Zieldateien. Deshalb macht es wenig Sinn, eine AC3-Datei von der DVD in eine andere AC3 (z. B. mit geringerer Bitrate) umzurechnen.

Das ältere MP2-Format ist höchstens für (S)VCDs interessant, mit denen wir und hier nicht beschäftigen. Und decodierte Waves taugen als Basis, um den Sound intensiver weiterzuverarbeiten, fürs direkte Muxing dagegen schon wegen der Größe überhaupt nicht.

AAC mit dem Nero-Encoder

Um den Nero-Encoder verwenden zu können, kopieren wir erst die *Aac.dll* und *aacenc32.dll* aus *C:\Programme\Gemeinsame Dateien\Ahead\AudioPlugins* in den BeS-weet-Ordner, bei älteren Nero-Versionen auch die *NeroIPP.dll* aus *C:\Programme\Gemeinsame Dateien\Ahead\Lib*.

Die Konfiguration erledigen wir auf der **AAC**-Registerkarte.



Unter **Output Channels** wählen wir, ob die Zieldatei **Stereo** oder 6 Kanäle (**5.1**) haben soll. Die **Quality** lassen wir immer auf **High** stehen.

Als **Profile** stellen wir **HE (High Efficiency) AAC** ein. **VBR Mode** regelt die Qualität/Dateigröße. Für Stereo-Ton liefert uns das **Streaming**-Profil kompakte 75 kbit/s, das ist wenig genug. Für 6 Kanäle können wir circa diese Bitraten erwarten:

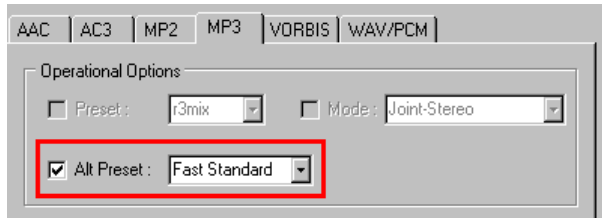
Tape	115
Radio	128
Internet	160
Streaming	210

Dabei klingt **Tape** immer noch gut. Meistens verwende ich **Internet**.

HE funktioniert übrigens nicht mit höheren Einstellungen als Streaming. Nero schaltet dann automatisch auf LC um.

MP3 mit dem Lame-Encoder

Lame encodiert den Sound in MP3. Wer sich (nicht nur über MP3) ein bisschen einlesen möchte, sollte einmal bei www.HydrogenAudio.org vorbeischaun.



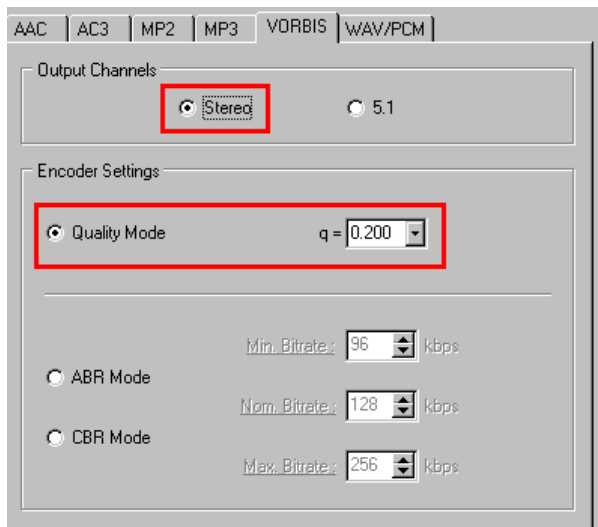
Mehr als die **Alt Presets** brauchen wir nicht. Die bieten für jeden Film eine passende Einstellungen und sind sehr intensiv getestet. Für einen normalen 2-CD-Rip eignet sich die **alt-preset fast standard**-Einstellung gut. Wer mit einem schnellen Rechner ausgestattet ist, kann das **fast** weglassen, um noch ein bisschen mehr Qualität heraus zu kitzeln. Der Unterschied ist aber marginal, dafür läuft **fast standard** mehr als doppelt so schnell. Das Preset erzeugt eine MP3-Datei mit variabler Bitrate und einwandfreier Qualität, deren Bitrate bei etwa 140 – 160 kbit/s liegt. Mehr ist wirklich nicht nötig.

Gerade für 1-CD-Rips kann das Standard-Preset aber schon einmal zu viel sein, dann müssen wir die Bitrate weiter drücken. **Alt Preset abr 128** dürfte sich dafür gut eignen.

Abr steht für *average bitrate*, d. h. die Bitrate ist variabel, ihr Durchschnitt wird vom Encoder aber auf den angegebenen kbit-Wert getrimmt wird. Schließlich können wir noch **cbr** auswählen. Das steht für *constant bitrate*, d. h. die Bitrate ist für die komplette Datei festgelegt und ändert sich nicht. Variable Bitrate erzeugt bei gleich großen Dateien bessere Qualität, deswegen sollten wir an CBR nicht zu viele Gedanken verschwenden.

Vorbis

Ähnlich simpel wie bei Lame sind die Vorbis-Einstellungen. Wir müssen uns nur für den gewünschten **Quality**-Level entscheiden. **0.200** ergibt etwa 80 kbit/s und ist als sichere Untergrenze recht brauchbar. **0.500** liefert uns etwa 135 – 140 kbit/s und taugt gut als Obergrenze.



Für 5.1-Output ist Vorbis nicht geeignet, da dieser Encodingmodus (noch?) reichlich unfertig und unoptimiert ist. Das heißt auch, um den ursprünglichen 6-Kanal-Ton beizubehalten bleiben uns zwei Möglichkeiten: AAC oder die originale AC3.

Transcoding starten

BeLight transcodiert immer in das Format, dessen Registerkarte gerade angezeigt wird. Da darf also nichts mehr verstellt werden, um keine unangenehmen Überraschungen zu erleben.

Da wir alle Einstellungen erledigt haben, bleibt nur noch ein Druck auf den **Start**-Button unten rechts, um BeSweet in Aktion treten zu lassen. Je nach Rechenpower kann das einige Zeit dauern.

Den ganzen Vorgang wiederholen wir für evtl. weitere Audiospuren.

Zwar hätten wir das Audio-Transcoding auch mit Gordian Knot erledigen können, allerdings stehen uns dort lange nicht alle Optionen von BeSweet zur Verfügung. Außerdem lässt sich das Video schöner kalkulieren, wenn die Größen der Audiospuren schon exakt feststehen.

2.4 Untertitel: Eine Einführung

Untertitel auf der DVD

Die einfachste Art Untertitel sind die, die im technischen Sinn gar keine Untertitel darstellen. Die sind auch auf der DVD schon fest ins Bild »eingebrennt«, d. h. sie liegen nicht getrennt von der Videospur vor. Solche Untertitel müssen wir so wie sie sind hinnehmen. Die Möglichkeiten, an denen etwas zu ändern, sind so gut wie nicht vorhanden. Es sei denn, wir wollten direkt ans Videobild Hand anlegen.

Interessant für uns sind echte Untertitel, die in einer separaten Untertitel-Spur auf der DVD liegen. Einer Bearbeitung steht dadurch nichts im Weg. Echte Untertitel teilen sich in drei Arten ein.

- **Dialog-Untertitel**, die die kompletten Dialoge des Films abbilden und auch alle zusätzlichen Untertitelungen enthalten (z. B. die Zeitangaben in *Spy Game*). Das ist die typische All-Inclusive-Untertitelspur.
- **Szene-Untertitel**, die nur Zusatzinformationen außerhalb der Dialoge und Übersetzungen fremder Sprachen liefern (wer wird da an *Herr der Ringe* denken ...). Auch Szene-Untertitel liegen in einer separaten Spur. Iirc sind die Elben-Übersetzungen in *Die Gefährten* genau auf diese Weise gespeichert.
- **Forced Subs**, die die Szene-Untertitel in der Spur der Dialog-Untertitel verpacken. Soll heißen, wir haben eine Dialog-Untertitel-Spur. Für einige der Untertitel ist ein Forced-Flag gesetzt, so dass die (und nur die) auch dann angezeigt werden, wenn beim Anschauen die Untertitel abgeschaltet sind. Jeder, der kein Russisch kann, wäre sonst bei *Der Anschlag* reichlich aufgeschmissen.

Für die Verarbeitung müssen wir erst einmal wissen oder herausfinden, auf welche Weise die Untertitel auf der DVD gespeichert sind. Am häufigsten sind All-Inclusive-Untertitelspuren mit einigen Forced Subs. Verlassen können wir uns darauf aber nicht. Leider lässt sich vor dem Rippen oft nicht erkennen, welche Untertitelspur exakt was enthält, so dass uns ein wenig Detektivarbeit nicht erspart bleibt. Ich habe mir deswegen angewöhnt, erst einmal sämtliche Untertitel auf die Platte zu ziehen und hinterher auszusortieren.

Untertitel im Encoding

Nun stehen wir vor der Wahl, welche Untertitel im endgültigen Film vorhanden sein sollen. Eines ist klar: Szene-Untertitel bzw. Forced Subs sollten wir nie weglassen, weil die für das Verständnis des Films wichtig sein können. Dialog-Untertitel dagegen sind hauptsächlich für Dinge wie *Originalton mit Untertiteln* interessant. Im Normalfall können wir auf die auch ganz gut verzichten.

Die zweite Entscheidung betrifft das Format der Untertitel im fertigen Film. Auf der DVD sind sie als Bilder gespeichert, die über den Film geblendet werden. Für unseren fertigen Film gibt es drei Möglichkeiten, Untertitel zu übernehmen.

- Bilduntertitel der DVD (VobSubs) fest ins Video einbrennen (sind dann nicht mehr ausblendbar).
- VobSubs als dynamische (ausblendbare) Untertitel übernehmen.

- Bilduntertitel in Text umwandeln und in dieser (dynamischen) Form übernehmen.

Wann ist denn was sinnvoll? Das kommt sehr darauf an, wie unser fertiger Film aussehen soll. Wenn wir uns auf eine Tonspur beschränken, können wir Szene/Forced-Untertitel (wenn der Film überhaupt solche hat) der Einfachheit halber fest ins Bild einbrennen. Bei mehreren Tonspuren in verschiedenen Sprachen bietet sich auch hier das dynamische Einbinden an, um später die Untertitel passend zur jeweiligen Sprache vorrätig zu haben. Für komplette Dialog-Spuren ist die dynamische Methode dagegen immer erste Wahl, um den Film auch einmal ohne Untertitel anschauen zu können.

Anmerkung: Ich mag eingebrannte Szene-Untertitel auch bei einsprachigen Filmen nicht, da die Schrift meistens doch spürbar unter dem Hochzoomen aufs Vollbild leidet. Ein Textuntertitel, der mit einer TrueType-Schrift in passender Größe dargestellt wird, sieht deutlich sauberer aus. Zweiter Vorteil: Dynamische Untertitel müssen nicht direkt im Bild erscheinen, sondern können in den schwarzen Balken über/unter dem Video eingeblendet werden. Der gleichmäßige Hintergrund verbessert die Lesbarkeit noch einmal.

Für die beiden dynamischen Varianten haben wir zusätzlich die Möglichkeit, die Untertitel als extra Datei zu speichern oder als Untertitel-Spur ins fertige Video zu muxen. Die extra Datei hat den Nachteil, dass wir den Film nicht gut verpackt als einzelne Datei vorliegen haben. Außerdem müssen die Untertitel-Dateien genau richtig benannt werden (meistens genauso wie die Filmdatei), damit sie der Player automatisch erkennt. Das Muxen vermeidet diese Probleme, deswegen würde ich separate Untertitel-Dateien nach Möglichkeit vermeiden. Allerdings lassen sich VobSubs nicht gleich über Gordian Knot muxen, sondern wir müssen das manuell erledigen. Mit diesem Grundwissen können wir uns jetzt an die Verarbeitung wagen. Das nächste Kapitel beschäftigt sich mit fest im Bild eingebrannten Untertiteln, das folgende dann mit dynamischen, die sich ein- und ausblenden lassen.

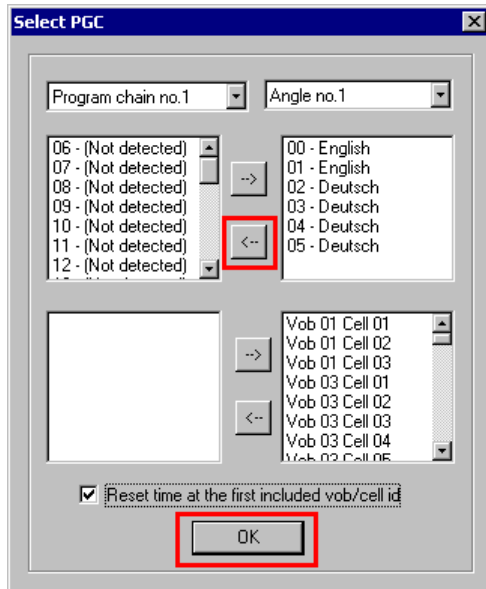
2.4.1 Vorbereiten von festen Untertiteln

Feste Untertitel sind permanent im Video verankert und nicht mehr ausblendbar. Damit wollen wir uns jetzt beschäftigen. Ausblendbare (dynamische) Untertitel behandelt das nächste Kapitel.

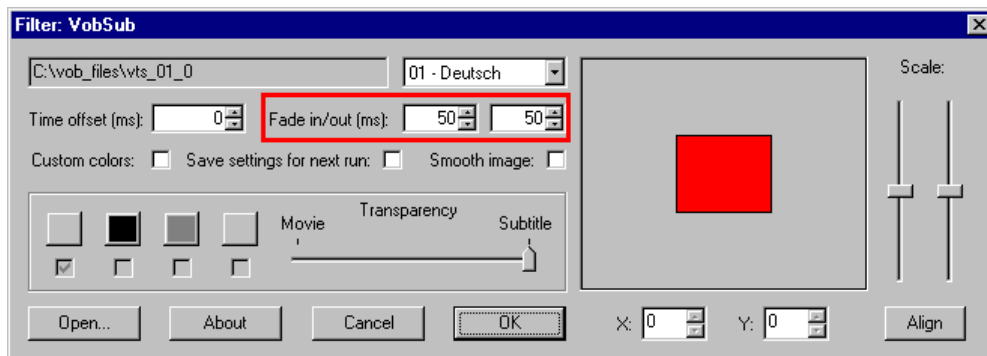
Wir starten VobSub, entweder über Gordian Knot, Register **Subtitles/Chapters** und den **Configure**-Button links oben, oder über **Start / Ausführen** mit diesem Befehl:
rundll32.exe vobsub.dll,Configure.

Über den **Open**-Button unten links gelangen wir in den Öffnen-Dialog, stellen bei Dateityp auf **ifo and vobs, for creating idx/sub** um und wählen die vom Rip-

per erstellte ifo-Datei aus, die sich `vts_01_0.ifo` oder ähnlich nennt. Es folgt ein Fenster, in dem wir den Speicherort der Untertitel angeben. Dann kommt dieses Fenster:



Im Bereich oben rechts stehen die Untertitel, die aus den VOBS gerippt werden sollen. Alles, was wir nicht beim Rippen von der DVD gezogen haben, können wir schon mal mit dem <-- Button abwählen. Wer genau weiß, in welcher Spur »seine« Untertitel liegen, kann auch gleich nur diese eine wählen. Wenn wir fertig sind, klicken wir auf **OK**. VobSub zieht jetzt die Untertitel aus den VOBS heraus und schreibt sie in eine sub-Datei. Zusätzlich wird eine idx-Datei erstellt, die alle Zeitmarken enthält, an denen Untertitel auftauchen sollen. Dann befinden wir uns wieder im Hauptfenster.



Normalerweise blendet VobSub jeden Untertitel kurz ein und wieder aus. Die Standard-Einstellung von 50ms sieht man kaum, sie lässt das Auftauchen und Verschwinden der Untertitel aber nicht ganz so hart erscheinen wie ohne Fade. Ändern lassen sich die zwei Werte über **Fade in/out**. Danach **OK** klicken.

