

SelfLinux-0.13.1



MPlayer - The Movie Player for Linux



Autor: Moritz Bunkus (moritz@bunkus.org)
Formatierung: Matthias Hagedorn (matthias.hagedorn@selflinux.org)
Lizenz: GPL

Inhaltsverzeichnis

1 Wie diese Dokumentation zu lesen ist

2 Einleitung

3 Geschichte

4 MPlayer Installation

- 4.1 Voraussetzungen an die Software
- 4.2 Codecs
- 4.3 Grafikkarten
 - 4.3.1 YUV-Karten
 - 4.3.2 Nicht-YUV-Karten
- 4.4 Soundkarten

5 Features von MPlayer

- 5.1 Was ist mit dem GUI?
- 5.2 Untertitel und das OSD
 - 5.2.1 Andere Untertitelformate
 - 5.2.2 Angleichung der Untertitel-Anzeigezeit und der Platzierung
- 5.3 MPlayers eigenes Untertitelformat (MPsub)
- 5.4 Installation des OSD und der Untertitel
- 5.5 Das OSD-Menü
 - 5.5.1 Installation
- 5.6 RTC

6 Features

- 6.1 Unterstützte Formate
 - 6.1.1 Videoformate
 - 6.1.1.1 MPEG-Dateien
 - 6.1.1.2 AVI-Dateien
 - 6.1.1.3 ASF/WMV-Dateien
 - 6.1.1.4 QuickTime/MOV-Dateien
 - 6.1.1.5 VIVO-Dateien
 - 6.1.1.6 FLI-Dateien
 - 6.1.1.7 RealMedia-(RM)-Dateien
 - 6.1.1.8 NuppelVideo-Dateien
 - 6.1.1.9 yuv4mpeg-Dateien
 - 6.1.1.10 FILM-Dateien
 - 6.1.1.11 RoQ-Dateien
 - 6.1.1.12 OGG/OGM-Dateien
 - 6.1.1.13 SDP-Dateien
 - 6.1.1.14 PVA-Dateien
 - 6.1.1.15 GIF-Dateien
 - 6.1.2 Audio Formate
 - 6.1.2.1 MP3-Dateien
 - 6.1.2.2 WAV-Dateien
 - 6.1.2.3 OGG/OGM-Dateien (Vorbis)

- 6.1.2.4 WMA/ASF-Dateien
- 6.1.2.5 MP4-Dateien
- 6.1.2.6 CD-Audio
- 6.1.2.7 XMMS
- 6.2 Unterstützte Codecs
 - 6.2.1 Video-Codecs
 - 6.2.1.1 DivX4/DivX5
 - 6.2.1.2 FFmpeg's DivX/libavcodec
 - 6.2.1.3 XAnim-Codecs
 - 6.2.1.4 VIVO-Video
 - 6.2.1.5 MPEG 1/2-Video
 - 6.2.1.6 MS Video 1
 - 6.2.1.7 Cinepak CVID
 - 6.2.1.8 RealVideo
 - 6.2.1.9 XviD
 - 6.2.1.10 Sorenson
 - 6.2.1.11 Kompilierung von MPLAYER mit Unterstützung für Quicktime-Bibliotheken
 - 6.2.2 Audio-Codecs
 - 6.2.2.1 AC3-Decodierung in Software
 - 6.2.2.2 AC3 Decodierung in Hardware
 - 6.2.2.3 libmad-Unterstützung
 - 6.2.2.4 VIVO-Audio
 - 6.2.2.5 RealAudio
 - 6.2.2.6 QDesign-Codecs
 - 6.2.2.7 Qualcomm-Codec
 - 6.2.3 Win32 Codec-Importierungs-HOWTO
 - 6.2.3.1 Vfw-Codecs
 - 6.2.3.2 DirectShow Codecs
- 6.3 Video & Audio-Ausgabegeräte
 - 6.3.1 Video-Ausgabe-Geräte
 - 6.3.1.1 MTRR
 - 6.3.1.2 Videoausgabegeräte für traditionelle Grafikkarten
 - 6.3.1.2.1 Xv
 - 6.3.1.2.1.1 3dfx-Karten
 - 6.3.1.2.1.2 S3-Karten
 - 6.3.1.2.1.3 nVidia-Karten
 - 6.3.1.2.1.4 Xv mit ATI-Karten
 - 6.3.1.2.1.5 NeoMagic-Karten
 - 6.3.1.2.1.6 Xv mit Trident-Karten
 - 6.3.1.2.1.7 Kyro/PowerVR-Karten
 - 6.3.1.2.2 DGA
 - 6.3.1.2.3 SDL
 - 6.3.1.2.4 SVGAlib
 - 6.3.1.2.5 Framebuffer-Ausgabe (FBdev)
 - 6.3.1.2.6 Matrox-Framebuffer (mga_vid)
 - 6.3.1.2.7 3dfx-YUV-Unterstützung (tdxfb)
 - 6.3.1.2.8 OpenGL-Ausgabe
 - 6.3.1.2.9 AAlib - Ausgabe im Textmodus
 - 6.3.1.2.10 VESA-Ausgabe über das VESA-BIOS
 - 6.3.1.2.11 X11
 - 6.3.1.2.12 VIDIX
 - 6.3.1.2.12.1 VIDIX mit ATI-Karten
 - 6.3.1.2.12.2 Matrox-Karten

- 6.3.1.2.12.3 VIDIX mit Trident-Karten
- 6.3.1.2.12.4 DLabs-Karten
- 6.3.1.2.13 DirectFB
- 6.3.1.2.14 DirectFB/Matrox (dfbmga)
- 6.3.1.3 MPEG-Dekoderkarten
 - 6.3.1.3.1 DVB
 - 6.3.1.3.2 DXR2
 - 6.3.1.3.3 DXR3/Hollywood+
- 6.3.1.4 Andere Anzeigehardware
 - 6.3.1.4.1 Zr
 - 6.3.1.4.2 Blinkenlights
- 6.3.1.5 Unterstützung für die TV- Ausgabe
 - 6.3.1.5.1 Matrox G400-Karten
 - 6.3.1.5.2 Matrox G450/G550-Karten
 - 6.3.1.5.3 TV-Ausgabe mit ATI-Karten
 - 6.3.1.5.4 Voodoo 3
 - 6.3.1.5.5 nVidia
- 6.3.2 Audio-Ausgabe-Geräte
 - 6.3.2.1 Audio-/VideoSynchronisation
 - 6.3.2.2 Soundkarten-Erfahrungen, -Empfehlungen
 - 6.3.2.3 Audiofilter
 - 6.3.2.3.1 Up/Down-sampling
 - 6.3.2.3.2 Ändern der Anzahl der Kanäle
 - 6.3.2.3.3 Formatkonvertierung
 - 6.3.2.3.4 Verzögerung - Audiofilter
 - 6.3.2.3.5 Lautstärkekontrolle mit Software
 - 6.3.2.3.6 Equalizer
 - 6.3.2.3.7 Panning-Filter
 - 6.3.2.4 Sub-woofer
 - 6.3.2.4.1 Surround-sound-Decoder
 - 6.3.2.5 Audio-Plugins (veraltet)
 - 6.3.2.5.1 Up/Downsampling
 - 6.3.2.5.2 Surround-Sound-Decodierung
 - 6.3.2.5.3 Sample-Format-Konvertierer
 - 6.3.2.5.4 Verzögerung - Audio-Plugins
 - 6.3.2.5.5 Software Lautstärkeregelung
 - 6.3.2.5.6 Extrastereo
 - 6.3.2.5.7 Lautstärkenormalisierer
- 6.4 TV input
 - 6.4.1 Kompilierung für TV input
 - 6.4.2 Tipps zur Benutzung
 - 6.4.3 Beispiele
- 6.5 Edit Decision Lists (EDL)
 - 6.5.1 Benutzung von EDL-Dateien
 - 6.5.2 Erstellen von EDL-Dateien

7 Verwendung

- 7.1 Kommandozeile
- 7.2 Steuerung
 - 7.2.1 Konfiguration der Steuerung
 - 7.2.1.1 Tastennamen
 - 7.2.1.2 Befehle

- 7.3 GUI-Aktionen
 - 7.3.1 Steuerung mittels LIRC
 - 7.3.2 Der "Slave"-Modus
- 7.4 Streaming über das Netzwerk oder Pipes

8 CD/DVD-Sektion

- 8.1 CD-ROM- und DVD-ROM-Laufwerke
- 8.2 DVD-Wiedergabe
- 8.3 VCD Wiedergabe

9 FAQs - Die häufigsten Fragen und Antworten

- 9.1 Kompilierung
- 9.2 Allgemeine Fragen
- 9.3 Dateiwiedergabe-Probleme
- 9.4 Video/Audio-Treiber-Probleme
- 9.5 DVD Wiedergabe
- 9.6 Feature-Wünsche
- 9.7 Konvertierung

10 Portierungen

- 10.1 Linux
 - 10.1.1 Debian-Pakete
 - 10.1.2 RPM-Pakete
 - 10.1.3 ARM

11 Encodierung mit MEncoder

- 11.1 Encodieren mit MEncoder
 - 11.1.1 MPEG4-Encodierung ("DivX") in zwei oder drei Durchläufen
 - 11.1.2 In das MPEG-Format encodieren
 - 11.1.3 Skalierung von Filmen
 - 11.1.4 Stream-Kopiermodus
 - 11.1.5 Reparieren von AVIs mit defektem Index oder schlechtem Interleaving
 - 11.1.5.1 Aneinanderhängen mehrerer AVI-Dateien
 - 11.1.6 Encodierung mit der libavcodec-Codec-Familie
 - 11.1.7 Encodierung von mehreren Einzelbildern (JPEGs, PNGs oder TGAs)
 - 11.1.8 Extrahieren von DVD-Untertiteln in eine Vobsub-Datei
 - 11.1.9 Das Höhen-/Breitenverhältnisses erhalten

12 Bekannte Probleme

13 Anhang "Aufschrei der Entwickler"

- 13.1 Aufschrei der Entwickler
 - 13.1.1 GCC 2.96
 - 13.1.2 Vorkompilierte (binäre) Pakete
 - 13.1.3 nVidia-Probleme

1 Wie diese Dokumentation zu lesen ist

Wenn du die Installation zum ersten Mal durchführst: Pass auf, dass du alles von hier bis zum Ende der Installationsanleitung durchliest, und folge den entsprechenden Links. Wenn du immer noch Fragen hast, kehre zum Inhaltsverzeichnis zurück und suche nach dem Begriff, lies die [FAQ](#) oder versuche, mit Hilfe von [grep](#) die Dateien zu durchsuchen.

Die Hauptregel dieser Dokumentation: Wenn es nicht dokumentiert ist, **existiert es nicht**. Wenn nichts von einer Möglichkeit der Audioencodierung von TV-Karten steht, kann man dies auch nicht tun. Wenn du gut kombinieren kannst, dann ist das hier wirklich gut für dich. Viel Glück. Du wirst es brauchen :) Ein weiterer guter Ratschlag ist folgendes Zitat von Chris Phillips, der auf der [mplayer-users](#) Mail-Liste folgendes schrieb:

Vor einiger Zeit habe ich gesagt, dass es einen Unterschied zwischen einem Neuling und einem Idioten gibt. Egal, wieviel du von einer Sache wirklich weißt (sei es Linux, Autos, Frauen :D), du solltest IMMER dazu fähig sein, einen Schritt zurück zu machen und objektiv die Lage zu betrachten. Andernfalls bist du meiner bescheidenen Meinung nach einfach nur dumm. Eine Frau, mit der ich zusammen wohne, nahm einmal an, dass der Staubsauger defekt war, nur weil er nichts mehr aufgesaugt hat. Sie hat nie daran gedacht, den Staubbeutel zu wechseln, weil sie das nie vorher gemacht hatte. Und das ist einfach nur doof und kein Fall von 'ich weiß nicht, was ich tun soll'. Mit etwas nicht vertraut zu sein ist absolut keine Entschuldigung für a) Faulheit und b) Unwissenheit. Es gibt so viele Menschen, die das Wort **Fehler** sehen und dann einfach aufhören zu denken. Tatsächlich scheinen nur wenige auch die Wörter hinter dem Punkt zu lesen.

2 Einleitung

MPlayer ist ein Video-Player für LINUX (er läuft auch auf vielen anderen Unix-Systemen und **nicht-x86**-CPUs, siehe ► [Sektion 6](#) . Er spielt die meisten MPEG, VOB, AVI, OGG/OGM, VIVO, ASF/WMA/WMV, QT/MOV/MP4, FLI, RM, NuppelVideo, yuv4mpeg, FILM, RoQ und PVA-Dateien ab. Dabeikommen viele eingebaute Codecs, Xanim-, RealPlayer- und Win32-DLL-Codecs zum Einsatz. Du kannst dir **VideoCDs, SVCDs, DVDs, 3ivx-, RealMedia-** und auch **DivX**-Filme sehen (und dafür wird nicht einmal die avifile-Bibliothek benötigt). Ein weiteres großes Feature von MPlayer ist die große Bandbreite unterstützter Ausgabetreiber. MPlayer funktioniert mit X11, Xv, DGA, OpenGL, SVGAlib, fbdev, AALib und DirectFB. Du kannst aber auch GGI und SDL (und dadurch alle deren Treiber) und einige grafikkartenspezifische Low-Level-Treiber verwenden - z.B. für Matrox, 3Dfx und Radeon, Mach64 oder Permedia3. Die meisten von ihnen unterstützen Software- oder Hardware-Skalierung, sodass du deine Videos im Vollbild genießen kannst. MPlayer unterstützt auch die Wiedergabe mit einigen Hardware-MPEG-Decoder-Karten wie der ► [DVB](#) und der ► [DXR3/Hollywood+](#)! Und was meinst du zu den schönen, großen, antialiasten und schattierten Untertiteln (**10 unterstützte Typen**) mit europäischen (ISO 8859-1,2, Ungarisch, Englisch, Tschechisch, usw.), kyrillischen und koreanischen Schriften? Und das onscreen display (OSD)?

Der Player spielt felsenfest beschädigte MPEG-Dateien (nützlich für einige VCDs) und schlechte AVI-Dateien, welche auch mit dem bekannten Windows Media Player nicht mehr abspielbar sind. Auch AVI-Dateien ohne Index sind abspielbar, und du kannst den Index entweder temporär mit der `-idx`-Option oder permanent mit MEncoder wiederherstellen. Beides ermöglicht wieder die Suchfunktion! Du siehst also, Stabilität und Qualität sind die wichtigsten Dinge für mich, aber die Geschwindigkeit ist genauso beeindruckend.

MEncoder (der Filme encodierende Partner von MPlayer) ist ein einfacher Filmencodierer, der dazu bestimmt ist, von MPlayer abspielbare Filme (**AVI/ASF/OGG/DVD/VCD/VOB/MPG/MOV/VIV/FLI/RM/NUV/NET/PVA**) in andere von MPlayer abspielbare Formate umzuwandeln (siehe unten). Er kann verschiedene Codecs zum Encodieren benutzen (**DivX4** (mit 1 oder 2 Durchgängen), **libavcodec**, **PCM/MP3/VBR MP3-Audio**). Er verfügt weiterhin über ein mächtiges Pluginsystem für die Manipulation des Videos.

Features von MEncoder:

- * verfügt über die gleiche große Bandbreite von Eingangsformaten wie MPlayer
- * kann alle von FFmpegs ► [libavcodec](#) zur Verfügung gestellten Codecs benutzen
- * Videoencodierung direkt von einem V4L-kompatiblen TV-Gerät
- * Encodieren/Multiplexen in ein verschachteltes (interleaved) AVI mit einem korrekten Index am Ende
- * Erstellen des Audio-Streams von externen Audiodateien
- * Encodieren mit einem, zwei oder drei Durchgängen
- * **VBR MP3-Audio - WICHTIGER HINWEIS:** VBR-MP3-Audio wird eventuell nicht immer korrekt von Windows-Playern wiedergegeben!
- * PCM-Audio
- * 1:1-Streamkopien
- * Audio-/Videosynchronisation basierend auf den PTS (kann mit der Option `-mc 0` deaktiviert werden)
- * Anpassung der Framerate mit der `-ofps`-Option (kann z.B. dazu benutzt werden, 29.95fps-VOBs in 24fps-AVIs umzuwandeln)
- * benutzt unser äußerst mächtiges Pluginsystem (beschneiden (`crop`), erweitern (`expand`), spiegeln (`flip`), Nachbearbeitung (`postprocess`), drehen (`rotate`), skalieren (`scale`), RGB/YUV-Konvertierung etc.)
- * kann DVD/Vobsub-Untertitel **UND** Textuntertitel direkt in das Bild einbetten
- * kann DVD-Untertitel in das Vobsub-Untertitel extrahieren

Geplante Features:

- * noch größere Bandbreite von verfügbaren En-/Decodier-Formate/-Codecs (z.B. die Erstellung von VOB-Dateien mit DivX4/Indeo5/VIVO-Streams ;))

MPlayer und MEncoder können unter den Bedingungen der GNU General Public License Version 2 vertrieben werden.

3 Geschichte

Es begann vor einem Jahr... Ich (A'rpi) hatte eine Menge von Playern unter Linux ausprobiert (mtv, xmps, dvdview, livid/oms, videolan, xine, xanim, avifile, xmp), aber sie hatten alle ihre Probleme. Meist mit speziellen Dateien oder mit der Audio-Video-Synchronisation. Die meisten waren unfähig, alle drei - MPEG1, MPEG2 und AVI (DivX) - abzuspielen. Einige Player hatten auch Probleme mit der Bildqualität und der Geschwindigkeit. So beschloss ich, einen Player zu schreiben/zu modifizieren...

- * **mpg12play v0.1-v0.3:** Sep 22-25, 2000
Der erste Versuch, innerhalb von nur 30 Minuten zusammengehackt! Ich verwendete `libmpeg3` von  www.heroinewarrior.com bis zu Version 0.3, aber es gab Bildqualitäts- und Geschwindigkeitsprobleme.
- * **mpg12play v0.5-v0.87:** Sep 28-Oct 20, 2000
Der MPEG-Codec wurde durch `DVDview` von *Dirk Farin* ersetzt. Es war ein großartiges Programm, aber es war langsam und in C++ geschrieben (ich hasse C++!).
- * **mpg12play v0.9-v0.95pre5:** Oct 21-Nov 2, 2000
Als MPEG-Codec wurde `libmpeg2` (`mpeg2dec`) von *Aaron Holtzman & Michel Lespinasse* verwendet. Es ist großartiger, optimierter und schneller C-Code mit perfekter Bildqualität und 100%ig zum MPEG-Standard kompatibel.
- * **MPlayer v0.01:** Nov 11, 2000
Der erste MPlayer.
- * **MPlayer v0.3-v0.9:** Nov 18-Dec 4, 2000
Diese Version war ein Paket aus 2 Programmen: `mpg12play v0.95pre6` und mein neuer, einfacher AVI-Player `avip`, der auf dem Win32-DLL-Loader von `avifile` basierte.
- * **MPlayer v0.10:** Jan 1, 2001
Der MPEG- und AVI-Player in einem einzigen Binary!
- * **MPlayer v0.11pre Serien:**
Einigen neue Entwickler sind beigetreten, und seit 0.11 war MPlayer Teamarbeit. ASF-Datei- und OpenDivX-Unterstützung (siehe  www.projectmayo.com) für en-/decoding hinzugefügt.
- * **MPlayer v0.17 "The IdegCounter"** Apr 27, 2001
Die Release-Version nach 0.11pre nach 4 Monaten schwerer Entwicklung. Probiere es und sei beeindruckt! Tausende von neuen Features hinzugefügt... Und natürlich wurde auch der alte Code verbessert, Fehler entfernt usw.
- * **MPlayer 0.18 "The BugCounter"** Jul 9, 2001
2 Monate seit 0.17, und hier ist das neue Release. Komplette ASF-Unterstützung, mehr Untertitel-Formate, `libao` wurde eingeführt (wie `libvo`, aber für Audio), stabiler denn je, und so weiter. Es ist ein muss!
- * **MPlayer 0.50 "The Faszom(C)ounter"** Oct 8, 2001
Hmmm. Wieder ein Release. Tonnen von neuen Features, Beta-Version vom GUI, Fehler beseitigt, neue vo- und ao-Treiber, Ports auf viele Systeme inklusive OpenSource-DivX-Codecs und vieles mehr. Probier's!
- * **MPlayer 0.60 "The RTFMCounter"** Jan 3, 2002
MOV/VIVO/RM/FLI/NUV-Dateiformat-Unterstützung, native CRAM-, Cinepak-, ADPCM-Codecs und Support für die Binary-Codecs von XAnim; DVD-Untertitel-Unterstützung, erstes Release des MEncoders, TV-Grabbing, Cache, liba52, unzählbare Fehlerbehebungen.
- * **MPlayer 0.90pre10 "The BirthdayCounter"** Nov 11, 2002
Obwohl diese Version kein Release ist werde ich sie hier erwähnen, da sie genau zwei Jahre nach MPlayer v0.01 herauskam. Herzlichen Glückwunsch, MPlayer!
- * **MPlayer 0.90 "?"**
Datum noch unbekannt
- * **MPlayer 0.90rc1 "The CodecCounter"** Dec 7, 2002
Schon wieder kein Release, aber MPlayer ist nach den neuen Features (Unterstützung für Sorenson 3 (QuickTime) und Windows Media 9) der einzige Player, der alle bekannten Videoformate unterstützt!

4 MPlayer Installation

In diesem Kapitel versuche ich, dich auf dem Weg der Konfiguration und Kompilierung von MPlayer zu begleiten. Es ist nicht einfach, aber auch nicht unnötig schwer. Wenn dir ein unterschiedliches Verhalten als hier beschrieben auffällt, versuche bitte diese Dokumentation zu durchsuchen, und du wirst eine Antwort erhalten. Wenn du Links siehst, folge ihnen bitte und lies, was sie enthalten. Es wird einige Zeit benötigen, aber es zahlt sich aus.

Du brauchst ein ziemlich aktuelles System. Unter Linux sind 2.4.x-Kernel empfohlen.

4.1 Voraussetzungen an die Software

- * **binutils** - vorgeschlagene Version ist **2.11.x**. Diese Programme sind für das Erzeugen von MMX/3DNow!/usw-Befehlen verantwortlich und deshalb sehr wichtig.
- * **gcc** - vorgeschlagene Version ist von **2.95.3**, (eventuell **2.95.4**) und **3.3**. Verwende **NIEMALS** 2.96 oder 3.0.x! Sie erzeugen für MPlayer fehlerhaften Code. Wenn du dich dazu entscheidest, den 2.96 über Bord zu werfen, dann nimm nicht einfach den 3.0.x, nur weil er neuer ist! Frühe Versionen von 3.0.x waren sogar schlechter als 2.96, downgrade also zu 2.95.x (downgrade auch die **libstdc++**, sie wird vielleicht von anderen Programmen benötigt) oder up-/downgrade überhaupt nicht (sei aber in diesem Falle auf Laufzeitfehler vorbereitet). Wenn du 3.x,x verwenden willst, dann solltest du die neuste Version ausprobieren. Ältere hatten verschiedene Fehler. Es sollte also darauf geachtet werden, dass wenigstens 3.1 verwendet wird, da diese Version getestet wurde und funktionieren sollte. Detaillierte Information dazu (die Fehler von gcc 2.96 sind immer noch **nicht** behoben, sondern wurden nur in MPlayer **umgangen**) können in der gcc [2.96-Sektion](#) und in den [FAQs](#) gefunden werden.
- * **XFree86** - vorgeschlagene Version ist **immer die neueste (4.2.1)**. Normalerweise will das jeder, da XFree86 seit Version 4.0.2 die [XVideo](#)-Erweiterung enthält (auch bezeichnet als Xv), die benötigt wird, um Hardware-YUV-Beschleunigung (schnelle Bildanzeige) auf Karten zu benutzen, die es unterstützen. Stelle sicher, dass die **Development-Pakete** installiert sind, sonst wird es nicht funktionieren. Bei einigen Grafikkarten wird XFree86 noch nicht einmal benötigt. Für eine Liste siehe unten.
- * **make** - vorgeschlagene Version ist **immer die neueste** (mindestens 3.79.x). Dies ist normalerweise nicht so wichtig.
- * **SDL** - SDL ist nicht vorgeschrieben, aber es ist in einigen Fällen hilfreich (bei schlechtem Ton oder Videokarten, welche Probleme mit dem Xv-Treiber haben). Verwende immer die aktuellste Version, mindestens aber 1.2.x).
- * **libjpeg** - optionaler (M)JPEG-Decoder, der bei -mf und einigen QT/MOV-Dateien benutzt wird. Sowohl für MPlayer als auch für MEncoder nützlich.
- * **libpng** - empfohlen und Standard-(M)PNG-Decoder. Wird für das GUI benötigt. Sowohl für MPlayer als auch für MEncoder nützlich.
- * **lame** - empfohlen, wird benötigt, um mit MEncoder MP3-Audio zu encodieren. Die empfohlene Version ist **immer die neuste** (wenigstens 3.90).
- * **libogg** - optional, wird für das OGG-Dateiformat benötigt.
- * **libvorbis** - optional, wird für Vorbis-Audio benötigt.
- * [LIVE.COM Streaming Media](#) - optional, wird für die Wiedergabe von RTSP/RTP-Streams benötigt.
- * **directfb** - optional, von <http://www.directfb.org>
- * **cdparanoia** - optional, für CDDA-Unterstützung
- * **libfreetype** - optional, für die Unterstützung von TrueType-Schriften. Mindestens Version 2.0.9 wird benötigt.
- * **libxmmms** - optional, für Unterstützung von XMMS-Inputplugins. Mindestens Version 1.2.7 oder neuer wird benötigt.

4.2 Codecs

- * **libavcodec**: Dieses Codec-Paket kann mit H263/MJPEG/RV10/DivX3/DivX4/DivX5/MP41/MP42/WMV1 codierte Filme und mit WMA (Windows Media Audio) v1/v2 codierte Audiostreams auf mehreren Plattformen decodieren. Es ist bekanntermaßen auch die schnellste Variante für diese Aufgaben. Details können in der [▶ libavcodec](#)-Sektion gefunden werden.
Features:
 - * Decodierung der genannten Codecs auf nicht-x86 Maschinen
 - * Encodierung mit den meisten der genannten Codecs
 - * Dies ist der **schnellste verfügbare** Codec für DivX/3/4/5 und andere MPEG4-Typen und wird deshalb sehr empfohlen!
- * **Win32-Codecs**: Wenn du planst, MPlayer auf einer x86 Architektur zu verwenden, wirst du sie möglicherweise brauchen. Lade dir dazu die [w32codecs.zip](#) herunter und entpacke sie nach [/usr/lib/win32](#).
Hinweis: Das Avifile-Projekt hat ein ähnliches Codecs-Paket, aber es unterscheidet sich von unserem. Wenn du also alle unterstützten Codecs verwenden willst, dann benutze unser Paket. Du kannst aber auch problemlos unserer Paket für [avifile](#) nutzen.
Features:
 - * Du wirst sie möglicherweise brauchen, um ein Video abzuspielen oder zu encodieren, das mit verschiedenen Hardware-Kompressoren wie Tuner-Karten und digitalen Kameras (z.B. DV, ATI VCR, MJPEG) aufgenommen wurde.
 - * Erforderlich, wenn du **WMV8- oder WMV9/WMA9**-Filme abspielen willst. Es wird nicht für alte ASFs mit MP41- oder MP42-Video benötigt (auch wenn VoxWare Audio für diese Dateien häufig anzutreffen ist und dieses von den Win32-Codecs behandelt wird), genausowenig für WMV7. Nicht einmal für WMA (Windows Media Audio) werden sie benötigt - [libavcodec](#) hat einen OpenSource-Decoder dafür.
- * **QuickTime-Codecs**: Auf der x86-Plattform können diese Codecs benutzt werden, um Sorenson v1/v3, RPZA und andere QuickTime-Videoformate sowie QDesign Audiostreams zu decodieren. Installationsanweisungen finden sich in der Sektion über den [▶ Sorenson Videocodec](#).
- * DivX4/DivX5: Wenn du MEncoder verwendest, solltest du auch [libavcodec](#) (siehe oben) verwenden, was schneller ist und eine bessere Qualität bietet.
Features:
 - * 1-Pass- oder 2-Pass-Encodierung mit [▶ MEncoder](#)
 - * Du kannst alte DivX3-Film schneller als mit Win32-DLL-Codecs aber langsamer als mit [libavcodec](#) ansehen.
 - * Es ist Closed-Source und nur in einer x86-Version verfügbar.
- * **Xvid**: Open Source-Encodierungs-Alternative zu Divx4Linux
Features:
 - * 1-Pass- oder 2-Pass-Encodierung mit [▶ MEncoder](#)
 - * Es ist Open-Source und nicht nur als x86 Version verfügbar.
 - * Es ist ungefähr zwei mal schneller als DivX4 bei der Encodierung - und das bei gleicher oder sogar besserer Qualität.
- * Die [▶ XAnim-Codecs](#) sind die besten Codecs (Vollbild, hardwareseitig unterstützter YUV-Zoom), um **3ivx**- und Indeo 3/4/5-Filme sowie einige alte Formate abzuspielen. Und sie sind für verschiedene Plattformen verfügbar, sodass sie die einzige Möglichkeit sind, wenn du Indeo-Video auf nicht-x86-Plattformen abspielen möchtest (nun mal abgesehen von XAnim :). Andere Codecs wie z.B. der Cinepak-Codec werden am besten mit dem MPlayer eigenen Cinepak-Decoder wiedergegeben.

- * Für **Ogg Vorbis**-Audiodecodierung wird eine korrekte Installation von `libvorbis` benötigt. Es sollten, falls vorhanden, `deb/rpm`-Pakete werden oder diese Quellen  http://ogg.org/ogg/vorbis/download/vorbis_nightly_cvstgz kompiliert werden (dies ist eine täglich erneuerter tarball des Vorbis CVS).
- * MPlayer kann die Libraries von RealPlayer 8 oder RealONE verwenden, um Dateien mit **RealVideo 2.0 bis 4.0** und Sipro/Cook-Audio wiederzugeben. Eine Installationsanleitung und weitere Informationen finden sich in der Sektion ► [RealMedia-Dateiformat](#).

4.3 Grafikkarten

Es gibt allgemein gesprochen zwei Arten von Grafikkarten. Die eine Art (die neueren Karten) besitzen **Hardware-Skalierung und YUV-Beschleunigung**, die anderen nicht.

4.3.1 YUV-Karten

YUV-Karten können das Bild anzeigen und auf jede beliebige Größe skalieren, die noch in ihren Speicher passt. Dabei kommt es zu sehr **sehr geringer CPU-Belastung** (Zoomen erhöht sie nicht!), weswegen Abspielen im Vollbild gut aussieht und sehr schnell ist.

- * **Matrox G200/G400/G450/G550-Karten:** Obwohl ein ► [Vidix](#) Treiber zur Verfügung gestellt wird, wird empfohlen, stattdessen die alten `mga_vid`-Kernelmodule zu verwenden, da diese viel besser funktionieren. in der ► [mga_vid](#) Sektion. Es ist wichtig, diese Schritte vor der Kompilierung von MPlayer vorzunehmen, da ansonsten keine `mga_vid`-Unterstützung einkompiliert wird. Du solltest auch die ► [Matrox TV-out](#)-Sektion lesen. **Nicht-Linux-Benutzer** können nur die Vidix-Treiber verwenden. Lies dazu die ► [Vidix](#)-Sektion.
- * **3Dfx Voodoo3/Banshee-Karten:** Du solltest die ► [Sektion](#) lesen, um eine möglichst hohe Geschwindigkeit zu erzielen. Es ist wichtig, diese Schritte **vor** der Kompilierung von MPlayer vorzunehmen, da ansonsten keine 3Dfx-Unterstützung einkompiliert wird. Du solltest auch die ► [3dfx TV-out](#)-Sektion lesen. Wenn X verwendet wird, solltest du mindestens Version 4.2.0 installiert haben, da die 3dfx Xv-Treiber in 4.1.0 und den früheren Versionen kaputt waren!
- * **ATI-Karten:** Der ► [Vidix](#)-Treiber wird für folgende Karten zur Verfügung gestellt: **Radeon, Rage128, Mach64** (Rage XL/Mobility, Xpert98). Du solltest auch die Sektion über ► [ATI-Karten](#) TV-out-Dokumentation lesen, um herauszufinden, ob der jeweilige TV-out der Karte unter Linux/MPlayer unterstützt wird.
- * **S3-Karten:** Die Savage und Virge/DX Chips besitzen Hardwarebeschleunigung. Verwende die neueste XFree86 Version, ältere Treiber sind fehlerhaft. Savage-Chips machen Problem mit YV12-Anzeige, siehe ► [S3-Xv](#)-Sektion für Details. ältere, Trio-Karten haben keinen oder nur langsamen Hardwaresupport.
- * **nVidia-Karten:** Solche können eventuell eine gute Wahl für Videowiedergabe sein. Wenn du weder eine GeForce2 noch eine neuere Karte hast, dann wird es wahrscheinlich nicht fehlerfrei funktionieren. **Der in XFree86 eingebaute nVidia-Treiber stellt bei manchen Karten keine YUV-Beschleunigung zur Verfügung!** Du musst dir also eventuell die Closed-Source-Treiber von  nvidia.com herunterladen. Nähere Details findest du in der ► [nVidia-Xv](#)-Treiber-Sektion. Wenn du den TV-Ausgang der Karte benutzen willst, dann schau auch in der ► [nVidia-TV-Out](#)-Sektion nach.
- * **3DLabs GLINT R3 und Permedia3:** Ein ► [Vidix](#)-Treiber steht zur Verfügung (`pm3_vid`).
- * **Andere Karten:** Du hast keine der oben genannten?
 - * Schau, ob dein XFree86-Treiber (und deine Karte) Hardwarebeschleunigung unterstützt.
 - * Wenn nicht, werden die Features deiner Karte unter deinem Betriebssystem nicht unterstützt :(Wenn die Hardware-Skalierung unter Windows läuft, bedeutet dies nicht, dass das auch unter Linux oder anderen Betriebssystem funktioniert. Es hängt von den Treibern ab! Die meisten Hersteller erstellen weder Linux-Treiber, noch geben sie die Chipspezifikationen frei - du bist also der Unglückliche, wenn du diese Karten verwendest.

4.3.2 Nicht-YUV-Karten

Vollbild kann durch Zoomen durch eine **Softwareskalierung** (benutze die Option `-zoom` oder `-vop scale`) erreicht werden. Aber sei gewarnt: Es ist **langsam**! Eventuell reicht es auch schon aus, in einen kleineren Videomodus zu wechseln, z.B. zu 352x288. Wenn keine YUV-Beschleunigung zur Verfügung steht, ist diese letztgenannte Methode zu empfehlen. Bei MPlayer wird dieses mit der `-vm`-Option angeschaltet und funktioniert mit folgenden Karten:

- * **Verwendung von XFree86:** siehe Details in der [DGA-Treiber](#) und der [X11-Treiber](#). DGA ist empfohlen! Versuche auch, DGA via SDL zu verwenden. Manchmal funktioniert das besser.
- * **Ohne XFree86:** versuche die Treiber in folgender Reihenfolge:
[vesa](#), [fbdev](#), [svgalib](#), [aalib](#).

Einige Karten:

* **Cirrus-Logic-Karten:**

- * GD 7548: In Compaq Armada 41xx-Notebook-Serien eingebaut und getestet.
- * XFree86 3: Funktioniert im 8/16bpp-Modus. Der Treiber ist aber in 800x600@16bpp zu langsam und fehlerbehaftet.
Empfohlen: 640x480@16bpp
- * XFree86 4: Der Xserver friert bald nach dem Start ein, es sei denn, die Beschleunigung ist deaktiviert. Dann wird jedoch alles langsamer als XFree86 3. Kein XVideo.
- * FBdev: Der Framebuffer kann mit dem `clgenfb`-Kerntreiber aktiviert werden, obwohl er bei mir nur im 8bpp-Modus funktioniert hat und somit für Videowiedergabe unbenutzbar ist. Ich musste vor der Kompilierung den Sourcen des `clgenfb` die ID des 7548-Chips hinzufügen.
- * VESA: Die Karte kann bloss VBE 1.2, sodass der VESA-Framebuffer nicht verwendet werden kann. Dieses Problem kann nicht mit UniVBE umgangen werden.
- * SVGAlib: Erkennt ältere Cirrus Chips. Es funktioniert mit mit `-bpp 8`, ist aber ziemlich langsam.