Nun öffnen wir mit einem Texteditor die idx-Datei. Sie nennt sich `vts_01_0.idx` oder ähnlich, jedenfalls mit dem `_0` hinten dran. Jetzt wird wichtig, wie Dialog- und Szene-Untertitel gespeichert sind.

Dialog- und Szene-Untertitel in *verschiedenen* Spuren

Jede der vielen Zeilen im Bild unten stellt eine Zeitmarke dar, an der ein Untertitel erscheint. Wir suchen die richtige Sprache aus den Daten oberhalb heraus. Gibt es mehrere Spuren für eine Sprache, ist ein bisschen Detektivarbeit angesagt. Die Spur mit den kompletten Untertiteln dürfte ziemlich umfangreich sein, die mit den Szene-

```
# English
id: en, index: 0
# Decoment next line to activate alternative name in DirectVobSub
# alt: English
# Vob/Cell ID: 1, 1 (PTS: 0)
timestamp: 00:00:45:880, filepos: 000000800
timestamp: 00:00:49:040, filepos: 000001000
timestamp: 00:00:51:120, filepos: 000003000
timestamp: 00:00:53:400, filepos: 000005000
timestamp: 00:00:56:120, filepos: 000007000
timestamp: 00:00:59:960, filepos: 000009000
timestamp: 00:01:03:960, filepos: 00000b800
timestamp: 00:01:07:120, filepos: 00000e800
```

Untertiteln müsste recht kurz ausfallen. Haben wir die richtige Spur gefunden, dann merken wir uns die Zahl hinter **index** und suchen diese Zeilen (direkt über der ersten Untertitel-Spur):

```
# Language index in use
langidx: 0
```

Hinter **langidx** tragen wir die gemerkte Zahl ein.

Dialog- und Szene-Untertitel in *einer* Spur (Forced Subs)

Auch in diesem Fall suchen wir erst einmal wie oben die passende Sprache heraus und tragen die **index**-Nummer bei **langidx** ein. Dann springen wir zu diesen Zeilen:

```
# ON: displays only forced subtitles, OFF: shows everything
Forced subs: OFF
```

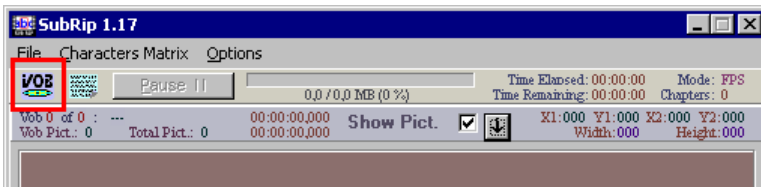
Da ersetzen wir das **OFF** durch **ON**, um die Forced Subs einzuschalten. Fertig. Zum Schluss die idx-Datei speichern. Damit sind die Untertitel erst einmal fertig.

2.4.2 Erzeugen von dynamischen Text-Untertiteln

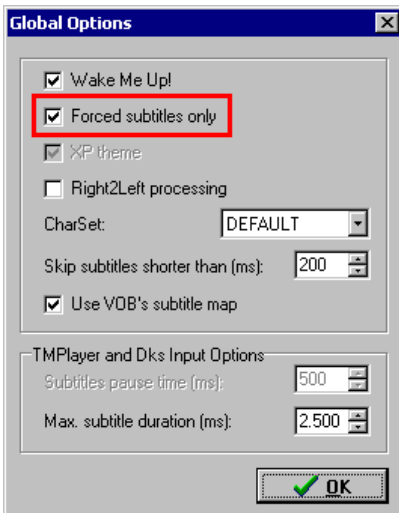
Dynamische Untertitel bestehen entweder aus Bildern in Form von VobSubs oder aus Text. VobSubs haben wir schon im letzten Kapitel besprochen. Das Vorgehen dort ist immer gleich, egal ob wir sie ins Video codieren oder als eigene Spur muxen wollen. Im zweiten Fall müssen wir auf das Muxing mit VirtualDubMod verzichten (das unterstützt keine VobSubs) und je nach Container⁴ mkvmerge (für Matroska) oder AVI-Mux GUI (für AVI) verwenden. OggMedia unterstützt keine VobSubs.

In diesem Kapitel kümmern wir uns um Text-Untertitel. Das benötigte Programm heißt SubRip, das nach dem Start dieses Fenster präsentiert:

4) vgl. 3.1.1

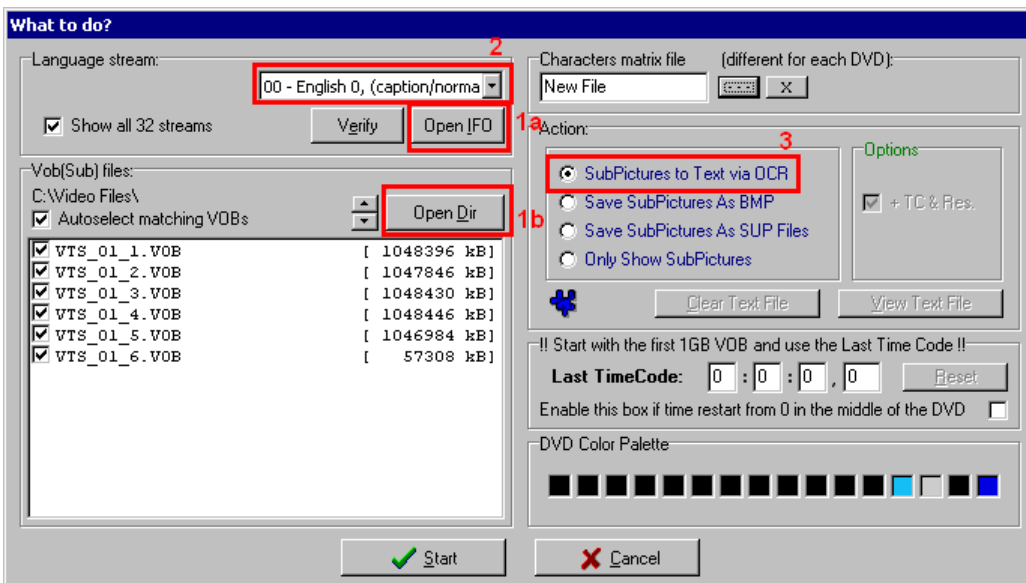


Über **Options** / **Global Options** gelangen wir in diesen Dialog:



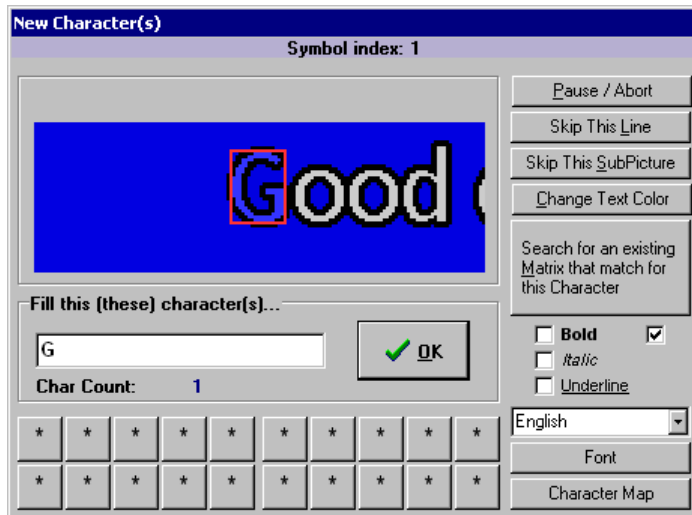
Mit einem Haken bei **Forced subtitles only** können wir, wie mit VobSub, aus einer kompletten Untertitelspur nur die Szene-Untertitel herausziehen, wenn die nicht in einer extra Spur gespeichert sind.

Im Hauptfenster rufen wir mit dem **VOB**-Button den Öffnen-Dialog auf.



Am einfachsten ist es nun, mit **open IFO** (1a) die vom Ripper erstellt Ifo-Datei zu öffnen. Alternativ können wir auch mit **Open Dir** (1b) die VOBs direkt laden. Unter (2) stellen wir die gewünschte Sprache ein und kontrollieren, dass bei (3) die gezeigte Option angehakt ist. Dann klicken wir auf **Start**.

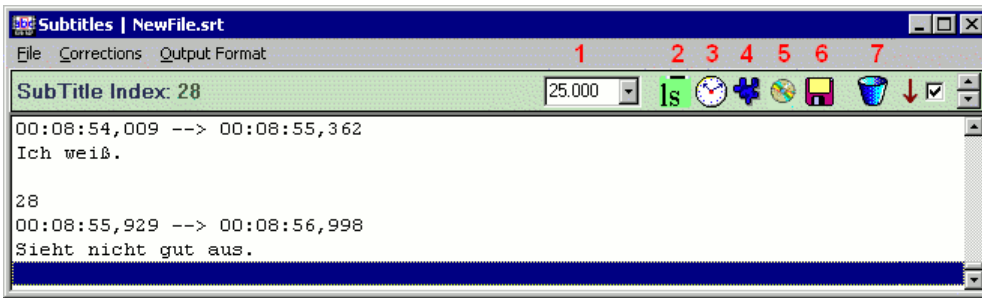
Da die Untertitel auf der DVD als Bilder gespeichert sind, muss sie SubRip per OCR-Texterkennung in Text umwandeln. Dabei erkennt er die Buchstaben aber nicht selbst wie die professionellen OCR-Programme, sondern fragt nach, welches Grafikmuster zu welchem Buchstaben gehört. Das geschieht in diesem Dialog:



Über den rot umrandeten Buchstaben (hier das »G«) ist sich SubRip im Unklaren. Also geben wir ihm im Eingabefeld in der Mitte den passenden Buchstaben an. Rechts daneben können wir noch fett, kursiv und unterstrichen als Auszeichnung wählen. Bei den ersten Untertiteln fragt SubRip noch bei fast jedem Buchstaben nach dessen Bedeutung. Eben so lange, bis mehr oder weniger jeder Buchstabe des Alphabets einmal vorgekommen und von uns identifiziert ist. Dann geht der Lesevorgang fast automatisch. Nur ab und zu stößt SubRip noch auf einen unbekanntes Buchstaben, macht sich dann aber mit blinkendem Fenster und Sound bemerkbar, wir können also zwischendurch ruhig am Computer weiterarbeiten. Viel länger als zehn Minuten dürfte der Vorgang insgesamt nicht dauern.

Wenn SubRip den letzten Untertitel erkannt hat, landen wir im Editor-Fenster, das sich am unteren Rand des Hauptfensters öffnet. Wer später noch Untertitel in anderen Sprachen rippen will, sollte jetzt im Hauptfenster mit **Characters Matrix / Save Characters Matrix File** die gerade beim Rippen erstellte Zeichen-Matrix sichern. In der sind die Informationen enthalten, welches Grafikmuster zu welchem Buchstaben gehört. Damit entfällt bei allen weiteren Untertiteln das erneute Eingeben der richtigen Buchstaben. Aber Achtung: Da die Schriftart der Untertitel von DVD zu DVD unterschiedlich ist, gilt eine Zeichen-Matrix immer nur für die gerade gerippte DVD. Für einen anderen Film müssen wir die Buchstabenerkennung wieder von vorne durchführen.

Damit zurück zum Editor-Fenster, in dem wir den gerade gerippte Untertitel in Textform (genauer: im SubRip-Format) sehen.



Ich gehe davon aus, dass wir die Untertitel-Spur am Ende auch im SubRip-Format speichern wollen. Dann können wir (1) ignorieren. Die richtige Bildrate (die uns DGIndex verraten hat) müssten wir dort nur einstellen, wenn wir die Untertitel in einem Frame-basierten Format wie MicroDVD speichern wollten. Mit dem Button (2) lassen sich die größten Fehler der Texterkennung korrigieren. Die ist zwar recht gut, aber von fehlerfrei weit entfernt. Bei (3) lassen sich Zeitkorrekturen vornehmen, wenn der Untertitel nicht synchron zum Film läuft. Das dürfte aber kaum vorkommen, wenn Untertitel und Film direkt von der DVD stammen. Mit (4) könnten wir die Untertitel in ein anderes Format konvertieren und (5) dient zum Splitten der Untertitel in mehrere Dateien. Nötig ist das nur, wenn wir die Untertitel in einer externen Datei abspeichern (ansonsten wird das Splitten später zusammen mit dem gesamten Film erledigt), und das wiederum dürfte nur nötig sein, wenn der Standalone-Player mit Untertitelspuren Stress macht. Und, nein, ich bin zu faul, das jetzt ausführlich zu erklären. Bleibt Button (6), der die aktuelle Untertitel-Datei speichert. Das sollten wir natürlich nicht vergessen.

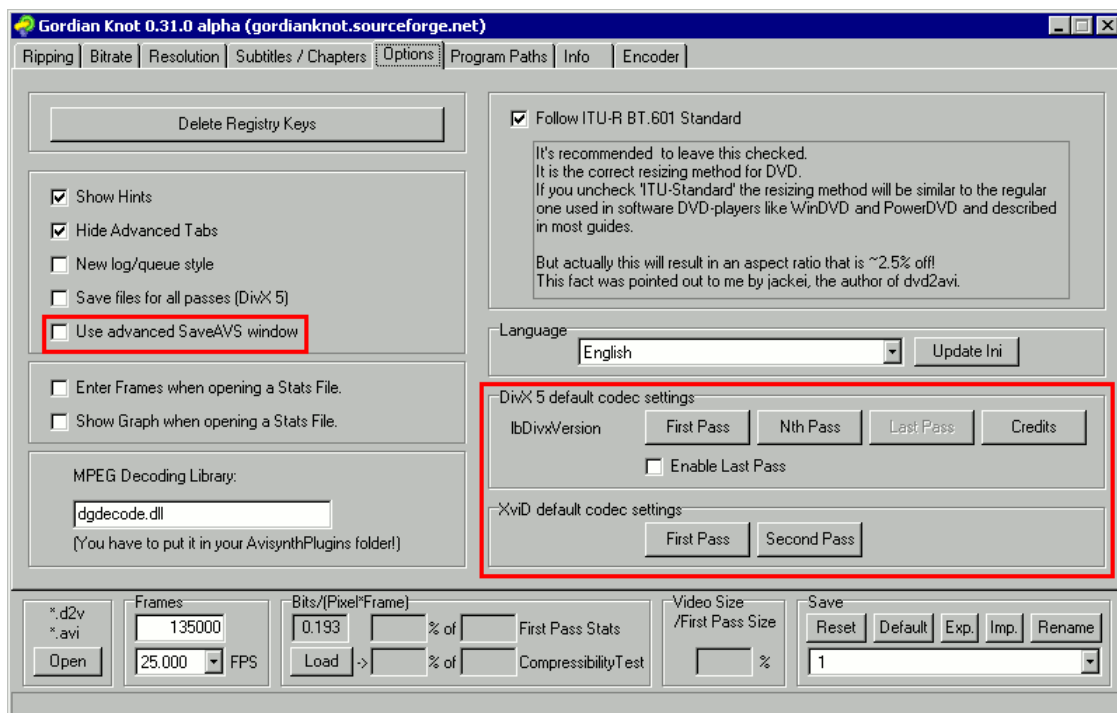
Wenn wir weitere Untertitel erkennen wollen: nach dem Speichern das Editorfenster über das Papierkorb-Symbol (7) leeren, mit dem **VOB**-Button den Öffnen-Dialog aufrufen, die neue Untertitelspur auswählen und auf der rechten Seite des Fensters oben die Character-Matrix-Datei laden und mit **Reset** den Timecode nullen, sonst passen die Zeitmarken nicht mehr.

Ganz zum Schluss bleibt nur noch, die Untertitel-Spur(en) in eine Textverarbeitung zu laden und durch die Rechtschreibprüfung zu jagen. Denn – wie gesagt – die Texterkennung ist zwar sehr gut, aber bei weitem nicht fehlerfrei. Beim Abspeichern müssen wir dann nur darauf achten, dass wir die Datei auch wieder als reinen Text sichern, denn mit Untertiteln im StarOffice- oder Word-Format kann kein Player etwas anfangen. ;-)

3 Encodieren mit Gordian Knot

3.1 Gordian Knot und Codecs konfigurieren

Nach dem Start von Gordian Knot wechseln wir erst einmal in das Register **Program Paths** und erklären GKnot, wo die verschiedenen Programme liegen. Die **Don't Check**-Checkbox verhindert, dass GKnot nach dem entsprechenden Programm sucht und sich beschwert, wenn kein Pfad dafür angegeben ist. Den Haken setzen wir bei allen Tools, die wir nicht verwenden und deshalb nicht installiert haben. Nandub ist ein Kandidat dafür, denn das bräuchten wir nur fürs DivX 3.11-Encoding. Dann wechseln wir ins Register **Options**.



Wichtig ist hier Zweierlei: Das abgeschaltete **advanced SaveAVS window**, da wir hier die einfachere Variante des Fensters verwenden, und die **default codec settings**. Die hier getroffenen Einstellungen gelten als Standard für alle Encodings, können aber im Einzelfall auch wieder überschrieben werden. Wer von vorne herein weiß, dass er nur einen der beiden Codecs verwendet, kann natürlich auf die Konfiguration des anderen verzichten.

Das Video wird unabhängig vom Codec in mindestens zwei Durchgängen (Passes) codiert. Im ersten Durchgang sammelt der Codec Informationen über das Video ohne den Film schon wirklich zu encodieren. Aus diesen Informationen wird eine möglichst gute Verteilung der verfügbaren Bitrate berechnet und im zweiten Durchgang der Film erzeugt. XviD ist an dieser Stelle fertig. Bei DivX sind noch zusätzliche Durchgänge möglich, um die Qualität weiter zu steigern.

Anmerkung: DivX unterstützt prinzipiell unendlich viele Passes, XviD beschränkt sich auf zwei. Ist das nicht ein Nachteil? Nein. XviD bearbeitet in den beiden Durchgängen das Video so intensiv und ausgeklügelt, dass weitere Passes gar nicht nötig sind. DivX dagegen erreicht sein Ziel über kontinuierliche Verbesserungen, also zusätzliche Passes.

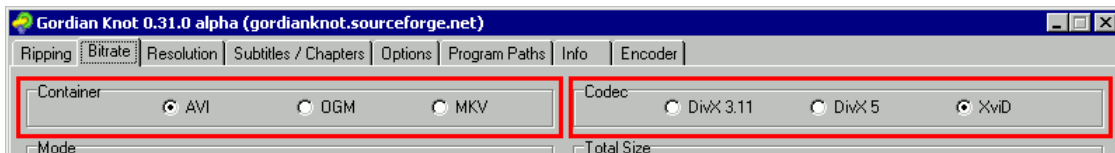
Für beide Codecs konfigurieren wir separat 1st und 2nd bzw. Nth Pass. Bei DivX haben wir zusätzlich die Möglichkeit, für den allerletzten Durchgang noch einmal veränderte Einstellungen zu wählen.

Bleiben die Credits, also der Abspann des Films. Da es sich dabei meistens nur um weiße Laufschrift auf schwarzem Hintergrund handelt, können die Credits viel stärker komprimiert werden als der restliche Film. Für XviD kümmert sich Gordian Knot um die richtige Konfiguration, DivX-Credits konfigurieren wir über den **Credits**-Button selbst.

Anmerkung: Details zur Codec-Konfiguration gibt's in den Kapiteln weiter unten.

Damit sind wir mit der Grundkonfiguration fertig und können ins Register **Bitrate** wechseln, um uns über gewünschten Container und Codec Gedanken zu machen.

3.1.1 Auswahl von Container und Codec



Die Container

Jeder Anfänger stolpert recht bald über den Begriff *Container*, die meisten dürften keine genau Vorstellung haben, was damit gemeint ist. Bildlich gesprochen stellt der Container die Schuhschachtel dar, in der Filmrolle und Tonband aufbewahrt werden, also die Verpackung um Bild und Ton außen herum. Das Format des Containers lässt noch nicht unbedingt auf das Format des Inhalts schließen. Begriffe wie *AVI-Video* sind streng genommen nicht richtig und auch nicht besonders aussagekräftig: AVI ist kein Video- sondern ein Container-Format. Innerhalb der AVI können Bild- und Tonspuren verschiedenster Formate liegen. Möglichkeiten für das Bild wären z. B. XviD, DivX, Huffuyuv; für den Ton z. B. MP3, Wave, AC3. Exakt müsste man also beispielsweise von *XviD und MP3 in AVI* sprechen. Heftig umständlich, deswegen werden im Alltag die Begriffe wild durcheinander geschmissen. Man sollte sich trotzdem immer im Klaren darüber sein, was eigentlich dahinter steckt.

Sehen wir uns doch die einzelnen Container etwas näher an.

AVI (Audio/Video Interleave). Der älteste und am weitesten verbreitete Container. Allerdings merkt man ihm sein Alter immer deutlicher an. AVI unterstützt nativ nur wenige Audioformate (CBR MP3, Wave), für VBR MP3 und AAC existieren gut funktionierende Hacks, Vorbis ist nicht möglich. Ob AC3 und DTS auch Hacks sind, bin ich mir gerade nicht sicher. AVI unterstützt leidlich Untertitel, allerdings keine Kapitel.

OGM (OggMedia). Da Vorbis in AVI unmöglich ist, warum sollte nicht der umgekehrte Weg funktionieren: also eine AVI in den Ogg-Container packen. Das Ergebnis heißt OggMedia, unterstützt natürlich Vorbis genauso wie MP3, AC3, DTS und AAC. Untertitel und Kapitel sind auch kein Problem.

Allerdings ist OGM ein aus der Not geborenes Format. Es existiert keine Dokumentation dafür und die Entwicklung steht seit einiger Zeit still. Durch das Aufkommen von Matroska und MP4 stellt sich immer mehr die Frage, wozu OGM noch gebraucht wird.

MKV (Matroska). Das ambitionierte freie Containerprojekt von www.matroska.org, angetreten mit dem Anspruch irgendwann AVI zu ersetzen. Das Potenzial dazu hat der Container allemal: Er unterstützt selbstverständlich eine Vielzahl Audio- und Videoformate, natürlich auch Untertitel und Kapitel. Das nächste große Feature könnten Menüs wie auf der DVD werden. Was fehlt ist die breite Unterstützung durch die Medienindustrie, deshalb wird wohl eher MP4 zum AVI-Nachfolger werden.

Empfehlung: Für Standalone-Player bleibt nur eine Wahl: AVI. Und auch dann muss es einer der DivX-fähigen Standalones sein. Ansonsten hängt die Auswahl hauptsächlich von den benötigten Features und dem persönlich Geschmack ab. – Ach, was sag ich! Matroska wird sie irgendwann alle in die Tasche stecken! :-) Soll heißen, ich habe meine Wahl getroffen.

Die Codecs

DivX ;-) 3.11 ist nichts anderes als der gehackte MS MPEG4-Codec. Im Gespann mit Nandub (was dann unter der Bezeichnung SBC läuft) hat er die Massenbewegung um digitales Video ausgelöst. Das Encoding ist gerade für Anfänger unübersehbar kompliziert, außerdem ist der Codec inzwischen veraltet. V. a. XviD bringt auf einfachere Weise mindestens genauso gute Ergebnisse.

DivX 5. Der offizielle und legale DivX ;-) Nachfolger von DivX Networks⁵ hat seine Wurzeln im DivX 3.11, ist allerdings von Grund auf neu programmiert. Er bringt gute Qualität und lässt sich recht einfach konfigurieren.

XviD. In Zeiten von DivX 4 hat sich XviD als eigenständiges Projekt abgespalten. Der Codec ist unkommerziell und sein Quellcode auf www.xvid.org frei zugänglich. XviD wird von einem enthusiastischen (und verdammt fähigen!) Team von Programmieren in deren Freizeit weiterentwickelt. Spätestens seit der Version 1.0 gilt er DivX gegenüber als überlegen. Allerdings kann sich gerade der Anfänger leicht in den vielen Einstellungsmöglichkeiten verirren.

Empfehlung: Der alte Hase, der seit 1999 schon Videos codiert, greift vielleicht noch zu DivX 3.11, alle anderen sollten sich an den neueren DivX oder XviD halten. Die Wahl hängt dabei hauptsächlich von persönlichen Vorlieben ab; genauso wie die Frage nach dem besten Container ist die Frage nach DivX oder XviD zu einem nicht unerheblichen Teil ein Glaubenskrieg.

Vielleicht als kleine Hilfe: Wer sich wenig Gedanken machen will – zwei oder drei Optionen einstellen und loslegen – der ist mit DivX gut beraten. Wer das letzte Quäntchen Bildqualität herausholen will und bereit ist, sich dafür auch mit den Hintergründen der ganzen Einstellungen zu beschäftigen, dürfte an XviD mehr Freude haben. Mal davon abgesehen, dass XviD in einer Mehrzahl der Fälle schneller sein dürfte ...

3.1.2 Konfiguration des XviD-1.0.2-Codecs

XviD ist ein sehr genau konfigurierbarer Codec, man könnte Bücher darüber schreiben, wann welche Kombination von Optionen sinnvoll ist. Um den Guide nicht ausufern zu lassen, erkläre ich die einzelnen Einstellungen nur kurz. Wer tiefer einsteigen will, dem kann ich nur Selurs »Wissenswertes rund um XviD«⁶ und crusty's (englische) XviD FAQ⁷ ans Herz legen.

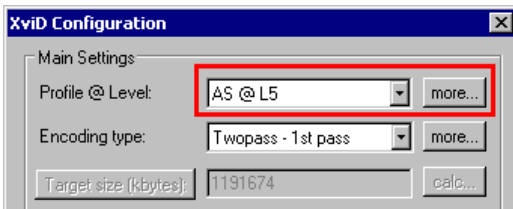
Einstellungen für den 1st Pass

Profile und deren Level schalten einzelne Optionen des Codecs frei oder sperren sie, um Konformität zu den gleichnamigen MPEG-4-Profilen zu gewährleisten. Für einen

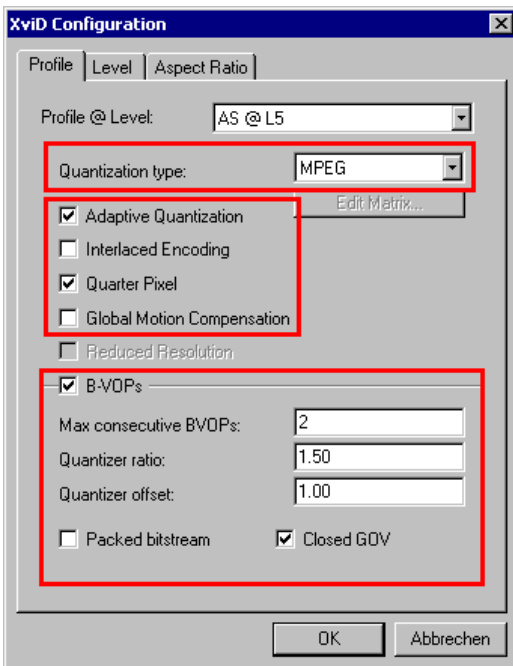
5) <http://www.divx.com>

6) <http://www.flaskmpeg.info/board/thread.php?threadid=4526>

7) <http://ronald.vslcatena.nl/docs/xvidfaq.html>



normalen DVD-Rip bietet sich **AS @ L5** (Advanced Simple, Level 5) an. Über den **More**-Button erreichen wir den Dialog für die Feineinstellungen.



Der **Quantization type** bestimmt die Matrix, die beim Encoden verwendet wird. **MPEG** erzeugt ein etwas schärferes Bild auf Kosten der Kompression und eignet sich eher für die höheren Datenraten von 2-CD-Rips. **H.263** ergibt dagegen ein etwas weiches Bild und erhöht damit die Kompression ein wenig; geeignet eher für die niedrigen Datenraten von 1-CD-Rips.

In sehr hellen und sehr dunklen Bereichen des Bilds nimmt das menschliche Auge weniger Details wahr. Das macht sich **Adaptive Quantization** zu nutze und komprimiert solche Bereiche stärker, um die eingesparte Datenrate an Stellen zu verwenden, die es nötiger haben. Gerade bei geringen

Bitraten macht sich das positiv bemerkbar.

Quarter Pixel erhöht die Genauigkeit, mit der Bewegungen gespeichert werden, von einem halben auf ein Viertel Pixel. Das wirkt sich positiv auf Details und Schärfe aus, deshalb sollten wir QPel normalerweise einschalten.

Global Motion Compensation versucht die Kompression zu erhöhen, indem es nach gemeinsamen Bewegungsvektoren in der Szene sucht. Zooms und Kameraschwenks bieten sich dafür besonders an. Ob GMC wirklich spürbare Unterschiede bringt, ist nicht ganz unumstritten. Aktivieren sollten wir es v. a. dann, wenn wir Qualität um jeden Preis wollen. Dann wählen wir später auch einen hohen VHQ-Wert in den **Advanced Options**.

Sowohl QPel als auch GMC bremsen nicht nur den Encoding-Vorgang, sondern benötigen auch beim Abspielen Rechenleistung. Auf alten Rechnern (500 MHz abwärts) sollten wir uns deshalb besonders den gemeinsamen Einsatz gut überlegen. Auch Standalone-Player kommen oft mit beiden nicht zurecht.

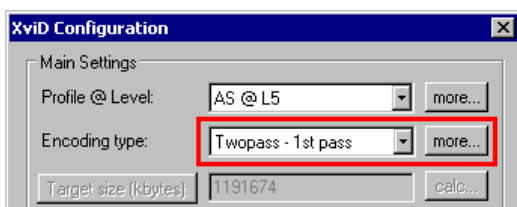
Um **B-VOPs** (technisch für B-Frames) zu verstehen, müssen wir einen kleinen (und vereinfachten) Ausflug in die Tiefen der Videokompression unternehmen. Es gibt in einem XviD-Stream drei Arten von Einzelbildern: I-Frames, P-Frames und B-Frames. I-Frames (Intra-Frames) heißen auch Keyframes und sind vollständige Einzelbilder, vergleichbar mit einem Jpeg-Bild. P-Frames (Predicted Frames) sind keine vollständigen Einzelbilder, sondern enthalten nur die Unterschiede zum vorhergehenden Bild, wodurch sie mit weniger Platz auskommen. B-Frames (Bidirectional Frames) sind eine Art erweiterter P-Frames. Sie beziehen sich nicht nur auf das Bild vorher, sondern auch auf das nachfolgende Bild. Dadurch können B-Frames ohne Qualitätseinbußen deutlich höher komprimiert werden.

B-Frames sollten wir ohne guten Grund nicht abschalten. **Max consecutive B-VOPs** gibt an, wie viele B-Frames maximal direkt hintereinander stehen dürfen. Mit der Standardeinstellung **2** wäre also eine solche Bildsequenz möglich: IPBBP, bei **3** wären es dann IPBBBP usw. Wichtig: Die Einstellung definiert nur das Maximum. **2** bedeutet also nicht, dass immer zwei B-Frames hintereinander stehen, es heißt nur, dass niemals mehr als 2 aufeinander folgen dürfen. Innerhalb dieser Grenze berechnet XviD dann die günstigste Möglichkeit. Die Standardeinstellung ist ein sinnvoller Wert, nur für Stalalone-Player müssen wir evtl. auf **1** herunter gehen, da Stalalones meistens auf DivX ausgelegt sind, der mehrere B-Frames hintereinander nicht unterstützt.

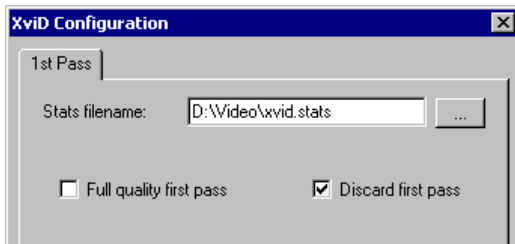
Quantizer ratio und **Quantizer offset** bestimmen, wie hoch B-Frames im Vergleich zu P-Frames komprimiert werden. Die Standards sind brauchbare Werte. Ohne einen Batzen Hintergrundwissen sollten wir die beiden lieber in Ruhe lassen.