4.4 Soundkarten

- * **Soundblaster Live!:** Diese Karte kann 4 oder 6 (5.1) Kanäle (von AC3) anstelle von 2 verwenden. Lies die Sektion über [AC3-Decodierung in Software](#). Für die unveränderte Ausgabe der AC3-Daten an den Ausgang **musst** du ALSA 0.9 mit der OSS-Emulation verwenden!
- * **C-Media mit SP/DIF Ausgang:** Hardware-AC3-Durchgabe ist mit diesen Karten möglich.
- * Die Features von **anderen Karten** werden von MPlayer nicht unterstützt.

5 Features von MPlayer

- * Entscheide dich, ob du das GUI brauchst. Wenn du es brauchst, wirf einen Blick auf die [GUI](#)-Sektion vor dem Kompilieren.
- * Wenn du MEncoder (unsen grossartigen Encoder-für-Alles) installieren willst, dann lies die [MEncoder](#)-Sektion.
- * Wenn du eine V4L-kompatible **TV-Tuner-Karte** hast und du mit MPlayer fernsehen oder encodieren willst, dann lies die [TV-Input](#)-Sektion.
- * Inzwischen gibt es ein hübsches **OSD-Menü**, das nur darauf wartet, ausprobiert zu werden. Schau dazu in der [OSD-Menü](#)-Sektion nach.

Nun kompiliere MPlayer:

```
user@linux ~/ $ ./configure
user@linux ~/ $ su

Password:

root@linux ~/ # make
root@linux ~/ # make install
```

Ab diesem Zeitpunkt ist MPlayer zur Benutzung bereit. Das Verzeichnis `$PREFIX/etc/mplayer` enthält die Datei `codecs.conf`, welche benutzt wird, um dem Programm alle Codecs und deren unterstützte Features mitzuteilen. Diese Datei sollte zusammen mit dem Hauptprogramm immer aktuell gehalten werden!

Du solltest außerdem überprüfen, ob eine `codecs.conf` deinem Home-Verzeichnis (`~/.mplayer/codecs.conf`) von einer alten MPlayer-Version vorhanden ist. Diese solltest du unbedingt löschen!

Debian-Benutzer können ein `.deb`-Paket für sich kompilieren, was sehr einfach ist. Du musst bloß `fakeroot debian/rules binary` in MPlayer's Sourceverzeichnis ausführen. Genauere Anleitungen können in der [Debian](#)-Sektion gefunden werden.

Schau dir immer die Ausgabe von `./configure` und die `configure.log`-Datei an. Beide enthalten Infos darüber, welche Features einkompiliert werden, und welche nicht. Du kannst dir auch die Dateien `config.h` und `config.mak` ansehen. Wenn einige Bibliotheken installiert sind, diese aber nicht von `./configure` erkannt werden, dann solltest du überprüfen, ob auch die dazugehörigen Header-Dateien installiert sind (welche sich meist in `-dev`-Paketen befinden), und ob die Versionsnummern übereinstimmen. Das `configure.log` sagt dir normalerweise, was fehlt.

Auch wenn sie nicht zwingend erforderlich sind, solltest du doch Schriften installieren, um das OSD und Untertitel benutzen zu können. Die empfohlene Methode ist, eine TrueType-Schrift zu installieren und MPlayer anzuweisen, diese zu benutzen. Wie das funktioniert, kannst du in der Sektion über [Untertitel und das OSD](#) nachlesen.

5.1 Was ist mit dem GUI?

Das GUI basiert auf GTK 1.2.x (nicht das ganze GUI benutzt GTK, sondern nur die Dialoge). Die Skins sind als PNG-Grafiken gespeichert, so dass du also auch gtk und die libpng (und deren `devel`-Pakete) installieren musst. Du kannst das GUI durch den Parameter `--enable-gui` bei `./configure` anschalten. Um MPlayer im GUI-Modus zu betreiben, kannst du entweder

- * `gui=yes` in deine Config-File eintragen oder
- * `ln -s $PREFIX/bin/mplayer $PREFIX/bin/gmplayer` ausführen und `gmplayer` aufrufen.

Im Moment kannst du die `-gui`-Kommandozeilenoption nicht verwenden.

Da MPlayer keine Skins enthält, musst du dir welche herunterladen, wenn du das GUI verwenden willst. Schau dazu auf die Download-Seite  <http://www.mplayerhq.hu/homepage/dload.html>. Diese sollten in das normale, systemweite Verzeichnis (`$PREFIX/share/mplayer/Skin`) oder nach `$HOME/.mplayer/Skin` extrahiert werden. MPlayer sucht standardmäßig nach einem Verzeichnis namens `default`, aber du kannst mit der Option `-skin` (neuer Skin) oder dem Eintrag `skin=(neuer Skin)` in der Konfigurationsdatei den Skin `/Skin/(neuer Skin)` auswählen.

5.2 Untertitel und das OSD

MPlayer kann Untertitel zu Filmen darstellen. Momentan werden folgende Formate unterstützt:

- * VobSub
- * OGM
- * CC (closed caption)
- * Microdvd
- * SubRip
- * SubViewer
- * Sami
- * VPlayer
- * RT
- * SSA
- * MPsub
- * AQTitle
- * JACOsub

MPlayer kann die oben aufgeführten Formate (bis auf die ersten drei) in die folgenden Formate konvertieren, wenn du die jeweils dahinter angegebene Option benutzt:

- * MPsub: `-dumpmpsub`
- * SubRip: `-dumpsrtsub`
- * Microdvd: `-dumpmicrodvds`
- * JACOsub: `-dumpjacsub`
- * Sami: `-dumpsami`

Die Kommandozeilenoptionen unterscheidet sich leicht für die unterschiedlichen Formate:

VobSub-Untertitel bestehen aus einer mehreren MB großen `.SUB`-Datei und optionalen `.IDX`- und/oder `.IFO`-Dateien.

Benutzung: wenn du Dateien wie `beispiel.sub`, `beispiel.ifo` (optional) und `beispiel.idx` hast, dann musst du die Optionen `-vobsub beispiel [-vobsubid <id>]` angeben (optional mit Pfadangabe natürlich). Die `-vobsubid`-Option funktioniert wie `-sid` bei DVDs. Du kannst mit ihr zwischen Untertitel-Sprache wählen. Falls keine `-vobsubid` angegeben wird, versucht MPlayer, die mit `-slang` angegebene Sprache zu benutzen und zur Not die durch `langidx` in der `.IDX`-Datei angegebene Sprache wählen. Wenn das ebenfalls fehlschlägt, werden keine Untertitel angezeigt.

5.2.1 Andere Untertitelformate

Die anderen Formate bestehen aus einer einzigen Textdatei, welche das Timing, die Platzierung und die Texte selber enthält. Verwendung:

Wenn man eine Datei wie `beispiel.txt` hat, übergibt man einfach die Option `-sub beispiel.txt` (natürlich optional mit Pfadname).

5.2.2 Angleichung der Untertitel-Anzeigezeit und der Platzierung

`-subdelay <sec>`

Verzögert die Untertitel um `<sec>` Sekunden. Der Wert kann negativ sein.

`-subfps <rate>`

Gibt die Bildrate in Bildern pro Sekunde für die Datei an (Fließkommazahl).

`-subpos <0 - 100>`

Legt die Position für die Untertitel fest.

Wenn du feststellst, dass bei Verwendung einer Micro DVD Untertiteldatei die Verzögerung im Laufe der Zeit immer größer wird, dann liegt das wahrscheinlich daran, dass die Bildraten für den Film und die Untertitel unterschiedlich sind. Es sei gesagt, dass das MicroDVD-Format absolute Bildnummern verwendet, um anzugeben, wann Text anzuzeigen ist. Deswegen funktioniert die `-subfps`-Option mit diesem Format nicht. Da MPlayer natürlich nicht in der Lage ist, die richtige Bildrate zu erraten, musst du die Bildrate manuell konvertieren. Auf dem MPlayer-FTP-Server liegt im `contrib`-Verzeichnis ein kleines Perl-Skript, das diese Konvertierung für dich vornehmen kann.

MPlayer wird versuchen, die Untertiteldatei zu erraten, wenn er einen Film abspielt. Wenn, wie das meistens der Fall ist, der Film und die Untertiteldatei den gleichen Namen haben und am gleichen Ort liegen, dann musst du nicht die Untertiteloption benutzen. Spiel einfach den Film ab, und MPlayer wird automatisch die Untertitel finden.

Für DVD Untertitel lies die [DVD-Sektion](#).

5.3 MPlayers eigenes Untertitelformat (MPsub)

MPlayer hat ein neues Untertitelformat eingeführt, das **MPsub** genannt wird. Es wurde von *Gabucino* entwickelt. Das Haupt-Feature ist, dass es **dynamische** Zeitabhängigkeiten benutzt (obwohl es auch einen Frame-basierende Modus besitzt). Ein Beispiel (aus [mpsub.sub](#)):

Beispiel aus mpsub.sub

```
# erste Nummer : warte so lange nach dem Verschwinden des vorherigen Untertitels
# zweite Nummer : zeige den momentanen Untertitel für so viele Sekunden

15 3
A long time ago...

0 3
in a galaxy far, far away...

0 3
Naboo was under attack.
```

Wie man erkennen kann, ist das Hauptziel, das **Bearbeiten/Timing/ Zusammenfügen/Schneiden von Untertiteln einfach zu machen**. Und wenn du mal angenommen einen SSA-Untertitel bekommst und er zeitlich schlecht zu deinem Video abgestimmt ist, kannst du einfach `mplayer dummy.avi -sub quelle.ssa -dumpmpsub` ausführen. Es wird eine `dump.mpsub`-Datei im aktuellen Verzeichnis erzeugt, die den Quell-Untertitel-Text enthält, jedoch im **MPSub**-Format. Dann kannst du dem Untertitel einfach Sekunden hinzufügen usw.

Untertitel werden mit einer Technik - genannt **OSD, On Screen Display** - angezeigt. OSD wird verwendet, um die aktuelle Zeit, die Lautstärken- und Such-Leisten etc. anzuzeigen.

5.4 Installation des OSD und der Untertitel

Du brauchst ein MPlayer-Schriftenpaket, um das OSD oder Untertitel anzeigen zu können. Es gibt mehrere Möglichkeiten, eines zu bekommen:

- * Lade die fertigen Schriftpakete von der MPlayer-Seite. Hinweis: Momentan sind nur Schriftarten verfügbaren, die den ISO 8859-1/2-Zeichensätzen entsprechen, aber es gibt auch ein paar von Benutzern erstellte Font-Pakete für andere Zeichensätze (einschließlich Koreanisch, Russisch, 8859-8 etc.). Diese kannst du im contrib/font-Verzeichnis auf dem FTP-Server finden.

Zu jeder Schrift muss eine passende font.desc-Datei existieren, die festlegt, wo im Zeichensatz die einzelnen Unicode-Zeichen zu finden sind. Eine andere Möglichkeit besteht darin, dass man bei in UTF8 kodierten Untertiteln die Option `-utf8` verwendet oder die Datei in `<video_name>.utf` umbenennt und im gleichen Verzeichnis liegen hat. Die Umwandlung zwischen verschiedenen Zeichensätzen und UTF8 kann mit den Programmen `konvert` (unter Debian) oder `iconv` (unter RedHat) vorgenommen werden. Einige URLs:

- * <ftp://ftp.mplayerhq.hu/MPlayer/releases/fonts/> - ISO Schriften
- * <ftp://ftp.mplayerhq.hu/MPlayer/contrib/fonts> - verschiedene Schriften von Anwendern
- * <http://realtime.ssu.ac.kr/~lethean/mplayer> - koreanische Schrift und RAW-Plugin
- * Verwende das Schrift-Erzeugungs-Tool, das in `TOOLS/subfont-c` liegt. Es ist ein komplettes Tool, um TTF/Type1/usw.-Schriften in ein MPlayer-Schriftpaket zu konvertieren (lies `TOOLS/subfont-c/README` für Details).
- * Verwende das GIMP-Plugin für Schrift-Erzeugung unter `TOOLS/subfont-GIMP` (Hinweis: Du musst auch das `HSI-RAW-Plugin` haben; siehe URL unten).
- * Mit Hilfe der freetype-Bibliothek kannst du auch TrueType-Schriften verwenden. Dafür wird mindestens Version 2.0.9 benötigt, und du musst `configure` die Option `--enable-freetype` mitgeben. Danach hast du zwei Möglichkeiten:
 - * Gib bei jedem Aufruf mit dem Parameter `-font /pfad/zu/arial.ttf` eine TrueType-Schrift an.
 - * Erstelle einen Symlink wie folgt: `ln -s /pfad/zu/arial.ttf ~/.mplayer/subfont.ttf`

Wenn du eine Nicht-TrueType-Schrift benutzt, dann entpacke die Dateien nach dem Download nach `~/.mplayer` oder `$PREFIX/share/mplayer`. Danach benennst du eine davon um oder erstellst einen symbolischen Links zu `font` (wie z.B.: `ln -s ~/.mplayer/arial-24 ~/.mplayer/font`). Nun solltest du die Zeit in oberen linken Ecke des Films sehen (kann man mit der Taste `o` ausschalten).

Das OSD hat 4 Anzeigarten, zwischen denen du mit `o` umschalten kannst:

(Untertitel werden **immer angezeigt**. Wie du sie deaktivierst, findest du in der man-Page.)

- * Lautstärken- + Suchleiste (Standard)
- * Lautstärken- + Suchleiste + und prozentuale Dateiposition beim Spulen
- * Lautstärken- + Suchleiste + die totale Länge der Datei

* nur Untertitel

Du kannst das Standard-Verhalten mit der `osdlevel=-` Variable in der Konfigurationsdatei oder der Kommandozeilenoption `-osdlevel` ändern.

5.5 Das OSD-Menü

MPlayer besitzt nun ein komplett vom Benutzer definierbares OSD-Menü.

ANMERKUNG:

das Optionsmenü ist momentan NICHT IMPLEMENTIERT!

5.5.1 Installation

1. Gib vor der Kompilierung `./configure` die Option `--enable-menu` mit.
2. überzeuge dich, dass du auch wirklich eine Schrift für das OSD installiert hast (siehe oben).
3. Kopiere die Datei `etc/menu.conf` in dein `~/mplayer`-Verzeichnis.
4. Kopiere die Datei `etc/input.conf` in dein `~/mplayer`-Verzeichnis oder in MPlayers systemweites Konfigurationsverzeichnis (standardmäßig `/usr/local/etc/mplayer`).
5. überprüfe den Inhalt der Datei `input.conf` und passe ihn nach deinen Wünschen an, um die Navigationstasten des Menüs zu aktivieren. Wie das geht, steht unten.
6. Starte MPlayer analog zu diesem Beispiel: `mplayer -menu file.avi`
7. Drücke die Tasten, die du vorher für die Navigation festgelegt hast.

5.6 RTC

Es gibt drei Methoden der Zeitmessung in MPlayer.

- * **Um die alte Methode zu verwenden**, musst du nichts tun. MPlayer verwendet `usleep()`, um Audio und Video synchron zu halten, und erreicht dabei eine Genauigkeit bis zu +/- 10ms. Manchmal muss die Synchronisation aber einfach feiner eingestellt werden können.
- * **Der neue Timercode** verwendet die RTC (Real Time Clock [Echtzeituhr]) für diese Aufgabe, da sie Timer mit genauen 1ms-Intervallen erzeugen kann. Dies wird automatisch aktiviert, falls verfügbar, erfordert jedoch Root-Privilegien, ein `setuid root` MPlayer-Binary oder einen passend eingerichteten Kernel. Wenn der Kernel 2.4.19pre8 oder später verwendet wird, kannst du die maximale RTC-Frequenz für normale Benutzer über das `/proc`-Dateisystem anpassen. Du kannst folgenden Befehl verwenden, um die RTC für alle normalen Benutzer zu aktivieren:

```
echo 1024 > /proc/sys/dev/rtc/max-user-freq
```

Wenn du einen älteren Kernel verwendest, dann kannst du auch eine Zeile in der Datei `drivers/char/rtc.c` des Kernelquelltextes verändern und den Kernel neu kompilieren. Dazu musst du folgenden Abschnitt finden:

Kernel Quelltexte `drivers/char/rtc.c`

```
We don't really want Joe User enabling more  
  
than 64Hz of interrupts on a multi-user machine.  
/  
if ((rtc_freq > 64) && (!capable(CAP_SYS_RESOURCE)))
```

Hier änderst du die 64 in 1024. Du solltest dabei aber wirklich wissen, was du tust. Du kannst die Effizienz

des neuen Timers in der Statusleiste sehen. Die Energieverwaltungs-Funktionen einiger Notebook-BIOSse mit Speedstep-CPU's haben Probleme mit der RTC. Audio und Video können dadurch aus der Synchronisation fallen. Das einstecken des externen Stromsteckers vor dem Booten soll scheinbar helfen. Du kannst die Verwendung der RTC kann jedoch jederzeit mit der Option `-nortc` deaktivieren. In einigen Hardwarekombinationen (bestätigt durch die Verwendung eines nicht-DMA-DVD-Laufwerkes auf einem ALi1541-Board) verursacht der RTC-Timer eine rucklige Wiedergabe. Du solltest in solchen Fällen die folgende Methode verwenden.

* **Der dritte Timercode** wird mit der Option `-softsleep` aktiviert. Er besitzt dieselbe Effizienz wie die RTC, braucht jedoch kein RTC. Er braucht jedoch auch mehr CPU-Leistung.

Warnung: Installiere NIEMALS MPlayer mit setuid root auf einem Mehrbenutzersystem! Das wär ein simpler Weg, um Root-Rechte zu bekommen.

6 Features

6.1 Unterstützte Formate

Es ist wichtig, einen häufigen Fehler zu berichtigen. Wenn Leute eine Datei mit **.AVI**-Erweiterung sehen denken sie sofort, dass es keine MPEG- Datei ist. Das ist nicht wahr. Zumindest nicht vollständig. Im Gegensatz zur verbreiteten Ansicht **kann** solche eine Datei MPEG1-Video enthalten.

Ein **Codec** ist nicht dasselbe wie ein **Dateiformat**.

Beispiele für **Video-Codecs** sind: MPEG1, MPEG2, DivX, Indeo5, 3ivx.

Beispiele für **Dateiformate** sind: MPG, AVI, ASF.

Theoretisch kannst du ein OpenDivX-Video und MP3-Audio in einer **.MPG**-Datei vereinigen. Die meisten Player könnten es jedoch nicht abspielen, da sie MPEG1-Video und MP2-Audio erwarten (im Gegensatz zu AVI besitzt MPG nicht die nötigen Felder, die die Video- und Audio-Streams beschreiben). Oder du könntest ein MPEG1-Video in ein **.AVI** geben.  [ffmpeg](#) und  [MEncoder](#) können solche Dateien erzeugen.

6.1.1 Videoformate

6.1.1.1 MPEG-Dateien

MPEG-Dateien kommen in unterschiedlichen Erscheinungen:

- * **MPG**: Dies ist die **grundlegende** Form des MPEG Dateiformates. Es enthält MPEG1-Video und MP2-(MPEG-1 layer 2) oder selten MP1- Audio.
- * **DAT**: Dies ist dasselbe Format wie MPG mit einer unterschiedlichen Endung. Es wird auf **Video CDs** benutzt. Auf Grund der Tatsache, wie VCDs erzeugt werden und wie Linux entworfen wurde, kann man die DAT- Dateien einer VCD weder abspielen noch kopieren. Du musst die **-vcd**-Option zum Abspielen verwenden.
- * **VOB**: VOB ist das MPEG Dateiformat bei **DVDs**. Es ist das gleiche wie MPG mit zusätzlicher Unterstützung für Untertitel oder nicht-MPEG-(AC3)-Audio. Es enthält MPEG2-Video und normalerweise AC3- Audio, aber auch DTS, MP2 und unkomprimierte LPCMs sind erlaubt. Lies die  [DVD-Sektion](#) !

Serien von Frames bilden unabhängige Gruppen in MPEG-Dateien. Das bedeutet, dass du eine MPEG-Datei mit einem Standard-Dateitool (wie **dd** oder **cut**) schneiden und zusammenfügen kannst und dieses trotzdem vollständig funktionsfähig bleibt.

Ein wichtiges Feature von MPGs ist, dass sie ein Feld besitzen, das das Breiten-/Höhenverhältnis des Video-Streams angibt. SVCDs haben zum Beispiel Video mit einer Auflösung von 480x480, und im Header ist dieses Feld auf 4:3 gesetzt. Also wird es als 640x480 abgespielt. AVI-Dateien haben dieses Feld nicht, sodass diese während des Encodierens skaliert werden müssen oder die Option **-aspect** bei der Wiedergabe verwendet werden muss.

6.1.1.2 AVI-Dateien

AVI (Audio Video Interleaved) ist ein von Microsoft entworfenes und weitverarbeitetes Multifunktionsformat, das zur Zeit hauptsächlich für DivX und DivX4 Videos verwendet wird. Es hat einige bekannte Nachteile, und ihm fehlen ein paar wichtige Features (zum Beispiel Streaming). Es unterstützt einen Videostream und 0 bis 99 Audiostreams. Die Dateien können bis zu 2 GB groß sein. Es existiert eine Erweiterung für größere Dateien,

genannt **OpenDMS**. Microsoft rät von der Verwendung stark ab und propergiert ASF/WMV. Als ob das irgendwen kratzen würde.

Es gibt einen Hack, der es AVI-Dateien erlaubt, Ogg Vorbis Audio-Streams zu enthalten, sie aber auch inkompatibel zu Standard-AVIs macht. MPlayer unterstützt die Wiedergabe solcher Dateien. Eine Spulfunktion ist auch implementiert, jedoch wird diese stark von schlecht konvertierten Dateien mit verwirrenden Headers gestört. Leider schreibt der einzige Encodierer, der fähig ist, solche Dateien zu erzeugen (NanDub), solch problematischen Header.

Hinweis:

DV-Kameras können Raw-DV-Streams erzeugen, welche von DV-Aufnahmetools in zwei unterschiedliche Typen von AVI-Dateien konvertiert werden. Die AVI-Datei enthält dabei entweder separate Audio- und Video-Streams, die MPlayer wiedergeben kann, oder den unveränderten DV-Stream. An dessen Unterstützung wird gerade gearbeitet.

Es gibt 2 Arten von AVI-Dateien:

- * **Interleaved:** Audio- und Video-Inhalt werden paketweise ineinandergeschachtelt. Die ist üblich, wird empfohlen und auch meist verwendet. Einige Tools erzeugen interleaved AVIs mit schlechtem Sync. MPlayer erkennt diese als interleaved, was zum Verlust von A/V-Sync führen kann (evtl. erst beim Vor-/Zurückspueln). Diese Dateien sollten non-interleaved (mit der `-ni`-Option) abgespielt werden.
- * **Non-interleaved:** Als erstes kommt der ganze Videostream, dann dann der ganze Audiostream. Dies erfordert viel Suchen in der Datei, was die Wiedergabe vom Netzwerk oder von CDROM schwierig macht.

MPlayer unterstützt zwei Arten des timings für AVI-Dateien:

- * **bps-basiert:** Diese Methode basiert auf der Bitrate/Samplerate des Video-/Audio-Streams. Diese Methode wird von den meisten Playern verwendet, inklusive  [avifile](#) und Windows Media Player. Dateien mit kaputten Headern und Datein, welche mit VBR-Audio aber nicht VBR-unterstützendem Encoder erstellt wurden, führen zu A/V-Desync mit dieser Methode (meistens beim Spulen).
- * **interleaving-basiert:** Diese Methode benutzt nicht mehr den Wert der Bitrate sondern die relative Position des Audio-Paketes zum aktuellen Video-Paket, was einige falsch codierte Dateien mit VBR-Audio abspielbar macht.

Jeglicher Audio- und Video-Codec ist erlaubt. Aber bedenke, dass VBR-Audio von den meisten Playern nicht gut unstützt wird. Das Dateiformat macht es möglich, VBR-Audio zu verwenden, aber die meisten Player erwarten CBR- Audio, weswegen VBR-Audio nicht funktioniert. VBR ist ungewöhnlich, und Microsofts Spezifikation zu AVI beschreiben nur CBR-Audio. Bedenke auch, dass die meisten AVI-Encoder/Multiplexer schlechte Dateien mit VBR-Audio erzeugen. Es gibt nur zwei Ausnahmen: NanDub und  [MEncoder](#).

6.1.1.3 ASF/WMV-Dateien

ASF (Active Streaming Format) kommt von Microsoft. Sie habe zwei Arten von ASF entwickelt, v1.0 und v2.0. v1.0 wird von ihren Media Tools (Windows Media Player und Windows Media Encoder) verwendet und ist sehr geheim. v2.0 ist veröffentlicht und patentiert :). Natürlich sind sie unterschiedlich und besitzen überhaupt keine Gemeinsamkeiten (es ist nur ein weiteres Spiel mit den Gesetzen). MPlayer unterstützt nur v1.0, weil noch niemand v2.0 Dateien gesehen hat :). Beachte, dass [.ASF](#)-Dateien heute auch mit der Extension [.WMA](#) oder [.WMV](#) vorkommen.

6.1.1.4 QuickTime/MOV-Dateien

Diese Formate sind von Apple designt und können jeden beliebigen Codec enthalten, CBR oder VBR. Normalerweise haben sie die Endung .QT oder .MOV. Seitdem die MPEG4-Gruppe QuickTime als das empfohlene Dateiformat für MPEG4 ausgewählt hat, kommen ihre .MOV-Dateien mit den Endungen .MPG oder .MP4. (Interessanterweise sind bei diesen Dateien die Video- und Audiostreams richtige MPG- und AAC-Dateien. Mit den den Optionen `-dumpvideo` und `-dumpaudio` kannst du sie sogar extrahieren.).

Hinweis:

Die meisten QuickTime-Dateien benutzen **Sorenson**- Video und QDesign-Audio. Lies dazu die [▶ Sorenson-Codec-Sektion](#).

6.1.1.5 VIVO-Dateien

MPlayer demuxt nun auch fröhlich VIVO-Dateiformate. Der grösste Nachteil des Formates ist, dass es keinen Index-Block besitzt, auch keine fixe Paketgröße oder Sync-Bytes. Die meisten Dateien besitzen nicht mal Keyframes, also vergiss das Spulen!

Der Videocodec von VIVO/1.0-Datei ist Standard-**h.263**. Der Videocodec der VIVO/2.0 Dateien ist veränderter, nichtstandard-**h.263**. Bei Audio ist es dasselbe, es ist **g.723** (Standard) oder **Vivo Siren**.

In den [▶ VIVO-Videocodec](#)- und [▶ VIVO-Audiocodec](#) -Sektionen findest du Installationsanweisungen.

6.1.1.6 FLI-Dateien

FLI ist ein sehr altes vom Autodesk Animator benutztes Dateiformat. Es ist aber immer noch ein im Internet gebräuchliches Dateiformat für kurze Animationen. MPlayer demuxt und decodiert FLI-Filme und ist sogar fähig, in ihnen vor- und zurückzuspulen (nützlich, wenn man looping mit der `-loop`-Option aktiviert hat). FLI-Dateien haben keine Keyframes, was zu verunstalteten Bildern kurz nach dem Spulen führt.

6.1.1.7 RealMedia-(RM)-Dateien

Ja, MPlayer kann RealMedia (.rm) Dateien lesen (demuxen). Spulen (seeking) funktioniert, jedoch muss die `-forceidx`-Option verwendet werden (das Format unterstützt Keyframes). Hier ist die Liste der unterstützten [▶ RealVideo](#)- und [▶ RealAudio](#)-Codecs.

6.1.1.8 NuppelVideo-Dateien

NuppelVideo ist, soweit ich weiss, ein TV-Aufnahmeprogramm. MPlayer kann dessen .NUV- Dateien (nur NuppelVideo 5.0) lesen. Diese Dateien können unkomprimierte YV12-, komprimierte YV12+RTJpeg-, komprimierte YV12+RTJpeg+lzo- und komprimierte YV12+lzo-Frames enthalten, welche MPlayer alle dekodiert (und diese auch mit MEncoder zu DivX/etc! encodiert). Spulen (seeking) funktioniert.

6.1.1.9 yuv4mpeg-Dateien

[🇬🇧 yuv4mpeg / yuv4mpeg2](#) ist ein Dateiformat, das von den [🇬🇧 mjpegtools- Programmen](#) verwendet wird. Du kannst damit Video in diesem Format einlesen, produzieren, filtern oder **encodieren**. Das Dateiformat ist eigentlich eine Sequenz von unkomprimierten YUV 4:2:0 Bildern.

6.1.1.10 FILM-Dateien

Dieses Format wird auf alten Sega Saturn CD-Rom-Spielen verwendet.

6.1.1.11 RoQ-Dateien

RoQ-Dateien sind Multimedia-Dateien, welche in gewissen ID-Spielen wie Quake III und Return to castle Wolfenstein benutzt werden.

6.1.1.12 OGG/OGM-Dateien

Dies ist ein neues Dateiformat von  [Xiphophorus](#). Es kann beliebige Video- und Audio-Codecs enthalten, sowohl CBR als auch VBR. Man benötigt dazu installierte `libogg`- und `libvorbis`-Bibliotheken vor der Kompilierung von MPlayer, um sie wiedergeben zu können.

6.1.1.13 SDP-Dateien

 [SDP](#) ist ein IETF-Standardformat, das Video- und/oder Audio-RTP-Streams beschreibt. (Dazu werden die  [LIVE.COM Streaming Media](#) -Bibliotheken benötigt.)

6.1.1.14 PVA-Dateien

PVA ist MPEG-ähnliches Format, das von verschiedenen Windowsprogrammen benutzt wird, die bei DVB-TV-Karten mitgeliefert werden (z.B. MultiDec, WinTV).

Die PVA-Spezifikationen können an der folgenden Adresse heruntergeladen werden:  http://www.technotr end.de/download/av_format_v1.pdf

6.1.1.15 GIF-Dateien

Das **GIF**-Format ist ein weit verbreitetes Format für Grafiken im Web. Es gibt zwei Versionen der GIF-Spezifikationen, GIF87a und GIF89a. Der größte Unterschied liegt darin, dass GIF89a Animationen unterstützt. MPlayer unterstützt beide Formate mit Hilfe der `libungif`-Bibliothek oder einer anderen `libgif`-kompatiblen Bibliothek. Nicht animierte GIFs werden als Ein-Bild-Videos dargestellt. (Mit den Optionen `-loop` und `-fixed-vo` können solche GIFs länger angezeigt werden.)

Momentan unterstützt MPlayer nicht das Spulen in GIF-Dateien. Die einzelnen Bilder in GIF-Dateien haben nicht zwangsläufig die gleichen Dimensionen, und auch nicht eine feste Bildrate. Jedes Bild hat vielmehr seine eigenen Dimensionen und soll an einer bestimmten Position auf einem Bereich angezeigt werden, der selber aber eine feste Größe hat. Die Bildrate wird von einem optionalen Block vor jedem Bild kontrolliert, der die Anzeigedauer des nachfolgenden Bildes in Zentisekunden angibt.

Standard-GIF-Dateien enthalten 24 Bit RGB-Bilder mit einer indizierten Palette, die höchstens bis 8 Bit geht. Die Bilder sind normalerweise mit dem LZW-Algorithmus komprimiert. Es gibt aber auch GIF-Encoder, die unkomprimierte Bilder erzeugen, um die Patentprobleme mit dem LZW-Algorithmus zu umgehen.

Detaillierte technische Informationen findest du in den  [GIF89a-Spezifikationen](#).

6.1.2 Audio Formate

MPlayer ist ein Movie-(Film-) und kein Media-Player. Er kann auch einige Audioformate wiedergeben (diese sind in der unteren Sektion beschrieben). Dies wird jedoch nicht empfohlen, und du solltest besser  [xmms](#)

verwenden.

6.1.2.1 MP3-Dateien

Du hast vielleicht Probleme mit der Wiedergabe einigen MP3-Dateien, die MPlayer falsch als MPEGs erkennt und dementsprechend falsch oder gar nicht wiedergibt. Dies kann nicht gelöst werden, ohne die Unterstützung für einige kaputte MPEG-Dateien aufzugeben und wird deshalb bis auf weiteres so bleiben. Die Option `-demuxer`, welche in der Manpage beschrieben wird, könnte dir in diesen Fällen helfen.

6.1.2.2 WAV-Dateien

6.1.2.3 OGG/OGM-Dateien (Vorbis)

Benötigt richtig installierte `libogg` und `libvorbis`.

6.1.2.4 WMA/ASF-Dateien

6.1.2.5 MP4-Dateien

6.1.2.6 CD-Audio

MPlayer kann die Bibliotheken von `cdparanoia` benutzen, um CDDA-Dateien wiederzugeben. Alle Kommandozeilenparameter von `cdparanoia` aufzuzählen gehört nicht zu den Aufgaben dieser Dokumentation. Schau in der Manpage bei der Option `-cdda` nach, wie mit ihr Parameter an `cdparanoia` übergeben werden können.

6.1.2.7 XMMS

MPlayer kann XMMS-Inputplugins benutzen, um eine Vielzahl von Dateiformaten abzuspielen. Es gibt Plugins für SNES-Spielemusik, SID-Musik (vom guten alten Commodore 64), viele Amiga-Formate, `.xm`, `.it`, VQF, musepack, Bonk shorten und viele weitere. Du findest sie auf der Seite für [XMMS-Inputplugins](#).

Um dieses Feature benutzen zu können, brauchst du XMMS und musst MPlayer mit `./configure --enable-xmms` kompilieren. Funktioniert das nicht, dann musst du eventuell die Pfade für die XMMS-Plugins und die XMMS-Bibliotheken explizit mit `--with-xmmsplugindir` und `--withxmmslibdir` angeben.

6.2 Unterstützte Codecs

6.2.1 Video-Codecs

Siehe [Codec-Status](#) -Seite für die komplette, täglich generierte Liste. Eine Vielzahl von Codecs kann von unserer Homepage heruntergeladen werden. Du findest sie auf der [Codec-Seite](#).

Die wichtigsten Video-Codecs:

- * MPEG1- (VCD) und MPEG2-Video (DVD)
- * eingebauter Dekoder für DivX ;-), OpenDivX, DivX4, DivX5, M\$ MPEG4 v1, v2 und andere MPEG4-Varianten

- * eingebauter Decoder für **Windows Media Video 7/8 (WMV1/WMV2)** und Win32-DLL-Decoder für **Windows Media Video 9 (WMV3)**. Beide werden in `.wmv`-Dateien verwendet.
- * eingebauter **Sorenson 1 (SVQ1)** Decoder
- * Win32/QT **Sorenson 3 (SVQ3)** Decoder
- * **3ivx**-Decoder für v1 und v2
- * Cinepak und **Intel Indeo** Codecs (3.1, 3.2, 4.1, 5.0)
- * **MJPEG**, AVID, VCR2, ASV2 und andere Hardwareformate
- * VIVO 1.0, 2.0, I263 und andere **h263(+)**-Varianten
- * FLI/FLC
- * **RealVideo 1.0** von libavcodec und **RealVideo 2.0, 3.0** und **4.0** durch die RealPlayer-Libraries
- * eingebauter Decoder für HuffYUV
- * Verschiedene alte, einfache RLE-ähnliche Formate

Wenn du einen Win32-Codec hast, der noch nicht aufgelistet und unterstützt ist, dann lies  [Win32 Codec-Importierungs-HOWTO](#) und hilf uns, die Unterstützung hinzuzufügen.

6.2.1.1 DivX4/DivX5

Diese Sektion enthält Informationen über den DivX4- und DivX5-Codec vom  [Project Mayo](#). Ihre ersten verfügbaren alpha-Versionen waren OpenDivX 4.0 alpha 47 und 48. Unterstützung dafür war früher standardgemäß in MPlayer eingebaut. Es wurde auch deren Postprocessing Code verwendet, um optional die Bildqualität von MPEG1/2-Filmen zu verbessern. Nun benutzen wir unseren eigenen für alle Dateitypen.

Die neue Version dieses Codecs heisst DivX4Linux und kann sogar Filme decodieren, die mit dem berühmten DivX-Codec (DivX 3 ;-)) erzeugt wurden! Dazu ist sie noch um einiges schneller als die traditionelle Win32-DivX-DLL aber immer noch langsamer als libavcodec. Vom Gebrauch dieses Codecs wird deshalb **abgeraten**. Er lässt sich jedoch trotzdem zum Encodieren verwenden. Ein Nachteil diese Codecs ist, dass er zurzeit nicht unter einer OpenSource-Lizenz verfügbar ist.

MPlayer erkennt automatisch, ob DivX4/DivX5 korrekt installiert ist, du kannst ihn wie gewöhnlich kompilieren. Wenn es nicht erkannt wird, wurde es nicht korrekt installiert oder konfiguriert.