Packed Bitstream bestimmt, wie die B-Frames in der Videodatei gespeichert werden und sollte normalerweise ausgeschaltet sein. Ausnahme sind wieder einmal die Stalalones, die bei deaktiviertem **Packed Bitstream** gerne Probleme machen. **Closed GOV** dagegen sollte immer aktiv sein, um Synchronisationsproblemen und Stress beim Splitten vorzubeugen.

Damit ist dieses Fenster abgehandelt und wir können zurück in den Hauptdialog.

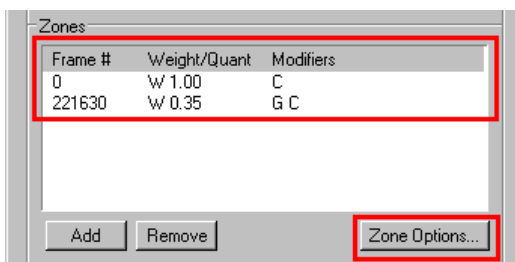


Mit **Twopass - 1st pass** stellen wir den ersten Durchgang ein und rufen über den **More**-Button die Details auf.

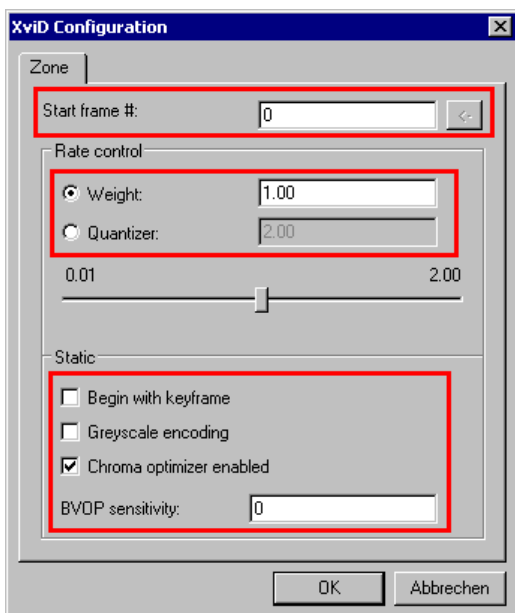


Im angegebenen **Stats file** werden die Informationen aus dem 1st Pass gespeichert, die der 2nd Pass dann weiterverwendet. **Full quality first pass** aktiviert sämtliche eingestellten Encoder-Optionen schon im ersten Durchgang. Normalerweise sind hier einige Optionen abgeschaltet (z. B. VHQ), die nur bremsen und im 1st pass noch nichts bringen. Die Option sollten wir nur dann einschalten, wenn wir die Videodatei des 1st Pass behalten wollen. Wollen wir aber nicht. Deshalb setzen wir auch den Haken bei **Discard first pass**. XviD schreibt dann keine komplette Videodatei sondern nur ein wenig Müll. Das spart Plattenplatz.

Damit kehren wir zurück ins Hauptfenster und kümmern uns um die **zones**.



Zones definieren Abschnitte innerhalb des Videos, für die unabhängig eine Reihe von Optionen definiert werden können. Offensichtlichster Anwendungsbereich für Zonen sind die Credits. Eine Zone, die sich über den ganzen Film erstreckt, existiert immer.



Über den **zone options**-Button gelangen wir in den Konfigurationsdialog.

Bei **start frame #** tragen wir das Bild ein, bei dem die Zone anfangen soll. Für die erste Zone ist das **0**, der Filmanfang. Das Ende einer Zone ist immer der Anfang der darauf folgenden oder das Ende des Films.

Zonen können entweder mit einem konstanten **quantizer** (vereinfacht: Kompressionsfaktor) oder einem relativen Gewicht (**weight**) versehen sein. **1.00** ist dabei der Standard. Höher gewichtete Zonen bekommen mehr Qualität zugeteilt, geringer gewichtete müssen sich mit weniger zufriede-

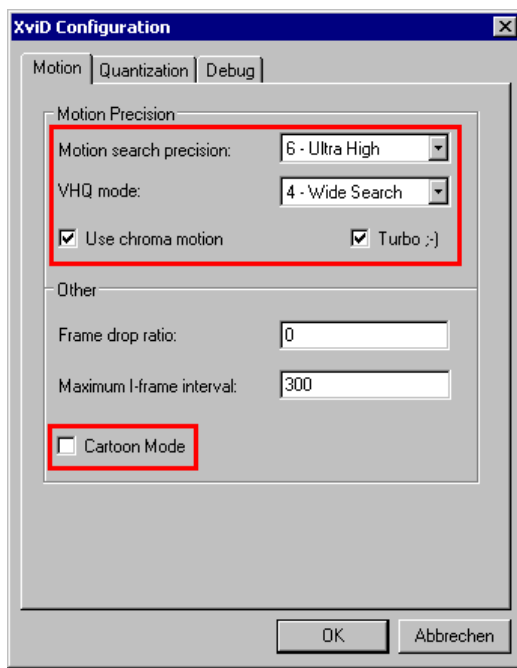
den geben. Im Normalfall können wir **Weight 1.00** einfach übernehmen.

Begin with a keyframe erzwingt am Anfang der Zone ein I-Frame. **Greyscale encoding** verwirft alle Farbinformationen, so dass wir ein Schwarzweiß-Video bekommen. **Chroma optimizer** wirkt Pixeltreppchen an scharfen Kanten entgegen, bremst aber den Encoding-Vorgang etwas. Obwohl ich mir nicht sicher bin, ob es wirklich sichtbare Verbesserungen bringt, lasse ich die Option immer eingeschaltet.

Die **BVOP sensitivity** schließlich wirkt sich darauf aus, wie gern XviD B-Frames setzt. Positive Werte ermutigen XviD zu mehr B-Frames, negative Werte schrecken den Codec eher ab. Die **0** können wir als sinnvollen Standardwert stehen lassen.

Normalerweise definieren wir nur eine einzige Zone über den ganzen Film. Um das Einrichten einer zusätzlichen Zone für die Credits kümmert sich Gordian Knot automatisch.

Über **OK** gelangen wir zurück in den Hauptdialog und rufen unten in der Mitte die **Advanced options** auf.



Die **Motion search precision** legt fest, wie intensiv XviD nach Bewegungen sucht. Es gibt kaum einen Grund von **6 - Ultra High** abzuweichen.

VHQ rechnet für die einzelnen Macroblocks verschiedene Szenarien durch und entscheidet sich dann für das mit der geringsten Anzahl Bits. Generell gilt: Jede höher der **VHQ mode** desto besser die Qualität und desto langsamer das Encoding. **1** oder **2** bringen im Vergleich zum Bremsseffekt den höchsten Qualitätsgewinn. Ich bin Qualitätsfanatiker genug, um meistens bei der höchsten Einstellung **4** zu bleiben. Wenn wir GMC aktiviert haben, sollte VHQ nicht abgeschaltet sein.

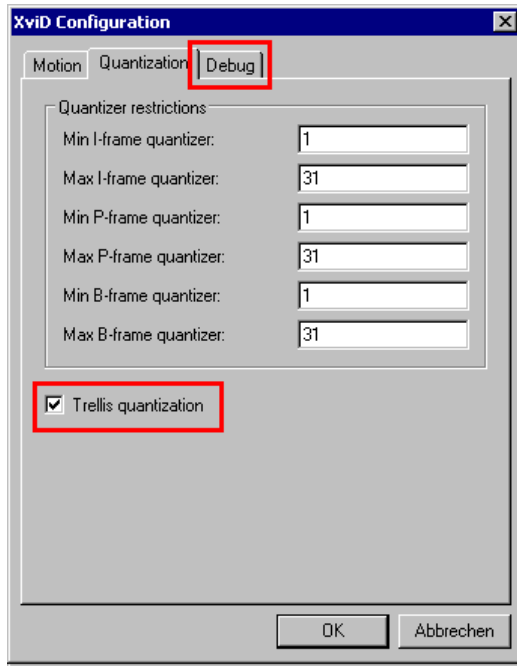
Use chroma motion veranlasst XviD, bei der Suche nach Bewegung nicht nur die Helligkeitsinformationen (Luminanz) zu berücksichtigen, sondern auch die Chrominanz (Farbe). Das steigert die Genauigkeit der Ergebnisse und damit die Qualität.

Turbo ; -) beschleunigt die Berechnung von B-Frames und QPel, erreicht dadurch aber nicht immer das absolute Qualitätsmaximum. Da der Unterschied kaum jemals sichtbar ist, lasse ich Turbo eingeschaltet.

Von der **Frame drop ratio** sollte jeder die Finger lassen, der nicht ganz genau weiß, was er da tut. Auch das **Max. I-frame interval** kann problemlos auf dem Standardwert bleiben. An günstigen Stellen setzt XviD sowieso automatisch I-Frames

(Man nennt sie auch Keyframes). Ein sehr niedriger Wert macht höchstens bei Captures direkt in XviD Sinn, die man hinterher noch schneiden will. Denn Schneiden funktioniert nur an I-Frames. Genaueres dazu steht im Splitting-Kapitel (Punkt 4.1).

Der **Cartoon Mode** ist nützlich bei Cartoons – wer hätte das gedacht. Cartoon meint dabei allerdings richtig klassisch gezeichneten Zeichentrick a la Tom & Jerry. Animationen wie Monster AG gelten für XviD nicht als Cartoon.



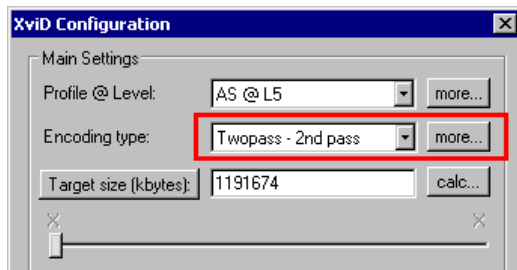
Die Optionen auf der **Quantization**-Seite sind sinnvoll eingestellt, so dass wir das meiste einfach übernehmen können. Interessant ist nur **Trellis quantization**. Trellis »überdenkt« die einmal getroffene Quantisierungsentscheidung und versucht sie zu verbessern. Heftige Qualitätsgewinne dürfen wir davon nicht erwarten, andererseits bremsst Trellis auch nicht besonders. Die Option können wir also ganz nach Lust und Laune ein- oder ausschalten.

Auf der **Debug**-Seite haben wir noch die Möglichkeit, das Statusfenster abzuschalten. Inzwischen bremsst es das Encoding zwar kaum noch, aber wer nicht die ganze Zeit vor dem Rechner sitzt und die Statistiken

bewundert, wird es kaum benötigen.

Und damit ist die Konfiguration des 1st Pass endlich beendet. Keine Angst, der 2nd Pass geht schneller.

Einstellungen für den 2nd Pass



Im Hauptdialog stellen wir **Twopass - 2nd pass** ein und stürzen uns per **More**-Button auf die Details.

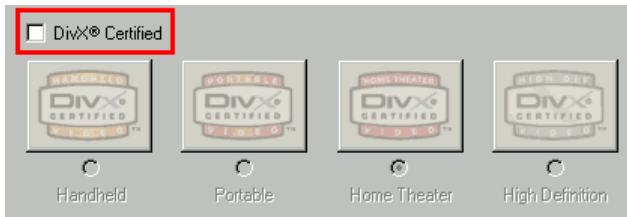
Eine kurze Kontrolle kann nicht schaden, ob XviD korrekt den Namen des Stats-Datei aus dem 1st Pass übernommen hat. Das sollte der Fall sein. Alle Optionen darunter können wir guten Gewissens auf den Standardwerten belassen. Genauso verändern

wir die restlichen Einstellungen im Vergleich zum 1st Pass nicht. Damit ist die Konfiguration des 2nd Pass auch schon abgeschlossen.

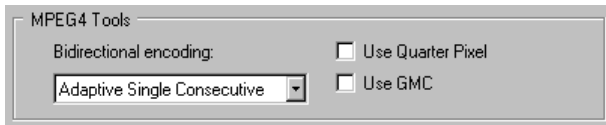
3.1.3 Konfiguration des DivX-5.2.1-Codecs

Einstellungen für den 1st Pass

Zuerst rufen wir über den Select **Divx Certified Profile**-Button den Profilassistenten auf.



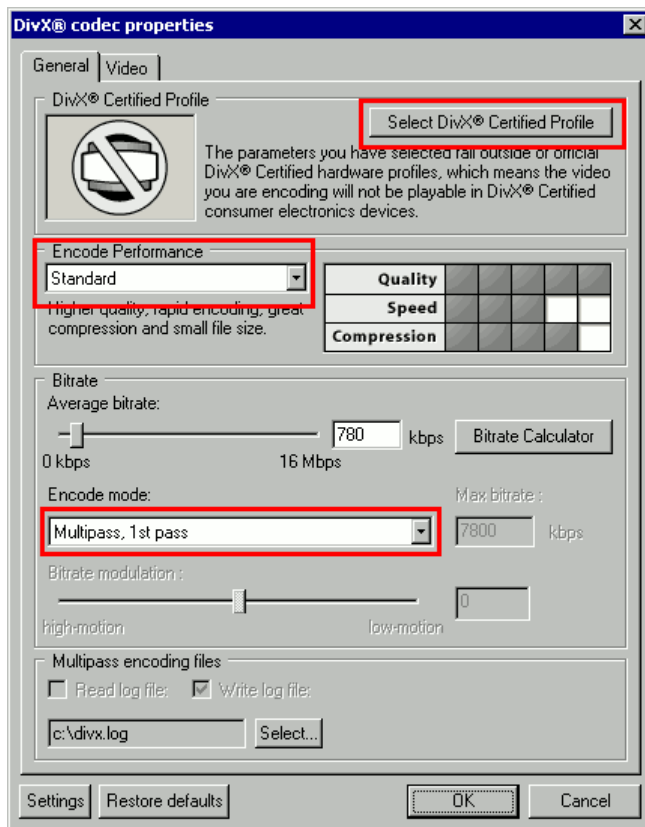
Um volle Kontrolle über alle Optionen zu erhalten, müssen wir den Haken bei **DivX Certified** entfernen und damit die Profile deaktivieren. Dann klicken wir uns auf die nächste Seite weiter.



Was die drei Optionen im Detail bedeuten, haben wir schon bei der XviD-Konfiguration gesehen. Bei DivX arbeiten sie prinzipiell genauso, von einigen Einschränkungen abgesehen. DivX-GMC benutzt nur einen Warppoint (im Gegensatz zu 3 bei XviD), QPel unterscheidet sich wenig. **Bidirectional encoding** konfiguriert die B-Frames. **off** dürfte sich selbst erklären ;-), **Adaptive Single Consecutive** entspricht dem Verhalten bisheriger DivX-Versionen, d. h. mehrere B-Frames hintereinander werden nicht gesetzt. Maximal zwei aufeinander folgende B-Frames erlaubt **Adaptive Multiple Consecutive**.

B-Frames sollten wir auf jeden Fall aktivieren. Wenn ich Doom9 durchstöbere, können das ruhig die **Multiple Consecutive** sein. Und wer B-Frames wirklich gezielt konfigurieren will, muss sowieso auf XviD umsteigen. QPel würde ich einschalten, GMC nicht. Achtung: Für Standalones sollten beide aus bleiben und maximal **Single Consecutive** B-Frames verwendet werden.

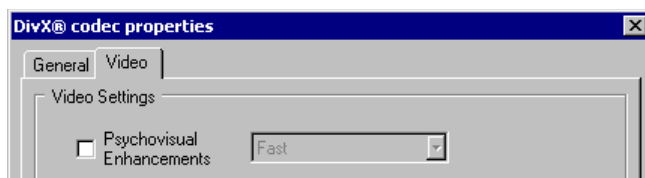
Damit verlassen wir den Assistenten und kommen wieder auf die **General**-Seite.