DivX4Linux arbeitet in zwei verschiedenen Modi:

`-vc odivx`

Spricht den Codec so wie den alten OpenDivX-Codec an. In diesem Fall produziert der Codec YV12-Bilder in seinem eigenen Puffer, und MPlayer nimmt die Farbraumkonvertierung mittels `libvo` vor (**schneller und empfohlener Modus!**).

`-vc divx4`

Benutzt die Farbraumkonvertierung des Codecs. In diesem Modus kannst du auch YUY2/UYYV benutzen (**LANGSAM**).

Der `-vc odivx`-Modus ist normalerweise schneller da er die Daten in einem YV12-Format (planares YUV 4:2:0) übermittelt, was weniger Bandbreite für den Bus bedeutet. Für gepackte YUV-Formate (YUY2, UYYV) verwende den `-vc divx4`-Modus. Für den RGB-Modus ist die Geschwindigkeit gleich und hängt am ehesten von deiner momentanen Farbtiefe ab.

Hinweis:

Wenn der `-vo`-Treiber **direct rendering** unterstützt, ist `-vc divx4` wahrscheinlich die schnellere oder sogar

die schnellste Lösung.

Die binäre DivX4/5-Bibliothek kann von  [avifile](#) oder  [divx.com](#) heruntergeladen werden. heruntergeladen werden. Entpacke ihn, starte `./install.sh` als root, und vergiss nicht, `/usr/local/lib` deiner `/etc/ld.so.conf` hinzuzufügen und `ldconfig` aufzurufen.

Besorge dir die CVS-Version der ALTEN OpenDivX-Kernbibliothek wie folgt:

```
1.cvs -d:pserver:anonymous@cvs.projectmayo.com:/cvsroot login
2.cvs -d:pserver:anonymous@cvs.projectmayo.com:/cvsroot co ivxcore
3.Diese Kernbibliothek inst in eine Dekodierbibliothek und eine Enkodierbibliothek aufgeteilt, sie separat
  kompiliert werden müssen. Bei der Dekodierbibliothek gibst du einfach folgendes ein:
  cd divxcore/decore/build/linux
  make
  cp libdivxdecore.so /usr/local/lib
  ln -s /usr/local/lib/libdivxdecore.so /usr/local/lib/libdivxdecore.so.0
  cp ../../src/decore.h /usr/local/include
4.Für die Enkodierbibliothek gibt es leider kein Linux-Makefile, und der MMX-optimierte Code funktioniert
  auch nur unter Windows. Du kannst die Bibliothek aber trotzdem kompilieren, wenn du dieses Makefile
  benutzt:
  cd ../../../../encore/build
  mkdir linux
  cd linux
  cp path/Makefile .
  make
  cp libdivxencore.so /usr/local/lib
  ln -s /usr/local/lib/libdivxdecore.so /usr/local/lib/libdivxdecore.so.0
  cp ../../src/encore.h /usr/local/include
```

MPlayer findet DivX4/DivX5 automatisch, sofern es richtig installiert wurde. Kompiliere also einfach wie sonst auch. Wenn `configure` die Bibliothek nicht findet, dann hast du DivX4/5 einfach nicht richtig installiert.

6.2.1.2 FFmpeg's DivX/libavcodec

Das  [FFmpeg](#)-Projekt enthält ein Open-Source-Codec-Paket, das fähig ist, mit H263/MJPEG/RV10/DivX3/DivX4/DivX5/MP41/MP42/WMV1/WMV2/HuffYUV codiertes Video und mit WMA (Windows Media Audio) codiertes Audio zu decodieren. Man kann nicht nur einige dieser Codecs zum Encodieren benutzen, sondern die Geschwindigkeit ist sogar um einiges höher als die der Win32-Codecs oder der ProjectMayo DivX4/5-Bibliothek!

FFmpeg beinhaltet auch viele nette Codecs. Die wichtigsten sind die MPEG4-Varianten: DivX3, DivX4, DivX5, Windows Media Video 7 (WMV1). Ein weiterer sehr interessanter Decoder ist der für WMA.

Wenn ein offizielles MPlayer-Release verwendet wird, befindet sich libavcodec (die Codec-Bibliothek des FFmpeg-Projektes) bereits im Paket, welches normal kompiliert werden kann. Wenn du die MPlayer-Sourcen per CVS runterlädst, musst du die libavcodec vom FFmpeg CVS-Server verwenden, da das offizielle FFmpeg-Release 0.4.5 **nicht** mit MPlayer zusammen läuft. Um dies zu erreichen gibst du folgendes ein:

```
1.cvs -d:pserver:anonymous@cvs.ffmpeg.sourceforge.net:/cvsroot/ffmpeg
  login
2.cvs -d:pserver:anonymous@cvs.ffmpeg.sourceforge.net:/cvsroot/ffmpeg co
  ffmpeg
3.Das libavcodec-Verzeichnis von den FFmpeg-Sourcen muss in's Hauptverzeichnis des MPlayer
```

CVS-Baums verschoben/kopiert werden. Es sollte etwa so aussehen:

```
main/libavcodec
```

Das Erstellen eines symbolischen Links ist nicht genug; es muss kopiert/verschoben werden!

4. Es folgt die Kompilierung. `configure` sollte Probleme vor der Kompilierung erkennen.

Hinweis:

Die MPlayer-Sourcen im CVS enthalten ein `libavcodec`-Unterverzeichnis. Es enthält jedoch nicht den Quellcode von `libavcodec`! Du musst - wie oben beschrieben - dir die CVS-Version von FFmpeg besorgen.

Mit FFmpeg und meiner Matrox G400 kann ich sogar DivX-Filme in hoher Auflösung auf meinem K6/2 500 ohne Framedrops betrachten.

6.2.1.3 XAnim-Codecs

Vorwort:

Es ist darauf hinzuweisen, dass die XAnim-Binary-Codecs mit einem Textstück geliefert werden, das eine legal bindende Softwarelizenz sein soll, welche (neben einigen anderen Einschränkungen) dem Benutzer das Verwenden der Codecs mit anderen Programmen als XAnim verbietet. Der XAnim-Autor hat jedoch noch keine rechtliche Schritte gegen irgendjemanden wegen Codec bezogenen Fragen unternommen.

Installation und Bedienung:

MPlayer kann XAnims Codecs für das Decodieren verwenden. Um diese zu aktivieren, solltest folgenden Anweisungen folgen:

1. Lade dir die Codecs, die du verwenden willst, von der [XAnim-Seite](#) herunter. Der 3ivx Codec befindet sich nicht dort. Er kann jedoch von der [3ivx-Seite](#) heruntergeladen werden.
2. **ODER** lade dir das Codec-Paket von unserer [Codecsseite](#) herunter.
3. Verwende die `--with-xanimlibdir`-Option, um `configure` zu sagen, wo es die XAnim-Codecs finden kann. Normalerweise werden diese in `/usr/local/lib/xanim/mods`, `/usr/lib/xanim/mods/` und `/usr/lib/xanim` gesucht. Weiterhin kannst du auch die Umgebungsvariable `XANIM_MOD_DIR` auf den Verzeichnisnamen der XAnim-Codecs setzen.
4. Entferne/Symlinke die Dateinamen, wobei du das Architektur-Zeugs löschst, sodass die Dateinamen etwa so aussehen: `vid_cvid.xa`, `vid_h263.xa`, `vid_iv50.xa`.

XAnim ist die Videocodec-Familie `xanim`. Du kannst also die `-vfm xanim`-Option verwenden, um MPlayer zu sagen, dass er sie, wenn möglich, verwenden soll.

Die getesteten Codecs beinhalten: Indeo 3.2, 4.1, 5.0, CVID, 3ivX, h263.

6.2.1.4 VIVO-Video

MPlayer kann VIVO (1.0 und 2.0) Videos abspielen. Der beste geeignete Codec für 1.0-Dateien ist FFmpegs H263-Decoder. Du kannst ihn mit der `-vc ffh263`-Option aufrufen. Für v2.0-Dateien musst du die Win32-DLLs mit der Option `-vc vivo` benutzen. Wenn du keine expliziten Angaben zum Codec auf der Kommandozeile machst, so wählt MPlayer automatisch den besten aus.

6.2.1.5 MPEG 1/2-Video

MPEG1 und MPEG2 werden von der eingebauten Multiplatform-Bibliothek `libmpeg2` decodiert, deren Source in MPlayer enthalten ist. Wir spielen fehlerhafte MPEG1/2-Videos durch das Abfangen des sig11 (segmentation

fault) und die anschließend schnelle Neuinitialisierung des Codecs ab. Dadurch kann exakt an der Stelle des Fehlers fortgefahren werden. Es ist dabei kein Geschwindigkeitsverlust messbar.

6.2.1.6 MS Video 1

Dies ist ein sehr alter und sehr schlechter Codec von Microsoft. In der Vergangenheit wurde es mit dem `msvidc32.dll` Win32-Codec decodiert, aber nun haben wir einen Open-Source-Codec (von  [Mike Melanson](#)) eingebaut.

6.2.1.7 Cinepak CVID

MPlayer verwendet normalerweise seinen eigenen, Open-Source multiplattformfähigen Cinepak-Decoder, der von  [Dr. Tim Ferguson](#) geschrieben wurde. Er unterstützt YUV-Ausgabe, was Hardwarescaling ermöglicht, wenn dies vom Videoausgabetreiber unterstützt wird.

6.2.1.8 RealVideo

MPlayer unterstützt die Dekodierung aller Versionen von RealVideo:

- * RealVideo 1.0 (fourcc RV10) - En/Dekodierung durch `libavcodec` unterstützt
- * RealVideo 2.0, 3.0, 4.0 (fourcc RV20, RV30, RV40) - Dekodierung durch die **RealPlayer Libraries** unterstützt

Es wird empfohlen, RealPlayer8 oder RealONE herunterzuladen und zu installieren, da MPlayer deren Bibliotheken verwenden kann, um RealVideo 2.0, 3.0 oder 4.0 zu decodieren. Das `configure`-Script von MPlayer sollte die RealPlayer-Bibliotheken an der normalen Stelle der Vollinstallation finden. Falls nicht, kannst du `configure` über die Option `--with-reallibdir` mitteilen, wo sie zu suchen/finden sind.

Hinweis:

Momentan funktionieren die RealPlayer-Bibliotheken nur mit Linux, FreeBSD, NetBSD und Cygwin auf der x86 oder der Alpha-Plattform.

6.2.1.9 XviD

 [XviD](#) ist eine abgespaltene Entwicklung des OpenDivX-Codecs. Es geschah, als ProjectMayo OpenDivX in das ClosedSource-Projekt DivX4 (jetzt DivX5) änderte, woraufhin die Leute, die an OpenDivX aber nicht beim ProjectMayo arbeiteten, sauer wurden und XviD starteten. Beide Projekte haben also den selben Ursprung.

Vorteile:

- * Open Source
- * Das API ist kompatibel mit DivX4, sodass es leicht ist, Unterstützung dafür einzubauen.
- * 2-Pass Encodierungs-Unterstützung
- * gute Encodierungsqualität, höhere Geschwindigkeit als DivX4 (man kann es während des Kompilierens für den jeweiligen Rechner optimieren)

Nachteile:

- * Es **decodiert** momentan nicht alle DivX/DivX4-Dateien sauber (kein Problem, da man sie ja immer noch mit der `libavcodec` abspielen kann).
- * Man muss sich beim Kompilieren für die DivX4- **ODER** XviD-Unterstützung entscheiden.

* in der Entwicklung

Installation der XVID-CVS-Version:

XviD ist momentan nur als CVS erhältlich. Hier sind die Download- und Installations-Anweisungen:

1. `cvcs -z3 -d:pserver:anonymous@cvs.xvid.org:/xvid login`
2. `cvcs -z3 -d:pserver:anonymous@cvs.xvid.org:/xvid co xvidcore`
3. `cd xvidcore/build/generic`
4. ändere `Makefile.linux` deinen Bedürfnissen entsprechend
5. `make -f Makefile.linux`
6. Nimm die `encore2.h` und `decore.h` vom Divx4linux-Paket und kopiere diese nach `/usr/local/include/`.
7. kompiliere MPlayer neu mit `--with-xvidcore=/pfad/zu/libcore.a`.

6.2.1.10 Sorenson

Sorenson ist eine von Sorenson Media entwickelte und von Apple lizenzierte Videocodecfamilie. Wir sind momentan in der Lage, alle Videos zu decodieren, die mit einer der folgenden Sorenson-Versionen erstellt wurden:

- * Sorenson 1 (fourcc /SVQ1/) - Decodierung durch **selbstentwickelte Codecs**. Tatsächlich gibt es zwei (qualitativ nahezu identische) Decoder für SVQ1: Einer ist der in MPlayer enthaltene OpenSource-Codec, der andere ist der in `libavcodec` vorhandene Decoder. Du kannst sie mit `-vc svq1` respektive `-vc ffsvq1` benutzen. Manche Dateien funktionieren eventuell nur mit einem der beiden Decoder. Probier als bei problematischen Dateien beide Decoder aus. Der Decoder wurde von den  `xine`-Autoren geschrieben.
- * Sorenson 3 (fourcc /SVQ3/) - Decodierung mit den **Win32-QuickTime-Bibliotheken**

6.2.1.11 Kompilierung von MPLAYER mit Unterstützung für Quicktime-Bibliotheken

Anmerkung:

Momentan wird nur die 32bit-Intel-Architektur unterstützt.

- * Lade die CVS-Version von MPlayer herunter.
- * Kompiliere MPlayer mit: `./configure --enable-qtx-codecs`
- * Lade dir das QuickTime-DLL-Paket von  <http://www.mplayerhq.hu/MPlayer/releases/codecs> herunter.
- * Entpacke die QuickTime-DLLs in dein Win32-Codec-Verzeichnis, das standardmäßig `/usr/lib/win32` ist.

6.2.2 Audio-Codecs

Die wichtigsten Audiocodecs als erstes:

- * MPEG layer 2 und layer 3 (MP3) Audio (**eingebauter** Code, mit MMX/SSE/3DNow!-Optimierungen)
- * MPEG layer 1 Audio (**eingebauter** Code, mit `libavcodec`)
- * Windows Media Audio 9 (WMAv3) (mit der DMO-DLL)
- * AC3 Dolby Audio (**eingebauter** Code, mit MMX/SSE/3DNow!-Optimierungen)
- * Ogg Vorbis Audiocodec (**eingebaute** Bibliothek)
- * RealAudio: DNET (AC3 mit niedriger Bitrate), Cook, Sipro, Atrac
- * QuickTime: Qualcomm und QDesign Audiocodecs
- * VIVO Audio (G723, Vivo Siren)

- * Voxware Audio (Verwendung der DirectShow-DLL)
- * alaw, MSGSM, PCM und andere einfache, alte Audioformate

6.2.2.1 AC3-Decodierung in Software

Dieser ist der standardmäßige Decoder für Dateien mit AC3-Audio.

Der AC3 Dekoder kann die Audio-Ausgabe für 2, 4 oder 6 Lautsprecher erzeugen. Wenn er für 6 Lautsprecher konfiguriert ist, unterstützt dieser Decoder die Ausgabe aller AC3-Kanäle an den Soundtreiber. Damit ermöglicht er die vollständige "Surround Sound"-Erfahrung ohne einen externen AC3-Decoder, der bei der Verwendung des hwac3-Codecs nötig ist.

Verwende die `-channels`-Option um die Anzahl der Ausgabekanäle zu wählen. Verwende `-channels 2` für Downmix auf Stereo. Für Downmix auf 4 Kanäle (links vorne, rechts vorne, links Surround und rechts Surround) verwende `-channels 4`. In diesem Fall werden alle Center-Kanäle gleichmäßig den Frontkanälen beigemischt. `-channels 6` gibt alle AC3-Kanäle wie Aufgezeichnet in der Reihenfolge links, rechts, links Surround, rechts Surround, Center und LFE aus.

Die Standardanzahl der Ausgabekanäle ist 2.

Um mehr als 2 Kanäle zu verwenden, musst du OSS verwenden und eine Soundkarte besitzen, die die Anzahl der Ausgabekanäle via `SNDCTL_DSP_CHANNELS` ioctl unterstützt. Ein Beispiel eines passenden Treibers ist `emu10k1` (verwendet bei SB Live! Karten) vom August 2001 oder neuer (ALSA CVS sollte auch funktionieren).

6.2.2.2 AC3 Decodierung in Hardware

Du brauchst eine AC3-fähige Soundkarte mit einem Digitalausgang (SP/DIF). Der Treiber muss das `AFMT_AC3` Format richtig unterstützen (wie z.B. der für die C-Media-Karten). Verbinde den AC3-Decoder mit dem SP/DIF-Ausgang, und benutze die Option `-ac hwac3`. Dieser Code ist noch experimentell, läuft aber bekanntermaßen mit C-Media-Karten, SoundBlaster Live!-Karten mit dem ALSA-Treiber (aber nicht mit dem OSS-Treiber) und mit den DXR3/Hollywood+-MPEG-Decoderkarten.

6.2.2.3 libmad-Unterstützung

 `libmad` ist eine Multiplattform-Bibliothek für die Decodierung von MPEG-Audio. Sie verarbeitet kaputte Dateien nicht so gut und hat teilweise Probleme mit dem Suchen. Um die Unterstützung zu aktivieren, verwende `configure` mit der Option `--enable-mad`.

6.2.2.4 VIVO-Audio

Der verwendete Audiocodec in VIVO-Dateien hängt davon ab, ob es eine VIVO/1.0- oder VIVO/2.0-Datei ist. VIVO/1.0-Dateien verwenden `g.723`-Audio und VIVO/2.0-Dateien **Vivo-Siren**-Audio. Beide werden unterstützt.

6.2.2.5 RealAudio

MPlayer unterstützt die Dekodierung fast aller RealAudio-Versionen:

- * RealAudio DNET - Decodierung durch `liba52`
- * RealAudio Cook/Sipro/Atrac - Decodierung durch die **RealPlayer-Bibliotheken**

Wie die RealPlayer Libraries installiert werden können, wird in der  [RealVideo-Sektion](#) behandelt.

6.2.2.6 QDesign-Codecs

QDesign-Audio (fourcc: **QDMC**, **QDM2**) findest du in MOV/QT-Dateien. Beide Versionen dieses Codecs können mit den QuickTime-Bibliotheken decodiert werden. Schau in der [Sorenson-Video-Codec-Sektion](#) für Installationsanweisungen nach.

6.2.2.7 Qualcomm-Codec

Qualcomm-Audiostreams (FourCC: **Qclp**) finden sich in MOV/QT-Dateien. Sie können mit den Quicktime-Bibliotheken decodiert werden. Installationsanweisungen findest du in der Sektion über den [Sorenson Videocodec](#).

6.2.3 Win32 Codec-Importierungs-HOWTO

6.2.3.1 Vfw-Codecs

VfW (Video for Windows) ist die alte Video-Schnittstelle für Windows. Diese Codecs haben die Erweiterung .DLL oder (eher selten) .DRV. Wenn MPlayer mit folgender Meldung fehlschlägt, ein AVI abzuspielen:

```
UNKNOWN video codec: HFYU (0x55594648)
```

...bedeutet es, dass das AVI mit einem Codec erstellt wurde, der den fourcc HFYU besitzt (HFYU = HuffYUV-Codec, DIV3 = DivX Low Motion, usw...). Jetzt musst du nur noch herausfinden, welche DLL Windows lädt, um diese Datei abzuspielen. In diesem Fall enthält die `System.ini` diese Information in solch einer Zeile:

```
VIDC.HFYU=huffyuv.dll
```

Also braucht man die Datei `huffyuv.dll`.

Hinweis:

die Audio-Codecs werden durch das MSACM-Prefix angegeben:

```
msacm.l3acm=L3codeca.acm
```

Dies ist ein MP3 Codec. Da du nun alle nötigen Informationen hast (fourcc, Codec-Datei, AVI-Beispiel), solltest du per Mail um Unterstützung für diesen Codec bitten und die Dateien auf folgenden FTP-Server hochladen:

```
ftp://ftp.mplayerhq.hu/MPlayer/incoming/[codecname]/
```

Anmerkung:

Auf Windows NT/2000/XP-Systemem solltest du in der Registry nach diesen Informationen suchen, also z.B. nach "VIDC.HFYU". Um herauszufinden, wie das geht, lies die Sektion "alte DirectShow-Methode".

6.2.3.2 DirectShow Codecs

DirectShow ist die neuere Video-Programmierschnittstelle, welche noch schlimmer als ihr Vorgänger ist. Den Namen der DLL herauszufinden ist bei DirectShow schwieriger, weil...

* die `system.ini` nicht mehr die benötigten Informationen enthält (stattdessen stehen diese in der Registry), und weil man die GUID des Codecs braucht.

Neue Methode: mittels Microsoft GraphEdit (schnell)

1. Besorg dir GraphEdit entweder aus dem DirectX-SDK oder von  [Doom9](#).
2. Starte `graphedit.exe`.
3. Wähle im Menü "Graph -> Insert Filters".
4. Klappe `DirectShow Filters` auf.
5. Suche und wähle den richtigen Codec, dann klappe auch diesen auf.
6. Der Eintrag `DisplayName` enthält die gesuchte GUID. Diese steht nach dem Backslash (\) in geschweiften Klammern und besteht aus fünf Blöcken mit Ziffern und Buchstaben, die jeweils mit Punkten voneinander getrennt sind. Schreibe diese GUID auf.
7. Der Dateiname des Codecs steht im Eintrag `Filename`.

Anmerkung:

Wenn es keinen Eintrag namens `Filename` gibt und der `DisplayName`-Eintrag etwas in der Art `device:dmo` enthält, so handelt es sich um einen DMO-Codec.

Alte Methode: Atme tief durch und beginne mit dem Durchsuchen der Registry...

1. Starte `regedit`
2. Drücke `Strg + f`, deaktiviere die ersten beiden Auswahlfelder und aktiviere das Dritte. Nun gib den `fourcc` des Codecs ein (z.B.: `TM20`).
3. Du solltest ein Feld sehen, welches den Pfad und Dateinamen enthält (z.B.: `C:\WINDOWS\SYSTEM\TM20DEC.AX`).
4. Jetzt da man die Datei hat, braucht man die GUID. Versuche erneut zu suchen, aber nun nach dem Codec-Namen, nicht nach der `fourcc`. Der Codec-Name kann bei der Wiedergabe mit dem Windows Media Player herausgefunden werden, indem man unter Datei -> Eigenschaften -> Erweitert nachschaut. Falls dies nichts bringt, hast du Pech. Versuche zu raten (suche z.B. nach: `TrueMotion`).
5. Wenn die GUID gefunden wurde, sollte ein `FriendlyName`- und ein `CLSID`-Feld auftauchen. Schreibe die 16 Byte lange `CLSID` auf, dies ist die von uns benötigte GUID.

Hinweis:

Wenn die Suche fehlschlägt, versuche alle Auswahlfelder zu aktivieren. Du wirst möglicherweise falsche Ergebnisse finden, aber vielleicht hast du auch Glück...

Da du nun alle nötigen Informationen hast (`fourcc`, GUID, Codec-Datei, AVI-Beispiel), solltest du per Mail um Unterstützung für diesen Codec bitten und die Dateien auf folgenden FTP-Server hochladen:
[ftp://ftp.mplayerhq.hu/MPlayer/incoming/\[codecname\]/](ftp://ftp.mplayerhq.hu/MPlayer/incoming/[codecname]/)

6.3 Video & Audio-Ausgabegeräte

6.3.1 Video-Ausgabe-Geräte

6.3.1.1 MTRR

Du solltest UNBEDINGT sicherstellen, dass die MTRR-Register richtig belegt sind, denn sie können einen großen Geschwindigkeits-Schub bringen.

Mache ein `cat /proc/mtrr`:

```
user@linux ~/ $ cat /proc/mtrr
reg00: base=0xe4000000 (3648MB), size= 16MB: write-combining, count=9
reg01: base=0xd8000000 (3456MB), size= 128MB: write-combining, count=1
```

Diese Anzeige ist richtig. Sie zeigt meine Matrox G400 mit 16MB Speicher. Ich habe die Einstellung von XFree 4.x.x, was die MTRR-Register automatisch einstellt.

Wenn nichts funktioniert, musst du sie manuell setzen. Als erstes musst du die Basisadresse finden. Dazu gibt es drei Möglichkeiten:

* durch die X11 Start-Meldungen, zum Beispiel:

```
(--) SVGA: PCI: Matrox MGA G400 AGP rev 4, Memory @ 0xd8000000, 0xd4000000
(--) SVGA: Linear framebuffer at 0xD8000000
```

* durch `/proc/pci` (verwende `lspci -v`):

```
root@linux ~/ # lspci -v
01:00.0 VGA compatible controller: Matrox Graphics, Inc.: Unknown device
0525
Memory at d8000000 (32-bit, prefetchable)
```

* durch die `mga_vid` Kerneltreiber-Meldungen (verwende `dmesg`):

```
root@linux ~/ # dmesg
mga_mem_base = d8000000
```

So, nun gilt es, die Speichergröße zu finden. Dies ist sehr einfach, konvertiere einfach die Video-RAM-Größe nach Hexdezimal oder verwende diese Tabelle:

1	MB	0x100000
2	MB	0x200000
4	MB	0x400000
8	MB	0x800000
16	MB	0x1000000
32	MB	0x2000000

Du weißt die Base-Adresse und die Speichergröße? Lass uns die MTRR Register einstellen! Für die Matrox Karte von oben (base=0xd8000000) mit 32MB RAM (size=0x2000000) führst du einfach das Folgende aus:

```
root@linux ~/ # echo "base=0xd8000000 size=0x2000000"
```

```
type=write-combining" >| /proc/mtrr
```

Nicht alle CPUs unterstützen MTRRs. Zum Beispiel ältere K6-2s [die bei ca. 266MHz, stepping 0] unterstützen kein MTRR, aber stepping 12 CPUs tun es (`cat /proc/cpuinfo` gibt Aufschluss).

6.3.1.2 Videoausgabegeräte für traditionelle Grafikkarten

6.3.1.2.1 Xv

Mit XFree86 4.0.2 oder neueren Versionen kannst du die Hardware-YUV-Unterstützung deiner Grafikkarte mit Hilfe der XVideo-Erweiterungen benutzen. Das ist die Technik, die `-vo xv` benutzt. Dieser Treiber unterstützt darüber hinaus die Anpassung von Helligkeit/Kontrast/Sättigung etc (es sei denn, du benutzt den alten und langsamen DirectShow DivX-Codec, welcher diese Anpassungen unabhängig vom Videoausgabetreiber unterstützt). Schau in der Manpage nach.

Um Xv zum Laufen zu bringen, musst du auf die folgenden Punkte achten:

- * Du musst XFree86 4.0.2 oder eine neuere Version benutzen, da die älteren Versionen XVideo noch nicht kannten.
- * XVideo) Deine Grafikkarte muss Hardware-Unterstützung für YUV bieten, was alle modernen Karten tun.
- * X muss die XVideo-Erweiterung auch tatsächlich laden, was zu Meldungen ähnlich den folgenden führt:

```
(II) Loading extension XVideo
```

Anmerkung:

Diese Meldung besagt nur, dass die XFree86-Erweiterung geladen wird. Bei einer guten Installation sollte das immer der Fall sein. Das heißt allerdings noch nicht, dass die **XVideo-Unterstützung der Grafikkarte** auch geladen wurde!

- * Deine Karte muss unter Linux Xv-Unterstützung haben. Du kannst dich dessen mit `xvinfo` vergewissern, das Teil der XFree86-Distribution ist. Es sollte einen längeren Text ausgeben, der ungefähr so aussieht:

```
X-Video Extension version 2.2
screen #0
  Adaptor #0: "Savage Streams Engine"
    number of ports: 1
    port base: 43
    operations supported: PutImage
    supported visuals:
      depth 16, visualID 0x22
      depth 16, visualID 0x23
    number of attributes: 5
  (...)
  Number of image formats: 7
    id: 0x32595559 (YUY2)
    guid: 59555932-0000-0010-8000-00aa00389b71
    bits per pixel: 16
    number of planes: 1
    type: YUV (packed)
    id: 0x32315659 (YV12)
    guid: 59563132-0000-0010-8000-00aa00389b71
    bits per pixel: 12
    number of planes: 3
```

```
type: YUV (planar)
(...etc...)
```

- * Damit MPlayer Xv benutzen kann, müssen die Pixelformate "YUY2 packed" und "YV12 planar" unterstützt werden.
- * Stell als letztes sicher, dass MPlayer mit Unterstützung für Xv kompiliert wurde. `configure` gibt eine entsprechende Meldung aus.

6.3.1.2.1.1 3dfx-Karten

ältere 3dfx-Treiber hatten bekanntermaßen Probleme mit der XVideo-Beschleunigung, die entweder YUY2 oder YV12 nicht unterstützte. Stell sicher, dass du XFree86 Version 4.2.0 oder neuer verwendest, da diese Versionen mit YV12 und YUY2 keine Probleme haben. Vorherige Versionen, auch 4.1.0, sind **bei Verwendung von YV12 abgestürzt**. Wenn du merkwürdige Effekte bei der Verwendung von `-vo xv` bemerkst, dann probier aus, ob mit SDL, das ebenfalls XVideo benutzen kann, diese Effekte verschwinden. In der [SDL-Sektion](#) stehen Details darüber.

Alternativ kannst du auch den NEUEN tdfxfb-Treiber mit `-vo tdfxfb` verwenden! Lies dazu die tdfxfb-Sektion.

6.3.1.2.1.2 S3-Karten

S3 Savage3D-Karten sollten problemlos funktionieren, aber bei Savage4- Chips solltest du XFree86 4.0.3 oder neuer verwenden. Probier bei Problemen den 16bpp-Farbmodus aus. Und der S3 Virge... Es gibt für ihn zwar Xv- Unterstützung, aber die Karte selber ist so langsam, dass du sie besser verkaufst.

Anmerkung:

Momentan ist nicht ganz klar, welche Savage-Modelle keine Unterstützung für YV12 in Hardware haben, sodass bei ihnen der Treiber diese Konvertierung sehr langsam vornimmt. Wenn du deine Karte deswegen verdächtigst, dann besorg dir einen neueren Treiber, oder frag auf der MPlayer-Users-Mailingliste freundlich nach einem Treiber, der MMX/3DNow unterstützt.

6.3.1.2.1.3 nVidia-Karten

nVidia ist für Linux keine optimale Wahl (auch wenn das laut nVidia nicht stimmt [<users_against_developers.html#nvidia>](#)). Du wirst die binären Closed-Source-Treiber benutzen müssen, die auf nVidias Webseite verfügbar sind. Die Standard-XFree86-Treiber bieten aufgrund der nicht öffentlichen Quellen und auch nicht öffentlich erhältlichen Spezifikationen der nVidia-Chips keine XVideo- Unterstützung für diese Karten.

Soweit ich weiß enthalten die neuesten in XFree86 enthaltenen Treiber XVideo-Unterstützung für GeForce2- und GeForce3-Modelle.

Riva128-Karten bieten nicht einmal mit den binären nVidia-Treibern XVideo-Unterstützung (beklag dich bei nVidia).

6.3.1.2.1.4 Xv mit ATI-Karten

- * Die [GATOS-Treiber](#), die du einsetzen solltest, sofern du keine Rage128- oder Radeon-Karte hast, haben

standardmäßig VSYNC angeschaltet. Das bedeutet, dass die Decodiergeschwindigkeit zur Bildwiederholrate des Monitors synchronisiert wird. Wenn dir die Wiedergabe langsam vorkommt, dann versuch, irgendwie VSYNC abzuschalten, oder setze die Bildwiederholrate des Monitors auf `n` (`fps des Films`) Hz.

- * Radeon VE - Momentan bietet nur die CVS-Version von XFree86 diese Karten, Version 4.1.0 tut dies noch nicht. Außerdem gibt es keine Unterstützung für den TV-Ausgang. Natürlich bekommst du mit MPlayer **hardwarebeschleunigte** Wiedergabe, das ganze wahlweise **mit oder ohne TV-Ausgang**, und es werden dabei nicht einmal weitere Bibliotheken oder X selber benötigt. Lies dazu die [VIDIX-Sektion](#).

6.3.1.2.1.5 NeoMagic-Karten

Diese Chips befinden sich in vielen Laptops. Leider unterstützt der in X 4.2.0 enthaltene Treiber nicht Xv, aber wir haben einen veränderten Treiber mit Xv-Unterstützung für dich. Lade ihn [hier herunter](#). Dieser Treiber wird von Stefan Seyfried zur Verfügung gestellt.

Um die Wiedergabe von Video in DVD-Auflösung zu ermöglichen, ändere deine `XF86Config` wie folgt:

XF86Config
<pre>Section "Device" /[...] Driver "neomagic" Option "OverlayMem" "829440" /[...] EndSection</pre>

6.3.1.2.1.6 Xv mit Trident-Karten

Wenn du Xv mit einer Trident-Grafikkarte benutzen willst, dann installiere XFree86 4.2.0, sofern Xv nicht schon mit 4.1.0 funktioniert. Version 4.2.0 enthält Unterstützung für Xv im Vollbild für Cyberblade XP-Karten.

6.3.1.2.1.7 Kyro/PowerVR-Karten

Wenn du Xv mit Kyro-Karten (wie z.B. der Hercules Prophet 4000XT) verwenden möchtest, dann solltest du die Treiber von der [PowerVR-Seite](#) benutzen.

6.3.1.2.2 DGA

Einleitung:

Dieser Abschnitt versucht, in wenigen Worten zu beschreiben, was DGA generell ist, und was der DGA-Videotreiber in MPlayer erreichen kann, und was nicht.

Was ist DGA?:

DGA ist die Abkürzung für Direct Graphics Access (direkter Zugriff auf die Grafikkarte) und gibt Programmen die Möglichkeit, unter Umgehung des X-Servers direkt den Framebuffer der Grafikkarte zu verändern. Technisch gesehen wird das dadurch realisiert, dass der Framebuffer in den virtuellen Adressraum des jeweiligen Prozesses abgebildet wird. Das wird vom Kernel aber nur dann zugelassen, wenn der Prozess Superuserprivilegien besitzt. Dazu musst du dich entweder als root anmelden oder das SUID-bit des

MPlayer-Binaries setzen (was **nicht empfohlen wird**).

Von DGA gibt es zwei Versionen: DGA1 kommt mit XFree 3.x.x, und DGA2 wurde mit XFree 4.0.1 eingeführt.

DGA1 bietet nur den oben beschriebenen Zugriff auf den Framebuffer. Die Umschaltung des Videomodus klappt nur mit der XVidMode-Erweiterung.

DGA2 beinhaltet die Features der XVidMode-Erweiterung und erlaubt außerdem, die Farbtiefe zu ändern. Damit kannst du also auf 32bit Farbtiefe umschalten, auch wenn der X-Server gerade mit 15bit Farbtiefe läuft.

DGA hat aber auch ein paar Nachteile. Die Funktionsweise scheint ein wenig von der Grafikkarte und dem Grafikkartentreiber im X-Server abhängig zu sein. Es funktioniert also nicht auf jedem System...

DGA-Unterstützung für MPLAYER Installieren:

Stell als erstes sicher, dass X die DGA-Erweiterung lädt. Schau in `/var/log/XFree86.0.log` nach:

```
/var/log/XFree86.0.log  
  
(II) Loading extension XFree86-DGA
```

Wie du siehst ist XFree86 4.0.x oder neuer SEHR ZU EMPFEHLEN! MPlayers DGA-Treiber wird von `./configure` automatisch erkannt. Alternativ kannst du seine Kompilierung mit `--enable-dga` erzwingen.

Falls der Treiber nicht zu einer kleineren Auflösung wechseln konnte, dann experimentier mit den Optionen `-vm` (nur bei X 3.3.x), `-fs`, `-bpp`, `-zoom`, um einen Videomodus zu finden, in den der Film reinpasst. Momentan gibt es keinen Konverter :(

Werde root. DGA braucht root-Privilegien, um direkt in den Grafikspeicher zu schreiben. Wenn du MPlayer als normaler Benutzer starten möchtest, dann installiere MPlayer mit dem SUID-Bit:

```
root@linux ~/ # chown root /usr/local/bin/mplayer  
root@linux / # chmod 750 /usr/local/bin/mplayer  
root@linux / # chmod +s /usr/local/bin/mplayer
```

Jetzt funktioniert es auch als normaler Benutzer.

Warnung: Sicherheitsrisiko!

Dieses ist ein **großes** Sicherheitsloch. Tue das **niemals** auf einem Server oder auf einem Computer, auf dem auch andere Leute Zugriff haben, da sie durch einen SUID-root-MPlayer root-Privilegien erlangen können.