Bei **Performance/Quality** müssen wir uns entscheiden, ob Geschwindigkeit oder Qualität wichtiger ist. **Standard** ist vergleichsweise flott und bringt durchschnittliche Qualität. **Slow** bremst deutlich ab, belohnt uns aber auch mit höherer Qualität. **Fast** und **Fastest** sind für ein auf Qualität getrimmtes 2-Pass-Encoding weniger interessant. Eine Einschränkung müssen wir im langsamen Modus hinnehmen: QPEL funktioniert nur mit **Standard**. Wenn wir Quarter Pixel aktivieren, stellt DivX den Modus automatisch um und blendet den Schieberegler ab.

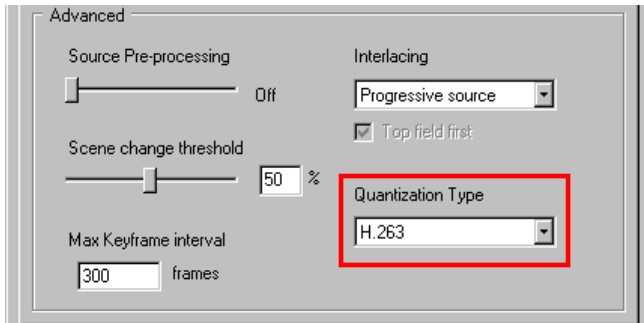
Multipass, 1st pass ist die passende Einstellung für den ersten Durchgang. Um die Bitrate darunter müssen wir uns nicht kümmern, denn die setzt Gordian Knot automatisch ein.

Und damit weiter zum **video**-Register.



Hier sind erst einmal die **Psychovisual Enhancements** interessant. Alles andere können wir bei den Standards belassen. PVE versucht die Bildteile zu errechnen, die das menschliche Auge sowieso nicht wahrnehmen kann, und lässt diese weg. Die frei werdende Bitrate kann dann für andere Teile des Bilds verwendet werden.

Der **Fast**-Modus ist aggressiver als der **Slow**-Modus, d. h. er bringt mehr, erhöht aber auch die Gefahr, dass sichtbare Artefakte im Bild auftauchen. Mit **Slow** sind wir da auf der sichereren Seite, allerdings um den Preis niedrigerer Geschwindigkeit und geringerer Effektivität der PVE.

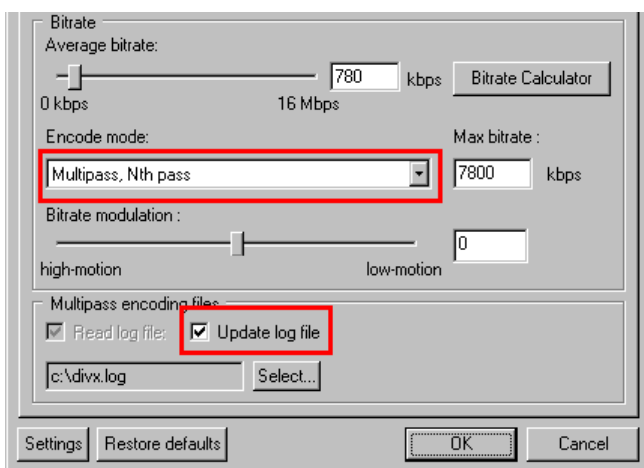


Unter **Quantization Type** lässt sich angeben, welche Quantisierungsmatrix DivX verwenden soll. Details dazu haben wir bei der XviD-Konfiguration schon gesehen. Da **MPEG-2** noch nicht richtig ausgereift ist, würde ich vorerst bei **H.263** bleiben.

In den **Settings** setzen wir unten die Haken bei **Do not prompt...** und **Disable the Feedback window**. So überschreibt DivX im 2nd Pass ungefragt die Log-Datei (was er ja tun soll) und schaltet das Statistik-Fenster während des Encodens ab. **Rotate artwork** und **Check for new version** kann jeder ganz nach seinen Vorlieben setzen. Naja ... das Statistik-Fenster eigentlich auch. :-)

Einstellungen für den Nth Pass

Den Großteil aller Einstellungen lassen wir für alle Durchgänge gleich. Nur auf der **General**-Seite ändert sich ab dem 2nd Pass etwas.



Hier stellen wir **Multipass, nth pass** ein und setzen die Haken bei **Update log file**. Die Aktualisierung der Logdatei ist wichtig, da dadurch erst die weiteren Qualitätsverbesserungen in den Passes 3 bis n ermöglicht werden.

Da kommt auch die Frage auf, wie viele Durchgänge denn sinnvoll sind. Drei Stück hat sich als vernünftiger Wert durchgesetzt. Mehr

Passes bringen nur noch minimale Qualitätssteigerungen und dürften höchstens bei hoch komprimierten 1-CD-Rips den Zeitaufwand wert sein.

Zurück zu Gordian Knot: Dort haben wir die Möglichkeit, den letzten Pass noch einmal extra zu konfigurieren. Das ist dann sinnvoll, wenn wir in den ersten Durchgängen mit **Performance/Quality Standard** eher auf Geschwindigkeit setzen und dann im letzten Durchlauf mit **slow** das Video auf Qualität trimmen. QPel-Liebhabern bleibt diese Möglichkeit natürlich verwehrt.

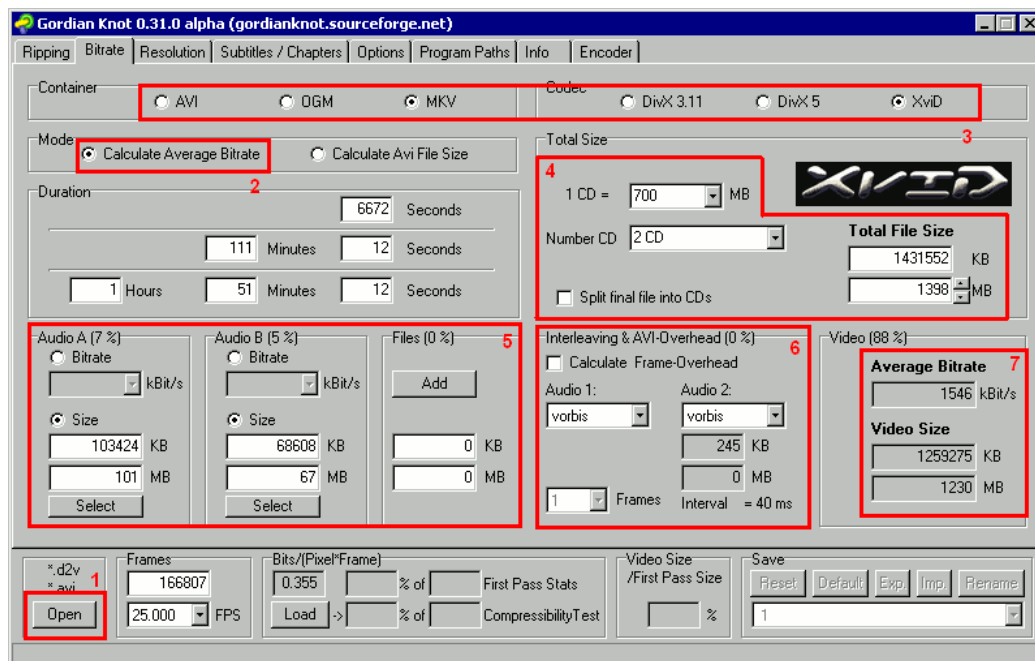
Einstellungen für Credits

Natürlich vertragen die Credits mit DivX genauso wie mit XviD eine höhere Kompression. DivX hat die Funktionalität, Credits extra zu behandeln, aber nicht eingebaut. Deshalb übernimmt das Gordian Knot.

Als Modus wählen wir **1-pass quality-based**. Den **Quantizer** stellen wir auf **15** bis **20**. Eine Logdatei ist für Credits nicht nötig und da wir nur einen Pass haben, macht auch die MV-Datei keinen Sinn. Um konform zum MPEG-Standard zu bleiben, sollten wir die anderen Optionen so wählen wie für den restlichen Film auch.

3.2 Kalkulieren der Bitrate

Im **Bitrate**-Register von Gordian Knot laden wir unseren Film, wählen den gewünschten Container und Video-Codec aus und berechnen die Bitrate, mit der später encodiert wird.



Mit **Open** (1) unten links laden wir die von DGIndex angelegte d2v-Datei. Es öffnet sich ein zusätzliches Fenster mit dem Video. Das ist im Moment noch uninteressant, darf aber nicht geschlossen werden. Neben dem **Open**-Button bei **FPS** sehen wir die

Framerate, die mit der aus DGIndex übereinstimmen sollte, und unter **Duration** steht die Länge des Films.

Bei **Mode** (2) stellen wir ein, ob wir die Bitrate oder die endgültige Dateigröße des Films berechnen wollen. Da die Größe durch die gewünschte Anzahl CDs schon feststeht, stellen wir **Calculate average bitrate** ein. Dann wählen wir unter (3) Container und Codec aus.

Bei **Total Size** (4) stellen wir die Rohling-Größe und gewünschte Anzahl CDs ein. Alternativ können wir auch rechts direkt die gewünschte Größe in MB angeben. Bei mehreren CDs macht es Sinn, die **Total File Size** ein paar MByte nach unten zu schrauben. Schließlich müssen wir den Film zum Schluss an einer sinnvollen Stelle schneiden, um ihn auf die CDs aufzuteilen. Da schadet es nicht, ein wenig Luft zu haben. Die **Split final file into CDs**-Funktion würde das zwar automatisch übernehmen, achtet dabei aber nur auf die Dateigröße, so dass es schon einmal vorkommen kann, dass der Held des Films mitten im Satz unterbrochen wird, weil die nächste CD fällig wird. Deshalb sollten wir das Splitten immer manuell erledigen.

Abschnitt (5) enthält alle Daten, die außer der Videospur auf die CD müssen. Mit **Select** wählen wir die erste erstellte Audiospur aus, deren Größe in die Felder darüber übernommen wird. Das Ganze nochmal für eine eventuelle zweite Audiospur. Der **Files**-Abschnitt ist für Dateien gedacht, die zusätzlich zum Film noch mit auf die CDs sollen, z. B. dynamische Untertitel oder Special Features von der DVD. Je mehr hier steht, desto geringer wird natürlich die Qualität des Films, das sollten wir immer bedenken.

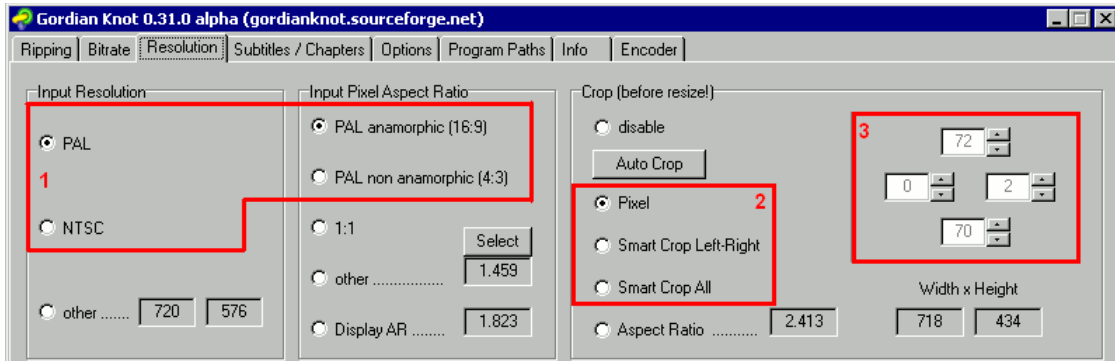
Schließlich wählen wir bei (6) noch die zum Audio passende Einstellung. Hier geht's um die Container-interne Synchronisation zwischen Audio- und Videospur. Je nach Audioformat braucht die unterschiedlich viel Platz. Dazu kommt noch der Platz, den die Struktur der Container-Datei selbst verbraucht. Das berücksichtigen wir mit dem Haken bei **Calculate Frame Overhead**. Bei XviD sollte der allerdings deaktiviert bleiben.

Anmerkung: Gordian Knot versteht kein AAC-Audio. Wollen wir trotzdem AAC verwenden ist **vorbis** eine gute Annäherung für die Berechnung.

Damit sind alle Einstellungen erledigt. Bei **Average Bitrate** (7) steht jetzt die Bitrate, mit der GKnot den Film erstellt. Der Wert gibt an, wie viele Kilobits (= 1000 Bits) pro Sekunde im Durchschnitt zur Verfügung stehen. Starke Schwankungen innerhalb des Videos sind nichts ungewöhnliches, da der Codec die Bitrate je nach Bedarf verteilt, schnelle Szenen brauchen für die gleiche Qualität mehr Bits als langsame.

Jetzt können wir uns um die Videospur kümmern. Dazu wechseln wir auf das Register **Resolution**.

3.3 Vorbereiten des Videos

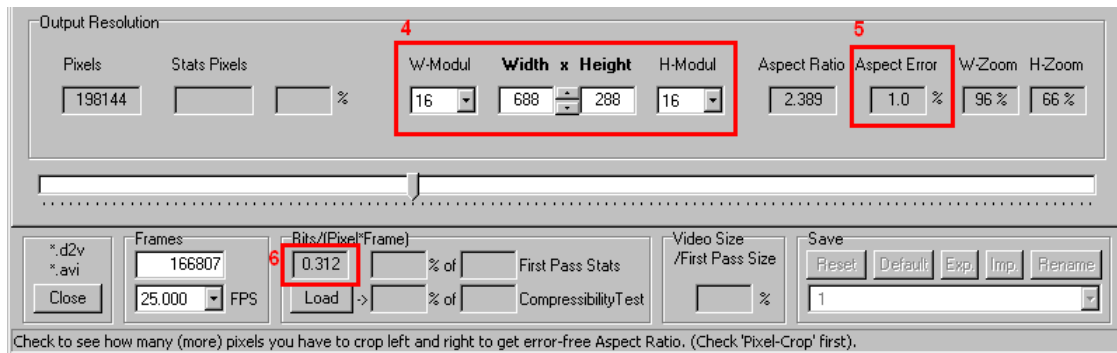


Im Abschnitt (1) stellen wir den Typ der DVD und die Art des Bildes ein. **PAL**-DVDs mit einer Auflösung von 720×576 Pixeln sind Standard in Europa, Japan und dem Nahen Osten, **NTSC** mit 720×480 Pixeln findet in Amerika Anwendung.

Zusätzlich müssen wir zwei Bildformate unterscheiden. Einmal **4:3**, das ist das Format eines ganz normalen Fernsehbildschirms. Kinofilme sind fast immer im schmalen **16:9** gedreht. Davon gibt es eine noch schmalere Abart, das 2,35:1. Zwischen den beiden Widescreen-Formaten brauchen wir aber nicht zu unterscheiden. 2,35:1 ist auf der DVD als 16:9 gespeichert. Der überschüssige Platz wird einfach mit schwarzen Balken aufgefüllt.

Aus DGIndex wissen wir schon, welches Format die DVD hat, z. B. PAL 16:9. Das dürfte für die meisten Region-2-DVDs (Europa) zutreffen.

Abschnitt (2) und (3) sind fürs Cropping zuständig, also für das Wegschneiden der schwarzen Ränder um das Video. Wer will, kann sich auf den **Auto Crop**-Mechanismus verlassen, ich traue solchen Automaten aber grundsätzlich nicht so recht über den Weg und mache das Cropping deshalb lieber selbst. Dafür stellen wir im Abschnitt (2) auf **Pixel**. Jetzt nehmen wir uns das Fenster mit dem Video vor, scrollen dort zu einer möglichst hellen Stelle und schneiden mit den Einstellungen bei (3) die schwarzen Ränder vollständig weg. **Smart Crop Left-Right** und **Smart Crop All** können wir später nach der Wahl der endgültigen Auflösung verwenden, um zusätzlich noch so viele Pixel wegzuschneiden, dass der unten erklärte **Aspect Error** (5) 0 wird.



Anschließend müssen wir eine geeignete Auflösung wählen. Dazu stellen wir bei (4) **W-Modul** und **H-Modul** beide auf **16**. Damit erlaubt Gordian Knot nur noch Auflösungen, die sowohl horizontal als auch vertikal durch 16 teilbar sind. Beim Einstellen der Auflösung mit dem großen Schieber müssen wir auf die Werte bei (5) und (6) achten. Der **Aspect Error** (5) gibt die Verzerrung des Bildes gegenüber dem Ursprungsformat an. Nicht über $\pm 2\%$ Fehler ist eine gute Einstellung, denn eine so geringe Verzerrung macht sich nicht nachteilig bemerkbar.

Das **Bits/(Pixel*Frame)**-Verhältnis (6) unten links gibt an, wie viele Bits zum Encoden eines einzelnen Pixels durchschnittlich zur Verfügung stehen, d. h. dieser Wert ist ein (wenn auch ungenauer) Indikator für die Qualität des fertigen Films, da er alle berechenbaren Einflussfaktoren berücksichtigt (Dateigröße, Filmlänge, Framerate, Auflösung). Allerdings sagt er wenig Definitives über die endgültig sichtbare Qualität aus. Ein Drama mit vielen langen Dialogen und hauptsächlich langsamen Szenen braucht für die gleiche sichtbare Qualität einen niedrigeren Wert als ein schneller Actionstreifen. Sehr nacht-lastige Filme geben sich auch mit einem niedrigeren Wert zufrieden, da Szenen im hellen Tageslicht viel mehr Details enthalten und deshalb auch mehr Bits benötigen. Deshalb kann das BPF-Verhältnis immer nur ein Anhaltspunkt für die sichtbare Qualität sein.

Der Wert sollte zwischen 0.20 und 0.30 liegen. Deutlich darunter äußert sich in schlechter Bildqualität, v. a. hässliche Macroblock-Artefakte werden dann sichtbar. Deutlich darüber erzeugt nur eine größere Datei bei kaum noch höherer Qualität. 1-CD-Rips orientieren sich dabei in Richtung der 0.20, 2-CD-Rips in Richtung 0.30.

Über den großen Schieber müssen wir jetzt eine Auflösung wählen, die einen akzeptablen Aspect Error mit einem sinnvollen BPF-Wert verbindet. Die Devise sollte dabei nicht unbedingt *möglichst hohe Auflösung* heißen. MPEG-4-Bilder sind in 16×16 Pixel große Einheiten aufgeteilt, die so genannten Macroblocks. Auf die wird auch die verfügbare Bitrate aufgeteilt. Je größer die Auflösung, desto mehr Macroblocks pro Einzelbild, und damit desto weniger Bitrate pro Macroblock. Das heißt, die Macroblocks können nicht mehr so viele Details speichern und die Qualität sinkt. Im Extremfall

werden im laufenden Video Macroblocks als diese hässlichen »Riesenpixel« sichtbar, die man von schlechten Encodings kennt.

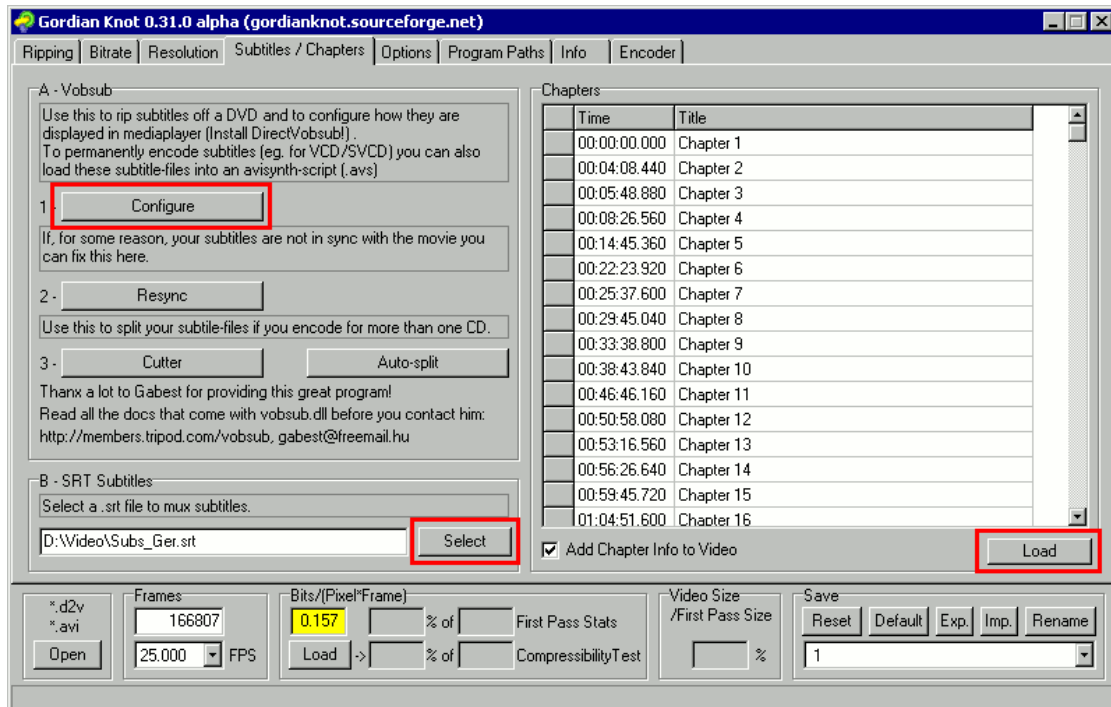
Wir sollten versuchen, mit der Auflösung nicht unter 512 Pixel in der Horizontalen zu kommen. Darunter leidet die Qualität immer stärker unter dem Hochzoomen aufs Vollbild. Die Grenze nach oben wird meistens vom Platz auf der CD vorgegeben. Sollte genug Raum sein, kann man ruhig auch über die ursprünglichen 720 horizontalen Pixel hinaus gehen. Das menschliche Auge ist für vertikale Auflösungen empfindlicher als für horizontale, und bei 720 horizontal ist in der vertikalen im Vergleich zur DVD noch gut Platz, was man auch am **H-Zoom**-Feld sieht. Solange das unter **100%** bleibt ist alles in Ordnung.

Trotzdem habe ich die Angewohnheit, im Zweifel lieber eine Stufe Auflösung für ein höheres BPF-Verhältnis zu opfern. Das erzeugt im Endeffekt die bessere Qualität.

Nun wird es Zeit, den Film in seiner endgültigen Auflösung anzusehen. Dazu stellen wir im Video-Fenster auf **View / Resized** um. Das Bild sollte jetzt ohne schwarze Ränder und ohne Verzerrungen erscheinen. Vorsicht dabei, auch ein verzerrtes Bild sieht auf den ersten Blick gern recht normal aus. Wenn's noch nicht passt, ist wahrscheinlich die **Input Resolution** und/oder die **Input Pixel Aspect Ratio (1)** verkehrt eingestellt.

Wenn wir dynamische Untertitel und/oder Kapitel haben, wechseln wir jetzt ins Register **Subtitles/Chapters**, ansonsten können wir gleich mit der Filterkonfiguration fortfahren.

3.4 Dynamische Untertitel und Kapitel einbinden



Mit dem **select**-Button können wir eine SubRip-Datei laden. Leider unterstützt Gordian Knot im Moment nur eine einzelne SubRip-Spur. Evtl. weitere müssen wir nach dem Encoding manuell einbinden.

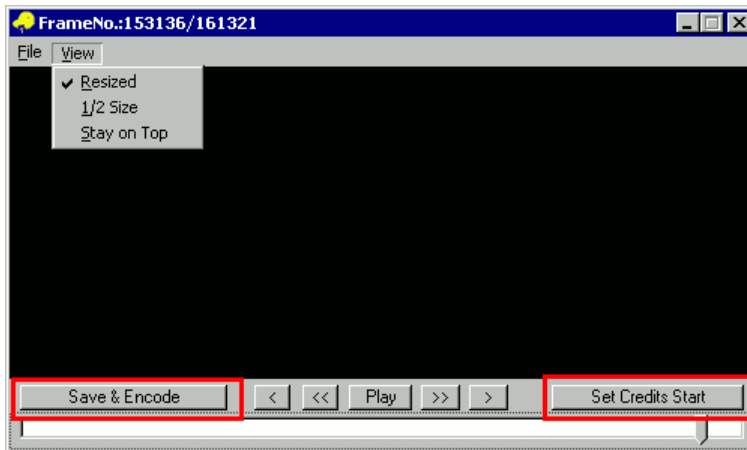
Über **Load** laden wir die von DVD Decrypter erzeugte Textdatei mit den Kapitelmarken. Im großen Fenster darüber können wir die Kapitel überarbeiten. Vorsicht bei Zeitänderungen, dass die Reihenfolge gewahrt bleibt. Für AVI können wir uns die Kapitel sparen, da der Container keine unterstützt.

Anmerkung: Den **Configure**-Button oben links kennen wir schon von den festen Untertiteln aus Kapitel 2.4.1. Damit haben wir VobSub aufgerufen.

3.5 Abschließende Arbeiten und Encoding

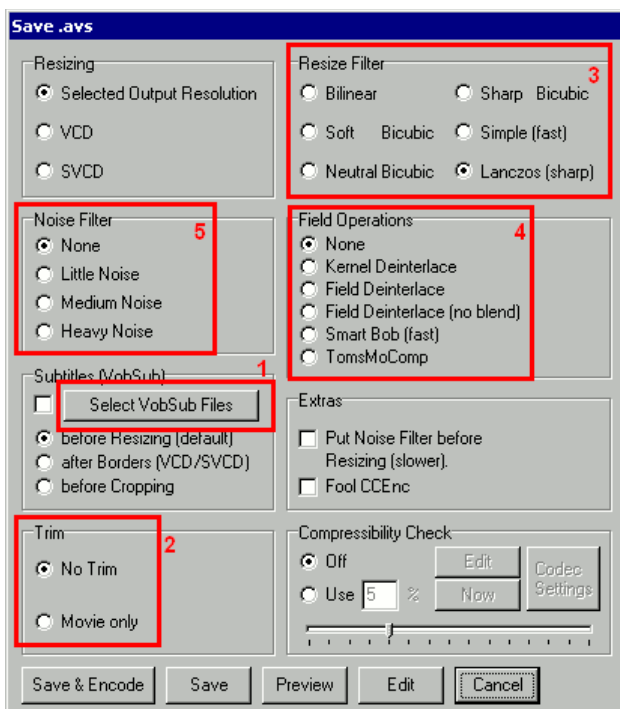
Filter konfigurieren

Im Video-Fenster müssen wir uns entscheiden, ob wir den Abspann mit anderen Einstellungen encoden als den Rest des Films. Wenn der Abspann nur aus weißer Laufschrift auf schwarzem Hintergrund besteht, können wir ihm deutlich weniger Bits gönnen als dem Film, ohne dass sich die niedrigere Qualität zu deutlich bemerkbar macht. Dadurch wird der Abspann kleiner und für den Rest bleibt mehr Platz übrig. Besteht der Abspann aber aus mehr als Laufschrift (z. B. bei der *Monster AG*), sollten



wir die Finger von der Funktion lassen. Ein mit Abspann-Einstellungen codiertes normales Videobild sieht grauenhaft aus.

Mit dem Schieberegler suchen wir den Anfang des Abspanns heraus. Das Bild sollte an der Stelle schon komplett schwarz sein, im Zweifel springen wir lieber noch etwas



weiter nach hinten. Dann klicken wir auf den Button **Set Credits Start**. GKnot weiß jetzt, dass hier die Credits beginnen sollen.

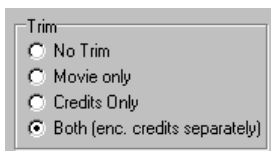
Über den **Save & Encode**-Button gelangen wir anschließend zum Fenster links, in dem wir alle Filter konfigurieren.

Hat unser Film feste Untertitel, stellen wir unter Subtitles (1) die mit VobSub erstellte und bearbeitete idx-Datei ein. Dynamische Untertitel interessieren hier nicht.

Hat unser Film feste Untertitel, stellen wir unter **Subtitles** (1) die mit

VobSub erstellte und bearbeitete idx-Datei ein. Dynamische Untertitel interessieren hier nicht.

Trim (2) kontrolliert das separate Encoding des Abspanns. Das Fenster oben gilt für XviD. Da der Codec mit den Zonen schon eine Möglichkeit eingebaut hat, die Credits zu codieren, müssen wir nichts Besonderes einstellen. **No Trim** ist die richtige Wahl.



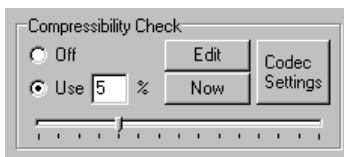
Anders bei DivX, der intern keine Möglichkeit fürs Credits-Encoding besitzt. Gordian Knot codiert deshalb Film und Credits getrennt und fügt die Dateien dann zusammen. **Both** ist

dafür die richtige Einstellung. **No Trim** bedeutet bei DivX, die Credits-Einstellung zu ignorieren und den kompletten Film unverändert am Stück zu codieren.

Mit dem **Resize Filter** (3) definieren wir die Methode, mit der das Video auf die eingestellte Zielauflösung umgerechnet wird. **Bilinear** und **Soft Bicubic** glätten das Bild dabei tendenziell. Dadurch geht zwar etwas Schärfe verloren, gleichzeitig steigt aber auch die Komprimierbarkeit. **Sharp Bicubic** und **Lanczos** zeichnen eher scharf, **Neutral Bicubic** und **Simple** liegen irgendwo dazwischen. Für 1-CD-Rips bietet sich ein weicher zeichnender Filter an, für 2-CD-Rips eher ein schärferer. Lanczos ist dabei sehr beliebt.

Die **Field Operations** (4) sind nur bei interlaced Video interessant, also wenn sich am Anfang in DGIndex Kammefekte im Bild gezeigt haben. In diesem Fall wählen wir einen der Deinterlacer: **Field Deinterlace** oder **TomsMoComp** bieten sich an.

Wenn das Bild rauscht, können wir das bei (5) mit einem **Noise Filter** bekämpfen. Gerade bei älteren Filmen kann das schon einmal nötig sein. *The Life of Brian* habe ich da als Beispiel in Erinnerung. Wichtig ist, nicht gleich in die Vollen zu gehen, denn ein Rauschfilter vernichtet immer nicht nur das Rauschen, sondern auch einige Details des Bildes.



Zu guter Letzt können wir noch einen **Compressibility Check** durchführen, der uns genauer als das BPF-Verhältnis Auskunft über die Qualität gibt. GKnot nimmt dazu in regelmäßigen Abständen einige Frames des Films (standardmäßig insgesamt 5 %) und encodiert ohne Rücksicht auf die Dateigröße mit einem festen Quantizer von 2, um maximale Qualität zu erreichen. Das Ergebnis lässt sich auf den ganzen Film hochrechnen und wir erhalten durch Vergleich mit dem BPF einen Prozentwert, an dem sich recht schön die Komprimierbarkeit dieses einen Films abschätzen lässt.

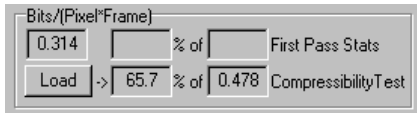
Mit **Codec settings** können wir überprüfen, ob die Einstellungen für den Comcheck passen. Für DivX heißt das:

- Modus **1-pass quality-based**.
- **Quantizer** von 2.

Und XviD sieht so aus:

- **Encoding type** auf **Single pass** stellen.
- **Target quantizer** auf **2.00** setzen.
- Den Rest so einstellen wie fürs Encoding des ganzen Films geplant.