Benutze jetzt die Option `-vo dga`, und ab geht's (hoffe ich zumindest :))! Du solltest auch ausprobieren, ob bei dir die Option `-vo sdl:dga` funktioniert. Sie ist viel schneller.

Ändern der Auflösung:

Der DGA-Treiber ermöglicht es, die Auflösung zu ändern. Damit entfällt die Notwendigkeit der langsamen Softwareskalierung und bietet gleichzeitig ein Vollbild. Idealerweise würde DGA in die gleiche Auflösung schalten, die das Video (natürlich unter Beachtung des Höhen-/Breitenverhältnisses) hat, aber der X-Server lässt

nur Auflösungen zu, die vorher in der `/etc/X11/XF86Config` bzw. `/etc/X11/XF86Config-4` definiert wurden. Diese werden durch sogenannte Modelines festgelegt und hängen von den Fähigkeiten deiner Grafikkarte ab. Der X-Server liest diese Konfigurationsdatei beim Start ein und deaktiviert alle Modelines, die sich nicht mit deiner Hardware vertragen. Du kannst die "überlebenden" Modelines anhand der X11-Logdatei herausfinden (normalerweise `/var/log/XFree86.0.log`).

Diese Einträge funktionieren mit einem Riva128-Chip und dem `nv.o`-X-Server-Treibermodul.

Riva128-Chip											
Section "Modes"											
Identifier	"Modes[0]"										
Modeline	"800x600"	40	800	840	968	1056	600	601	605	628	
Modeline	"712x600"	35.0	712	740	850	900	400	410	412	425	
Modeline	"640x480"	25.175	640	664	760	800	480	491	493	525	
Modeline	"400x300"	20	400	416	480	528	300	301	303	314	Doublescan
Modeline	"352x288"	25.10	352	368	416	432	288	296	290	310	
Modeline	"352x240"	15.750	352	368	416	432	240	244	246	262	Doublescan
Modeline	"320x240"	12.588	320	336	384	400	240	245	246	262	Doublescan
EndSection											

DGA & MPLAYER:

DGA wird bei MPlayer an zwei Stellen benutzt: beim SDL-Treiber mit `-vo sdl:dga` oder beim DGA-Treiber selber (`-vo dga`). Das oben gesagte gilt für beide Treiber. In den folgenden Abschnitten erkläre ich, wie der DGA-Treiber von MPlayer selber arbeitet.

Features des DGA-Treibers:

Der DGA-Treiber wird durch die Option `-vo dga` aktiviert. Sein Standardverhalten sieht vor, dass er in die Auflösung schaltet, die der Videoauflösung am nächsten kommt. Der Treiber ignoriert absichtlich die Optionen `-vm` (Videomodumschaltung aktivieren) und `-fs` (Vollbildmodus erzwingen) - er versucht immer, so viel des Bildes wie möglich durch eine Änderung der Auflösung zu bedecken. Dadurch wird nicht ein einziger weitere CPU-Takt für die Skalierung des Bildes verwendet. Wenn du mit dem Modus nicht zufrieden bist, den der Treiber gewählt hat, dann kannst du ihn zwingen, denjenigen Modus zu wählen, der am besten zu dem mit den Optionen `-x` und `-y` angegebenen Werten passt. Die Option `-v` veranlasst den DGA-Treiber, neben einigen anderen Dingen auch alle von deiner XF86-Konfiguration unterstützten Videomodi aufzulisten. Wenn DGA2 verwendet wird, dann kannst du mit der `-bpp`-Option die Verwendung einer bestimmten Farbtiefe erzwingen. Gültige Werte sind 15, 16, 24 und 32. Es hängt dann von deiner Hardware ab, ob der Modus nativ unterstützt wird, oder ob eine (möglicherweise langsame) Konvertierung stattfindet.

Wenn du Glück hast und dir genug unbenutzter Grafikspeicher zur Verfügung steht, um ein komplettes Bild aufzunehmen, dann wird der DGA-Treiber Doppelpufferung verwenden, was zu regelmäßigerer Wiedergabe führt. Der DGA-Treiber wird dir mitteilen, ob Doppelpufferung angeschaltet ist oder nicht.

Doppelpufferung bedeutet, dass das nächste Bild deines Videos bereits an einer anderen Stelle im Grafikspeicher aufgebaut wird, während das aktuelle Bild angezeigt wird. Wenn das nächste Bild fertig ist, so wird dem Grafikchip nur noch mitgeteilt, wo er das neue Bild im Speicher finden kann. Somit holt sich der Chip seine Daten einfach von dort. In der Zwischenzeit wird der andere, jetzt unbenutzte Puffer wieder mit neuen Videodaten gefüllt.

Doppelpufferung kann mit der Option `-double` aktiviert und mit `-nodouble` deaktiviert werden. Momentan ist die Doppelpufferung standardmäßig deaktiviert. Wenn der DGA-Treiber verwendet wird, dann funktioniert das Onscreen-Display (ODS) nur dann, wenn auch die Doppelpufferung aktiviert ist. Andererseits kann die

Doppelpufferung auch einen großen Einbruch bei der Geschwindigkeit hervorrufen, was sehr von der DGA-Implementierung der Treiber für deine Hardware abhängt (auf meinem K6-II+ 525 benötigt Doppelpufferung weitere 20% CPU-Zeit!).

Punkte bezüglich der Geschwindigkeit:

Generell gesehen sollte der Zugriff auf den DGA-Framebuffer genauso schnell sein wie der X11-Treiber, wobei man zusätzlich noch ein Vollbild erhält. Die prozentualen Geschwindigkeitswerte, die MPlayer ausgibt, müssen mit Vorsicht genossen werden, da sie z.B. beim X11-Treiber nicht die Zeit beinhalten, die der X-Server tatsächlich zum Anzeigen des Bildes benötigt. Klemme ein Terminal an deinen seriellen Port und starte `top`, wenn du wissen willst, wie's wirklich mit der Geschwindigkeit aussieht.

Allgemein betrachtet hängt die Geschwindigkeitsverbesserung von DGA gegenüber dem "normalen" X11-Treiber sehr von deiner Grafikkarte und davon ab, wie gut das X-Servermodul optimiert ist.

Wenn du ein langsames System hast, dann benutze besser eine Farbtiefe von 15 oder 16bit, da sie nur die halbe Bandbreite des 32bit-Farbmodus benötigen.

Eine gute Idee ist auch die Verwendung von 24bit Farbtiefe selbst dann, wenn deine Grafikkarte nativ nur 32bit unterstützt, da bei 24bit 25% weniger Daten über den Bus transferiert werden müssen.

Ich habe schon gesehen, wie einige AVI-Dateien auf einem Pentium MMX 266 wiedergegeben werden konnten. AMD K6-2-CPU's werden ab ca. 400 MHz oder höher funktionieren.

Bekannte Fehler:

Die Entwickler von XFree sagen selber, dass DGA ein ganz schönes Monstrum ist. Sie raten eher davon ab, es zu benutzen, da seine Implementation bei bestimmten Treibern nicht immer ganz fehlerfrei war.

- * Bei der Kombination aus XFree 4.0.3 und dem nv.o-Treiber gibt es einen Fehler, der zu merkwürdigen Farben führt.
- * Die ATI-Treiber müssen den Videomodus mehrmals zurückstellen, nachdem der DGA-Modus verlassen wurde.
- * Einige Treiber schaffen es manchmal einfach nicht, in die vorherige Auflösung zurückzuschalten. Benutze in solch einem Fall `Ctrl + Alt + Keypad + und -`, um manuell die Auflösung zu ändern.
- * Einige Treiber zeigen einfach nur merkwürdige Farben an.
- * Einige Treiber lügen, was die von ihnen in den Prozessorspeicher eingeblendete Menge Grafikspeicher angeht, weswegen `vo_dga` nicht die Doppelpufferung verwendet (SIS?).
- * Einige Treiber schaffen es nicht einmal, auch nur einen einzigen gültigen Grafikmodus bereitzustellen. In solchen Fällen gibt der DGA-Treiber schwachsinnige Modi wie z.B. 100000x100000 oder so ähnlich aus.
- * Das OSD funktioniert nur, wenn auch die Doppelpufferung aktiviert ist.

6.3.1.2.3 SDL

SDL (Simple Directmedia Layer, Simpele Schicht für den direkten Zugriff auf Mediageräte) bietet grundsätzlich eine einheitliche Schnittstelle zu Audio- und Videogeräten. Programme, die die SDL benutzen, kennen nur die SDL und brauchen kein Wissen darüber, welche Video- oder Audiotreiber die SDL tatsächlich benutzt. So kann z.B. eine Doom-Portierung mit der SDL die Sglib, aalib, X11, fbdev und andere Treiber benutzen. Dazu musst du z.B. nur den Videotreiber angeben, indem du die Umgebungsvariable `SDL_VIDEODRIVER` setzt. So lautet zumindest die Theorie.

Bei MPlayer benutzten wir damals die Softwareskaliererroutinen der X11-Treiber von SDL bei Grafikkarten, die keine Unterstützung für XVideo hatten, bis wir unsere eigenen schrieben, die schneller und hübscher waren. Wir

benutzten damals außerdem SDLs aalib-Ausgabe. Auch davon haben wir selber eine komfortablere Version geschrieben. SDLs DGA-Code war besser als unserer - zumindest bis vor kurzem. Verstehst du, worauf ich hinauswill? :)

SDL ist auch bei einigen fehlerbehafteten Treibern/Karten nützlich, wenn das Video ruckelig abgespielt wird (und es nicht an einem langsamen System liegt), oder wenn der Ton hinterherhinkt.

Die SDL-Videoausgabe unterstützt die Anzeige von Untertiteln unterhalb des Films auf den schwarzen Balken (sofern diese vorhanden sind).

Es gibt einige Kommandozeilenparameter bezüglich SDL:

`-vo sdl:name`

Gibt den Namen des SDL-Vidiotreibers an (z.B. aalib, dga, x11)

`-ao sdl:name`

Gibt den Namen des SDL-Audiotreibers an (z.B. dsp, esd, arts)

`-noxv`

Deaktiviert die XVideo-Hardwarebeschleunigung

`-forcexv`

Versucht, die Verwendung der XVideo-Beschleunigung zu erzwingen

SDL-Tasten:

F

C

W/S

Funktion:

Umschalten zwischen Vollbild/Fenstermodus

Wechselt zwischen den vorhandenen Vollbildmodi

gleiche Funktionen wie und / (Mixersteuerung)

Bekannte Fehler:

- * Tasten, die bei `sdl:aalib` einmal gedrückt werden, bleiben aus Sicht des Programmes gedrückt (benutz also `-vo aa!`). Das ist ein Fehler in SDL, den ich nicht beheben kann (mit SDL 1.2.1 getestet).
- * Benutze AUF KEINEN FALL SDL mit dem GUI! Das funktioniert nicht so, wie es sollte.

6.3.1.2.4 SVGalib

Installation:

Du must zuerst die `svgalib` und die dazugehörigen Devel-Pakete installieren, bevor du MPlayer kompilierst, da es die `svgalib` sonst nicht automatisch findet und den Treiber dazu nicht kompiliert (das kann aber trotzdem erzwungen werden). Vergiss auch nicht, in `/etc/vga/libvga.config` richtige Werte für deine Grafikkarte & deinen Monitor anzugeben.

Anmerkungen:

Verwende nicht die `-fs`-Option, da sie die Benutzung des Softwareskalierers erzwingt, und das ganze dann langsam wird. Wenn du diese Option wirklich brauchst, dann verwende auch `-sws 4`, welche zwar schlechte

Qualität produziert, dafür aber auch ein wenig schneller ist.

EGA(4bpp)-Unterstützung:

SVGAlib beinhaltet die EGAlib, und MPlayer kann damit jeden Film in 16 Farben bei den folgenden Modi anzeigen:

- * EGA-Karte mit EGA-Monitor: 320x200x4bpp, 640x200x4bpp, 640x350x4bpp
- * EGA-Karte mit CGA-Monitor: 320x200x4bpp, 640x200x4bpp

Der bpp-Wert (Bits pro Pixel) muss von Hand auf vier gesetzt werden:

```
-bpp 4
```

Die Auflösung des Films muss wahrscheinlich verkleinert werden, damit er in den EGA-Modus reinpasst:

```
-vop scale=640:350 oder  
-vop scale=320:200
```

Dafür brauchen wir eine schnelle, aber schlechte Qualität produzierende Skalieroutine:

```
-sws 4
```

Eventuell muss die automatische Anpassung des Höhen-/Breitenverhältnisses ausgeschaltet werden:

```
-noaspect
```

ANMERKUNG:

Die besten Ergebnisse bei EGA-Bildschirmen erhält man meiner Erfahrung nach, wenn man die Helligkeit ein wenig verringert: `-vop eq=-20:0`. Ich musste auch die Audiosamplerate erniedrigen, weil bei 44KHz der Sound nicht richtig funktionierte: `-srate 22050`.

Du kannst das OSD und Untertitel mit dem `expand`-Filter aktivieren. Die man-Page enthält die exakten Parameter.

6.3.1.2.5 Framebuffer-Ausgabe (FBdev)

`./configure` erkennt automatisch, ob es den Framebuffertreiber (fbdev) kompilieren soll oder nicht. Lies die Framebufferdokumentation in den Kernelquellen ([Documentation/fb/](#)); dort stehen mehr Informationen.

Falls deine Karte den VBE 2.0-Standard nicht unterstützt (wie z.B. ältere ISA-/PCI-Karten wie die S3 Trio64) oder nur VBE 1.2 und älter unterstützt: Tja, dann kannst du immer noch VESAfb benutzen, benötigst aber den SciTech Display Doctor (ehemals UniVBE), der vor dem Booten von Linux geladen werden muss. Benutze dazu eine DOS-Bootdiskette oder was auch immer. Vergiss nicht, deine Kopie von UniVBE zu registrieren ;).

Die fbdev-Ausgabe kann neben den üblichen Parametern noch einige andere:

```
-fb
```

Gibt das zu benutzende Framebuffergerät an (Standard `/dev/fb0`)

```
-fbmode
```

Gibt zu benutzenden Modusnamen an (wie sie in `/etc/fb.modes` stehen)

```
-fbmodeconfig
```

Konfigurationsdatei für die Modi (Standard `/etc/fb.modes`)

```
-monitor_hfreq
-monitor_vfreq
-monitor_dotclock
```

Wichtige Werte, schau dir die `example.conf` an.

Wenn du in einen speziellen Modus wechseln willst, dann benutze

```
user@linux ~/ $ mplayer -vm -fbmode (Modusname) Dateiname
```

- * `-vm` ohne weitere Optionen wird den am besten passenden Modus aus `/etc/fb.modes` auswählen. Kann auch zusammen mit `-x` und `-y` benutzt werden. Die Option `-flip` wird nur dann unterstützt, wenn das Pixelformat des Films mit dem Pixelformat des Videomodus übereinstimmt. Pass auf den `bpp`-Wert auf. `fbdev` wird den aktuell eingestellten benutzen, wenn du nicht mit `-bpp` einen bestimmten angibst.
- * `-zoom` wird nicht unterstützt (Softwareskalierung ist langsam).
- * `-fs` wird nicht unterstützt. Du kannst keine Modi mit 8bpp oder weniger benutzen.
- * Wahrscheinlich wirst du den Cursor (`echo -e '\033[?25l'` oder `setterm -cursor off`) und den Bildschirmschoner (`setterm -blank 0`) deaktivieren wollen. Um den Cursor wieder zu aktivieren: `echo -e '\033[?25h'` oder `setterm - cursor on`.

Anmerkung:

`fbdev` kann den Videomodus in Verbindung mit dem VESA-Framebuffer nicht ändern. Frag auch nicht danach - das ist keine Einschränkung seitens des MPlayers.

6.3.1.2.6 Matrox-Framebuffer (mga_vid)

Dieser Abschnitt beschäftigt sich mit der Unterstützung für den BES (Back-End Scaler, Hardwareskalierungseinheit) bei Karten mit dem Matrox-G200/G400/G450/G550-Chip durch das `mga_vid`-Kernelmodul. Es wird von A'rpi aktiv entwickelt und bietet Unterstützung für Hardware- VSYNC und Dreifachpufferung. Dieser Treiber funktioniert sowohl unter der Framebufferconsole als auch unter X.

Anmerkung:

Das Modul ist nur für Linux-Systeme verfügbar! Auf nicht-Linux-Systemen solltest du statt dessen  [VIDIX](#) benutzen!

Installation:

1. Um den Treiber benutzen zu können, musst du erstmal `mga_vid.o` kompilieren:

```
cd drivers
make
```

2. Jetzt erstelle das Gerät `/dev/mga_vid`:

```
mknod /dev/mga_vid c 178 0
```

und lade das Kernelmodul:

```
insmod mga_vid.o
```

3. Du solltest sicherstellen, dass das Modul die Größe des Grafikkartenspeichers korrekt ermittelt hat. Benutze dazu `dmesg`. Wenn die Angabe nicht stimmt, dann gib nach `rmmmod mga_vid` mithilfe der Option `mga_ram_size` die Größe explizit an:
`insmod mga_vid.o mga_ram_size=16`
4. Wenn das Modul automatisch geladen und entladen werden soll, sobald es benötigt wird, so füge die folgende Zeile in der Datei `/etc/modules.conf` ein:
`alias char-major-178 mga_vid`
Jetzt kopiere `mga_vid.o` in das entsprechende Verzeichnis unterhalb von `/lib/modules/<Kernelversion>/irgendwo`. Jetzt gib ein:
`depmod -a`
5. Schließlich musst du noch MPlayer (erneut) kompilieren. `configure` wird automatisch `/dev/mga_vid` finden und den 'mga'-Treiber erstellen. Die entsprechende Option für MPlayer lautet `-vo mga`, wenn du mit dem `matroxfb` auf der Console arbeitest, oder `-vo xmga`, wenn du unter XFree 3.x.x oder XFree 4.x.x arbeitest.

Der `mga_vid`-Treiber kooperiert mit Xv.

Das `/dev/mga_vid`-Gerät kann z.B. mit `cat /dev/mga_vid` ausgelesen werden, um ein paar Informationen über den aktuellen Zustand zu erhalten. Die Helligkeit kann zusätzlich mit z.B. `echo "brightness=120" > /dev/mga_vid` angepasst werden.

6.3.1.2.7 3dfx-YUV-Unterstützung (tdxfb)

Dieser Treiber benutzt den `tdfx`-Framebuffertreiber des Kernels, um Filme mit YUV-Beschleunigung abzuspielen. Deswegen benötigst du einen Kernel mit `tdxfb`-Unterstützung. Danach musst du MPlayer kompilieren und `configure` mit der Option `--enable-tdxfb` aufrufen.

6.3.1.2.8 OpenGL-Ausgabe

MPlayer unterstützt die Ausgabe von Filmen via OpenGL. Wenn aber deine Plattform/dein Treiber Xv unterstützt (was bei PCs mit Linux praktisch immer der Fall ist), dann benutze besser Xv, da die OpenGL- Geschwindigkeit deutlich geringer als die von Xv ist. Wenn du dagegen eine X11-Implementation hast, die Xv nicht unterstützt, so mag OpenGL eine brauchbare Alternative sein.

Leider unterstützen nicht alle Treiber die erforderlichen Features. Die Utah-GLX-Treiber (für XFree86 3.3.6) unterstützen sie für alle Karten. Auf  <http://utah-glx.sourceforge.net> findest du Details zur Installation.

XFree86(DRI) 4.0.3 oder neuer unterstützt OpenGL mit Matrox- und Radeon-Karten, 4.2.0 und neuer unterstützen zusätzlich Rage128. Auf <http://dri.sourceforge.net> findest du Details zur Installation.

6.3.1.2.9 AALib - Ausgabe im Textmodus

AALib ist eine Bibliothek, mit der Grafiken im Textmodus angezeigt werden, wobei ein mächtiger Textmodusrenderer benutzt wird. Es gibt SEHR viele Programme, die das bereits unterstützen, wie z.B. Doom, Quake etc. MPlayer enthält einen sehr gut brauchbaren Treiber für AALib. Wenn `./configure` feststellt, dass die AALib installiert ist, dann wird anschließend der AALib-Treiber gebaut.

Du kannst diese Tasten im AA-Fenster benutzen, um die Render-Optionen zu beeinflussen:

- 1 Kontrast verringern
- 2 Kontrast erhöhen

- 3 Helligkeit verringern
- 4 Helligkeit erhöhen
- 5 Schnelles Rendern an-/ausschalten
- 6 Wahl des Farbverteilungsmodus (keiner, Fehlerverteilung, Floyd Steinberg)
- 7 Bild invertieren
- a schaltet zwischen den MPlayer- und den AA- Tastenbelegungen um

Die folgenden Kommandozeilenparameter stehen zur Verfügung:

`-aaosdcolor=V`

OSD-Farbe ändern

`-aasubcolor=V`

Farbe der Untertitel ändern

V kann folgende Werte annehmen: (0/normal, 1/dark (= dunkel), 2/bold (= fett), 3/bold font (= fette Schrift), 4/reverse (= negative Farben), 5/special)

Die AALib selber bietet ebenfalls eine große Anzahl von Optionen. Hier sind die wichtigsten:

`-aadriver`

wählt den empfohlenen aa-Treiber (X11, curses, Linux)

`-aaextended`

benutze alle 256 Zeichen

`-aaeight`

benutze auch ASCII-Zeichen mit dem achten Bit

`-aahelp`

gibt alle aalib-Optionen aus

Anmerkung:

Das Rendern ist sehr CPU-intensiv, vor allem, wenn AA unter X benutzt wird. AALib braucht auf einer Nicht-Framebuffer-Console am wenigstens CPU-Zeit. Benutze `SVGATextMode`, um einen möglichst großen Textmodus zu wählen, und genieß den Film! (Hercules- Karten als zweitem Ausbebergerät rocken :) (Kann irgendjemand fbdev/hgafb beibringen, Konvertierung/Farbverteilung vorzunehmen? Wäre nett :))

Wenn dein Computer nicht schnell genug ist, um alle Bilder anzuzeigen, dann benutz `-framedrop`.

Wenn du auf einem Terminal abspielst, dann erzielst du mit dem Linux- Treiber (`-aadriver linux`) bessere Ergebnisse als mit dem curses- Treiber. Allerdings benötigst du dafür auch Schreibrechte auf `/dev/vcsa<terminal>`. Das wird nicht automatisch von aalib festgestellt, aber `vo_aa` versucht, den besten Modus herauszufinden. Lies <http://aa-project.sourceforge.net/tune/> für weitere Tuningtipps.

6.3.1.2.10 VESA-Ausgabe über das VESA-BIOS

Dieser Treiber ist vom Design her ein **generischer Treiber** für alle Grafikkarten, deren Bios VESA VBE 2.0+ unterstützt. Ein weiterer Vorteil dieses Treibers liegt darin, dass er versucht, den TV-Ausgang anzuschalten.

VESA BIOS EXTENSION (VBE) Version 3.0 Date: September 16, 1998 (Seite 70) hat folgendes zu sagen:

Design für zwei Controller

VBE 3.0 unterstützt zwei Controller dadurch, dass angenommen wird, dass beide Controller vom gleichen OEM (Hardwarehersteller) stammen und unter Kontrolle desselben BIOS auf derselben Grafikkarte sitzen. Somit ist es möglich, die Tatsache, dass zwei Controller vorhanden sind, vor der Anwendung zu verbergen. Das verhindert zwar, dass beide Controller unabhängig voneinander gesteuert werden, erlaubt andererseits aber, dass Anwendungen weiterhin problemlos funktionieren, die vor Erscheinen der VBE-3.0-Spezifikation geschrieben wurden. Die VBE-Funktion 00h (Auskunft über die Controller, Return Controller Information) gibt dementsprechend die kombinierten Informationen über beide Controller zurück, was auch eine kombinierte Liste der vorhandenen Grafikmodi einschließt. Sobald eine Anwendung einen Grafikmodus wählt, wird der entsprechende Controller aktiviert. Alle weiteren VBE-Funktionen werden dann auf diesem Controller ausgeführt.

Somit hast du also eine Chance, den TV-Ausgang mit diesem Treiber zum Laufen zu bringen. (Ich vermute, dass der TV-Ausgang normalerweise auf einer separaten Grafikkarte oder zumindest ein separater Ausgang ist.)

Vorteile:

- * Du hast die Möglichkeit, selbst dann Filme anzusehen, wenn **Linux nichts von deiner Grafikkarte** weiß.
- * Du musst keine einzige Grafikanwendung installiert haben (wie X11/XFree86, fbdev usw.). Dieser Treiber wird im **Textmodus** benutzt.
- * Die Chancen stehen gut, dass der **TV-Ausgang funktioniert**. (Es funktioniert nachweislich zumindest auf ATI-Karten.)
- * Dieser Treiber ruft wirklich die **int 10h**-Routine auf und ist dementsprechend kein Emulator - er ruft **echte** Funktionen des **echten** BIOS im **Real**-Modus auf (bzw. im vm68-Modus).
- * Du kannst den Treiber zusammen mit VIDIX benutzen und erhältst dadurch gleichzeitig hardwarebeschleunigte Grafikanzeige **UND** den TV-Ausgang! (für ATI-Karten empfohlen)
- * Wenn du ein VESA-VBE-3.0+-BIOS hast und irgendwo die Optionen `monitor_hfreq`, `monitor_vfreq`, `monitor_dotclock` angegeben werden (Kommandozeile, Konfigurationsdatei), dann bekommst du die höchstmögliche Bildwiederholrate (mit den generischen Timingformeln). Um dieses Feature zu aktivieren, müssen alle Monitoroptionen angegeben werden.

Nachteile:

- * Der Treiber funktioniert nur auf **x86-Systemen**.
- * Er kann nur von **root** benutzt werden.
- * Momentan ist er nur für **Linux** verfügbar.

Benutz diesen Treiber nicht mit **GCC 2.96**! Das wird nicht funktionieren!

Bei VESA Verfügbare Kommandozeilenoptionen:

`-vo vesa:opts`

momentan erkannte Optionen `dga`, um den DGA-Modus zu erzwingen und `nodga`, um ihn zu deaktivieren. Im DGA-Modus kannst du den Doppelpuffermodus mit `-double` aktivieren. Anmerkung: Du kannst diese Parameter auch weglassen, um die **automatische Erkennung** des DGA-Modus zu ermöglichen.

Bekannte Probleme und wie man sie umgeht:

- * Wenn du unter Linux eine **NLS**-Schrift verwendest und du den VESA-Treiber aus dem Textmodus heraus aufrufst, dann wird nach dem Beenden von MPlayer die **ROM-Schrift** anstelle der nationalen geladen sein. Du kannst die nationale Schriftart erneut mit `setsysfont`-Tool laden, das z.B. bei Mandrake zur Distribution gehört.
(**Tipp:** Das gleiche Tool wird für die Lokalisation von fbdev verwendet.)
- * Eine **Linux-Grafiktreiber** aktualisieren nicht den aktiven **BIOS-Modus** im DOS-Speicher. Wenn du also so ein Problem hast, dann benutze den VESA-Treiber nur aus dem **Textmodus** heraus. Andernfalls wird immer der Textmodus (#03) aktiviert werden, und du wirst den Computer neustarten müssen.
- * Oftmals siehst du nur einen **schwarzen Bildschirm**, wenn der VESA-Treiber beendet wird. Um die Anzeige wieder in den richtigen Zustand zu versetzen, wechsele einfach zu einer anderen Console (mit `Alt + Fx`) und wieder zurück.
- * Um eine **funktionierenden TV-Ausgabe** zu erhalten, musst du das TV-Kabel eingesteckt haben, bevor du deinen PC bootest, da das BIOS nur einmal während der POST-Phase initialisiert wird.

6.3.1.2.11 X11

Vermeide diesen Treiber, wenn's geht. Er benutzt X11 (mit den Shared-Memory-Erweiterungen) ohne jegliche Hardwarebeschleunigung. Unterstützt MMX-/3DNow/SSE-beschleunigte Softwareskalierung mit den Optionen `-fs - zoom`, aber die ist trotzdem langsam. Die meisten Karten bieten Unterstützung für Hardwareskalierung. Benutze also `-vo xv` in den meisten Fällen bzw. `-vo xmg` bei Matrox-Karten.

Ein Problem liegt darin, dass die meisten Grafikkartentreiber Hardwarebeschleunigung nicht beim zweiten Ausgang/beim TV-Ausgang unterstützen. In diesen Fällen siehst du nur ein grünes/blaueres Fenster anstelle des Films. Hier ist der X11-Treiber ganz praktisch, aber du brauchst trotzdem eine schnelle CPU für die Softwareskalierung. Benutze nicht den SDL-Ausgabetreiber und SDLs Skalierer, da dieser eine schlechtere Qualität bietet!

Softwareskalierung ist sehr langsam. Versuch also besser, vorher in einen anderen Videomodus zu schalten. Das ist sehr einfach. Such die Modelines in der [DGA-Sektion](#) und füge sie in deine `XF86Config` ein.

- * Wenn du XFree86 4.x.x hast, dann benutze die Option `-vm`. MPlayer wird dann die Auflösung in diejenige ändern, in die dein Film am besten hineinpasst. Wenn das nicht funktioniert:
- * Unter XFree86 3.x.x musst du mit `CTRL + ALT + plus` und `-minus` die Auflösung ändern.

Wenn du die soeben eingefügten Modi nicht wiederfindest, dann schau dir die Ausgabe von XFree86 an. Einige Treiber können nicht die niedrigen Pixelclock-Werte benutzen, die für niedrige Auflösungen von Nöten sind.

6.3.1.2.12 VIDIX**Einleitung:**

VIDIX ist die Abkürzung für **VIDeo Interface for niX** (Video-Schnittstelle für nx). VIDIX wurde entworfen, um eine Schnittstelle für schnelle Userspacetreiber für Grafikkarten zur Verfügung zu stellen, so wie es `mga_vid` für Matroxkarten tut. VIDIX ist ebenfalls sehr portabel.

Diese Schnittstelle wurde als Versuch entworfen, den vorhandenen Schnittstellen für Videobeschleunigung (`mga_vid`, `rage128_vid`, `radeon_vid`, `pm3_vid`) ein einheitliches Dach zu geben. Sie stellt einen einheitlichen Highlevel-Zugang zu BES- und OV-Chips zur Verfügung (BackEnd Scaler und Video Overlays). Sie stellt keine Lowlevel-Funktionen für z.B. Grafikserver zur Verfügung. (Ich möchte nicht mit dem X11-Leuten in Sachen Grafikmodusumschaltung konkurrieren.) Das Ziel dieser Schnittstelle liegt also einfach darin, die höchstmögliche

Geschwindigkeit bei der Videowiedergabe zu erreichen.

Benutzung:

- * Du kannst den eigenständigen Videotreiber benutzen: `-vo vidix` Dieser Treiber wurde als das X11-Frontend für die VIDIX-Technologie entwickelt. Er benötigt dementsprechend einen X-Server und funktioniert auch nur unter X. Beachte, dass der Pixmap-Cache korumpiert werden kann, weil der Treiber unter Umgehung des X-Treibers direkt auf die Hardware zugreift. Du kannst das dadurch verhindern, dass du die von X verwendete Menge des Grafikspeichers verringerst. Benutze dafür die Option "VideoRam" in der "device"-Sektion der `XF86Config`. Du solltest dort die installierte Menge Grafikspeicher minus 4MB eintragen. Wenn du über weniger als 8MB Grafikspeicher verfügst, dann solltest du stattdessen die Option "XaaNoPixmapCache" in der "screen"-Sektion verwenden.
- * Du kannst auch das VIDIX-Untergerät verwenden, das bei vielen Treibern zur Verfügung steht: `-vo vesa:vidix` (**nur unter Linux**) und `-vo fbdev:vidix`

Es ist in der Tat nicht wichtig, welcher Videoausgabetreiber mit VIDIX verwendet wird.

Anforderungen:

- * Die Grafikkarte sollte sich gerade im Grafikmodus befinden (ich schreibe sollte, weil ich's mal im Textmodus ausprobiert habe - es funktioniert zwar, produziert aber hässliche Ausgabe ;) Benutze in diesem Fall `AALib`).
Anmerkung: Jeder kann diesen Trick ausprobieren, indem er den Moduswechsel im `vo_vesa`-Treiber auskommentiert.
- * MPlayers Videoausgabetreiber sollte den aktiven Videomodus kennen und in der Lage sein, dem VIDIX-Untergerät ein paar Charakteristika des X-Servers mitzuteilen.

Ich hoffe, dass jeder von MPlayers Videoausgabetreibern das `:vidix`-Untergerät erkennt.

Bedienung:

Wenn VIDIX als **Untergerät** (`-vo vesa:vidix`) benutzt wird, dann wird die Konfiguration des Videomodus vom Videoausgabegerät erledigt (kurz `vo_server`). Deswegen kannst du die gleichen Kommandozeilenparameter wie für `vo_server` verwenden. Zusätzlich ist die Option `-double` global verfügbar. (Ich empfehle diese Option zumindest bei VIDIX und ATI-Karten.) `-vo xvidix` erkennt momentan die folgenden Optionen: `-fs -zoom -x -y -double`.

Du kannst den VIDIX-Treiber auch direkt als drittes Teilargument auf der Kommandozeile angeben:

```
root@linux ~/ # mplayer -vo xvidix:mga_vid.so -fs -zoom -double file.avi
```

oder

```
root@linux ~/ # mplayer -vo vesa:vidix:radeon_vid.so -fs -zoom -double  
-bpp 32 file.avi
```

Das ist allerdings gefährlich, und du solltest das lieber nicht tun. Hierbei wird die Verwendung des angegebenen Treibers erzwungen, und das Resultat ist unklar (dein Computer könnte sogar **abstürzen**). Du solltest das wirklich NUR DANN tun, wenn du absolut sicher bist, dass es funktioniert und MPlayer es nicht eh schon automatisch auswählt. Berichte den Entwicklern von deinen Erfahrungen. Die korrekte Art, VIDIX zu benutzen, ist ohne das dritte Teilargument, sodass MPlayer automatisch den richtigen Treiber aussucht.

VIDIX ist eine sehr junge Technologie. Es ist deshalb gut möglich, dass sie auf deinem System nicht funktioniert. In diesem Fall liegt deine einzige Möglichkeit darin, VIDIX auf dein System zu portieren (hauptsächlich die `libdha`). Aber es gibt immer noch die Hoffnung, dass es auf den Systemen funktioniert, auf denen auch X11 funktioniert.

Da VIDIX direkten Zugriff auf die Hardware benötigt, musst du MPlayer entweder als root starten oder der Programmdatei das SUID-Bit setzen (**WARNUNG: Das ist ein Sicherheitsrisiko!**). Alternativ kannst du auch spezielle Kernelmodule benutzen:

1. Lade dir die  [Entwicklerversion](#) der `svgalib` herunter (z.B. 1.9.17), **ODER** lade dir eine von Alex speziell für die Benutzung mit MPlayer modifizierte Version herunter (die nicht die `svgalib`-Sources zum Kompilieren benötigt): [svgalib_helper-1.9.17-mplayer.tar.bz2](#).
2. Kompiliere das Modul im `svgalib_helper`-Verzeichnis (das im `svgalib-1.9.17/kernel/`-Verzeichnis gefunden werden kann, wenn du die Sources von der `svgalib`-Seite heruntergeladen hast) und lade es mit `insmod`.
3. Verschiebe das `svgalib_helper`-Verzeichnis nach `mplayer/main/libdha/svgalib_helper`.
4. Wenn du die Sources von der `svgalib`-Seite heruntergeladen hast, dann musst du den Kommentar vor der `CFLAGS`-Zeile entfernen, die "`svgalib_helper`" enthält, und die sich in `libdha/Makefile` befindet.
5. Kompiliere und installiere `libdha`.

6.3.1.2.12.1 VIDIX mit ATI-Karten

Momentan werden die meisten ATI-Karten unterstützt, von der Mach64 bis hin zur neuesten Radeon.

Es gibt zwei kompilierte Binaries: `radeon_vid` für Radeons und `rage128_vid` für Rage128-Karten. Du kannst entweder eine der beiden erzwingen oder das VIDIX-System automatisch alle verfügbaren Treiber ausprobieren lassen.

6.3.1.2.12.2 Matrox-Karten

Matrox G200, G400, G450 und G550 funktionieren. Der Treiber unterstützt Videoequalizer und sollte fast genauso schnell, wie der  [Matrox-Framebuffer](#) sein.