Mit einem Klick auf **Now** öffnet sich VirtualDubMod und führt den Test durch. GKnot ist solange blockiert. Hinterher landen wir auf dem Register **Resolution**. Der Bereich unten in der Mitte sieht jetzt etwa so aus:



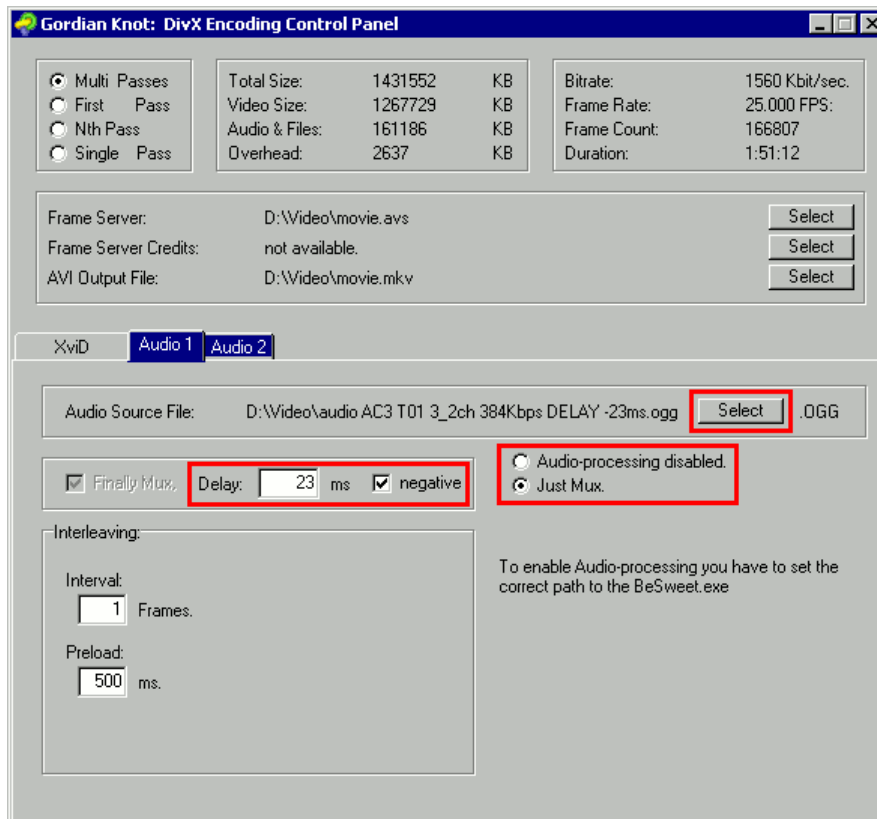
Der wichtige Wert sind die **65.7** Prozent. Gut sind zwischen 60 und 80 Prozent. Darunter wird die Qualität schnell schlechter, darüber vergrößert sich nur die Datei ohne spürbare Steigerung der Qualität. Alles analog zum BPF-Wert, nur ein Stück genauer.

Bei zu wenigen Prozenten (unter 50 % dürfte es so richtig kritisch werden) müssen wir entweder Platz schaffen (kleinere Audiospuren, mehr CDs) oder die Auflösung verringern. Bei zu hohen Werten können wir umgekehrt die Auflösung erhöhen oder Platz verschwenden. Vielleicht reicht's doch für den Original-AC3-Sound. Im günstigsten Fall lässt sich sogar eine ganze CD streichen.

Anmerkung: Beim Ändern der Auflösung verliert der Prozent-Wert einen Teil seiner Aussagekraft, weil der Compcheck nur für ein ganz spezielles Encoding-Setup gilt, zu dem eben auch die Auflösung gehört. Genau genommen müssten wir also bei jeder Auflösungs-Änderung einen neuen Kompressionscheck durchführen. Da aber auch der Compcheck nur ein Schätzwert für die Qualität ist, fällt diese zusätzliche Ungenauigkeit kaum ins Gewicht.

Haben wir schließlich und endlich passende Einstellungen gefunden, gelangen wir über den **Save & Encode**-Button im **Videofenster** wieder in den **Save .avs**-Dialog zurück und läuten mit dem dortigen **Save & Encode**-Button die letzte Konfigurationsphase vor dem Encoding ein. Gordian Knot speichert die avs-Datei, die anschließend VirtualDubMod braucht, und springt dann in den Encoding-Dialog, wo wir die Einstellungen für die Ton und Bild vervollständigen.

Audio- und Video-Setup vervollständigen

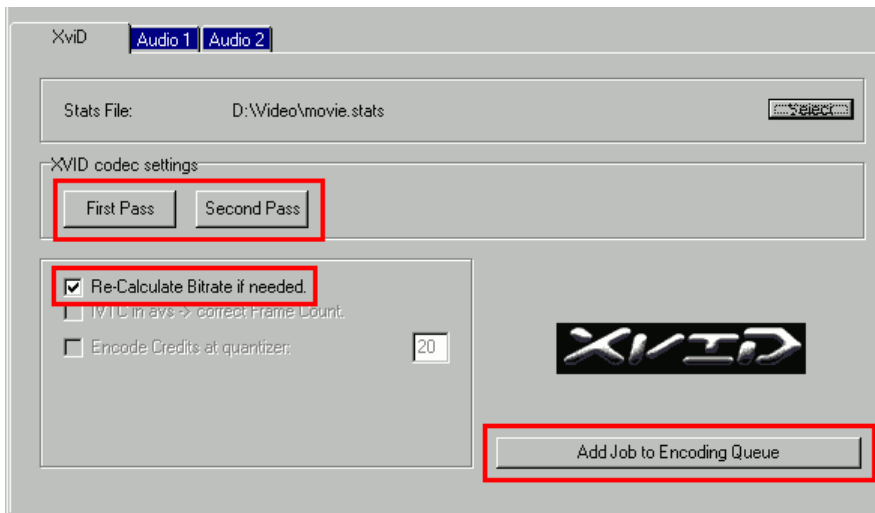


Mit **Select** wählen wir die erste Audiospur aus. GKnot hat hier schon automatisch die Datei eingetragen, die wir im Register **Bitrate** bei **Audio A** angegeben haben. Ist das eine AAC-Datei, müssen wir **Audio processing disabled** auswählen und uns später von Hand um die Tonspuren kümmern. Gordian Knot kann ja nicht mit AAC umgehen.

Bei allen anderen Audioformaten stellen wir auf **Just mux**, das Transcoding hat ja BeSweet schon erledigt. Im Dateinamen steht ein DELAY-Wert. Der sagt uns, um wie viele Millisekunden die Spur nach vorne oder hinten verschoben werden muss, um synchron mit dem Video zu laufen. Achtung: Wer das **Delay** schon bei BeSweet berücksichtigt hat, trägt bei Delay immer **0** ein. Ansonsten übernehmen wir den ms-Wert und setzen für negative den **negative**-Haken. Im Textfeld dann kein Minuszeichen angeben!

Wenn wir eine zweite Audiospur haben, wechseln wir ins **Audio 2**-Register und wiederholen dort die Einstellungen für die zweite Spur.

Die Audiospuren sind nun fertig und wir wechseln ins Register **XviD** bzw. **DivX 5**, um dem Codec den letzten Schliff zu verpassen.

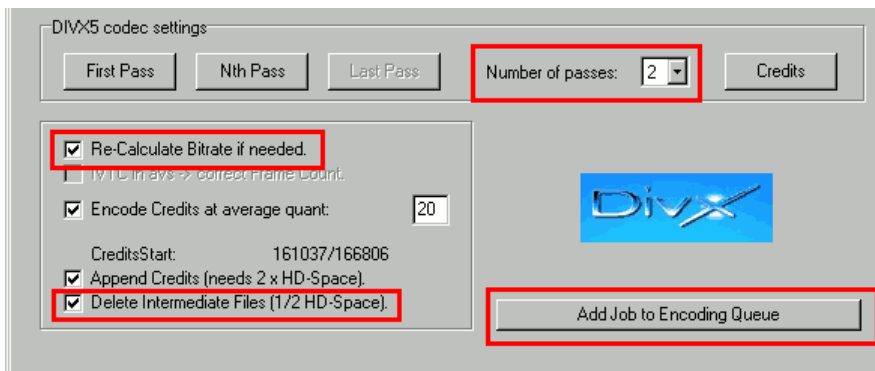


Für XviD gibt es wenig zu tun. Über die Buttons **First Pass** und **Second Pass** können wir das Codec-Setup (einschließlich der Einstellungen für Credits) anpassen. Das wirkt sich nur auf den aktuellen Film aus. Die Standardeinstellungen im **Options**-Register bleiben davon unberührt.

Re-Calculate Bitrate sollte angehakt bleiben, um Problemen mit falschen Dateigrößen aus dem Weg zu gehen. Nötig ist die Option eigentlich nur dann, wenn Gordian Knot auch das Audio-Transcoding übernimmt, was wir ja schon vorher erledigt haben. Aber man kann nie wissen.

Mit **Add Job to Encoding Queue** fügen wir den Film zur Encoding-Warteschlange hinzu.

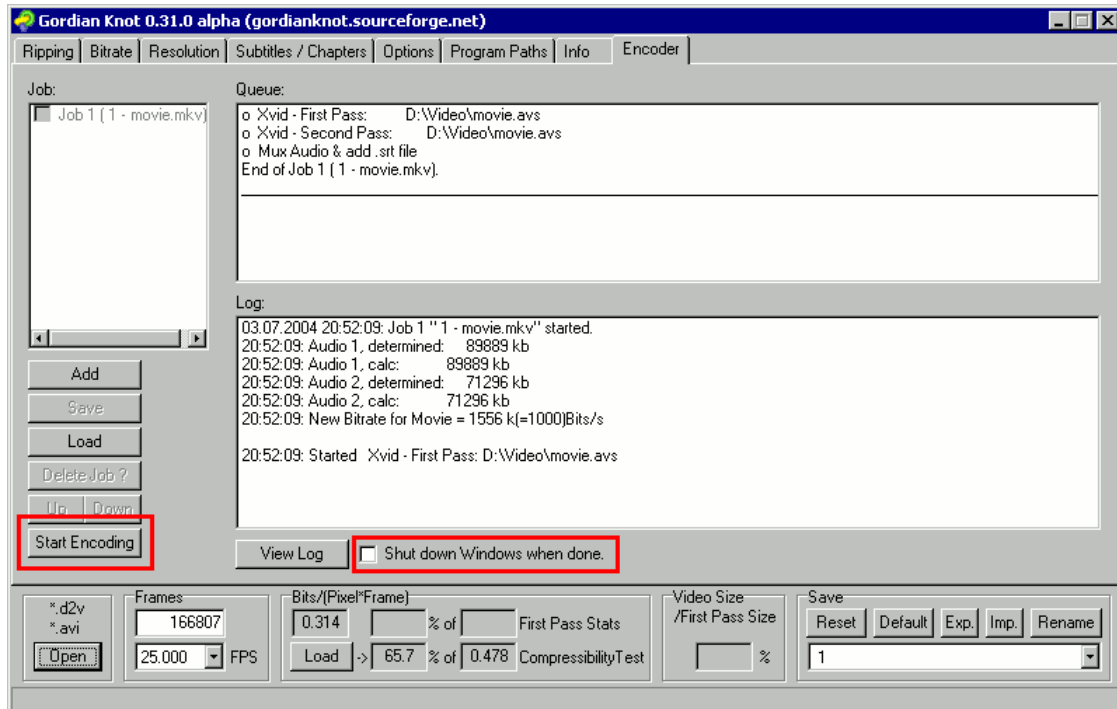
Für DivX sieht der Dialog ähnlich aus.



Mit **Number of passes** können wir auswählen, wie viele Passes Gordian Knot durchführen soll. Nicht übertreiben, mehr als drei dürfte meistens nicht nötig sein. Auch die Codec-Konfiguration können wir noch einmal anpassen.

Delete Intermediate Files löscht unnötige Dateien, die beim Credits-Encoding entstehen, automatisch wieder. Wer unter notorischen Platznot auf der Platte leidet, dürfte froh über diese Option sein. ;-)

Unglaublich aber wahr: wir sind endlich bereit zum Encoding. Über den Button **Add Job to Encoding Queue** fügen wir den Auftrag in die Warteschlange von GKnot ein und landen anschließend im Register **Encoder**.



Wenn wir GKnots Nachfrage nicht schon bestätigt haben, starten wir über **Start Encoding** den Codiervorgang. Im **Queue**-Fenster sehen wir unseren aktuellen Auftrag, darunter bei **Log** sämtliche Details dazu. GKnot ruft jetzt VirtualDubMod auf, um den Film zu encoden und anschließend die Audiospuren einzubinden.

Solange VirtualDubMod arbeitet war früher herrlich Zeit, mal wieder mit der Freundin auszugehen oder ein paar Stunden Schlaf nachzuholen. Heutzutage kann man gerade noch in Ruhe ein Stück Kuchen vom Bäcker holen und Kaffee trinken. Jedenfalls reicht es noch zum Ausspannen vor den Abschlussarbeiten: den Film schnippeln und brennen.

4 Film fertigstellen

4.1 Codierten Film splitten

2-CD-Rips müssen wir nach dem Encoding noch auf die CDs aufteilen. Manchmal sind vielleicht auch drei CDs nötig, dann müssen wir eben einmal mehr splitten. Vom Vorgehen her macht das keinen Unterschied.

Anmerkung: Die AAC-Fans sollten sich erst einmal mit den nächsten beiden Kapiteln beschäftigen und dann wieder hierher zurück kommen.

Beim Aufteilen auf die CDs gibt es zwei Dinge zu beachten.

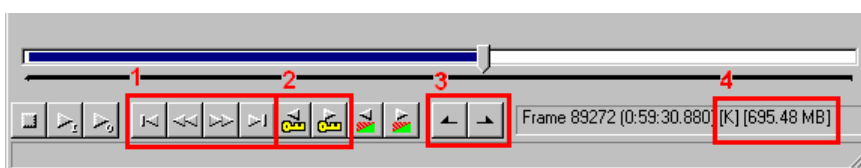
- Der Film muss *immer* an einem Keyframe geteilt werden, sonst entstehen am Anfang der zweiten Datei Bildfehler. Das MPEG-4-Format (XviD/DivX sind nichts anderes) speichert keine kompletten Bilder, sondern nur die Veränderung zum vorhergehenden Bild. Allein die Keyframes stellen vollständige Einzelbilder dar. Fehlt am Anfang der zweiten CD das Keyframe, fehlen dem Player erst einmal die Informationen, um ein komplettes Bild darzustellen. Das äußert sich in meist grünen Flächen anstatt der fehlenden Bildinformation und sieht natürlich gewaltig hässlich aus.

- Ein sinnvoller Schnitt erfolgt bei einem Szenenwechsel. Es ist unheimlich lästig, wenn der Held der Geschichte mitten im Satz unterbrochen wird, weil es Zeit für die zweite CD wird. Solche Schnitzer geben auch dem professionellst codierten Film einen laienhaften Beigeschmack.

Eine Einschränkung muss ich gelten lassen. Wenn der Film gerade so auf 2 CDs passt, ist kaum Spielraum zum Splitten vorhanden. Deswegen ist es manchmal doch nötig, den Schnitt innerhalb einer Szene zu setzen. Ein bisschen Fingerspitzengefühl dabei aber auch nicht schaden. Vor allem nie, nie mitten im Satz schneiden!

Zuständig für das Aufteilen ist entweder komplett VirtualDubMod oder – bei dynamischen VobSubs oder AAC-Audio – AVI-Mux GUI bzw. mkvtoolnix (dazu in den nächsten Kapiteln mehr). Im letzteren Fall suchen wir mit VirtualDubMod nur eine passende Stelle zum Schneiden. Und das geht so:

Wir laden den fertigen Film (die Datei incl. Audio!) über **File / Load Video file** und stellen im **Video**-Menü auf **Direct stream copy** um. In diesem Modus kopiert VDubMod den Film nur und encodiert nicht noch einmal neu. Dann springen wir 700 MB weit in den Film hinein (bei Rohlingen anderer Größe natürlich den entsprechenden Wert nehmen). Das funktioniert per **Strg+Shift+J** oder **Edit / Go to last Keyframe**. VirtualDubMod hüpfet zum gewünschten Keyframe.



Mit den **Keyframe-Buttons** (2) können wir keyframe-weise durch den Film springen, um eine günstige Stelle zum Schneiden auszusuchen. Vorwärts springen ist dabei nur

bedingt sinnvoll, schließlich ist da die CD zu Ende. Die genaue Dateiposition sehen wir unter (4), so wissen wir exakt, bei wie viel Megabyte wir uns befinden (fürs Splitting über mkvtoolnix ist das besonders wichtig). die Exaktheit bezieht sich leider nicht auf AVIs mit VBR-MP3-Audio. Bei solchen Dateien müssen wir ca. 5 MB dazurechnen, um auf den richtigen Wert zu kommen.