6.3.1.2.12.3 VIDIX mit Trident-Karten

Es gibt einen Treiber für den Trident Cyberblade/i1-Chipsatz, der auf VIA Epia-Mainboards eingesetzt wird.

Der Treiber wurde von Alastair M. Robinson geschrieben und weiterentwickelt, welcher auch die neuesten Treiberversionen auf seiner  [Homepage](#) zum Download anbietet. Dieser Treiber wird bei MPlayer immer sehr schnell nach Updates implementiert, sodass die CVS-Version von MPlayer immer auf dem aktuellen Stand sein sollte.

6.3.1.2.12.4 DLabs-Karten

Auch wenn es einen Treiber für 3DLabs GLINT R3-Chips und Permedia3-Chips gibt, so hat noch niemand diese getestet. Feedback wird deswegen gern gesehen.

6.3.1.2.13 DirectFB

DirectFB ist eine Grafikkbibliothek, deren Zielplattform eingebettete Systeme sind. Sie bietet maximale

Hardwarebeschleunigung bei minimalem Ressourcenverbrauch und minimalem Overhead. - Zitat von 
<http://www.directfb.org>.

Ich lasse die DirectFB-Features in dieser Sektion weg.

Obwohl MPlayer nicht als "Videoprovider" bei DirectFB unterstützt wird, bietet dieser Treiber Videowiedergabe mittels DirectFB. Die Wiedergabe ist - natürlich - hardwarebeschleunigt. Bei meiner Matrox G400 war der DirectFB genauso schnell wie XVideo.

Versuche, immer die neueste Version von DirectFB zu verwenden. Du kannst DirectFB-Optionen mit der `-dfbopts`-Option auf der Kommandozeile angeben. Layer-Auswahl erfolgt durch Angabe als Teilargument, z.B. mit `-vo directfb:2` (Layer -1 ist der Standardwert: automatische Layerauswahl).

6.3.1.2.14 DirectFB/Matrox (dfbmga)

Bitte lies die  [DirectFB-Sektion](#) für generelle Informationen über DirectFB.

Dieser Videoausgabetreiber wird auf einer Matrox G400 den CRTC2 (des zweiten Ausgangs) aktivieren und damit das Video **unabhängig** vom primären Ausgang anzeigen.

Anweisungen, um das zum Laufen zu bringen, stehen in der  [Techniksektion](#).

Anmerkung:

Uns ist es nicht gelungen, das zum Laufen zu bringen, aber anderen schon. Wie auch immer, eine Portierung des CRTC2-Codes für mga_vid ist bereits in Arbeit.

6.3.1.3 MPEG-Dekoderkarten

6.3.1.3.1 DVB

MPlayer unterstützt Karten mit dem Siemens-DVB-Chipsatz von Herstellern wie Siemens, Technotrend, Galaxis oder Hauppauge. Die neuesten DVB-Treiber gibt's auf der  [Linux TV- Seite](#). Wenn du in Transcodierung in Software machen willst, dann brauchst du eine CPU mit mindestens 1GHz.

`configure` sollte automatisch deine DVB-Karte erkennen. Wenn es das nicht tut, dann erzwingen DVB-Unterstützung mit

```
root@linux ~/ # ./configure --enable-dvb
```

Wenn die ost-Headerdateien nicht an ihrem normalen Platz liegen, dann gib explizit den Pfad zu ihnen an:

```
root@linux ~/ # ./configure  
--with-extraincdir=<DVB-Quellenverzeichnis>/ost/include
```

Dann kompiliere und installiere wie sonst auch.

Bedienung:

Hardwaredecodierung (Abspielen von Standard-MPEG1/2-Dateien) geschieht mit diesem Kommando:

```
user@linux ~/ $ mplayer -ao mpegpes -vo mpegpes file.mpg | vob
```

Softwaredecodierung oder die Transcodierung verschiedener Formate nach MPEG1 klappt so:

```
user@linux ~/ $ mplayer -ao mpegpes -vo mpegpes -vop lavc deinedatei.ext
user@linux / $ mplayer -ao mpegpes -vo mpegpes -vop fame,expand
deinedatei.ext
```

Beachte, dass DVB-Karten nur bestimmte Bildhöhen unterstützen: 288 und 576 für PAL und 240 und 480 für NTSC. Du **musst** das Bild vorher skalieren, wenn die Höhe nicht einer der oben erwähnten entspricht: `-vop scale=width:height`. DVB-Karten unterstützen eine Vielzahl von horizontalen Auflösungen wie z.B. 720, 704, 640, 512, 480, 352 etc. Sie skalieren horizontal selber in Hardware, sodass du meistens nicht in horizontaler Richtung skalieren musst. Bei einem 512x384-DivX kannst du folgendes probieren:

```
user@linux ~/ $ mplayer -ao mpegpes -vo mpegpes -vop lavc,scale=512:576
```

Wenn du einen Widescreen-Film hast und du ihn nicht auf die volle Höhe skalieren möchtest, dann kannst du den `expand=w:h`-Filter benutzen, um schwarze Balken hinzuzufügen. Um ein 640x384-DivX anzuschauen:

```
user@linux ~/ $ mplayer -ao mpegpes -vo mpegpes -vop lavc,expand=640:576
file.avi
```

Wenn deine CPU für 720x576-DivX zu langsam ist, dann skalier herunter:

```
user@linux ~/ $ mplayer -ao mpegpes -vo mpegpes -vop lavc,scale=352:576
file.avi
```

Wenn sich die Geschwindigkeit nicht verbessert, dann skalier auch in vertikaler Richtung:

```
user@linux ~/ $ mplayer -ao mpegpes -vo mpegpes -vop lavc,scale=352:288
file.avi
```

Für ein OSD und Untertitel kannst du das OSD-Feature des `expand`-Filters benutzen. Anstelle von `expand=w:h` oder `expand=w:h:x:y` benutzt du dafür `expand=w:h:x:y:1` (der fünfte Parameter `:1` schaltet die OSD-Anzeige an). Eventuell willst du das Bild ein wenig nach oben schieben, um unten mehr Platz für die Untertitel zu haben. Vielleicht willst du auch die Untertitel hochschieben, wenn sie ansonsten außerhalb des Sichtbereiches des Fernsehers liegen. Das kannst du mit `-subpos <0-100>` erreichen, wobei `-subpos 80` meistens eine gute Wahl darstellt.

Um Filme mit weniger/mehr als 25 Bildern pro Sekunde auf einem PAL- Fernseher abzuspielen, oder wenn du eine langsame CPU hast, verwende die Option `-framedrop`.

Um das Höhen-/Breitenverhältnis des DivX beizubehalten und trotzdem die optimalen Skalierungsparameter zu verwenden (Hardwareskalierung in horizontaler Richtung und Softwareskalierung in vertikaler Richtung unter Beibehaltung des richtigen Höhen-/Breitenverhältnisses), benutze den neuen `dvbscale`-Filter:

für 3:4 TV: `-vop lavc,expand=-1:576:-1:-1:1,scale=-1:0,dvbscale`

für 16:9 TV: `-vop lavc,expand=-1:576:-1:-1:1,scale=-1:0,dvbscale=1024`

Ausblick:

Wenn du Fragen hast oder an der Diskussion über zukünftige Features teilnehmen willst, dann melde dich an unserer [MPlayer-DVB](#) Mailingliste an. Denk bitte daran, dass dort Englisch gesprochen wird.

Für die Zukunft kannst du mit der Möglichkeit, dass OSD und die Untertitel mit den eingebauten Funktionen der DVB-Karten anzuzeigen, mit flüssigerer Wiedergabe von Filmen mit weniger/mehr als 25 Bildern pro Sekunde und mit Echtzeittranscodierung zwischen MPEG2 und MPEG4 (partielle Decompression) rechnen.

6.3.1.3.2 DXR2

TODO: Bitte schreib hier jemand ein paar Informationen rein.

6.3.1.3.3 DXR3/Hollywood+

MPlayer unterstützt die hardwarebeschleunigte Wiedergabe mit den Karten Creative DXR3 und Sigma Designs Hollywood Plus. Beide Karten basieren auf dem em8300-MPEG-Decoderchip von Sigma Designs.

Als erstes brauchst du korrekt installierte DXR3/H+-Treiber, Version 0.12.0 oder neuer. Diese Treiber und weitere Installationsanweisungen findest du auf der Seite [DXR3 & Hollywood Plus for Linux](#). `configure` sollte die Karte automatisch finden. Die Kompilierung sollte auch problemlos funktionieren.

Bedienung:

```
-vo dxr3:prebuf:sync:norm=x:<device>
```

`overlay` aktiviert das Overlay anstelle des TV-Ausgangs. Dafür brauchst du ein korrekt konfiguriertes Overlaysetup. Am einfachsten konfigurierst du das Overlay mit dem Tool `autocal`. Danach starte MPlayer mit dxr3-Ausgabe und ohne Overlay anzuschalten. Starte `dxr3view`. Mit `dxr3view` kannst du die Overlayeinstellungen verändern und siehst die Auswirkungen sofort. Eventuell wird dieses Feature irgendwann vom MPlayer-GUI unterstützt. Wenn du das Overlay richtig eingestellt hast, dann brauchst du `dxr3view` nicht mehr laufen zu lassen. `prebuf` schaltet Prebuffering ein. Das ist ein Feature des em8300-Chips, das es ihm ermöglicht, mehr als nur ein Bild gleichzeitig zu speichern. Das bedeutet, dass MPlayer in diesem Modus versucht, den Puffer ständig mit Daten gefüllt zu halten. Wenn du einen langsamen Rechner hast, dann wird MPlayer wahrscheinlich die meiste Zeit über knapp oder genau 100% der CPU-Zeit belegen. Das ist vor allem dann der Fall, wenn du echte MPEG-Streams (z.B. DVDs, SVCDs etc) abspielt, da MPlayer nicht nach MPEG encodieren muss und den Puffer sehr schnell wird füllen können.

Mit Prebuffering ist die Videowiedergabe viel weniger gegenüber anderen CPU-intensiven Programmen anfällig. Frames werden nur dann verworfen, wenn eine andere Applikation für eine sehr lange Zeit die CPU belegt.

Wenn kein Prebuffering verwendet wird, dann ist der em8300 viel anfälliger gegenüber CPU-Last. Somit wird dringend empfohlen, MPlayers `-framedrop`-Option zu verwenden, um die A/V-Sync zu erhalten. `sync` aktiviert die neue sync-Methode. Dieses Feature ist momentan noch experimentell. Bei dieser Methode beobachtet MPlayer ständig die interne Uhr des em8300-Chips. Weicht diese von MPlayers Uhr ab, so wird die des em8300-Chips zurückgesetzt, sodass dieser alle Frames verwirft, die hinterherhängen.

`norm=x` setzt den TV-Standard der DXR3-Karte, ohne dafür externe Programme wie `em8300setup` zu benötigen. Gültige Werte sind 5 = NTSC, 4 = PAL-60, 3 = PAL. Spezielle Standards sind 2 (automatische Erkennung mit PAL/PAL-60) und 1 (automatische Erkennung für PAL/NTSC), da sie den Standard in Abhängigkeit der FPS des Films setzen. `norm = 0` (Standard) ändert den momentan eingestellten TV-Standard nicht. `<device>` = Gerätenummer wählt die zu verwendene em8300-Karte, falls du mehrere davon hast.

Jede dieser Optionen kann auch weggelassen werden.

`:prebuf:sync` scheint sehr gut zu funktionieren, wenn du DivX abspielst. Es gab Berichte von Leuten, die Probleme mit `prebuf` bei der Wiedergabe von MPEG1/2-Dateien hatten. Du solltest es also zuerst ohne Optionen probieren. Wenn du Sync-Probleme hast, dann probier `:sync` aus.

`-ao oss:/dev/em8300_ma-X`

Audioausgabe, wobei `X` die Gerätenummer ist (0 bei nur einer Karte). Der em8300 kann keine Sampleraten niedriger als 44100Hz abspielen. Wenn die Samplerate weniger als 44100Hz beträgt, dann wähle 44100Hz oder 48000Hz, je nachdem, welche davon besser passt. Beispiel: Wenn der Film 22050Hz benutzt, dann wähle 44100Hz, da $44100 / 2 = 22050$ ist. Bei 24000Hz nimmst du 48000Hz etc. Das funktioniert nicht mit der digitalen Audioausgabe (`-ac hwac3`).

`-vop lavc/fame`

Wenn du nicht-MPEG-Filme mit dem em8300 ansehen möchtest (z.B. DivX oder RealVideo), dann musst du einen MPEG1-Videofilter wie `lavcodec` (`lavc`) oder `libfame` (`fame`) verwenden. Momentan ist `lavc` sowohl schneller als auch qualitativ besser, sodass die Empfehlung `lavc` lautet. Schau in der Manpage nach. Dort stehen weitere Informationen zu `-vop lavc/fame`.

Die Benutzung von `lavc` wird empfohlen. Momentan gibt es keine Möglichkeit, die Anzahl der Bilder pro Sekunde des em8300 zu setzen, was bedeutet, dass sie fest bei 29.97 liegt. Aus diesem Grund solltest du `-vop lavc=<quality>:25` verwenden, besonders dann, wenn du auch Prebuffering verwendest. Warum aber 25 und nicht 29.97? Tja, die Sache ist, dass das Bild bei 29.97 unruhig wird. Wir wissen leider nicht, warum das so ist. Wenn du Werte zwischen 25 und 27 benutzt, dann wird das Bild stabil. Momentan können wir das nur als gegeben hinnehmen.

`-vop lavc,expand=-1:-1:-1:-1:1`

Obwohl der DXR3-Treiber ein OSD über das MPEG1-/2-/4-Video projizieren kann, ist es qualitativ deutlich schlechter als MPlayers traditionelles OSD, und es hat diverse Probleme mit der Erneuerung der Anzeige. Das oben angegebene Kommando konvertiert das Video erst nach MPEG4 (das ist leider erforderlich) und wendet dann den `expand`-Filter an, der zwar das Bild nicht vergrößert (`-1: = Standardwerte`) aber dafür das normale OSD auf das Bild stanzt (die "1" am Ende).

`-ac hwac3`

Der em8300 unterstützt die Audiowiedergabe von AC3-Streams (Surroundsound) über den digitalen Ausgang der Karte. Schau oben bei der `-ao oss`-Option nach. Sie muss angegeben werden, um den DXR3-Ausgang anstelle der Soundkarte anzugeben.

6.3.1.4 Andere Anzeigehardware

6.3.1.4.1 Zr

Dieser Treiber ist ein Anzeigetreiber (`-vo zr`), der verschiedene MJPEG-Aufnahme-/Wiedergabekarten unterstützt. Getestet wurde er mit DC10+ und Buz, und er sollte auch mit der LML33 und der Original-DC10 funktionieren. Dieser Treiber encodiert jedes Bild nach JPEG und schickt es dann an die Karte. Für die Encodierung wird `libavcodec` benutzt und dementsprechend auch benötigt. Mit dem speziellen `/cinemara`-Modus kannst du Filme auch tatsächlich im Breitbildformat anschauen, wenn du zwei Beamer und zwei MJPEG-Karten hast. Abhängig von der Qualität und Auflösung braucht dieser Treiber eine Menge CPU-Power. Benutz also besser die `-framedrop`-Option, wenn deine Maschine zu langsam ist. Anmerkung: Mein AMD K6-2 350MHz ist durchaus in der Lage, Filme in VCD-Größe mit `- framedrop` wiederzugeben.

Dieser Treiber benutzt den Kerneltreiber, den du unter  <http://mjpeg.sourceforge.net> herunterladen kannst. Dieser muss also vorher schon funktionieren. `configure` erkennt automatisch vorhandene MJPEG-Karten. Wenn nicht, dann erzwinge zr mit

```
user@linux ~/ $ ./configure --enable-zr
```

Die Ausgabe kann mit diversen Optionen gesteuert werden. Eine vollständige Liste findest du in der Manpage. Eine kurze Auflistung gibt dir auch

```
user@linux ~/ $ mplayer -zrhelphelp
```

Sachen wie das OSD und Skalierung werden nicht von diesem Treiber erledigt, aber sie können natürlich durch Filter realisiert werden. Beispiel: Angenommen, du hast einen Film mit einer Auflösung von `512x272`, und du möchtest ihn im Vollbild auf deiner DC10+ anschauen. Du hast dann drei Möglichkeiten: den Film auf eine Breite von `768`, `384` oder `192` zu skalieren. Aus Geschwindigkeits- und Qualitätsgründen würde ich empfehlen, den Film auf `384x204` mit dem bilinearen Algorithmus zu skalieren. Die Kommandozeile sieht dazu wie folgt aus:

```
user@linux ~/ $ mplayer -vo zr -sws 0 -vop scale=384:204 movie.avi
```

Das Beschneiden des Bildes kann mit dem `crop`-Filter geschehen oder vom Treiber selber vorgenommen werden. Angenommen, der Film ist zu breit für die Anzeige deiner Buz, und du möchtest `-zrcrop` benutzen, um den Film schmaler zu machen. Dann benutzt du folgendes Kommando:

```
user@linux ~/ $ mplayer -vo zr -zrcrop 720x320+80+0 benhur.avi
```

Mit dem `crop`-Filter sieht es so aus:

```
user@linux ~/ $ mplayer -vo zr -vop crop=720:320:80:0 benhur.avi
```

Mehrfache Verwendung von `-zrcrop` aktiviert den `/cinerama/`-Modus. Das heißt, du kannst das Bild über mehrere Fernseher oder Beamer verteilen, um eine größere Anzeigefläche zu erreichen. Angenommen, du hast zwei Beamer. Der linke hängt an deiner Buz an `/dev/video1`, und der rechte hängt an deiner DC10+ an `/dev/video0`. Der Film hat eine Auflösung von `704x288`. Nehmen wir weiter an, dass du den rechten Beamer schwarz/weiß betreiben möchtest, und dass du auf dem linken Beamer Bilder mit der Qualitätsstufe `10` haben möchtest. Dann benutzt du dafür das folgende Kommando:

```
user@linux ~/ $ mplayer -vo zr -zrdev /dev/video0 -zrcrop 352x288+352+0  
-zrxdooff 0 -zrbw \ -zrcrop 352x288+0+0 -zrdev /dev/video1 -zrquality 10  
movie.avi
```

Wie du siehst gelten die Optionen vor dem zweiten `-zrcrop` nur für die DC10+ und die Optionen nach dem zweiten `-zrcrop` nur für die Buz. Die maximale Anzahl an MJPEG-Karten, die am `/cinerama/`-Modus teilnehmen, liegt bei vier, sodass du dir eine `2x2`-Videowand basteln kannst.

Zuletzt ein wirklich wichtiger Hinweis: Starte oder beende auf keinen Fall XawTV während der Wiedergabe, da das deinen Computer zum Absturz bringen wird. Du kannst aber problemlos **ZUERST** XawTV, **DANN** MPlayer starten, warten, bis MPlayer fertig ist und **ZULETZT** XawTV beenden.

6.3.1.4.2 Blinkenlights

Dieser Treiber kann Video mit dem Blinkenlights UDP-Protokoll wiedergeben. Wenn du nicht weißt, was  [Blinkenlights](#) ist, dann brauchst du diesen Treiber auch nicht.

6.3.1.5 Unterstützung für die TV- Ausgabe

6.3.1.5.1 Matrox G400-Karten

Unter Linux hast du zwei Möglichkeiten, den TV-Ausgang deiner G400 anzuschalten:

WICHTIG: Anweisungen für die Matrox G450/G550 und deren TV-Ausgänge findest du in der nächsten Sektion!

- * **XFree86:** mit dem alten Treiber und dem HAL-Modul, welches es auf der  [Matrox-Seite](#) gibt. Damit bekommst du X auf dem Fernseher, aber **keine Hardwarebeschleunigung** wie unter Windows! Der zweite Ausgang besitzt nur einen YUV-Framebuffer. Der **BES** (BackEnd Scaler, die YUV-Skalierungseinheit des G200/G400/G450/G550) funktioniert mit ihm nicht! Der Windows-Treiber umgeht das irgendwie, wahrscheinlich dadurch, dass er die 3D-Engine für die Skalierung und den YUV-Framebuffer zur Anzeige des skalierten Bildes verwendet. Wenn du unbedingt X benutzen willst, dann probier `-vo x11 -fs -zoom`, aber das wird **LANGSAM** sein und den **Macrovision**-Kopierschutz aktiviert haben. (Du kannst Macrovision mit diesem  [Perlscript](#) umgehen.)
- * **Framebuffer:** Mit den **matroxfb-Modulen** in den 2.4er Kernen. 2.2er Kernel kennen den TV-Ausgang noch nicht und sind somit hierfür nicht geeignet. Du musst ALLE matroxfb-spezifischen Features bei der Kompilierung anschalten (bis auf MultiHead). Kompiliere sie als Module! Du musst ebenfalls I2C anschalten.

1. Gehe nach `TVout/matroxset` und gib `make`. ein. Installiere `matroxset` in ein Verzeichnis, das in deinem PATH liegt.
2. Wenn du `fbset` nicht installiert hast, dann gehe nach `TVout/fbset` und gib `make` ein. Installiere `fbset` in ein Verzeichnis, das in deinem PATH liegt.
3. Geh jetzt in das Verzeichnis `TVout/` in den MPlayer-Quellen und führe dort `./modules` als root aus. Deine Textmodusconsole wird danach in den Framebuffermodus umschalten, aus dem es keinen Weg zurück gibt!
4. Editiere als nächstes das Script `./matroxtv`. Es wird dir ein simples Menü präsentieren. Drücke **2** gefolgt von **ENTER**. Jetzt solltest du auf dem Fernseher das gleiche Bild wie auf dem Monitor sehen. Wenn das TV-Bild (PAL ist die Standardeinstellung) merkwürdige Streifen enthält, dann war das Script nicht in der Lage, die Auflösung richtig zu setzen (standardmäßig 640x512). Probier andere im Menü angebotene Auflösungen aus und/oder experimentier mit `fbset`.

So. Die nächste Aufgabe ist es, den Cursor auf `tty1` (oder `woauchimmer`) verschwinden zu lassen, und den Bildschirmschoner auszuschalten. Führe folgende Kommandos aus:

```
echo -e '\033[?25l'
```

oder

```
setterm -cursor off setterm -blank 0
```

Wahrscheinlich möchtest du das in ein Script packen und dabei gleich den Bildschirm löschen. Um den Cursor wieder anzuschalten:

```
echo -e '\033[?25h'
```

oder

```
setterm -cursor on
```

Yeah, cool! Starte die Wiedergabe mit `mplayer -vo mga -fs -screenw 640 -screenh 512 <Dateiname>`

(Wenn du X benutzt, dann wechsel jetzt auf den matroxfb mit z.B. CTRL + ALT + F1!) ändere die 640x512, wenn du eine andere Auflösung verwendest.

Genieß die ultra-schnelle und featurereiche Wiedergabe mit dem Matrox-TV-Ausgang (sogar noch besser als Xv)!

6.3.1.5.2 Matrox G450/G550-Karten

Unterstützung für den TV-Ausgang dieser Karten wurde erst kürzlich implementiert und ist noch nicht in den Standardkernen enthalten. Momentan kann das **mga_vid**-Modul nicht benutzt werden, wenn ich recht informiert bin, da der G450/G550-Treiber nur in einer Konfiguration arbeitet: Der erste CRTC-Chip (mit den vielen Features) am ersten Display (meistens der Monitor), und der zweite CRTC (kein **BES** - Erläuterungen zum BES gibt's in der G400-Sektion oben) am Fernseher. Somit kannst du momentan nur den **fbdev**-Treiber benutzen.

Der erste CRTC kann momentan nicht an den zweiten Ausgang umgeleitet werden. Der Author des matroxfb-Kernelmoduls, Petr Vandrovec, wird auch das irgendwann unterstützen, indem die Ausgabe des ersten CRTC auf beiden Ausgängen angezeigt wird, wie es momentan auch für die G400 empfohlen wird (siehe oben).

6.3.1.5.3 TV-Ausgabe mit ATI-Karten

Vorwort:

Momentan möchte ATI keinen einzigen ihrer TV-Ausgabe-Chips unter Linux unterstützen, da sie die Macrovision-Technologie lizenziert haben.

Status der ATI-TV-Ausgabeunterstützung unter Linux:

- * **ATI Mach64:** Von  [gatos](#) unterstützt.
- * **ASIC Radeon VIVO:** Von  [gatos](#) unterstützt.
- * **Radeon and Rage128:** Von MPlayer unterstützt! Lies die  [VESA-Treiber](#)- und  [VIDIX](#)-Sektionen.
- * **Rage Mobility P/M, Radeon, Rage 128, Mobility M3/M4:** Von  [atitvout](#) unterstützt.

Benutze bei anderen Karten einfach den  [VESA-Treiber](#) ohne  [VIDIX](#). Dafür brauchst du aber eine schnelle CPU.

Nur eines musst du tun - das TV-Kabel vor dem Booten eingesteckt haben, da das BIOS sich nur einmal während der POST-Prozedur initialisiert.

6.3.1.5.4 Voodoo 3

Lies diese  [URL](#).

6.3.1.5.5 nVidia

Zuerst MUSST du die Closed-Source-Treiber von  <http://nvidia.com> herunterladen. Ich werde Installation und Konfiguration nicht im Detail beschreiben, da diese außerhalb der Aufgabe dieses Dokuments liegt.

Nachdem du sichergegangen bist, dass XFree86, XVideo und die 3D- Beschleunigung funktionieren, ändere die **XF86Config**, und passe das folgende Beispiel deiner Karte an:

```
XF86Config

Section "Device"
    Identifier      "GeForce"
    VendorName     "ASUS"
    BoardName      "nVidia GeForce2/MX 400"
    Driver         "nvidia"
    #Option        "NvAGP" "1"
    Option         "NoLogo"
    Option         "CursorShadow" "on"

    Option         "TwinView"
    Option         "TwinViewOrientation" "Clone"
    Option         "MetaModes" "1024x768,640x480"
    Option         "ConnectedMonitor" "CRT, TV"
    Option         "TVStandard" "PAL-B"
    Option         "TVOutFormat" "Composite"

EndSection
```

Natürlich ist der wichtige Teil die TwinView-Optionen.

6.3.2 Audio-Ausgabe-Geräte

6.3.2.1 Audio-/VideoSynchronisation

MPlayers Audio-Interface wird `libao2` genannt. Es enthält zur Zeit diese Treiber:

`oss`

OSS (ioctl)-Treiber (unterstützt das Durchreichen von AC3)

`sdl`

SDL-Treiber (unterstützt Sound-Dämonen wie **ESD** und **ARTS**)

`nas`

NAS (Network Audio System)-Treiber

`alsa5`

ALSA 0.5-Treiber

`alsa9`

ALSA 0.9-Treiber (unterstützt das Durchreichen von AC3)

`sun`

SUN Audiotreiber (`/dev/audio`) für BSD und Solaris8

`arts`

ARTS-Treiber (hauptsächlich für KDE-Benutzer)

esd

ESD-Treiber (hauptsächlich für GNOME-Benutzer)

Linux-Soundkartentreiber haben Probleme mit der Kompatibilität. MPlayer benutzt einen in vernünftig implementierten Treibern eingebauten Mechanismus, um die Audio-/Video-Synchronisation korrekt beizubehalten. Leider sind einige Treiberautoren nicht sonderlich gründlich bei der Implementation dieses Mechanismus, da er nicht benötigt wird, um MP3s abzuspielen oder Soundeffekte zu erzeugen.

Andere Medienplayer wie z.B. [aviplay](#) oder [xine](#) funktionieren bei dir wahrscheinlich so, wie sie sind, weil sie eine "simplere" Methode wie internes Timing für die Audio-/Video-Synchronisation verwenden. Verschiedene Tests haben gezeigt, dass ihre Methoden nicht ganz so effizient wie MPlayers Methoden sind.

Wenn du MPlayer zusammen mit einem korrekt geschriebenen Soundkartentreiber verwendest, dann wirst du niemals A/V-Desyncs aufgrund des Audiotreibers erleben (höchstens mit sehr schlecht erstellten Dateien. Schau in einem solchen Fall in die Dokumentation, wie du das beheben kannst).

Wenn du einen solchen nicht gut implementierten Audiotreiber verwendest, dann probier die Option `-autosync` aus, die deine Probleme lösen sollte. Schau in der man-Page nach; dort findest du mehr Informationen dazu.

Einige Anmerkungen:

- * Wenn du einen OSS-Treiber hast, versuche als erstes `-ao oss` (dies ist die Standardeinstellung). Wenn du kleine Fehler, Verzögerungen oder irgendetwas anderes ungewöhnliches hörst, versuche mal `-ao sdl` (HINWEIS: die SDL-Bibliotheken und das dazugehörige Devel-Paket müssen installiert sein). Der SDL-Audiotreiber hilft in vielen Fällen und unterstützt auch ESD (GNOME) und ARTS (KDE).
- * Wenn du ALSA Version 0.5 verwendest, musst du fast immer `-ao alsa5` verwenden, weil ALSA 0.5 einen fehlerhaften OSS- Emulationscode hat. MPlayer wird mit etwa so einer Meldung **abstürzen**:
`DEMUXER: Too many (945 in 8390980 bytes) video packets in the buffer!`
- * Unter Solaris solltest du den SUN-Audiotreiber mittels `-ao sun` benutzen, da ansonsten weder Video noch Audio funktionieren.
- * Wenn beim Ton Klickgeräusche beim Abspielen vom CD-ROM-Laufwerk zu hören sind, dann schalte "IRQ unmasking" an, z.B. mit `hdparm -u1 /dev/cdrom` (man `hdparm`). Das ist auch generell von Vorteil, wie du in der [CD-ROM-Sektion](#) nachlesen kannst.

6.3.2.2 Soundkarten-Erfahrungen, -Empfehlungen

Unter Linux wird ein 2.4.x Kernel sehr empfohlen. Kernel 2.2 wird nicht getestet.

Linux-Soundtreiber werden hauptsächlich von der freien Version von OSS bereitgestellt. Diese Treiber werden inzwischen von denen des [ALSA-Projektes](#) abgelöst (Advanced Linux Sound Architecture), die bereits im Entwicklerbaum des Kernels (2.5) zu finden sind. Wenn deine Distribution nicht bereits ALSA verwendet und du Probleme mit dem Sound hast, dann solltest du die ALSA-Treiber mal ausprobieren. Die ALSA-Treiber sind den OSS-Treibern meistens in Sachen Kompatibilität, Performance und Features überlegen. Leider werden einige Soundkarten dagegen nur von OSS-Treibern von [4Front Technologies](#) unterstützt. Sie unterstützen ebenfalls einige Nicht-Linux-Systeme.

SOUNDKARTE TREIBER

	OSS/Free	ALSA	OSS/Pro	andere	max kHz	max Anz. Kanäle	max Anz. Zugriffe
VIA onboard (686/A/B, 8233, via82cxxx_audio)	via82cxxx_audio	snd-via82xx			4-48 kHz		oderr nur 48

					kHz, abhängig vom Chipsatz		
8235)							
Aureal Vortex 2	keine	keine	OK	 Linux Aureal-Treiber buffer size increased to 32k	48	4.1	5+
SB Live!	Analog OK, SP/DIF funktioniert nicht	beide OK	beide OK	 Creatives OSS-Treiber (mit SP/DIF)	192	4.0/5.1	32
SB 128 PCI (es1371)	OK	?			48	stereo	2
SB AWE 64	max 44kHz	48kHz klingt schlecht			48		
GUS PnP	keine	OK	OK		48		
Gravis	nicht OK	OK			44		
UltraSound ACE							
Gravis	OK	OK (?)			48		
UltraSound MAX							
ESS 688	OK	OK (?)			48		
C-Media-Karten (welche genau?)	nicht OK (zischt) (?)	OK			?		
Yamaha-Karten (*ymf*)	nicht OK (?) (vielleicht mit <CODE>-ao sdl</CODE>)	nur mit der OSS-Emulation von ALSA 0.5 OK UND -ao sdl (!) (?)			?		
Karten mit envy24-Chips (wie z.b. Terratec EWS88MT)	?	?	OK		?		
DAC oder PC-Lautsprecher	OK	none		 Der Treiber mono emuliert für den PC - 44.1, Lautsprecher vielleicht auch mehr.			1

Die Anzahl der Anwendungen, die **zur gleichen Zeit** auf das Gerät zugreifen können.

Rückmeldungen zu diesem Dokument sind willkommen. Bitte sag uns, wie MPlayer und deine Soundkarte(n) zusammen funktionieren.

6.3.2.3 Audiofilter

Die alten Audioplugins wurden von einer neuen Filterschicht abgelöst. Audiofilter werden benutzt, um die Parameter der Daten zu ändern, bevor sie die Soundkarte erreichen. Die Aktivierung der Filter geschieht normalerweise automatisch, kann aber auch erzwungen oder verhindert werden. Die Filter werden aktiviert, wenn die Parameter der Daten von denen der Soundkarte abweichen und deaktiviert, wenn sie gleich sind. Die Option `-af filter1,filter2,...` wird benutzt, um Filter einzufügen, die nicht automatisch eingefügt werden. Diese Filter werden in der Reihenfolge abgearbeitet, wie sie angegeben wurden.

Beispiele:

```
user@linux ~/ $ mplayer -af resample,pan movie.avi
```

Schickt die Audiodaten durch das Resample-Filter und danach durch den Pan-Filter. Die Liste darf keine Leerzeichen enthalten, sonst klappt's nicht.

Viele Filter kennen Parameter, mit denen ihr Verhalten geändert werden kann. Diese Parameter werden unten detailliert beschrieben. Wenn keine Parameter angegeben werden, dann verwendet der Filter seine Standardwerte. Hier ist ein Beispiel, wie Filter mit Parametern verwendet werden können:

```
user@linux ~/ $ mplayer -af resample=11025,pan=1:0.5:0.5 -channels 1
-srate 11025 media.avi
```

Das setzt die Ausgabefrequenz des Resampleplugins auf 11025Hz und downmixt das Audio auf einen Kanal mit dem Pan-Filter.

Die allgemeine Kontrolle der Filterschicht geschieht über `-af-adv`. Diese Option kennt zwei Parameter:

`force` ist ein Bitfeld, das angibt, wie die Filter einzufügen sind und welche Geschwindigkeits-/Genauigkeitsoptimierungen sie verwenden:

- | | |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | Benutzt automatisches Einfügen und optimiert in Abhängigkeit der CPU-Geschwindigkeit. |
| 1 | Benutzt automatisches Einfügen und optimiert für die höchstmögliche Geschwindigkeit.
Warnung: Einige Features der Audiofilter werden kommentarlos deaktiviert, und die Soundqualität könnte niedriger sein als erwartet. |
| 2 | Benutzt automatisches Einfügen der Filter und optimiert für Qualität. |
| 3 | Fügt nicht automatisch Filter ein.
Warnung: Hiermit kann MPlayer eventuell zum Absturz gebracht werden. |
| 4 | Fügt automatisch Filter so wie bei 0 hinzu, benutzt aber Fließkommaarithmetik, wo dies möglich ist. |
| 5 | Fügt automatisch Filter so wie bei 1 hinzu, benutzt aber Fließkommaarithmetik, wo dies möglich ist. |
| 6 | Fügt automatisch Filter so wie bei 2 hinzu, benutzt aber Fließkommaarithmetik, wo dies möglich ist. |
| 7 | Fügt keine Filter automatisch hinzu, benutzt aber Fließkommaarithmetik, wo dies möglich ist. |

`list` ist ein Synonym für `-af`.

Die Filterschicht wird auch von den folgenden allgemeinen Optionen beeinflusst:

<code>-v</code>	Gibt ausführlichere Meldungen aus. Die meisten Filter geben hiermit auch mehr Statusinformationen aus.
<code>-channels</code>	Diese Option setzt die Anzahl der Ausgabekanäle, die du von deiner Soundkarte ausgegeben haben möchtest. Sie beeinflusst auch die Anzahl der Kanäle, die vom Quellstream decodiert werden. Wenn der Stream weniger als die hiermit geforderte Anzahl Kanäle enthält, so werden automatisch Filter eingefügt (siehe unten). Das Routing ist das Standardrouting des channels-Filters.
<code>-srate</code>	Diese Option gibt die Samplerate an, die die Soundkarte benutzen soll, sofern sie diese Rate unterstützt. Wenn die Samplefrequenz deiner Soundkarte anders ist als die des aktuellen Quellstreams, so wird der resample-Filter (siehe unten) automatisch eingefügt, um diese Differenz auszugleichen.
<code>-format</code>	Diese Option setzt das Sampleformat zwischen Audiofilterschicht und der Soundkarte. Wenn das angeforderte Format der Soundkarte nicht das gleiche wie das des Quellstreams ist, dann wird der format-Filter (siehe unten) automatisch eingefügt, um das Format zu konvertieren.

6.3.2.3.1 Up/Down-sampling

MPlayer unterstützt vollständig Up-/Downsampling mit dem resample-Filter. Dieser Filter kann benutzt werden, wenn du eine Soundkarte hast, die nur eine feste Frequenz unterstützt, oder die nur maximal 44.1kHz unterstützt. Dieser Filter wird automatisch aktiviert, wenn er benötigt wird. Er kann aber auch explizit auf der Kommandozeile aktiviert werden. Er kennt drei Optionen:

<code>srate <8000-192000></code>	wird benutzt, um die Ausgabesamplefrequenz in Hz anzugeben. Der gültige Bereich liegt zwischen 8kHz und 192kHz. Wenn Eingangs- und Ausgabefrequenz gleich sind, oder wenn dieser Parameter nicht angegeben wurde, so wird der Filter wieder entladen. Hohe Samplefrequenzen verbessern normalerweise die Audioqualität, vor allem, wenn auch andere Filter benutzt werden.
<code>sloppy</code>	Mit dieser Option wird zugelassen, dass die Ausgabefrequenz leicht von der geforderten Frequenz (<code>srate</code>) abweicht. Diese Option kann benutzt werden, wenn das Playback nur sehr langsam startet. Diese Option ist standardmäßig aktiviert.
<code>type <0-2></code>	ist ein optional anzugebender Integer zwischen 0 und 2, der den zu benutzenden Resamplealgorithmus auswählt. Hier wählt 0 lineare Interpolation und 1 mehrphasige Filterbänke mit Ganzzahlarithmetik. 2 wählt mehrphasige Filterbänke mit Fließkommaarithmetik. Lineare Interpolation ist sehr schnell, liefert dafür aber auch schlechte Qualität, vor allem, wenn sie benutzt wird, um die Samplerate zu erhöhen. Die beste Qualität liefert 2. Dafür benötigt sie aber auch die meiste CPU-Zeit.

Beispiele:

```
user@linux ~/ $ mplayer -af resample=44100:0:0
```

Dies setzt die Ausgabefrequenz des Resample-Filters auf exakt 44100Hz mit linearer Interpolation.

6.3.2.3.2 Ändern der Anzahl der Kanäle

Der `channels`-Filter kann benutzt werden, um Kanäle hinzuzufügen oder zu entfernen. Er kann auch benutzt werden, um Kanäle umzuleiten oder sie zu kopieren. Er wird automatisch aktiviert, falls die Anzahl der Kanäle bei Eingang und Ausgang der Audiofilterschicht unterschiedlich ist, oder wenn ein anderer Filter diesen Filter benötigt. Wenn er nicht gebraucht wird, so entlädt sich dieser Filter automatisch. Die Anzahl der Parameter ist dynamisch:

`nch <1-6>` ist eine ganze Zahl zwischen 1 und 6, die die Anzahl der Ausgabekanäle festlegt. Dieser Parameter wird benötigt. Wenn er weggelassen wird, dann erhält man einen Laufzeitfehler.

`nr <1-6>` ist eine ganze Zahl zwischen 1 und 6, die die Anzahl der Umleitungen festlegt. Dieser Parameter ist optional. Wenn er weggelassen wird, dann werden keine Umleitungen vorgenommen.

`von1:nach1:von2:nach2...` sind paare von Nummern zwischen 0 und 5, die festlegen, wohin jeder Kanal umgeleitet werden soll.

Wenn nur `nch` angegeben wird, dann werden die Standardumleitungen benutzt, die so aussehen: Wenn mehr Ausgabekanäle gefordert sind als Eingangskanäle existieren, so werden leere Kanäle erzeugt (bis auf's Mixen von Mono auf Stereo: hier wird der Monokanal auf beide Kanäle kopiert). Ist die Anzahl dagegen kleiner, so werden die überschüssigen Kanäle entfernt.

Beispiel 1:

```
user@linux ~/ $ mplayer -af channels=4:4:0:1:1:0:2:2:3:3 media.avi
```

ändert die Anzahl der Kanäle auf 4 und legt vier Routen fest, die die Kanäle 0 und 1 vertauschen und die Kanäle 2 und 3 intakt lassen. Wenn die Quelldatei nur zwei Kanäle hat, so sind die Kanäle 2 und 3 leise, aber 0 und 1 werden trotzdem vertauscht.

Beispiel 2:

```
user@linux ~/ $ mplayer -af channels=6:4:0:0:0:1:0:2:0:3 media.avi
```

ändert die Anzahl der Kanäle auf 6 und gibt 4 Routen an, die Kanal 0 auf Kanal 0 und 3 kopieren. Die Kanäle 4 und 5 bleiben stumm.

6.3.2.3.3 Formatkonvertierung

Der `format`-Filter konvertiert zwischen verschiedenen Sampleformaten. Er wird automatisch aktiviert, wenn er von der Soundkarte oder einem anderen Filter benötigt wird.

`bps <Nummer>` kann 1, 2 oder 4 sein und gibt die Anzahl der Bytes pro Sample an. Dieser Parameter wird benötigt und führt zu einem Laufzeitfehler, wenn er weggelassen wird.

`f <format>` ist ein Textstring, der das Sampleformat beschreibt. Der String besteht aus einem Mix der folgenden Schlüsselwörter: `alaw`, `mulaw` oder `imaadpcm`, `float` oder `int`, `unsigned` oder `signed`, `le` oder `be` (little oder big endian). Dieser Parameter wird benötigt und führt zu einem

Laufzeitfehler, wenn er weggelassen wird.

Beispiel:

```
user@linux ~/ $ mplayer -af format=4:float media.avi
```

setzt das Ausgabeformat auf 4 Bytes pro Sample Fließkommandaten

6.3.2.3.4 Verzögerung - Audiofilter

Der `delay`-Filter verzögert den Sound auf dem Weg zum Lautsprecher, damit die verschiedenen Kanäle zur gleichen Zeit bei der Sitzposition ankommen. Dieser Filter ist nur dann nützlich, wenn du mehr als zwei Lautsprecher hast. Dieser Filter erhält eine variable Anzahl Parameter:

`d1:d2:d3...` sind Fließkommazahlen, die die Verzögerung in ms für den jeweiligen Kanal angeben. Minimum ist 0ms, das Maximum 1000ms.

Um die benötigte Verzögerung für die verschiedenen Kanäle zu berechnen, tu folgendes:

1. Miss die Entfernung zwischen den Lautsprechern und deiner Sitzposition in Metern. Dies gibt dir die Distanzen `s1` bis `s5` (bei einem 5.1-System). Es lohnt sich nicht, auch für den Subwoofer zu kompensieren, weil man den Unterschied eh nicht hört.
2. Subtrahiere die Distanzen `s1` bis `s5` von der maximalen Distanz, z.B.
 $s[i] = \max(s) - s[i]; i = 1...5$
3. Berechne die benötigte Verzögerung wie folgt:
 $d[i] = 1000s[i]/342; i = 1...5$

Beispiel:

```
user@linux ~/ $ mplayer -af delay=10.5:10.5:0:0:7:0 media.avi
```

verzögert die Kanäle vorne links und vorne rechts um 10.5ms, die beiden hinteren Kanäle um 0ms und den zentrierten Kanal um 7ms.

6.3.2.3.5 Lautstärkekontrolle mit Software

Lautstärkeregelung in Software wird mit dem `volume`-Filter realisiert. Sei bei der Benutzung dieses Filters vorsichtig, da er den Signal-zu-Rausch-Abstand verringern kann. In den meisten Fällen ist es besser, beim Mixer deiner Soundkarte den PCM-Regler auf's Maximum zu setzen und diesen Filter wegzulassen. ändere dann die Lautstärke mit dem Hauptregler des Mixers. Wenn deine Soundkarte einen digitalen PCM-Mixer anstelle eines analogen besitzt und du Verzerrungen hörst, dann benutze statt dessen den MASTER-Mixer. Wenn der Computer an einen externen Verstärker angeschlossen ist (was meistens der Fall ist), dann kann der Rauschpegel minimiert werden, indem der Hauptregler und der Lautstärkereglere des Verstärkers angepasst werden, bis das Hintergrundzischen verschwunden ist. Der Filter selber kennt zwei Parameter:

`v <-200 - +60>` ist eine Fließkommazahl zwischen `-200` und `+60` und gibt die Lautstärke in dB an. Der Standardlevel ist 0dB. `c` ist ein Binärwert, der soft clipping an- oder ausschaltet. Soft clipping kann den Sound glätten, wenn sehr laute Passagen auftreten. Aktivier diese Option, wenn die Dynamikbandbreite

deiner Lautsprecher gering ist. Sei gewarnt, dass dieses Feature Verzerrungen hervorruft. Es sollte nur als letzte Möglichkeit benutzt werden.

Beispiel:

```
user@linux ~/ $ mplayer -af volume=10.1:0 media.avi
```

verstärkt den Sound um 10.1dB und schneidet hart ab, wenn die Lautstärke zu hoch wird.

Dieser Filter besitzt noch ein zweites Feature: Er misst die maximale Lautstärke und gibt sie aus, wenn MPlayer beendet wird. Dieser Wert kann benutzt werden, um die Verstärkung in MEncoder so zu setzen, dass die Dynamikbandbreite vollständig ausgenutzt wird.

6.3.2.3.6 Equalizer

Der `equalizer`-Filter ist ein 10-Band Graphicequalizer, der durch 10 IIR-Bandpassfilter (infinite impulse response, rekursive Filter) realisiert wird. Das bedeutet, dass er unabhängig vom wiederzugebenden Audiotyp funktioniert. Die Mittenfrequenzen der zehn Bändern sind:

Band Nr.	Mittenfrequenz
0	31.25 Hz
1	62.50 Hz
2	125.0 Hz
3	250.0 Hz
4	500.0 Hz
5	1.000 kHz
6	2.000 kHz
7	4.000 kHz
8	8.000 kHz
9	16.000 kHz

Wenn die Samplefrequenz des Sounds niedriger als die Mittenfrequenz eines Bandes ist, so wird dieses Band deaktiviert. Ein bekannter Fehler dieses Filters liegt darin, dass die Charakteristiken des höchsten Bandes nicht ganz symmetrisch sind, wenn die Samplefrequenz nahe der Mittenfrequenz des Bandes liegt. Das kann umgangen werden, wenn vorher ein Upsampling mit dem `resample`-Filter vorgenommen wird.

Dieser Filter bekommt zehn Parameter:

`g1:g2:g3...g10` sind Fließkommazahlen zwischen `-12` und `+12`, die die Verstärkung in dB des jeweiligen Bandes angeben.

Beispiel:

```
user@linux ~/ $ mplayer -af equalizer=11:11:10:5:0:-12:0:5:12:12
media.avi
```

verstärkt den Sound in den oberen und unteren Frequenzbereichen und löscht ihn um 1kHz beinahe komplett

aus.

6.3.2.3.7 Panning-Filter

Der `pan`-Filter kann benutzt werden, um Kanäle beliebig zu mischen. Er ist grundlegend nur eine Kombination des Lautstärkefilters und des `channels`-Filters. Es gibt zwei Haupteinsatzgebiete für diesen Filter:

1. Heruntermischen vieler Kanäle auf wenige, z.B. Stereo zu Mono.
2. Verändern der **Breite** des zentrierten Kanals bei Sourround-Sound-Systemem.

Dieser Filter ist nicht ganz leicht zu bedienen und benötigt viel Ausprobieren, bevor die gewünschten Ergebnisse erzielt werden. Die Anzahl der Parameter hängt von der Anzahl der Ausgabekanäle ab:

`nch <1-6>` ist eine ganze Zahl zwischen 1 und 6 und wird benutzt, um die Anzahl der Ausgabekanäle zu setzen. Dieser Parameter wird benötigt. Wenn er weggelassen wird, erhält man einen Laufzeitfehler.

`100:101:102:...` sind Fließkommazahlen zwischen 0 und 1. `1[[i]][j]` gibt an, wieviel vom Eingangskanal `j` in den Ausgabekanal `i` zu mischen ist.

Beispiel 1:

```
user@linux ~/ $ mplayer -af pan=1:0.5:0.5 -channels 1 media.avi
```

mischt stereo auf mono.

Beispiel 2:

```
user@linux ~/ $ mplayer -af pan=3:1:0:1:0.5:0.5 -channels 3 media.avi
```

gibt drei Kanäle aus, lässt dabei Kanäle 0 und 1 intakt und mischt die Quellkanäle 0 und 1 in den Ausgabekanal 2 (welcher z.B. an einen Sub-woofer geschickt werden kann).

6.3.2.4 Sub-woofer

Der `sub`-Filter fügt dem Audiostream einen Sub-woofer-Kanal hinzu. Die dazu verwendeten Audiodaten sind der Durchschnitt der Kanäle 0 und 1. Der resultierende Sound wird dann mit einem Butterworth-Filter vierten Grades tiefpassgefiltert. Der Filter schneidet bei 60Hz ab.

Warnung:

deaktivier diesen Filter, wenn du DVDs mit Dolby Digital 5.1-Sound abspielst, weil dieser Filter ansonsten den bereits vorhandenen Sub-woofer-Kanal stört. Dieser Filter bekommt zwei Parameter:

`fc <20-300>` ist eine optionale Fließkommazahl, die die Frequenz in Hz angibt, bei der Tiefpass abschneidet. Gültig sind Werte zwischen 20 und 300. Das beste Ergebnis erhältst du, wenn du die Frequenz so niedrig wie möglich setzt. Das wird das Stereo- und Surround-Sound-Erlebnis verbessern. Der Standardwert für die Frequenzgrenze liegt bei 60Hz.

`ch <0-5>` ist eine optionale ganze Zahl zwischen 0 und 5, die angibt, auf welchem Kanal der Sub-woofer-Kanal auszugeben ist. Der Standardkanal ist 5. Die gesamte Anzahl an Kanälen wird auf `ch` erhöht, falls dies notwendig sein

sollte.

Beispiel:

```
user@linux ~/ $ mplayer -af sub=100:4 -channels 5 media.avi
```

fügt einen Sub-woofer-Kanal mit Grenzfrequenz von 100Hz hinzu und gibt ihn auf Kanal 4 aus.

6.3.2.4.1 Surround-sound-Decoder

Matrix encodierter Surround-Sound kann mit dem `surround`-Filter decodiert werden. Dolby Surround ist ein Beispiel für ein matrix encodiertes Format. Viele Dateien mit Zweikanalaudio enthalten tatsächlich matrix encodierten Surround-Sound. Um diesen Filter zu benutzen, brauchst du eine Soundkarte, die mindestens vier Kanäle unterstützt. Dieser Filter kennt einen Parameter:

`d <0-1000>` ist eine optionale Fließkommazahl zwischen 0 und 1000, die die Verzögerung in ms für den hinteren Lautsprecher angibt. Die Verzögerung sollte wie folgt gewählt werden: Wenn `d1` die Entfernung zwischen der Sitzposition und den vorderen Lautsprechern ist, und wenn `d2` die Entfernung zwischen der Sitzposition und den hinteren Lautsprechern ist, dann sollte `d` auf 15ms gesetzt werden, wenn $d1 \leq d2$ ist, und auf $15 + 5(d1-d2)$, wenn $d1 > d2$ ist. Der Standardwert `d` ist 20ms.

Beispiel:

```
user@linux ~/ $ mplayer -af surround=15 -channels 4 media.avi
```

decodiert Surround-Sound mit einer Verzögerung von 15ms für die hinteren Lautsprecher.

6.3.2.5 Audio-Plugins (veraltet)

Anmerkung:

Audioplugins sind veraltet und wurden durch Audiofilter ersetzt. Sie werden bald entfernt werden.

MPlayer bietet Unterstützung für Audio-Plugins. Audio-Plugins können benutzt werden, um die Einstellungen der Audio-Daten zu ändern, bevor sie die Soundkarte erreichen. Sie aktiviert durch die Option `-aop` aktiviert, welche `list=plugin1,plugin2,...` als Argument hat. Das `list` wird benötigt, um festzulegen, in welcher Reihenfolge die Plugins benutzt werden sollen.

Beispiel:

```
user@linux ~/ $ mplayer media.avi -aop list=resample,format
```

würde den Ton durch das Resampling-Plugin laufen lassen, gefolgt vom Format-Plugins.

Die Plugins können auch Schalter haben, welche ihr Verhalten ändern. Diese Schalter werden im unteren Abschnitt im Detail beschrieben. Ein Plugin wird mit den Standardeinstellungen gestartet, wenn dem Plugin

Optionen mitgegeben werden. Hier ist ein Beispiel, wie man Plugins in Kombination mit pluginspezifischen Schaltern verwendet:

```
user@linux ~/ $ mplayer media.avi -aop
list=resample,format:fout=44100:format=0x8
```

würde die Ausgabefrequenz des Resample-Plugins auf 44100Hz and das Ausgabeformat des Format-Plugins auf AFMT_U8 stellen.

Momentan können Audio-Plugins nicht im MEncoder verwendet werden.

6.3.2.5.1 Up/Downsampling

MPlayer unterstützt vollständiges up/downsampling von Audiostreams. Dieses Plugin kann verwendet werden, wenn man eine Soundkarte mit fester Frequenz besitzt, oder wenn man eine alte Soundkarte am Hals hat, welche nur max 44.1kHz verarbeitet. Ob die Verwendung dieses Plugins benötigt wird oder nicht erkennt MPlayer **automatisch**. Dieses Plugin hat einen Schalter, `fout`, welcher benutzt wird, um die gewünschte Ausgabefrequenz zu setzen. Der Standard ist 48kHz und wird in `<Hz>` angegeben.

Verwendung:

```
user@linux ~/ $ mplayer medium.avi -aop list=resample:fout=<erforderliche
Frequenz in Hz, z.B. 44100>
```

Hinweis:

Die Ausgabefrequenz sollte nicht vom Standard-Wert hochskaliert werden. Die Skalierung führt dazu, dass der Audio- und Video-Stream in Zeitlupe wiedergegeben werden, und man erhält zusätzlich Audioverzerrungen.

6.3.2.5.2 Surround-Sound-Decodierung

MPlayer hat ein Audio-Plugin das matrix-codierten Surround Sound decodieren kann. Dolby Surround ist ein Beispiel eines matrix-codierten Formats. Viele Dateien mit zwei Audiokanälen enthalten tatsächlich matrix Surround Sound. Um dieses Feature zu benutzen, brauchst du eine Soundkarte mit mindestens vier Kanälen.

Verwendung:

```
user@linux ~/ $ mplayer medium.avi -aop list=surround
```

6.3.2.5.3 Sample-Format-Konvertierer

Wenn deine Soundkarte keinen signed 16bit `int` Datentyp unterstützt, kann dieses Plugin benutzt werden, um das Format in eines zu ändern, das deine Soundkarte versteht. Es hat nur einen Schalter `format`, welcher auf eine Nummer gestellt werden kann, die man der Datei `libao2/afmt.h` findet. Dieses Plugin wird praktisch nie gebraucht und ist für erfahrene Anwender gedacht. Bedenke, dass dieses Plugin nur das Sampleformat und nicht die Samplefrequenz oder die Anzahl Kanäle ändert.

Verwendung:

```
user@linux ~/ $ mplayer medium.avi -aop list=format:format=<required
output format>
```

6.3.2.5.4 Verzögerung - Audio-Plugins

Dieses Plugin verzögert den Ton und ist als Beispiel gedacht, wie man neue Plugins schreibt. Es kann für nichts sinnvolles aus der Anwenderperspektive verwendet werden und ist hier nur aus Gründen der Vollständigkeit erwähnt. Verwende dieses Plugin nicht, wenn du kein Entwickler bist.

Wenn du eine Datei mit einer konstanten A/V-Sync-Differenz hast, dann benutze einfach die Tasten + und -, um das Timing während des Abspielens anzupassen. Wenn das OSD aktiviert ist, dann werden die aktuellen Werte angezeigt, was die Anpassung erleichtert.

6.3.2.5.5 Software Lautstärkeregelung

Dieses Plugin ist ein Software-Ersatz für die Lautstärkeregelung und kann in Maschinen mit kaputtem Mixer-Device verwendet werden. Es kann auch verwendet werden, um die Ausgabelautstärke von MPlayer zu ändern, ohne die Einstellung der PCM-Lautstärke im Mixer zu ändern. Es gibt einen Schalter `volume`, welcher verwendet wird, um den anfänglichen Schallpegel festzulegen. Der anfängliche Schallpegel kann auf Werte zwischen 0 und 255 festgelegt werden (normal auf 101, was 0db Verstärkung entspricht). Benutze dieses Plugin mit Vorsicht, da es den Signal/Rauschabstand verringern kann. In den meisten Fällen ist es das beste, den Regler für PCM auf das Maximum zu stellen, dieses Plugin nicht zu verwenden und die Lautstärke zu deinen Boxen mit der Master-Lautstärkeregelung zu kontrollieren. Wenn deine Soundkarte einen digitalen PCM-Mixer anstelle eines analogen verwendet und du Verzerrungen hörst, so verwende statt dessen den MASTER-Mixer. Wenn ein externer Verstärker mit dem Computer verbunden ist (was fast immer der Fall ist), kann der Geräuschpegel durch die Anpassung des Master-Stufe und der Lautstärkeknöpfe des Verstärkers minimiert werden, bis das Hintergrundrauschen verschwindet.

Verwendung:

```
user@linux ~/ $ mplayer media.avi -aop list=volume:volume=<0-255>
```

Dieses Plugin besitzt auch einen Kompressor oder "soft-clipping"- Fähigkeiten. Die Kompression kann benutzt werden, wenn der Dynamikbereich des Sounds sehr hoch ist, oder wenn der Dynamikbereich der Lautsprecher sehr niedrig ist. Sei dir bewusst, dass dieses Feature Verzerrungen verursacht. Es sollte erst als letzte Möglichkeit in Betracht gezogen werden.

Verwendung:

```
user@linux / $ mplayer media.avi -aop list=volume:softclip
```

6.3.2.5.6 Extrastereo

Dieses Plugin erhöht linear die Differenz zwischen dem linken und rechten Kanal (wie das XMMS Extrastereo-Plugin), was zum Teil zu "live"- Effekten bei der Wiedergabe führt.

Verwendung:

```
user@linux ~/ $ mplayer media.avi -aop list=extrastereo
```

```
user@linux / $ mplayer media.avi -aop list=extrastereo:mul=3.45
```

Der Koeffizient (`mul`) ist eine Fließkommazahl, welche standardmäßig auf 2.5 gestellt ist. Wenn man den Wert auf 0.0 setzt, erhält man einen Mono-Ton (Durchschnitt beider Kanäle). Wenn man den Wert auf 1.0 setzt, bleibt der Ton gleich. Wenn man den Wert auf -1.0 setzt, werden der linke und rechte Kanal vertauscht.

6.3.2.5.7 Lautstärkenormalisierer

Dieses Plugin maximiert die Lautstärke, ohne den Ton zu verzerren.

Verwendung:

```
user@linux ~/ $ mplayer media.avi -aop list=volnorm
```

6.4 TV input

Dieser Teil der Docs zeigt, wie man mit einem V4L-kompatiblen Gerät (z.B. einer TV-Karte) **fernsehen oder Sendungen aufnehmen** kann. In der man-page findest du eine Beschreibung aller TV-Optionen sowie die Belegung der Tastatur für's Fernsehen.

6.4.1 Kompilierung für TV input

1. Als erstes musst du MPlayer neu kompilieren. `./configure` erkennt automatisch die Kernel-Header von v4l und ob die `/dev/video`-Einträge existieren. Wenn sie existieren, wird die TV-Unterstützung eingebaut (siehe Ausgabe von `configure`).
2. Stelle sicher, dass dein Tuner mit anderer TV-Software unter Linux funktioniert, zum Beispiel xawtv.

6.4.2 Tipps zur Benutzung

Die vollständige Auflistung der vorhandenen Optionen ist in der man-page verfügbar. Hier nur ein paar Tipps:

- * Benutze die Option `channels`. Ein Beispiel: `-tv on:channels=26-MTV1,23-TV2` Erklärung: Mit dieser Option sind nur die Kanäle 26 und 23 anwählbar, und das OSD zeigt eine nette Meldung beim Kanalwechsel an, nämlich den Namen des Kanals. Leerzeichen im Namen müssen durch "_" ersetzt werden.
- * Wähle vernünftige Bildabmessungen. Höhe und Breite des resultierenden Bildes (nach Anwendungen aller Filter) sollten durch 16 teilbar sein.
- * Wenn du Video mit einer Höhe aufnimmst, die mehr als die Hälfte der Fernsehauflösung beträgt (z.B. 288 für PAL oder 240 für NTSC), dann schalte Deinterlacing an. Andernfalls wirst du Video bekommen, das bei Szenen mit viel vertikaler Bewegung sehr viele Artefakte aufweist. Zusätzlich wird der zur gerechten Verteilung der Bitrate verwendete Algorithmus damit wahrscheinlich nicht zurecht kommen, da Interlacing-Artefakte sehr viele Details erzeugen und der Codec somit eine Unmenge an Bandbreite für die Encodierung benötigt. Du kannst Deinterlacing mit der Option `-vop pp=DEINT_TYPE` anschalten. Normalerweise funktioniert `pp=lb` sehr gut, aber das hängt auch vom persönlichen Geschmack ab. Schau in der man-page nach. Dort findest du eine Beschreibung der einzelnen Deinterlacing-Algorithmen. Probier sie einfach aus.
- * Schneide nicht benutze Bildteile weg. Wenn du Video aufnimmst, dann hast du an den Rändern oftmals schwarze Bereiche oder Bereiche, in denen viele Bildstörungen auftreten. Diese benötigen wiederum eine sehr große Bandbreite. Genauer gesagt sind es nicht die schwarzen Bereiche, die soviel Bandbreite

- benötigen, sondern der Übergang zwischen Bild und schwarzem Bereich, sprich die Kanten. Bevor du also mit der Aufnahme anfängst, solltest du die Argumente des `crop`-Filters so anpassen, dass all der Müll an den Rändern weggeschnitten wird. Vergiss aber wiederum nicht, die Bildabmessungen sinnvoll zu wählen.
- * Achte auf die CPU-Auslastung. Sie sollte die 90%-Barriere die meiste Zeit über nicht überschreiten. Wenn du große Aufnahmebuffer verwendest, dann kann MEncoder auch ein paar Sekunden mit höherer Auslastung überleben, aber auch nicht mehr. Somit solltest du besser deine 3D-OpenGL-Bildschirmschoner und allgemein alles andere, was viel Performance kostet, vor der Aufnahme ausschalten.
 - * Spiel nicht mit der Systemuhr herum. MEncoder benutzt die Systemuhr, um Audio und Video synchron zu halten. Wenn du die Uhrzeit veränderst (besonders, wenn du sie zurücksetzt), dann wird das MEncoder verwirren, und er wird Bilder wegwerfen. Das ist besonders dann wichtig, wenn dein Rechner die Systemzeit automatisch über das Netzwerk anpasst, z.B. mit einer Software wie NTP. Schalte also NTP während der Aufnahme aus, wenn du verlässliche Resultate erzielen willst.
 - * ändere das `outfmt` nicht, es sei denn, du weißt wirklich, was du da tust, oder wenn deine Karte/dein Treiber wirklich das Standardausgabeformat nicht beherrscht (YV12-Farbraum). In älteren Versionen von MPlayer/MEncoder musste das `outfmt` immer mit angegeben werden, aber heute nicht mehr, und die Standardwerte passen eh in den meisten Fällen. Wenn du z.B. nach mit DivX aufnehmen willst, dafür die `libavcodec` benutzt und `outfmt=RGB24` angibst, um die Bildqualität zu verbessern, so wird das aufgenommene Bild eh später wieder in den YV12-Farbraum zurückkonvertiert. Das einzige, was du somit erreichst, ist die Verschwendung von CPU-Zeit.
 - * Wenn du den I420-Farbraum verwenden möchtest (`outfmt=i420`), dann musst du auch die Option `-vc rawi420` angeben, weil es ansonsten einen Konflikt mit der FourCC des Intel Indeo-Videocodecs gibt.
 - * Es gibt mehrere Möglichkeiten, den Ton aufzunehmen. Zum einen kannst du den Ton mit der Soundkarte aufnehmen und die TV-Karte mit einem Kabel an die Soundkarte anschließen. Zum anderen kannst du den im bt878-Chip eingebauten ADC verwenden. Im zweiten Fall musst du das `btaudio`-Kernelmodul verwenden. Lies die Datei `linux/Documentation/sound/btaudio`, die in den Kernelquellen zu finden ist. Sie enthält Informationen darüber, wie du diesen Treiber benutzt.
 - * Falls MEncoder das Audiogerät nicht öffnen kann, dann geh sicher, dass es auch wirklich verfügbar ist. Probleme machen vor allem die Soundserver wie z.B. arts vom KDE oder esd vom GNOME. Wenn du eine Full-Duplex-Soundkarte hast (und das ist heutzutage praktisch jede Karte) und KDE verwendest, dann versuche, die Option "full duplex" im Optionsmenü des KDE-Soundservers zu aktivieren.

6.4.3 Beispiele

Dummy-Ausgabe mittels AALib ;)

```
user@linux ~/ $ mplayer -tv on:driver=dummy:width=640:height=480 -vo aa
```

Standard-V4L-Gerät als Quelle

```
user@linux ~/ $ mplayer -tv on:driver=v4l:width=640:height=480 -vo xv
```

Hier ein sinnvolleres und umfassenderes Beispiel. MEncoder wird mit voller PAL-Größe aufnehmen, das Bild an den Rändern beschneiden und mit einem linear-blend-Algorithmus ein Deinterlacing vornehmen. Audio wird mit einer Bitrate von 64kbps mit dem LAME-Codec in MP3 komprimiert. Diese Einstellungen sind dazu geeignet, um Filme aufzunehmen.

```
user@linux ~/ $ mencoder -tv on:driver=v4l:width=768:height=576 -ovc lavc  
-lavcopts vcodec=mpeg4:vbitrate=900 -oac mp3lame -lameopts cbr:br=64 -vop  
pp=lb,crop=720:544:24:18 -o output.avi
```

In diesem Beispiel wird das Bild noch auf 384x288 verkleinert und mit einer Bitrate von nur 350kbps im Qualitätsmodus komprimiert. Die `vqmax`-Option lockert die Beschränkungen der Quantizer und erlaubt es dem Codec erst, wirklich so eine niedrige Bitrate zu erreichen, was natürlich auf Kosten der Bildqualität geschieht. Dieses Beispiel kann verwendet werden, um z.B. lange TV-Serien aufzuzeichnen, wobei es auf die Qualität nicht so sehr wie auf den Platzbedarf ankommt.

```
user@linux ~/ $ mencoder -tv on:driver=v4l:width=768:height=576 -ovc lavc
-lavcopts vcodec=mpeg4:vbitrate=350:vhq:vqmax=31:keyint=300 -oac mp3lame
-lameopts cbr:br=48 -vop scale=384:288,pp=tn/lb,crop=720:540:24:18 -sws 1
-o output.avi
```

Du könntest auch in der `-tv`-Option eine kleinere Bildgröße angeben und dafür auf die Softwareskalierung verzichten. Aber der oben gewählte Ansatz gibt MEncoder die maximal zur Verfügung stehende Information zur Verarbeitung und ist ein bisschen resistenter gegen Bildrauschen. Dazu kommt, dass der bt8x8-Chip wegen einer Hardwarebeschränkung nur in der Horizontale zwischen Pixeln interpolieren kann.

6.5 Edit Decision Lists (EDL)

Die Edit Decision Lists (EDL) erlauben es, bestimmte Stellen bei der Wiedergabe automatisch zu überspringen oder stummzuschalten. Dabei werden EDL-Konfigurationsdateien ausgewertet.

Das ist dann ganz nützlich, wenn du einen Film in einer "familienfreundlichen" Version sehen möchtest. Du kannst damit Gewalt, Obszönitäten oder Jar-Jar Binks rausschneiden, ganz so, wie du's willst. Davon abgesehen kannst du z.B. auch automatisch Werbung überspringen.

Das EDL-Dateiformat ist ziemlich simpel gestrickt. Wenn das EDL erst einmal einen gewissen Reifegrad erreicht hat, wird wahrscheinlich ein auf XML basierendes Dateiformat eingeführt (aber die Kompatibilität zu alten EDL-Formaten trotzdem gewahrt) werden.

Die maximale Anzahl von EDL-Einträgen beträgt momentan 1000. Falls du mehr davon brauchst, dann ändern einfach `#define MAX_EDL_ENTRIES` in der Datei `edl.h`.

6.5.1 Benutzung von EDL-Dateien

Gib mit der Option `-edl <Dateiname>` die EDL-Datei an, die du auf dein Video angewandt haben möchtest.

6.5.2 Erstellen von EDL-Dateien

Das aktuelle EDL-Dateiformat sieht so aus:

```
[Start in Sekunden] [Ende in Sekunden] [Aktion]
```

Die Sekundenangaben sind Fließkommazahlen, und die Aktion ist entweder `0` (überspringen) oder `1` (stummschalten).

Beispiel:

5.3	7.1	0
15	16.7	1
420	422	0

Hier wird von Sekunde 5.3 bis Sekunde 7.1 ausgelassen. Von Sekunde 15 bis Sekunde 16.7 wird stummgeschaltet, bei 16.7 wieder lautgestellt, und von Sekunde 420 bis Sekunde 422 wird das Video übersprungen. Diese Aktionen werden ausgeführt, sobald der Abspieltimer die angegebenen Werte erreicht.

Wenn du eine EDL-Datei erstellen möchtest, dann benutze `-edlout <filename>`. Während der Wiedergabe kannst du `i` drücken. Es wird dann ein Eintrag erstellt, der an dieser Stelle 2 Sekunden überspringt. Du kannst dann mit einem Editor deiner Wahl die generierte EDL-Datei verfeinern.

7 Verwendung

7.1 Kommandozeile

MPlayer benützt einen komplexen Abspielbaum. Er besteht aus globalen Optionen am Anfang (z.B. `mplayer -vfm 5`) und Optionen nach dem Dateinamen, welche nur für die jeweilige Datei/URL/etc gelten (z.B. `mplayer -vfm 5 movie1.avi movie2.avi -vfm 4`).

Mit `{` und `}` können Dateinamen/URL-Gruppen gebildet werden. Dies ist mit der Option `-loop` nützlich: `mplayer { 1.avi -loop 2 2.avi } -loop 3` spielt die Dateien in dieser Reihenfolge: 1 1 2 1 1 2 1 1 2

Datei	<code>mplayer [Optionen] [Pfad/]Dateiname</code>
Datei	<code>mplayer [standard Optionen] [Pfad/]Dateiname [Optionen für datei1] datei2 datei3 [Optionen für datei3]</code>
VCD	<code>mplayer [Optionen] -vcd Tracknr [-cdrom-device /dev/cdrom]</code>
DVD	<code>mplayer [Optionen] -dvd Titelnr [-dvd-device /dev/dvd]</code>
Netz	<code>mplayer [Optionen] http://site.com/datei.asf</code> (playlist kann auch benutzt werden)

Die neusten Versionen von MPlayer akzeptieren auch VCD- und DVD-Tracks im URL-Format, so wie z.B. auch Xine: `mplayer dvd:///1` oder `mplayer vcd:///1`

```
user@linux ~/ $ mplayer -vo x11 /mnt/Filme/Contact/contact2.mpg
user@linux ~/ $ mplayer -vcd 2
user@linux ~/ $ mplayer -afm 3 /mnt/DVDtrailers/alien4.vob
user@linux ~/ $ mplayer -dvd 1
user@linux ~/ $ mplayer -abs 65536 -delay -0.4 -nobps ~/movies/test.avi
```

7.2 Steuerung

MPlayer besitzt eine vollständig konfigurierbare, durch Befehle angesteuerte Steuerungsschicht (control layer), der es erlaubt, MPlayer mit Tastatur, Maus, Joystick oder Fernbedienung (unter Verwendung von LIRC) zu steuern. Schau in die man-page. Dort steht eine vollständige Liste aller Tastaturkommandos.

7.2.1 Konfiguration der Steuerung

MPlayer ermöglicht es, dass du jede Taste/jeden Button mit einem MPlayer-Befehl verbindest. Dieses wird in einer simplen Konfigurationsdatei festgelegt. Die Syntax besteht einfach aus einem Tastennamen gefolgt vom auszuführenden Befehl. Die Standardkonfigurationsdatei ist `$HOME/.mplayer/input.conf`, aber diese Position kann auch mit der Option `-input` geändert werden. Pfadangaben beziehen sich hier relativ zu `$HOME/.mplayer`.