Haben wir eine passende Stelle gefunden, setzen wir dort den **Auswahl-Ende-Marker** (bei (3) der rechte Button). Dann springen wir mit dem linken Button bei (1) zum Anfang des Films und setzen hier den **Auswahl-Anfang-Marker** (bei (3) der linke Button). In der Zeitleiste ist jetzt der vordere Teil des Films markiert.

Jetzt ist auch der Zeitpunkt, an dem wir spätestens über **Streams / Stream List / Add** evtl. zusätzliche Untertitelspuren hinzufügen sollten. Anschließend speichern wir über **F7**. VirtualDubMod sichert immer nur den markierten Bereich des Films, so dass wir eine Datei erhalten, die genau auf die erste CD passen sollte.

Anmerkung: XviD (genau genommen eigentlich VfW) hält ein Hindernis bereit, wenn wir B-Frames verwenden. Dann erscheint die Meldung B-Frame Decoder Lag und wir bekommen erst beim ersten Drücken eines **Keyframe-Buttons** ein Bild. Dumm daran ist, dass das angezeigte Bild dann nicht unbedingt zur angesprungenen Framenummer passt, wir wissen also nicht exakt, wo wir uns im Film befinden. Abhilfe schafft nur ausprobieren: Schnittpunkt suchen, dort den **Anfangsmarker** setzen und ein kurzes Stück weiter hinten das Ende markieren. Diese Ministück Film dann abspeichern und im Software-Player anschauen. Dort sehen wir, wie groß die Abweichung war. Meistens liegen wir nur ein oder zwei Keyframes daneben, es artet also kaum in endloses Probieren aus.

Über **Edit / Move to Selection End** springen wir zurück zum Ende der ersten CD und setzen jetzt hier den **Auswahl-Anfang-Marker**. Ein Sprung zum Filmende mit dem rechten Button von (1) und hier den **Auswahl-Ende-Marker** setzen. Dann können wir die zweite CD speichern (mehr CDs brauchen natürlich entsprechend mehr Schnitte).

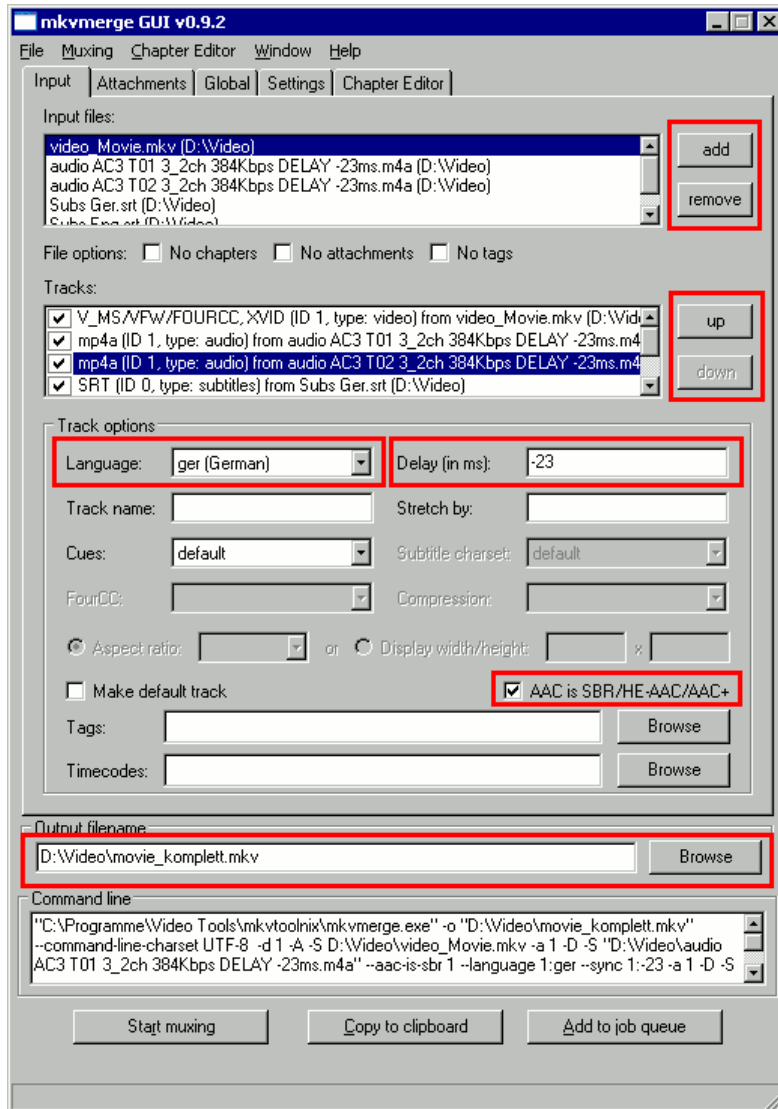
Das war's. Der Film ist fertig und bereit zum Brennen. Ein Test vorher, ob alles synchron ist und sämtliche Untertitelspuren, Kapitel etc. vorhanden sind, kann natürlich nicht schaden.

4.2 Muxen und splitten mit mkvmerge

Mkvtoolnix ist ein Toolpaket für den Matroska-Container. Für uns ist das interessant, wenn wir uns für Matroska und AAC-Audio entschieden haben, denn VirtualDubMod ist nicht AAC-fähig. In dem Fall konnten wir auch die Tonspur(en) nicht schon mit Gordian Knot zum Film dazumuxen. Wir haben also einen fertigen Film, der nur das Video enthält, und separat dazu AAC-Ton und evtl. Untertitel und Kapitel.

Muxen

Das Programm aus mkvtoolnix, um alles zusammenzufügen, heißt mkvmerge und ist genauso wie BeSweet eine Konsolenanwendung. Die grafische Oberfläche dazu nennt sich mmg, und die starten wir.



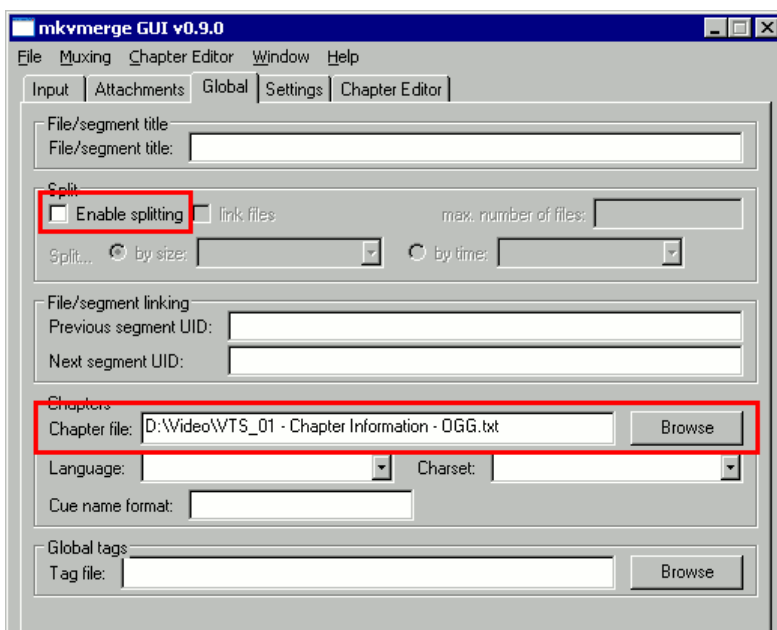
Mit **Add** oder per Drag&Drop können wir alle benötigten Dateien hinzufügen, im Bild sind das die von GKnot erzeugte Videodatei, zwei AAC-Audiospuren und zwei Sub-Rip-Untertitel. Die Spuren sind am Ende in der gleichen Reihenfolge in der Datei gespeichert wie sie hier erscheinen. Mit **Up** und **Down** können wir Spuren verschieben. Unter **Language** stellen wir für Audios und Untertitel die passende Sprache ein. Die erscheint dann beim Abspielen auch wieder im Player.

Delay hatten wir schon einmal bei BeSweet angesprochen. Um diese Anzahl Millisekunden muss die Audiospur verschoben werden, um synchron zum Video zu laufen. Den Wert aus dem Dateinamen (**DELAY -23ms**) übertragen wir incl. Vorzeichen in das

Delay-Feld. Wichtig: Wenn wir am Anfang bei BeSweet das Delay schon einmal berücksichtigt haben, gilt jetzt *immer* ein Delay von 0.

Den Haken bei **AAC is SBR...** müssen wir setzen, wenn unsere AAC-Tonspuren den High-Efficiency-Modus (HE) verwenden und nicht im MP4-Container verpackt sind. Dann kann mkvmerge nämlich das Vorhandensein von HE nicht automatisch erkennen. Da BeSweet nur MP4-AACs erstellt, sollte uns die Checkbox keine Probleme bereiten und kann deaktiviert bleiben. mkvmerge erkennt die BeSweet-AACs auch ohne unsere Hilfe korrekt.

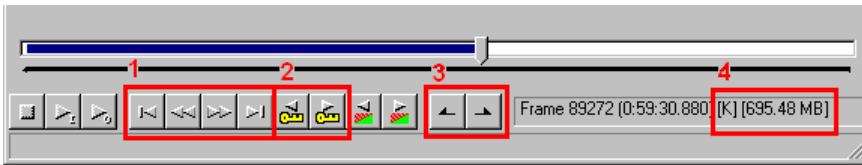
Ganz unten geben wir noch den Namen der Ausgabedatei an und wechseln dann ins Register **Global**.



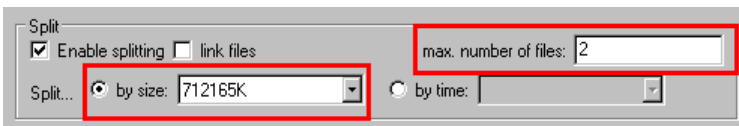
Enable splitting bleibt (bei 2-CD-Rips) erst einmal aus, da wir noch nicht wissen, wo der Film geteilt werden muss. Unter **Chapter file** können wir die Datei mit den Kapitelinformationen angeben, die wir ganz am Anfang mit DVD Decrypter erstellt haben. **File/segment title** gibt dem Film einen Namen (hat nichts mit dem Dateinamen zu tun), dann starten wir ganz links unten mit **Start muxing** den Muxing-Vorgang. Der dauert ein paar Minuten, dann liegt der Film komplett fertig in einem Stück auf der Platte und muss noch gesplittet werden. mmg können wir solange offen lassen.

Splitten

Den fertigen Film laden wir in VirtualDubMod und suchen wie unter Film splitten beschrieben einen passenden Punkt zum Schneiden heraus. Das Schneiden selbst muss dann mkvmerge übernehmen.



Leider lässt sich in mkvmerge keine exakte Framenummer als Schnittpunkt angeben, sondern nur Zeit oder Größe. mkvmerge sucht automatisch das nächste Keyframe nach der angegebenen Position und schneidet dort. Mit diesem Wissen können wir den Schnitt genauso exakt steuern wie über eine Framenummer, schließlich gibt uns VirtualDubMod an, bei wie viel Megabyte der gewünschte Schnittpunkt liegt (4). Diesen Wert rechnen wir erst in Kilobyte um (im Beispiel $695,48 \times 1024 = 712.171,52$) und runden dann ein wenig ab (z. B. auf 712.165).



Dann gehen wir zurück zu mmg, setzen den Haken bei **Enable splitting**, lassen **link files** deaktiviert (das wird afaik noch von keinem Player unterstützt) und geben bei **max number of files** die 2 für 2 CDs ein. Unter **Split by size** tragen wir unseren abgerundeten Wert ein und setzen ein **K** dahinter für Kilobyte. Alle anderen Einstellungen bleiben wie am Anfang, mit **Start muxing** erzeugen wir die endgültigen Dateien. Damit ist der Film komplett fertig.

Anmerkung: Für drei oder mehr CDs wird es noch ein wenig aufwändiger, da wir mkvmerge immer nur einen Schnittpunkt angeben können. D. h. wir nehmen die komplette Datei, splitten wie oben beschrieben. Dann haben wir einen Teil für die erste CD und einen zweiten größeren Teil für die restlichen CDs. Für diesen suchen wir in VirtualDubMod den nächsten Schnittpunkt und laden ihn dann in mmg (Achtung! Nicht wieder die ursprüngliche Nur-Video-Datei nehmen!). Einstellungen zu Tonspuren usw. sind nicht mehr nötig, denn die stecken in der Datei ja schon drin. Wir konfigurieren wieder das Splitting und teilen die Datei. Für alle weiteren CDs geht's genauso. Wichtig: **max. number of files** bleibt immer auf 2!

4.3 AAC im OGM- oder AVI-Container

Und nochmal AAC. Ja, VirtualDubMods Unfähigkeit, das neue Audioformat zu verarbeiten, kann einem ganz schön zu schaffen machen. Da hilft nur selbst Hand anzuzeigen, das haben wir weiter oben ja schon gemerkt. Nach der ausführlichen Anleitung für AAC in MKV fällt dieses Kapitel aber etwas schmaler aus. Ich bin nun mal überzeugter Matroska-User und habe daher mit OGM und AVI nicht mehr besonders viel zu tun. Eines vorne weg: AAC lässt sich in beide dieser Container muxen. Kein Pro-

blem bei OGM, AAC in AVI dagegen ist ein genauso dreckiger Hack wie VBR MP3 in AVI. Man sollte sich also mit dem Gedanken anfreunden, dass es funktionieren kann und sehr wahrscheinlich auch funktionieren wird, man aber nie eine Garantie bekommt.

Wie kriegt man also **AAC in OGM**? Für LC-AAC reicht Ogg Mux Nic⁸, für HE-AAC muss man den Weg über die 3ivx-Filter und Graphedit gehen.⁹ Das funktioniert auch für LC.

AAC in AVI kann nur ein einziges Programm: AVI-Mux GUI¹⁰. Das versteht leider die von BeSweet erzeugten MP4-AACs nicht, sondern möchte sie gern im ADTS-Format haben. Eine Umwandlung ist mit Ivan & Menno¹¹ möglich.

Anmerkung: Ein guter Einstiegspunkt für weitere Infos und Links ist die Doom9.org-Audio-FAQ¹².

5 Changelog

05.07.2004

- Erste Veröffentlichung.

08.07.2004

- Wegen Rechtslage Link zu DVD Decrypter entfernt.
- Unklarheiten bei der DivX-Konfig beseitigt (thx Hybrid).
- Guide als Zip-Archiv (Link auf der Startseite).

22.07.2004

- Untertitelkapitel überarbeitet.
- Rippingkapitel überarbeitet.
- einige Screenshots aktualisiert.
- HE-AAC-Checkbutton in mmg erwähnt.

8) http://nic.dnsalias.com/OggMux_Nic.zip

9) <http://www.3ivx.com> und <http://home.comcast.net/~appleguru/graphedt9.zip>

10) <http://www-user.tu-chemnitz.de/~noe/Video-Zeug/AVIMux%20GUI/index-eng.html>

11) <http://members.home.nl/w.speek/ivan&menno.htm>

12) <http://forum.doom9.org/showthread.php?s=&threadid=68300>

22.10.2004

- Abschnitt zum Kompressionstest aktualisiert.
- DivX-5.2.1-Konfig eingefügt.
- Layout: neues Format, das ohne Framesets auskommt.
- Ein paar Links gefixt.

02.12.2004

- Der Guide heißt jetzt »Brother Johns gesammeltes Encodingwissen«, da er über reines Gordian Knot doch schon ein Stück raus geht.
- Ein paar Links gefixt.
- Kurzen Abschnitt zum I-Frame-Intervall eingefügt (bei der XviD-Konfig).
- Audiokapitel umgebaut und um BeLight erweitert.
- Grundlagenkapitel eingefügt.

03.12.2004

- PDF-Version online gestellt.