Beispiel:

Beispiel .mplayer

```
##
## MPlayer input control file
##

RIGHT seek +10
LEFT seek -10
- audio_delay 0.100
+ audio_delay -0.100
q quit
> pt_step 1
< pt_step -1
ENTER pt_step 1 1
```

7.2.1.1 Tastennamen

Mit folgendem Befehl erhältst du eine vollständige Liste: `mplayer -input keylist`

Tastatur

- * Irgendein druckbares Zeichen
- * SPACE
- * ENTER
- * TAB
- * CTRL
- * BS
- * DEL
- * INS
- * HOME
- * END
- * PGUP
- * PGDWN
- * ESC
- * RIGHT
- * LEFT
- * UP
- * DOWN

Maus (wird nur unter X unterstützt)

- * MOUSE_BTN0 (Linke Taste)
- * MOUSE_BTN1 (Rechte Taste)
- * MOUSE_BTN2 (Mittlere Taste)
- * MOUSE_BTN3 (Mausrad)
- * MOUSE_BTN4 (Mausrad)
- * ...
- * MOUSE_BTN9

Joystick (Joystick-Unterstützung muss bei der Kompilierung aktiviert sein)

- * JOY_RIGHT oder JOY_AXIS0_PLUS
- * JOY_LEFT oder JOY_AXIS0_MINUS

- * JOY_UP oder JOY_AXIS1_MINUS
- * JOY_DOWN oder JOY_AXIS1_PLUS
- * JOY_AXIS2_PLUS
- * JOY_AXIS2_MINUS
- *
- * JOY_AXIS9_PLUS
- * JOY_AXIS9_MINUS

7.2.1.2 Befehle

Mit folgendem Befehl erhältst du eine vollständige Liste: `mplayer -input cmdlist`

- * **seek** (int) Wert [(int) type=0]
Sucht bis zu einer gewissen Stelle im Film.
Type 0 ist eine relative Suche von +/- Wert Sekunden.
Type 1 sucht bis Wert % im Film.
- * **audio_delay** (float) Wert
Korrigiert die Audioverzögerung um Wert Sekunden
- * **quit** Beendet MPlayer
- * **pause** Wechselt zwischen Pause und Weiterspielen
- * **grap_frames** Weiß das jemand?
- * **pt_step** (int) Wert [(int) force=0]
Springt zum nächsten/vorherigen Eintrag im playtree. Das Vorzeichen von Wert bestimmt die Richtung.
Wenn kein neuer Eintrag mehr vorhanden ist, geschieht gar nichts, es sei denn, force ist nicht 0.
- * **pt_up_step** (int) Wert [(int) force=0]
Wie pt_step, springt jedoch zum nächsten/vorherigen Eintrag in der übergeordneten Liste. Dies ist nützlich, um eine innere Schleife im playtree zu unterbrechen.
- * **alt_src_step** (int) Wert
Wenn mehr als eine Quelle vorhanden ist, wählt dies die nächste/vorherige (nur unterstützt für asx Wiedergabelisten).
- * **sub_delay** (float) Wert [(int) abs=0]
Korrigiert die Untertitel-Verzögerung um +/- Wert Sekunden oder setzt sie auf Wert Sekunden, wenn abs nicht null ist.
- * **osd** [(int) level=-1]
Wechselt den OSD-Modus oder setzt in auf level, wenn level > 0.
- * **volume** (int) dir
Erhöht/erniedrigt die Lautstärke
- * **contrast** (int) Wert [(int) abs=0]
- * **brightness** (int) Wert [(int) abs=0]
- * **hue** (int) Wert [(int) abs=0]
- * **saturation** (int) Wert [(int) abs=0]
Setzt/korrigiert die Video-Parameter. Die Wertreichweite reicht von -100 bis 100.
- * **frame_drop** [(int) type=-1]
Wechselt den Framedrop-Modus.
- * **sub_pos** (int) Wert
Korrigiert die Untertitel-Position.
- * **vobsub_lang**
ändert die Sprache der VobSub-Untertitel.
- * **vo_fullscreen**
Wechselt den Vollbildschirm-Modus.
- * **tv_step_channel** (int) dir
Wählt nächsten/vorherigen Tv-Kanal.

```
* tv_step_norm
  ändert TV-Norm.
* tv_step_chanlist
  ändert Kanalliste.
* gui_loadfile
* gui_loadsubtitle
* gui_about
* gui_play
* gui_stop
* gui_playlist
* gui_preferences
* gui_skinbrowser
```

7.3 GUI-Aktionen

7.3.1 Steuerung mittels LIRC

Linux Infrared Remote Control - verwendet einen einfach zu bauenden, selbstgebauten IR-Empfänger, (fast) jede beliebige Fernbedienung und steuere deinen Linux-Rechner damit. Mehr dazu auf www.lirc.org.

Wenn du das lirc-Paket installiert hast, erkennt es `configure` automatisch. Wenn alles gut geht, gibt MPlayer während des Starts eine Nachricht wie etwa **Setting up lirc support...** aus. Wenn ein Fehler auftritt, wird dies auch gemeldet. Wenn gar nichts gesagt wird, ist auch keine LIRC-Unterstützung einkompiliert. Das ist alles :-)

Der Anwendungsname für MPlayer ist - oh, ein Wunder - `mplayer`. Du kannst jeden beliebigen MPlayer-Befehl benutzen und sogar mehrere nacheinander ausführen lassen, indem du sie durch `\n` voneinander trennst. Vergiss nicht, die `repeat`-Option in der `.lircrc` zu benutzen, wenn es Sinn macht (beim Suchen, bei der Lautstärke etc.). Hier ist ein Auszug aus meiner `.lircrc`:

```
.lircrc

begin
  button = VOLUME_PLUS
  prog = mplayer
  config = volume 1
  repeat = 1
end

begin
  button = VOLUME_MINUS
  prog = mplayer
  config = volume -1
  repeat = 1
end

begin
  button = CD_PLAY
  prog = mplayer
  config = pause
end

begin
  button = CD_STOP
  prog = mplayer
  config = seek 0 1\npause
end
```

Wenn du die Standard-Position der LIRC-Konfigurationsdatei (`~/.lircrc`) nicht magst, verwende die Option `-lircconf <Dateiname>`, um eine andere Datei anzugeben.

7.3.2 Der "Slave"-Modus

Der Slave-Modus erlaubt es dir, ein einfaches Frontend für MPlayer zu erstellen. Wenn er mit der Option `-slave` aktiviert wurde, dann liest MPlayer durch neue Zeilen unterbrochene Befehle (`\n`) von der Standardeingabe.

7.4 Streaming über das Netzwerk oder Pipes

MPlayer kann Videos über Netzwerke unter Verwendung des HTTP- MMS- oder RTSP/RTP-Protokolls abspielen.

Abspielen geht einfach durch das Hinzufügen der URL zu der Kommandozeile. MPlayer berücksichtigt auch die `http_proxy`-Umgebungsvariable und benutzt einen Proxy, falls vorhanden. Der Proxygebrauch kann auch erzwungen werden:

```
user@linux ~/ $ mplayer
http_proxy://proxy.micorsops.com:3128/http://micorsops.com:80/stream.asf
```

MPlayer kann auch von `stdin` lesen (nicht von benannten Pipes). Dies ist ein Beispiel für die Verwendung, um von FTP abzuspielen:

```
user@linux ~/ $ wget ftp://micorsops.com/etwas.avi -O - <command> mplayer
-
```

Hinweis:

Bei der Wiedergabe übers Netzwerk empfiehlt sich, den Cache zu aktivieren:

```
user@linux ~/ $ wget ftp://micorsops.com/etwas.avi -O - <command> mplayer
-cache 8192 -
```

8 CD/DVD-Sektion

8.1 CD-ROM- und DVD-ROM-Laufwerke

Auszug aus der Linux-Dokumentation:

Moderne CDROM-Laufwerke können sehr hohe Abtastgeschwindigkeiten erreichen. Einige können jedoch auch ihre Geschwindigkeit reduzieren. Es gibt mehrere Gründe, dies zu tun:

- * Es gibt Berichte, dass diese Laufwerke bei hohen Geschwindigkeiten Lesefehler erzeugen, vor allem bei schlecht gepressten CD-ROMs. Das Herabsetzen der Geschwindigkeit kann in diesen Fällen die Datenverluste verringern.
- * Viele CD-ROM Laufwerke erzeugen störend laute Geräusche, die eine niedrigere Geschwindigkeit ebenfalls reduzieren kann.

Die Lesegeschwindigkeit eines IDE-CD-Laufwerkes kann mit `hdparm` oder einem Programm namens `setcd` verringert werden. Das geht so:

```
root@linux ~/ # hdparm -E <Geschwindigkeit> <CD-ROM-Device>
```

```
root@linux ~/ # setcd -x <Geschwindigkeit> <CD-ROM-Device>
```

Du kannst auch folgendes probieren:

```
root@linux / # echo current_speed:4 > /proc/ide/[CDROM-Device]/settings
```

Aber dafür brauchst du Root-Privilegien. Der folgende Befehl könnte ebenfalls helfen:

```
root@linux / # echo file_readahead:200000 > /proc/ide/[CDROM-Device]/settings
```

Dies setzt einen 2MB Lesebuffer für die Daten, was sich bei zerkratzten CDs als nützlich erweist. Wenn du die Größe zu hoch wählst, dann wird das Laufwerk ständig beschleunigen und abbremsen und dadurch die Geschwindigkeit dramatisch verringern. Es ist auch empfehlenswert, das CD-ROM Laufwerk via `hdparm` auf performantere Werte einzustellen:

```
root@linux / # hdparm -d1 -a8 -u1 <CDROM-Device>
```

Dies aktiviert den DMA-Zugriff, Vorauslesen und IRQ-Unmasking (genauere Details dazu findest du in der `hdparm`-Manpage)

Bitte schau dir `/proc/ide/[CDROM-Device]/settings` an, um das CD-ROM feineinzustellen.

SCSI-Laufwerke kennen keine einheitliche Schnittstelle zum Setzen dieser Werte (oder kennst du eine? Dann sag sie uns!) Es gibt ein Tool, das mit  [Plextor-SCSI-Laufwerken](#) funktioniert.

8.2 DVD-Wiedergabe

Bitte schau in der man-Page nach, wenn du eine Auflistung aller verfügbaren Optionen willst. Die Syntax bei einer normalen Digital Versatile Disc (DVD) sieht wie folgt aus:

```
user@linux / $ mplayer -dvd <track> [-dvd-device <Gerät>]
```

Beispiel:

```
user@linux ~/ $ mplayer -dvd 1 -dvd-device /dev/hdc
```

Das Standard-DVD-Gerät ist `/dev/dvd`. Wenn das bei dir anders ist, so erstelle einen symbolischen Link auf das richtige Gerät, oder gib es auf der Kommandozeile mit der Option `-dvd-device` an.

Unterstützung für DVD-Navigation:

An einer Unterstützung für DVD-Navigation via `dvdnav` wird gearbeitet. Sie ist jedoch noch nicht vollendet und wird momentan nicht erwartet. Wer weiß - vielleicht lässt sie sich sogar kompilieren.

DVD-Strukturen:

DVDs benutzen alle 2048 Bytes/Sektor mit ECC/CRC. Meistens beinhalten sie ein UDF-Dateisystem in einem einzigen Track. Dieses Dateisystem enthält mehrere Dateien (kleine `.IFO`- und `.BUK`-Dateien sowie die großen (1GB) `.VOB`-Dateien). Sie sind echte Dateien und können von einem in's Dateisystem eingehängten Dateisystem einer nicht verschlüsselten DVD abgespielt oder kopiert werden.

Die `.IFO`-Dateien enthalten Filmnavigationsinformationen (z.B. Kapitel, Titel, verschiedene Kameraeinstellungen sofern vorhanden, Tabelle mit den Sprachen etc.) und werden benötigt, um den Inhalt der `.VOB`-Dateien (den eigentlichen Film) lesen und interpretieren zu können. Die `.BUK`-Dateien sind Backups der `.IFO`-Dateien. Sie benutzen ausschließlich **Sektornummern** zur Adressierung, sodass man zur DVD-Navigation das direkte Ansprechen einzelner Sektoren der DVD implementieren muss. `.IFO`s werden ebenfalls benötigt, um den Inhalt entschlüsseln zu können.

Aus diesem Grund braucht die alte DVD-Unterstützung ein in den Dateibaum eingehängtes DVD-Dateisystem sowie zusätzlich Zugriff auf das Device, um einzelne Sektoren der DVD lesen zu können. Leider musst du (unter Linux) root-Rechte besitzen, um die Sektorenadresse einer Datei zu erhalten. Du hast nun die folgenden Möglichkeiten:

- * Sei root oder gib der mplayer-Programmdatei das SUID-Bit.
- * Lass MPlayer das Wrapper-Programm `fibmap_mplayer` ausführen, welches das SUID-Bit gesetzt hat, um Zugriff auf die DVD zu bekommen.
- * Lass die Dateisystemtreiber des Kernels links liegen und programmiere den Support dafür erneut im User-Space. `libdvdread` 0.9.x und `libmpdvdkit2` machen das genau so (neue DVD-Unterstützung). Der UDF-Treiber des Kernels wird nicht benötigt, da diese Bibliotheken ihren eigenen UDF-Dateisystemtreiber mitbringen. Die DVD muss nicht einmal in den Dateibaum eingehängt sein, da direkt auf die Sektoren der DVD zugegriffen wird.

Manchmal hat ein Benutzer keinen Zugriff auf das Device (z.B. `/dev/dvd`). Deswegen haben die Autoren der `libdvdread` einen Emulationsmodus spendiert, der eine Umsetzung zwischen Sektornummern und Dateinamen/Offsets vornimmt. Damit wird Sektorzugriff auch bei in das Dateisystem eingehängten DVDs und sogar bei auf die Festplatte kopierten Dateien ermöglicht.

`libdvdread` akzeptiert sogar Mountpoints anstelle von Devicenamen. Dabei schaut sie in `/proc/mounts` nach, um den tatsächlichen Devicenamen für den Sektorzugriff herauszufinden. Entwickelt wurde dies

Verfahren für Solaris, da Solaris Devicenamen dynamisch vergibt.

Der Standardname für das DVD-Device ist `/dev/dvd`. Sollte das bei dir anders sein, so erstelle einen symbolischen Link auf das richtige Device, oder gib es beim Starten mit der Option `-dvd-device` separat an.

RPC-1-konforme DVD-Laufwerke sind nur über den Gebietsschutz geschützt, den der Softwareplayer bietet. RPC-2-konforme Laufwerke haben dagegen einen Hardwareschutz, der es nur fünf mal erlaubt, den Gebietscode zu ändern. Es ist evtl. nötig oder empfehlenswert, dass du Firmware für dein Laufwerk findest, die nur einen RPC-1-Schutz implementiert hat. Firmwareupdates können auf dieser [Firmwareseite](#) gefunden werden.

***Warnung:** Du kannst nur fünf mal den Gebietscode ändern.

8.3 VCD Wiedergabe

Abspielen von Standard Video-CDs:

```
user@linux ~/ $ mplayer -vcd <Tracknummer> [-cdrom-device <Gerät>]
```

Beispiele:

```
user@linux ~/ $ mplayer -vcd 2 -cdrom-device /dev/hdc
```

Das Standard-CD-Rom-Laufwerk ist `/dev/cdrom`. Falls das bei dir nicht der Fall ist, erstelle einen symbolischen Link oder gib das richtige Gerät mit `-cdrom-device` auf der Kommandozeile an.

Anmerkung:

Zumindest einige Plextor- und Toshiba SCSI-CD-Rom-Laufwerke haben eine grauenvolle Geschwindigkeit beim Lesen von VCDs. Das liegt daran, dass der `ioctl`-Aufruf `CDROMREADRAW` nicht vollständig implementiert ist. Wenn du Ahnung von der SCSI-Programmierung hast, dann hilf uns `<../tech/patches.txt>`, generische Unterstützung für VCD auf SCSI-Laufwerken zu implementieren.

VCD-Struktur:

Eine VCD besteht aus einem oder mehreren Spuren:

- * Die erste Spur ist eine kleine Datenspur mit 2048 Bytes pro Sektor, die ein ISO9660-Dateisystem enthält, auf dem meist ein VCD-Player für Windows und eventuell auch andere Informationen (Bilder, Texte etc.) gespeichert sind.
- * Die zweite Spur und alle folgenden sind MPEG-Spuren mit 2324 Bytes pro Sektor, die ein MPEG-PS-Datenpaket pro Sektor anstelle des Dateisystems enthalten. ähnlich wie bei Audio-CDs können diese Spuren **nicht gemountet** werden (hast du jemals eine Audio-CD gemountet, um sie abzuspielen?). Da die meisten Filme sich auf der zweiten Spur befinden, solltest du zuerst `-vcd 2` ausprobieren.
- * Es existieren auch VCDs ohne die erste Spur (nur eine Spur, überhaupt kein Dateisystem). Diese lassen sich abspielen aber **nicht** mounten.

über .DAT-Dateien:

Die ca. 600MB große sichtbare Datei auf der ersten Spur der gemounteten VCD ist keine echte Datei! Es handelt sich dabei um ein sogenanntes ISO-Gateway, das erstellt wurde, um Windows zu ermöglichen, solche Spuren zu lesen (Windows erlaubt Anwendungen den direkten Zugriff auf das Laufwerk nämlich nicht). Unter Linux kannst du diese Datei weder einfach kopieren noch einfach abspielen (sie enthalten dann nur Müll). Unter Windows geht das, weil der ISO9660-Treiber den direkten Zugriff auf die Spur emuliert. Um eine .DAT-Datei

abzuspielen, brauchst du einen Kerneltreiber, der in der Linux-Version von PowerDVD enthalten ist. Er enthält einen modifizierten Treiber für das ISO9660-Dateisystem (`vcdfs/isofs-2.4.X.o`), der ebenfalls in der Lage ist, das direkte Auslesen über diese .DAT-Datei zu emulieren. Wenn du die Disc mit diesem Treiber mountest, dann kannst du die .DAT-Datei kopieren und sogar direkt mit MPlayer abspielen. Das wird aber **nicht** mit dem Standard-ISO9660-Treiber funktionieren! Benutze dabei `-vcd`. Alternativen für's Kopieren einer VCD sind der neue  [cdfs](#)-Kerneltreiber (der nicht Teil des offiziellen Kernels ist), der **CD-Sessions** als Abbilder anzeigt, und  [cdrdao](#), ein Anwendung zum Bit-für-Bit-Grabben/Kopieren von CDs.

9 FAQs - Die häufigsten Fragen und Antworten

9.1 Kompilierung

Q: Die Kompilierung bricht mit in etwa folgender Meldung ab:

```
In file included from mplayer.c:34:
mw.h: In function `mplMainDraw':
mw.h:209: Internal compiler error in print_rtl_and_abort, at
flow.c:6458
Please submit a full bug report,
with preprocessed source if appropriate.
```

A: Dies ist ein bekanntes Problem des gcc 3.0.4. Upgrade auf 3.1, um es zu beheben. Wie gcc installiert werden kann, ist in der gcc 2.96-Sektion beschrieben.

Q: Configure beendet sich mit diesem Text, und MPlayer kann nicht kompiliert werden!

```
Your gcc does not support even i386 for '-march' and '-mcpu'.
```

A: Dein gcc ist nicht richtig installiert, prüfe [config.log](#) für Details.

Q: Was bedeutet "No such file or directory"?

A: Vermutlich gibt es keine solche Datei oder kein solches Verzeichnis.

Q: Was ist das Problem mit gcc 2.96?

A: Wir raten sehr davon abgeraten, den gcc 2.96 einzusetzen! Lies [▶ dieses](#) Dokument, in dem die Gründe stehen, warum RedHat den gcc 2.96 herausgegeben hat, und welche Probleme es damit gibt. Wenn du diese Version aber trotzdem wirklich einsetzen willst, dann stell sicher, dass du die neueste Version verwendest, und übergib die Option `--disable-gcc-checking` an [configure](#). Bedenke aber, dass du auf dich alleine gestellt bist. Sende **keine** Meldungen oder Bugs, und frag in der Mailingliste nicht um Hilfe. Wir werden sie nicht zur Verfügung stellen, noch Unterstützung bei Problemen anbieten.

Q: Großartig, ich habe gcc 3.0.1 von Red Hat/Mandrake, dann ist alles fein!

A: Nein, denn auch mit diesen Compilern gab/gibt es Probleme. Der Status der momentanen Compilernerunterstützung findet sich in der [▶ Installationssektion](#).

Q: Ich habe versucht, MPlayer zu kompilieren, aber ich bekomme folgende Ausgabe:

```
In file included from /usr/include/g++-v3/bits/std_cwchar.h:42,
                 from /usr/include/g++-v3/bits/fpos.h:40,
                 from /usr/include/g++-v3/bits/char_traits.h:40,
                 from /usr/include/g++-v3/bits/std_string.h:41,
```

```
from /usr/include/g++-v3/string:31,  
from libwin32.h:36,  
from DS_AudioDecoder.h:4,  
from DS_AudioDec.cpp:5:  
/usr/include/wchar.h: In function Long long int wcstog(const  
wchar_t, wchar_t, int)':  
/usr/include/wchar.h:514: cannot convert `const wchar_t  
__restrict' to `const
```

A: Update deine glibc auf die neueste Release. Bei Mandrake verwende 2.2.4-8mdk.

Q: ... 2.96 ... (ja, einige Leute beschwerten sich NOCH IMMER wegen gcc 2.96!)

A: Zitat aus einer  [Mail](#), die A'rpi an die  [mplayer-users](#) Mailing-Liste geschrieben hat [übersetzung] (Das Wort 'ideg' ist unten beschrieben): Aber wir hatten idegs. Und unser Idegcounter hatte wieder und wieder Overflows. Leider entweicht MPlayer unsere Kontrolle. Er wird von Idioten verwendet, die nicht mal Windows verwenden können und auch noch nie versucht, haben einen Kernel zu kompilieren. Sie installieren Mandrake, Red Hat oder Suse (mit Standardeinstellungen) und schreiben uns 'Es funktioniert nicht! Helft mir! Bitte! Ich bin neu bei Linux! Hilfe! Oh! Helft mir!', ohne vorher die Dokumentation zu lesen. Wir können sie nicht aufhalten, aber wir versuchen, sie zum Lesen der Dokumentation und der Ausgabe von ./configure und MPlayer zu zwingen.

Und ihr cleveren Leute kommt und beleidigt uns mit gcc 2.96 und binären Paketen. Anstatt anderen Users zu helfen oder Patches zu erzeugen um, Probleme zu lösen.

Die Hälfte unserer Freizeit verbrauchen wir beim Beantworten sinnloser Mails und beim Einbauen von neuen Tricks für configure, um solche Mails zu vermeiden. Und hier gibt es Gleichgewicht. Auf der einen Seite seid ihr cleveren Jungs, die sagen, dass wir böse sind, da wir das fehlerhafte gcc 2.96 nicht mögen, und auf der anderen Seite sind die 'neu bei Linux'-Leute, welche uns zeigen, wie schlecht der gcc 2.96 ist. Schlussfolgerung: wir können nicht gut sein. Die Hälfte der Leute sagt, dass wir böse sind.

Möglicherweise sollten wir das Projekt schließen, kommerzielles Closed-Source daraus machen und euch dann Installationssupport anbieten. Dann könnten wir unsere aktuelle Arbeitsstätte verlassen, die Entwicklung ginge schneller, and wir könnten eine Menge Geld verdienen und ein großes Haus kaufen, usw. usw. Willst du das wirklich? Es scheint so.

Q: SDL Ausgabe funktioniert/kompiliert nicht. Das Problem ist ...

A: Es wurde mit SDL 1.2.x getestet und läuft vielleicht auf SDL 1.1.7+. Es funktioniert nicht mit irgendeiner früheren Version. Wenn du eine solche einsetzen willst, dann bist du auf dich allein gestellt.

Q: Ich habe immer noch Probleme mit dem SDL Support. gcc sagt etwas über "undefined reference to `SDL_EnableKeyRepeat'". Was ist nun?

A: Wo hast du die SDL Library installiert? Wenn du es in /usr/local (Standard) installiert hast, editiere config.mak und füge -L/usr/local/lib nach X_LIBS= ein. Nun gib make ein. Es sollte funktionieren!

Q: Es wird nicht kompiliert, und es vermisst uint64_t, inttypes.h oder ähnliche Dinge ...

A: Kopiere etc/inttypes.h in das MPlayer-Verzeichnis (cp etc/inttypes.h) und versuche es erneut ...

Q: Ich habe Linux auf einem Pentium III, aber ./configure erkennt SSE nicht ...

A: Nur Kernel mit der Version 2.4.x unterstützen SSE (oder versuche 2.2.19 oder neuer, aber sei auf Probleme vorbereitet).

Q: Ich habe eine G200/G400, wie kompiliere/verwende ich den mga_vid-Treiber?

A: Lies die [mga_vid-Dokumentation](#).

Q: Gibt es rpm/deb/...-Pakete von MPlayer?

A: Du kannst ein .deb Paket für dich selbst machen, lies die [Debian-Pakete-Sektion](#). Es gibt Links zu offiziellen RedHat-RPM-Paketen. Sie sind auf der [Download-Seite](#) zu finden.

Q: Während `make` beschwert sich MPlayer über die X11- Librarys. Ich verstehe das nicht, ich habe X installiert!?

A: ... aber du hast die X Development Pakete nicht installiert. Oder nicht richtig. Diese heißen unter Red Hat XFree86-devel und unter Debian xlibs-dev . Prüfe auch, ob die symbolischen Links `/usr/X11` und `/usr/include/X11` existieren (dies kann ein Problem auf Mandrake-Systemen sein). Diese können mit diesen Befehlen erzeugt werden:

```
$ ln -sf /usr/X11R6 /usr/X11 $ ln -sf /usr/X11R6/include/X11
/usr/include/X11 Deine Distribution unterscheidet sich vielleicht vom 
Dateisystem-Hierarchie-Standard.
```

Q: Ich kann SVGAlib nicht kompilieren. Ich verwende eine 2.3/2.4er Kernel.

A: Du musst die Makefile.cfg von SVGAlib editieren. Kommentier `BACKGROUND = y` aus.

Q: Hmm, eigenartig. Beim Laden des mga_vid.o-Kernelmodules kann ich folgendes in den Logs finden:

Warning: loading mga_vid.o will taint the kernel: no license

A: Die neuesten Kernelmodule benötigen einen Parameter, um ihre Lizenz anzuzeigen (hauptsächlich, um Kernel-Hacker am Debuggen von Closed-Source-Treibern zu hindern). Update deinen Kernel, modutils und MPlayer.

Q: Während des Linkens von MEncoder passiert ein segfault!

A: Dies ist ein Linker-Problem. Ein Upgrade deiner binutils sollte helfen (2.11.92. oder neuer wäre gut). Melde den Fehler nicht, da er nicht von uns ist!

Q: MPlayer bricht mit einem segmentation fault bei der pthread-überprüfung ab!

A: `chmod 644 /usr/lib/libc.so`

Q: Ich möchte gerne MPlayer auf Minix kompilieren!

A: Ich auch. :)

9.2 Allgemeine Fragen

Q: Wie erzeuge ich einen vernünftigen Patch für MPlayer?

A: Wir haben dafür ein [kleines Dokument](#) erstellt, das alle notwendigen Details erklärt. Folge bitte seinen Anweisungen.

Q: Wie kann ich die Entwicklung von MPlayer unterstützen?

A: Wir nehmen sehr gerne  [Hard- und Software Spenden](#) an. Sie helfen uns, MPlayer ständig zu verbessern.

Q: Wie kann ich ein MPlayer-Entwickler werden?

A: Wir nehmen Programmierer und Leute auf, die Dokumentation schreiben wollen. Lies die  [technische Dokumentation](#) <../tech/>, um einen ersten Überblick über MPlayer zu bekommen. Dann solltest du dich bei der  [mplayer-dev-eng](#)-Mailingliste anmelden und mit dem Programmieren anfangen.

Q: Kann ich MPlayer mehrmals aufrufen?

A: Ja. Einige Video-Ausgabetreiber wie xv, dga oder (x)mga erlauben jedoch nur eine Instanz, andere tun dies in Kombination mit gewissen Grafikkarten oder -treibern. Du wirst in diesen Fällen nur jeweils eine Instanz mit diesen Videoausgabetreibern laufen lassen können. Etwas weniger stark trifft das auch auf die Audioausgabetreiber. Nicht alle erlauben die gleichzeitige Wiedergabe mehrerer Streams auf dem gleichen Gerät.

Q: Ich sehe in der linken oberen Ecke einen Timer. Wie werde ich den los?

A: Drücke `o` oder verwende die `-osdlevel`-Option.

Q: Die `-xy` oder `-fs`-Option funktioniert nicht mit dem X11-Treiber (`-vo x11`)

A: Doch, das funktioniert, aber du musst explizit die Softwareskalierung mit der `-zoom`-Option aktivieren (sehr langsam). Du solltest besser die XF86VidMode-Unterstützung verwenden. Du musst dazu die `-vm`-Option und die `-fs`-Option verwenden. Stelle sicher, dass du die richtigen modelines in deiner XF86Config-Datei hast, und versuche, den DGA-Treiber und SDLs  [DGA-Treiber](#) für dich zum Laufen zu bringen. Er ist um einiges schneller. Wenn  [SDL-DGA](#) auch funktioniert, verwende das, es ist nochmals schneller.

Q: Was ist die Bedeutung der Nummern auf der Statusleiste?

A: Beispiel: **A:** 2.1 **V:** 2.2 **A-V:** -0.167 **ct:** 0.042 57/57 41% 0% 2.6% 0 4 49%

- * A: Audioposition in Sekunden
- * V: Videoposition in Sekunden
- * A-V: Audio-Video Differenz in Sekunden
- * ct: insgesamt geleistete A-V Synchronisation
- * abgespielte Frames (gezählt ab der letzten Suche)
- * dekodierte Frames (gezählt ab der letzten Suche)
- * Video-Codec CPU-Auslastung in Prozent (bei "slices" und DR ist auch die Video-Ausgabe enthalten)
- * Video-Ausgabe CPU-Auslastung
- * Audio-Codec CPU-Auslastung in Prozent
- * ausgelassene Frames für die A-V Sync
- * aktuelle Level des Bild-Postprocessing (wenn -autoq verwendet)
- * aktuelle benutzte Cachegröße (normal ist ~50%)

Die meisten sind für debugging-Zwecke und werden irgendwann entfernt.

Q: Was ist, wenn ich die nicht sehen will?

A: Dann verwende die `-quiet`-Option, und lies die Manpage.

Q: Warum ist die Video-Ausgabe CPU-Auslastung null (0%) für einige Dateien?

A1: Es ist nicht null, aber der Codec ruft die Videoausgabe selber auf, weswegen sie nicht genau gemessen werden kann. Du solltest versuchen, die Datei mit `-vo null` und dann mit `-vo ...` abzuspielen. Die Differenz ist die Videoausgabegeschwindigkeit.

A2: Du verwendest Direct Rendering, der Codec schreibt also selbst in den Videospeicher. In diesem Fall beinhaltet die Decodierungs-Prozentrate auch die Wiedergabe-Prozentrate.

Q: Ich habe Fehlermeldung über "file not found /usr/lib/win32/ ..."

A: Downloade und installiere die [Win32 Codecs](#) von unserem Server (avifiles Codec-Paket hat unterschiedliche DLLs).

Q: Gibt es Mailing-Listen über MPlayer?

A: Ja. Lies dazu das Ende der Infoseite auf unserer [Homepage](#), um dich anzumelden.

Q: Ich habe einen bösen Fehler beim Abspielen meines Lieblingsvideos gefunden. Wen soll ich informieren?

A: Bitte lies die [Leitlinien zu Fehlerberichten](#) und folge den Anweisungen.

Q: Ich habe Probleme beim Abspielen von Dateien mit dem ... Codec. Kann ich sie trotzdem verwenden?

A: Prüfe den [Codecs-Status](#). Wenn dein Codec nicht aufgelistet ist, lies die [Codec-Dokumentation](#), vor allem das [Codec-Importierungs-HOWTO](#) und kontaktiere uns.

Q: Hmm, was ist "IdegCounter"?

A: Eine Kombination eines ungarischen und eines englischen Wortes. "Ideg" in Ungarn bedeutet etwa das gleiche wie "Nerv" in Deutschland "ydaegh" ausgesprochen. Es wurde zuerst benutzt, um die Nervosität von A'rpí nach eigenartigem, mystischen Verschwinden von Code aus dem CVS auszudrücken ;)

Q: Und was ist "Faszom(C)ounter"?

A: "Fasz" ist ein ungarisches Wort, das du nicht kennen willst, die anderen stehen in Verbindung zu den perversen Gedanken der MPlayer-Entwickler.

Q: LIRC funktioniert nicht, weil ...

A: Bist du sicher, dass du `mplayer` anstatt `mplayer_lirc` verwendest? Eine lange Zeit lang hieß das Executable `mplayer_lirc`, sogar im 0.60 Release, wurde jedoch neulich wieder zu `mplayer` geändert.

Q: Die Untertitel sind sehr nett, die wunderschönsten die ich jemals gesehen habe, aber sie bremsen das abspielen! Ich weiss, das klingt unwahrscheinlich ...

A: Editiere `config.h` nach dem Ablaufen von `./configure` und ändere `#undef FAST_OSD` in `#define FAST_OSD`. Kompiliere MPlayer danach neu.

Q: Das OSD (onscreen display) flackert!

A: Du verwendest einen vo-Treiber mit einfachem-Buffering (x11, xv). Verwende bei xv die `-double`-Option. Versuche auch `-vop expand`.

Q: Was genau ist dieses `libavcodec`-Dingens?

A: Siehe  [libavcodec](#)-Sektion.

Q: Aber configure sagt mir "Checking for `libavcodec` ... no"!

A: Du brauchst `libavcodec` vom FFmpeg CVS. Lies die Anweisungen in der  [libavcodec](#)-Sektion.

Q: Icewm's Taskbar bleibt im Vollbildschirm-Modus ständig über der Filmausgabe!

A: Dies sollte nicht mehr geschehen, falls doch ist die `-icelayer`-Option zu verwenden und ein Bericht an die  [mplayer-users](#) zu senden.

Q: Ich kann nicht auf das GUI Menü zugreifen. Ich drücke die rechte Maustaste, kann aber auf kein Menü zugreifen.

A: Verwendest du FVWM? Versuche folgendens: Start -> Settings -> Configuration -> Base Configuration Setze "Use Applications position hints" auf "Yes".

Q: Wie kann ich MPlayer im Hintergrund laufen lassen?

A: Verwende: `mplayer <Optionen> <Dateiname> < /dev/null &`

9.3 Dateiwiedergabe-Probleme

Q: Ich kann einige AVIs wiedergeben, erhalte jedoch keine Tonausgabe und diese Fehlermeldung:

```
Detected audio codec: [divx] afm:4 (DivX audio (WMA))
Requested audio codec family [divx] (afm=4) not available (enable
it at compile time!)
```

A: Wahrscheinlich wird ein Audiocodec verwendet, der nicht in MPlayer eingebaut ist. Zur Abhilfe sollte das Win32-Codec Paket wie im README oder in der Installation beschrieben installiert werden.

Q: ... funktioniert mit `avifile/aviplay`, aber nicht bei MPlayer.

A: MPlayer ist nicht `avifile`. Der einzige gemeinsame Teil zwischen diesen beiden Playern ist der Win32-DLL-Loader. Das Codec-(DLL)-Setup, die Synchronisation, das Demultiplexing usw sind total unterschiedlich und sollten nicht verglichen werden. Wenn etwas mit `aviplay` funktioniert, bedeutet das nicht, dass es bei MPlayer auch funktionieren wird (und umgekehrt).

Q: Audio-Video-Synchronisation geht beim Abspielen eines AVIs verloren.

A: Versuche die `-bps-` und `-nobps-` Optionen. Lies die  [Bugreports Sektion](#), falls es zu keiner Verbesserung führt, und lade die Datei auf den FTP-Server hoch.

Q: MPlayer beendet sich mit einem Fehler bei der Verwendung von `l3codeca.acm`.

A: Prüfe die `ldd /usr/local/bin/mplayer-Ausgabe`. Wenn sie `libc.so.6 => /lib/libc.so.6 (0x4???????)` enthält, wobei "?" irgendeine Nummer ist, ist es OK, und der Fehler liegt nicht hier. Wenn aber die Ausgabe `libc.so.6 => /lib/libc.so.6 (0x00???????)`

enthält, hast du ein Problem mit deinem Kernel und/oder deiner libc. Möglicherweise verwendest du auch einige Sicherheits-Patches (z.B. Solar Designer's OpenWall Patch), welcher das Laden von Libs in sehr niedrige Speicheradressen erzwingt. Aufgrund der Tatsache, dass a.acm eine nicht-verlegbare DLL ist, muss sie auf den Speicherbereich 0x00400000 geladen werden. Wir können das nicht ändern. Du solltest einen nicht-gepatchten Kernel oder MPlayers `-afm 1` Option verwenden, um die l3codeca.acm auszuschalten.

Q: Mein Computer spielt M\$-DivX-AVIs mit Auflösungen von ~ 640x300 und Stereo-MP3-Sound zu langsam ab. Wenn ich den `-nosound` -Parameter verwende ist alles OK, aber ich höre halt keinen Ton.

A: Dein Computer ist zu langsam oder dein Soundkarten-Treiber kaputt. Lies in der Dokumentation, um zu sehen, wie sich die Geschwindigkeit noch steigern lässt.

Q: MPlayer beendet sich mit "MPlayer interrupted by signal 4 in module: decode_video".

A: Versuche MPlayer auf dem Computer zu starten, auf dem es kompiliert wurde. Oder kompiliere MPlayer mit Laufzeit -CPU- Erkennung (runtime CPU detection) neu (`./configure --enable-runtime-cpudetection`). Verwende MPlayer auf keiner anderen CPU als auf der, auf der MPlayer kompiliert wurde ohne dieses Feature zu aktivieren.

Q: Ich habe Probleme mit [dein Window-Manager] und Vollbild-xv/xmga/sdl/x11.

A: Lies die  [Richtlinien für Fehlerberichte](#) und schreib einen Fehlerbericht.

Q: Ich bekomme diese Meldung beim Abspielen von MPEG-Dateien: Can't find codec for video format 0x10000001!

A: Du verwendest eine alte Version der codecs.conf in `~/.mplayer/`. Upgrade sie von `etc/`, oder du verwendest die `vc=` Option oder eine ähnliche in deiner Config-Datei.

Q: Nach dem Starten von MPlayer unter KDE I bekomme ich einen schwarzen Bildschirm, und nichts passiert. Nach etwa einer Minute beginnt das Video zu spielen.

A: Der KDE Arts Sound Daemon blockiert das Sound-Device. Du kannst also die Zeit warten, bis das Video startet, oder den Arts-Daemon im Kontroll-Zentrum ausschalten. Um Arts zu verwenden, sollte die Audio-Ausgabe über den eingebauten Arts-Audiotreiber (`-ao arts`) aktiviert werden. Falls dies fehlschlägt oder der Treiber nicht einkompiliert wurde, kann SDL (`-ao sdl`) verwendet werden. Dabei muss jedoch sichergestellt werden, dass SDL mit Arts umgehen kann. Eine weitere Möglichkeit besteht darin, MPlayer mit `artsdsp` zu starten.

Q: Ich habe eine AVI-Datei, die mit `-vc odivx` einen grauen und mit `-vc divx4` einen grünen Bildschirm wiedergibt.

A: Es ist keine DivX-Datei, sondern eine M\$ MPEG4v3. Update deine codecs.conf.

Q: Wenn ich diesen Film abspiele erhalte ich eine Audio/Video-Desynchronisation und/oder MPlayer stürzt mit folgender Nachricht ab:

DEMUXER: Too many (945 in 8390980 bytes) video packets in the buffer!

A: Dies kann mehrere Gründe haben.

* Deine CPU **und/oder** deine Grafikkarte und/oder dein BUS ist/sind zu **langsam**. MPlayer zeigt in diesem Fall eine Meldung (und der Zähler für die ausgelassenen Frames steigt).

* Wenn es eine AVI-Datei ist, hat sie möglicherweise ein schlechtes Interleaving. Versuche die `-ni` Option.

* Dein Soundtreiber paßt nicht gut zu Deinem Chipsatz, oder du verwendest ALSA 0.5 mit `-ao oss`. Siehe in der [▶ Soundkarten-Sektion](#).

* Deine AVI-Datei hat einen defekten Header, versuche die `-nobps` Option und/oder `-mc 0`.

Q: Ich habe hier eine MJPEG-Datei, die bei anderen Playern funktioniert. MPlayer zeigt jedoch nur ein schwarzes Bild an.

A: Deaktiviere die Windows-DLL in `codecs.conf`, oder verwende die `-vc ffmpegjpeg`-Option (kompiliere dazu `libavcodec!`).

Q: Wenn ich versuch, von meinem Tuner zu grabben, funktioniert es, aber die Farben sind merkwürdig. Mit anderen Anwendungen sehen sie aber normal aus.

A: Deine Karte gibt vielleicht falsche Farbraum-Fähigkeiten an. Versuche es mit YUY2 anstatt des Standards YV12 siehe [▶ TV Input Sektion](#).

Q: Wenn ich beginne abzuspielen, bekomme ich diese Meldung, aber die Wiedergabe ist gut: `Linux RTC init: ioctl (rtc_pie_on): Permission denied`

A: Du benötigst root-Privilegien oder einen speziell vorbereiteten Kernel, um den neuen Timing-Code zu verwenden. Für Details lies die [▶ RTC-Sektion](#) der Dokumentation.

Q: Ich habe A/V-Sync-Probleme. Einige meiner AVIs spielen problemlos, aber andere werden mit doppelter Geschwindigkeit abgespielt!

A: Du hast eine fehlerhafte Soundkarte / einen fehlerhaften Treiber. Wahrscheinlich kann die Soundkarte nur 44100Hz und du versuchst, eine Datei mit 22050Hz Audio abzuspielen. Probier das Audio-Resample-Plugin aus.

Q: Alle WMV-Dateien (oder andere..), die ich abspiele, erzeugen ein grün/graues Fenster und nur Ton! MPlayer gibt folgendes aus: `Detected video codec: [null] drv:0 (NULL codec (no decoding))`

A: Update deine `codecs.conf`.

Q: Ich erhalte während der Wiedergabe von Dateien auf meinem Notebook sehr seltsame Prozentangaben (viel zu hohe)?

A: Dies ist ein Effekt des Power Managements / Power Saving Systems auf deinem Notebook (BIOS, nicht Kernel). Stecke das externe Netzteil ein, bevor du dein Notebook einschaltest. Du kannst auch ausprobieren, ob dir [cpufreq](#) hilft (ein SpeedStep-Interface für Linux).

Q: Ton/Bild geraten total aus der Synchronisation, wenn ich MPlayer als root starte auf meinem Notebook starte. Als Benutzer läuft jedoch alles normal.

A: Dies ist wieder ein Power-Management-Effekt (siehe oben). Stecke den externen Stromstecker ein, bevor du dein Notebook einschaltest, oder benutze die Option `-nortc`.

9.4 Video/Audio-Treiber-Probleme

Q: Ich erhalte keinen Ton bei der Wiedergabe eines Filmes und folgende Fehlermeldung:

```
AO: [oss] 44100Hz 2ch Signed 16-bit (Little-Endian)
```

```
audio_setup: Can't open audio device /dev/dsp: Device or resource
busy couldn't open/init audio device -> NOSOUND
Audio: no sound!!!
Start playing...
```

A: Verwendest du KDE oder GNOME mit dem arts- oder esd-Sounddaemon? Versuche ihn zu deaktivieren, oder verwende die `-ao arts`-Option, um MPlayer arts verwenden zu lassen.

Q: Was ist mit dem DGA-Treiber? Ich kann ihn nicht finden!

A: `./configure` erkennt den DGA-Treiber automatisch. Wenn `-vo help` DGA nichts anzeigt, gibt es ein Problem mit der X-Installation. Versuche `./configure --enable-dga` und lies die  [Dokumentation](#). Alternativ kannst du auch versuchen, den SDL-DGA Treiber mit der Option `-vo sdl:dga` zu verwenden.

Q: `-vo help` zeigt den DGA-Treiber, aber es beschwert sich über die Zugriffsrechte! Helft mir!

A: DGA funktioniert nur als Root! Dies ist eine DGA-Beschränkung. Du solltest root werden (`su -`) und es erneut versuchen. Eine andere Lösung ist, MPlayer auf SUID root zu stellen, aber das ist nicht empfehlenswert!

```
chown root /usr/local/bin/mplayer
chmod 755 /usr/local/bin/mplayer
chmod +s /usr/local/bin/mplayer
```

Warnung: Sicherheitsrisiko

Dies ist ein grosses Sicherheitsrisiko! Tu das nie auf einem Server oder einem Computer, den du nicht vollständig kontrollierst. Die User bekommen durch einen SUID root MPlayer Root-Privilegien! Du bist also gewarnt worden.

Q: Wenn ich Xvideo verwende, meldet meine Voodoo 3/Banshee:

```
X Error of failed request: BadAccess (attempt to access private
resource denied)
Major opcode of failed request: 147 (MIT-SHM)
Minor opcode of failed request: 1 (X_ShmAttach)
Serial number of failed request: 26
Current serial number in output stream:27
```

A: Der "tdfx"-Treiber in XFree86 4.0.2/4.0.3 hatte diesen Bug. Er wurde durch den bugfix #621 in  [XFree86 4.1.0](#) behoben. Upgrade also auf XFree86 4.1.0 oder neuer. Alternativ kannst du auch entweder DRI Version 0.6 (oder neuer) von der  [DRI-Homepage](#) oder CVS DRI verwenden.

Q: OpenGL-`(-vo gl)`-Ausgabe funktioniert nicht (aufhÃ¤t;ngen/schwarzes Fenster/X11-Fehler/...).

A: Dein OpenGL-Treiber unterstützt keine "Dynamic Texture Changes" (`glTexSubImage`). Es ist bekannt, dass es nicht mit  [nVidias](#) Binär-Sch*** funktioniert. Es funktioniert mit Utah-GLX/DRI und Matrox-G400-Karten ebenso wie mit Radeon-Karten und DRI. Es wird nicht mit DRI und anderen Karten funktionieren. Es wird auch nicht mit 3DFX-Karten funktionieren, denn sie haben eine Texture-Größenbeschränkung auf 256x256.

Q: Ich habe eine nVidia TNT/TNT2-Karte und Probleme mit eigenartigen Farben rechts unter dem Film! Was ist der Auslöser?

A: Es ist ein Bug des binÃ¤t;ren nVidia X-Treibers. Diese Bugs treten NUR mit TNT/TNT2-Karten auf. Wir können nichts dagegen tun. Um das Problem zu lösen, sollte die neuste nVidia Treiber-Version verwendet werden. Wenn dies nichts nützt: Beklag dich bei nVidia!

Q: Ich habe eine nVidia XYZ-Karte. Wenn ich auf der GUI-Fenster klicke, erscheint an der Stelle, wo ich geklickt habe, ein schwarzes Quadrat. Ich verwende den neusten Treiber.

A: Ja ... nVidia hat einen vorherigen Bug (siehe oben) korrigiert und einen neuen implementiert. Lasst uns nVidia gratulieren. UPDATE: Nach nVidia wurde dieser bereits behoben.

Q: Wenn ich das GUI mit SDL Videoausgabe verwende, wird ein zweites Fenster erzeugt.

A: Bekannt, verwende SDL für GUI zur Zeit nicht.

Q: Oh die Welt ist grausam ...! SDL besitzt nur `x11` aber kein `xv` als Ausgabe!

A: Versuch nochmals die `x11` Ausgabe. Nun versuch `-vo x11 -fs -zoom`. Siehst du irgendwelche Unterschiede? Nein?! OK, hier die Erleuchtung: SDLs `x11`-Ausgabe verwendet `xv`, falls vorhanden. Du musst dir keine Sorgen dazu machen ... Hinweis: Du kannst `Xv` via SDL mit den Optionen `-forcexv` und `-noxv` erzwingen bzw. deaktivieren.

9.5 DVD Wiedergabe

Q: Was steht's mit DVD-Navigation?

A: Die Unterstützung für DVD-Navigation in MPlayer ist momentan kaputt. Die normale Wiedergabe funktioniert natürlich. Wenn du unbedingt tolle Menüs haben möchtest, dann benutze einen anderen Player wie z.B.  [xine](#) oder  [Ogle](#). Wenn dir DVD-Navigation wichtig ist, dann schick uns einen Patch `<./tech/patches.txt>`.

Q: Beim Abspielen der DVD erscheint folgender Fehler:

```
mplayer: ifo_read.c:1143: ifoRead_C_ADT_internal: Assertion
ifo_length / sizeof(cell_adr_t) >= c_adt->nr_of_vobs' failed.
```

A: Dies ist ein bekannter Fehler in libdvread 0.9.1/0.9.2. Verwende libmpdvdkit2, welche sich in den MPlayer-Sourcen befindet und standardmäßig verwendet wird.

Q: Was ist mit den Untertiteln? Kann MPlayer sie anzeigen?

A: Ja. Lies das  [DVD-Kapitel](#) der Dokumentation.

Q: Wie kann ich den Regional-Code meine DVD-Laufwerks einstellen? Ich habe kein Windows!

A: Verwende das 'regionset'-Tool.

Q: Muss ich root oder fibmap_mplayer setuid root sein, um DVDs abspielen zu können?

A: Nein, das ist nur bei der alten DVD-Unterstützung notwendig. Natürlich brauchst du aber die Zugriffsrechte für das DVD-Laufwerk (in `/dev`).

Q: Ist es möglich, nur ausgewählte Kapitel abzuspielen/zu encodieren?

A: Ja, mit der `-chapter`-Option.

Q: Meine DVD-Wiedergabe ist träge!

A: Benutze die `-cache`-Option (siehe Manpage). Versuche auch, den DMA-Modus für das DVD-Laufwerk mit `hdparm` zu aktivieren lies die [DVD-Sektion](#) der Dokumentation.

9.6 Feature-Wünsche

Q: Wenn MPlayer pausiert wird und ich vorspule oder zurückspule oder eine andere Taste drücke, dann beendet MPlayer die Pause. Ich würde gerne im pausierten Film spulen.

A: Das ist sehr schwierig zu implementieren, ohne die A/V-Synchronisation zu verlieren. Alle Versuche scheiterten bisher. Patches sind jedoch willkommen.

Q: Ich möchte gerne 1 Frame vorwärts oder rückwärts springen anstatt den 10 Sekunden.

A: Dies ist nicht implementiert. Es war bereits implementiert, aber es verursachte A/V-Desynchronisationsfehler. Implementiere es und schick uns einen Patch ... aber frag nicht danach.

Q: Wo ist die Windows Version?

A: Sie ist nicht erschienen und wird auch nicht erscheinen. Schau dich in der Windows-Szene um (Opensource-ASF-Parser, Opensource-ASF-Encoders, usw), und du weißt warum. Wir wollen nicht eingesperrt werden. Du kannst es übrigens für dich kompilieren. eine Anleitung findet sich unter [Cygwin](#)

9.7 Konvertierung

Q: Wie kann ich konvertieren?

A: Lies diese [Dokumentation](#).

Q: Wie erstelle ich VCDs?

A: Verwende das `mencvcd`-Script im `TOOLS`-Unterverzeichnis. Damit können DVDs oder andere Filme ins VCD- oder SVCD-Format konvertiert oder sogar direkt auf eine CD gebrannt werden.

Q: Wie kann ich zwei Videodateien aneinanderhängen?

A: Das wurde schon unglaublich oft auf der MPlayer-Users-Mailingliste diskutiert. Durchsuche das [Archiv](#). Dort findest du vollständige Antworten. Das Aneinanderhängen ist ein sehr komplexes Thema, und dein (Miss-)Erfolg hängt sehr von der Art deiner aneinanderzuhängenden Dateien ab. MPEGs lassen sich noch relativ einfach aneinanderhängen. Für AVIs gibt es zwei Tools, die das eventuell hinkriegen können: [avidemux](#) und `avimerge`, das Teil von [transcode](#) ist. Wenn du zwei Dateien hast, deren Auflösung und Codec identisch sind, dann kannst du eventuell auch MEncoder dafür benutzen:

```
cat file1 file2 > file3
mencoder -ovc copy -oac copy -o out.avi -forceidx file3.avi
```

Q: Mein Tuner funktioniert. Ich höre Ton und sehe Video mit MPlayer, aber MEncoder nimmt keinen Ton auf!

A: TV-Audioaufnahme für Linux ist momentan nicht implementiert. Wir arbeiten daran. Es läuft momentan nur auf BSD.

Q: Ich kann keine DVD-Untertitel in ein AVI kodieren!

A: Du musst die `-sid` Option korrekt verwenden!

Q: MEncoder segfaultet beim Start!

A: Besorg dir neue Versionen von DivX4Linux/XviD/etc.

Q: Wie kann ich nur ein bestimmtes Kapitel von einer DVD kodieren?

A: Verwende die `-chapter`-Option korrekt, z.B. so: `-chapter 5-7`

Q: Ich versuche, mit 2GB+-Dateien auf einem VFAT-Dateisystem zu arbeiten. Geht das?

A: Nein, VFAT unterstützt keine Dateien, welche grösser als 2Gb sind.

Q: Wieso ist die von MEncoder vorgeschlagene Bitrate negativ?

A: Weil die Bitrate, die bei der Audiokonvertierung verwendet wurde, zu hoch war, um den Film auf eine CD bringen. Du solltest überprüfen, ob libmp3lame korrekt installiert wurde.

Q: Wie kann ich MPlayer für eine bestimmte Datei Optionen beibringen, ohne sie jeweils auf der Kommandozeile angeben zu müssen?

A: Schreib all die dateispezifischen Einstellungen in eine Datei namens `movie.avi.conf`, wenn dein Film `movie.avi` heißt, und leg diese Datei nach `~/mplayer` oder in das gleiche Verzeichnis wie der Film selber.

10 Portierungen

10.1 Linux

Die hauptsächliche Entwicklungsplattform ist Linux auf x86, obwohl MPlayer auch auf anderen Linuxportierungen läuft.

10.1.1 Debian-Pakete

Um selber ein Debianpaket zu erstellen, starte den folgenden Befehl im Source-Verzeichnis:

```
user@linux ~/ $ fakeroot debian/rules binary
```

Und nun einfach root werden und:

```
root@linux ~/ # dpkg -i ../mplayer_0.90-1_i386.deb
```

als root.

Christian Marillat hat seit einiger Zeit inoffizielle Debianpakete von MPlayer, MEncoder und den Schriften erstellt. Du kannst sie (per apt-get) von seiner  [Homepage](#) herunterladen. Diese Pakete sind definitiv inoffiziell, da Christian bereits damals diese Pakete erstellt und verteilt hat, als MPlayer noch nicht komplett unter der GPL stand und Binärpakete [<users_against_developers.html#binary>](#) nicht erlaubt waren. Christian ignorierte unsere Bitte, das zu unterlassen, was einiges an bösem Blut zwischen ihm und den MPlayer-Entwicklern hinterließ. Inzwischen sind Binärpakete kein Problem mehr. Trotzdem supporten wir diese Pakete nicht!

10.1.2 RPM-Pakete

Dominik Mierzejewski erstellt und wartet die offiziellen RedHat RPM-Pakete für MPlayer. Diese können von seiner  [Homepage](#) heruntergeladen werden. Bitte lies die Anweisungen auf seiner Seite, und schreib ihm wegen Problemen und nicht uns. Inzwischen gibt es auch andere RPM-Versionen (SuSE liefert MPlayer mit ihrer offiziellen Distribution aus, Mandrake-Pakete gibt es von  [P.L.F](#)) von MPlayer, aber keine dieser Versionen wird offiziell unterstützt.

10.1.3 ARM

MPlayer läuft auf Linux-PDAs mit einer ARM-CPU, z.B. dem Sharp Zaurus oder dem Compaq Ipaq. Am einfachsten ist es, sich MPlayer von  [Openzaurus](#) runterzuladen. Wenn du MPlayer selber Kompilieren möchtest, dann schau dir die Verzeichnisse für  [mplayer](#) und  [libavcodec](#) bei der OpenZaurus-Distribution an. Sie enthalten immer die aktuellen Makefiles und Patches, die benutzt werden können, um die CVS-Version von MPlayer mit [libavcodec](#) zu kompilieren. Wenn du das GUI brauchst, dann kannst du [xmms-embedded](#) benutzen.

11 Encodierung mit MEncoder

11.1 Encodieren mit MEncoder

Eine komplette Auflistung aller MEncoder-Optionen und aller Beispiele findest du in der Manpage. Einige sinnvolle Beispiele und detaillierte Anleitungen, wie die diversen Parameter benutzt werden, findest du in den Encodierungstipps (englisch) <../tech/encoding-tips.txt>, die von mehreren Emails auf der  [mplayer-users](#) zusammengetragen wurden. Durchsuche auch das  [Archiv](#). Dort findest du eine Unmenge an Diskussionen über alle möglichen Themen rund um die Encodierung mit MEncoder.

11.1.1 MPEG4-Encodierung ("DivX") in zwei oder drei Durchläufen

2-pass-Encodierung: Der Name kommt in der Tat daher, dass diese Methode die Datei **zweimal** encodiert. Das erste Encodieren (1. Pass [=Durchlauf]) erzeugt einige temporäre Dateien (.log) mit einer Dateigröße von nur wenigen MB. Lösche sie nicht (das AVI kannst du aber löschen). Im zweiten Pass wird die Ausgabedatei unter Verwendung der Bitraten-Daten der temporären Datei erzeugt. Die endgültige Datei wird deswegen eine weitaus bessere Bildqualität besitzen. Wenn dies das erst Mal ist, dass du davon hörst, solltest du einige Guides im Internet lesen.

Dieses Beispiel zeigt dir, wie du mit 2 Passes eine DVD in ein MPEG4-AVI ("DivX") encodierst. Nur 2 Befehle sind erforderlich: `rm frameno.avi` - Lösche diese Datei, welche von einer früheren 3-Pass-Encodierung kommen kann (Sie führt zu einem Konflikt mit der momentanen Encodierung).

```
root@linux / # mencoder -dvd 2 -lavcopts vcodec=mpeg4:vpass=1 -oac copy
-o film.avi
root@linux / # mencoder -dvd 2 -lavcopts vcodec=mpeg4:vpass=2 -oac copy
-o film.avi
```

3-Pass-Encodierung: Dies ist eine Erweiterung der 2-Pass-Encodierung, in der die Encodierung von Audio in einem separaten Durchlauf erledigt wird. Diese Methode erlaubt es MEncoder, Vorschläge für die Videobitrate zu machen, sodass das Ergebnis auf eine CD passt. Auch wird der Ton im Unterschied zum 2-Pass nur einmal encodiert. Das Schema:

1. Lösche die alte Datei:

```
CODE>rm frameno.avi
```

2. Erster Durchlauf:

```
mencoder <file/DVD> -ovc frameno -oac mp3lame -lameopts vbr=3 -o
frameno.avi
```

Hierbei wird ein AVI erstellt, das nur die gewählte Audiospur enthält. Vergiss nicht, die `-lameopts` deinen Wünschen anzupassen. Wenn du einen langen Film encodierst, so gibt dir MEncoder am Schluss die empfohlenen Videobitraten aus, damit ein resultierender Film auf eine 650MB-, eine 700MB- oder eine 800MB-CD passt.

3. Zweiter Durchgang:

```
mencoder <file/DVD> -oac copy -ovc lavc -lavcopts
vcodec=mpeg4:vpass=1:vbitrate=<bitrate>
```

Genauso wie der erste Durchgang vom DivX4-Beispiel oben. Gib hier die nach dem ersten Durchlauf von MEncoderempfohlene oder eine andere von dir gewählte Videobitrate an.

4. Dritter Durchgang:

```
mencoder <file/DVD> -oac copy -ovc lavc -lavcopts
vcodec=mpeg4:vpass=2:vbitrate=<bitrate>
```

Genauso wie der zweite Durchgang vom DivX4-Beispiel oben. Gib hier die gleiche Bitrate wie im ersten Durchgang an, es sei denn, du weißt genau, was du tust. In diesem Durchgang wird die im ersten Durchgang encodierte Audiospur in die endgültige Datei eingefügt. Nun ist alles fertig!

Beispiel einer 3-Pass-Encodierung:

```
root@linux ~/ # rm frameno.avi
```

Lösche diese Datei, welche von einer früheren 3-Pass-Encodierung kommen kann (Sie führt zu einem Konflikt mit der momentanen Encodierung).

```
user@linux / $ mencoder -dvd 2 -ovc frameno -o frameno.avi -oac mp3lame
-lameopts vbr=3
user@linux / $ mencoder -dvd 2 -ovc lavc -lavcopts vcodec=mpeg4:vpas=1
-oac copy -o movie.avi
user@linux / $ mencoder -dvd 2 -ovc lavc -lavcopts vcodec=mpeg4:vpas=2
-oac copy -o movie.avi
```

11.1.2 In das MPEG-Format encodieren

MEncoder kann MPEG-Dateien (MPEG-PS) erstellen. Das ist wahrscheinlich nur in Kombination mit dem /mpeg1video/-Codec der `libavcodec` sinnvoll, da alle Player (bis auf MPlayer) MPEG1-Video und MPEG1 Layer 2 (MP2)-Audio in MPEG-Dateien erwarten.

Dieses Feature ist momentan nicht wirklich nützlich, mal davon abgesehen, dass es wahrscheinlich noch einige Fehler enthält. Schlimmer ist aber, dass MEncoder momentan kein MPEG1 Layer 2 (MP2)-Audio erzeugen kann, das alle anderen Player in einer MPEG-Datei erwarten.

Das Ausgabeformat von MEncoder kann mit `-of mpeg` auf MPEG geändert werden.

Beispiel:

```
root@linux / # mencoder -of mpeg -ovc lavc -lavcopts vcodec=mpeg1video
-oac copy <weitere Optionen> media.avi -o ausgabe.mpg
```

11.1.3 Skalierung von Filmen

Oftmals ist die Änderung der Bildgröße erforderlich. Die Gründe dafür können verschieden sein: Dateigröße verringern, Netzwerkbandbreite nicht überlasten etc. Viele Leute skalieren das Bild auch, wenn sie von DVD zu SVCD oder DivX-AVI konvertieren. Das ist allgemein schlecht. Wenn du das vermeiden willst, dann lies den Abschnitt über das Beibehalten des Höhen-/Breitenverhältnisses `<aspect>`. Der Skalierungsprozess erfolgt durch den /scale/-Videofilter: `-vop scale=breite:hoehe`. Die Qualität kann durch die Option `-sws` angegeben werden. Wenn nichts angegeben ist, verwendet MEncoder 0: schnelle bilienare Filterung ('fast bilinear'). Verwendung:

```
user@linux ~/ $ mencoder beispiel-svcd.mpg -ovc lavc -lavcopts
vcodec=mpeg4 -vop scale=640:480 -sws 2 -oac copy -o ausgabe.avi
```

11.1.4 Stream-Kopiermodus

MEncoder kann Input-Streams auf zwei Arten bearbeiten: encodieren oder kopieren (copy). Diese Sektion erklärt die Stream-Kopie.

- * Videostreams (Option `-ovc copy`): Es können nette Sachen gemacht werden :)
Wie das kopieren (nicht konvertieren) von FLI- oder VIVO- oder MPEG1-Video in eine AVI-Datei. Natürlich ist nur MPlayer in der Lage, solche Dateien abspielen :) Und vermutlich ist das auch absolut nicht praxisrelevant. Das Kopieren von Videostream kann nützlicher sein, wenn man zum Beispiel nur den Audiostream encodieren will (z.B.: unkomprimiertes PCM zu MP3).
- * Audiostreams (Option `-oac copy`): geht genauso. Es ist möglich, eine externe Audiodatei (MP3, AC3, Vorbis) zu nehmen, und diese dem Ausgabestream hinzuzufügen. Benutze dazu die Option `-audiofile <dateiname>`.

11.1.5 Reparieren von AVIs mit defektem Index oder schlechtem Interleaving

Trivial. Wir kopieren einfach die Video- und Audiostreams, und MEncoder erzeugt den Index. Natürlich kann das keine Fehler in den Video- und/oder Audiostreams beheben. Es kann auch Dateien mit kaputten Interleaving reparieren, sodass die Option `-ni` für diese nicht mehr gebraucht wird.

Befehl:

```
user@linux ~/ $ mencoder -idx eingabe.avi -ovc copy -oac copy -o
ausgabe.avi
```

11.1.5.1 Aneinanderhängen mehrerer AVI-Dateien

Ein Nebeneffekt des oben erwähnten Reparierens ist, dass MEncoder damit auch zwei oder mehrere aneinandergehängte AVI-Dateien reparieren kann:

Kommando:

```
user@linux ~/ $ cat 1.avi 2.avi | mencoder -noidx -ovc copy -oac copy -o
ausgabe.avi -
```

Anmerkung:

Hier wird angenommen, dass `1.avi` und `2.avi` den gleichen Codec verwenden, die gleiche Auflösung und Streamrate besitzen etc., und wenigstens `1.avi` darf nicht defekt sein. Eventuell musst du, wie oben <fixing> beschrieben, die Quelldateien erst reparieren.

11.1.6 Encodierung mit der libavcodec-Codec-Familie

`libavcodec` bietet eine einfache Encodierung für viele interessante Video- und Audio-formate (momentan werden deren Audio-Codecs nicht unterstützt). Du kannst zu folgenden Codecs encodieren:

- * mjpeg - Motion JPEG
- * h263 - H263
- * h263p - H263 Plus
- * mpeg4 - ISO-Standard-MPEG4 (kompatibel zu DivX5, XviD)
- * msmpeg4 - pre-Standard-MPEG4-Variante von MS, v3 (aka DivX3)
- * msmpeg4v2 - pre-Standard-MPEG4-Variante von MS, v2 (wird in alten ASF-Dateien benutzt)

- * rv10 - ein alter RealVideo-Codec
- * mpeg1video - MPEG1-Video :)
- * huffyuv - verlustfreie Kompression

Die erste Spalte enthält den Codec-Namen, den du nach dem `vcodec`-Stichwort übergibst, wie z.B.:
`-lavcopts vcodec=msmpeg4` Ein Beispiel mit MJPEG-Kompression:

```
user@linux ~/ $ mencoder -dvd 2 -o titel2.avi -ovc lavc -lavcopts
vcodec=mjpeg -oac copy
```

11.1.7 Encodierung von mehreren Einzelbildern (JPEGs, PNGs oder TGAs)

MEncoder kann aus mehreren JPEG-, PNG- oder TGA-Einzelbildern einen Film erstellen. Durch einfaches Kopieren der Bilder kann er MJPEG- (Motion JPEG), MPNG- (Motion PNG) oder MTGA-Filme (Motion TGA) erzeugen.

So läuft dieser Prozess ab:

1. MEncoder **decodiert** das/die Eingangsbild(er) mit `libjpeg` (wenn PNGs decodiert werden, wird die `libpng` benutzt).
2. MEncoder führt die decodierten Bilder danach dem gewählten Videokompressor zu (DivX4, Xvid, ffmpeg msmpeg4, etc...).

Beispiele:

Die Erklärung der `-mf`-Option kann in der Manpage gefunden werden.

Erstellung einer DivX4-Datei aus allen JPEG-Dateien im aktuellen Verzeichnis:

```
user@linux ~/ $ mencoder -mf on:w=800:h=600:fps=25 -ovc divx4 -o
ausgabe.avi \.jpg
```

Erstellung einer DivX4-Datei aus einigen JPEG-Dateien im aktuellen Verzeichnis:

```
user@linux ~/ $ mencoder -mf on:w=800:h=600:fps=25 -ovc divx4 -o
ausgabe.avi frame001.jpg,frame002.jpg
```

Erstellung einer Motion JPEG-Datei (MJPEG) aus allen JPEG-Dateien im aktuellen Verzeichnis:

```
user@linux / $ mencoder -mf on:w=800:h=600:fps=25 -ovc copy -o
output.avi\.
```

Erstellung einer unkomprimierten Datei aus allen PNG-Dateien im momentanen Verzeichnis:

```
user@linux / $ mencoder -mf on:w=800:h=600:fps=25:type=png -ovc rawrgb -o
output.avi\.
```

Anmerkung:

Die Breite muss ein Vielfaches von 4 betragen, was eine Beschränkung des RGB-AVI-Formates ist.

Erstellung einer Motion PNG-Datei (MPNG) aus allen PNG-Dateien im momentanen Verzeichnis:

```
user@linux / $ mencoder -mf on:w=800:h=600:fps=25:type=png -ovc copy -o
output.avi\.png
```

Erstellung einer Motion TGA-Datei (MTGA) aus allen TGA-Dateien im aktuellen Verzeichnis:

```
user@linux / $ mencoder -mf on:w=800:h=600:fps=25:type=tga -ovc copy -o
output.avi\.tga
```

11.1.8 Extrahieren von DVD-Untertiteln in eine Vobsub-Datei

MEncoder kann Untertitel von einer DVD extrahieren und sie als Datei im Vobsub-Format speichern. Dieses besteht aus einem Dateipaar (.sub und .idx) und wird gewöhnlich in ein einziges .rar Archiv verpackt. MPlayer kann diese mit den Optionen `-vobsub` und `-vobsubid` wiedergeben. Du gibst mit `-vobsubout` den Basisnamen der Ausgabedatei (ohne .idx- oder .sub-Endung) und mit `-vobsuboutindex` den Index des Untertitels der Ausgabedatei an. Wenn die Eingabe nicht von einer DVD stammt, solltest du mit der Option `-ifo` die Datei angeben, die zur Erstellung der resultierenden .idx-Datei benötigt wird. Wenn die Eingabe nicht von einer DVD stammt und keine .ifo-Datei vorhanden ist, benötigst du die `-vobsuboutid`-Option, um MPlayer zu sagen, welche Sprach-ID für die .idx Datei verwendet werden soll. Jeder Aufruf fügt neue Untertitel dazu, falls die .idx- und .sub-Dateien bereits existieren. Sie sollten im Bedarfsfall also vor dem Starten gelöscht werden.

Beispiele:

Kopieren zweier Untertitel von einer DVD mit 3-Pass-Encodierung

```
user@linux ~/ $ rm untertitel.idx untertitel.sub
user@linux / $ mencoder -dvd 1 -vobsubout untertitel -vobsuboutindex 0
-sid 2 -o frameno.avi -ovc frameno -oac mp3lame -lameopts vbr=3
user@linux / $ mencoder -dvd 1 -oac copy -ovc divx4 -divx4opts pass=1
user@linux / $ mencoder -dvd 1 -oac copy -ovc divx4 -divx4opts pass=2
-vobsubout untertitel -vobsuboutindex 1 -sid 5
```

Kopieren eines französischen Untertitels von einer MPEG-Datei

```
user@linux ~/ $ rm untertitel.idx untertitel.sub
user@linux / $ mencoder film.mpg -ifo film.ifo -vobsubout untertitel
-vobsuboutindex 0 -vobsuboutid fr -sid 1
```

11.1.9 Das Höhen-/Breitenverhältnisses erhalten

DVDs und SVCDs (z.B. MPEG1/2) enthalten einen Wert, der das Verhältnis zwischen der Ausgebehöhe und der Ausgebreite enthält. Der Player sorgt dann durch Skalierung dafür, dass dieses Verhältnis erreicht wird, sodass wir Menschen keine Eierköpfe sehen (Beispiel: die Bildmaße betragen 480x480, das gespeicherte Verhältnis beträgt 4:3, und der Player spielt den Videostream dann bei 640x480 ab). Wenn du allerdings solche Dateien in ein AVI umwandelst, dann musst du beachten, dass das AVI-Format das Höhen-/Breitenverhältnis nicht speichern kann. Die Skalierung beim Encodieren ist ein zeitraubender und qualitätsmindernder Prozess. Es sollte

doch eine bessere Möglichkeit geben!?

Ganz genau - die gibt es. MPEG4 enthält ein einzigartiges Feature: Im Videostream selber kann das Höhen-/Breitenverhältnis gespeichert werden. Jop, genauso wie bei MPEG1/2 (DVD, SVCD). Leider gibt es bisher keinen Videoplayer, der dieses Feature unterstützt. Bis auf MPlayer natürlich.

Dieses Feature kann nur in Verbindung mit dem `mpeg4`-Codec der `libavcodec` genutzt werden. Denke immer daran, dass - auch wenn MPlayer die Datei richtig wiedergeben wird - alle anderen Player ein falsches Höhen-/Breitenverhältnis anzeigen werden.

Du solltest ernsthaft in Erwägung ziehen, die eventuell vorhandenen schwarzen Balken oberhalb und unterhalb des Bildes wegzuschneiden. Lies in der man-Page dazu, wie du die `cropdetect`- und `crop-Filter` benutzt.

Benutzung:

```
user@linux ~/ $ mencoder beispiel-svcd.mpg -ovc lavc -lavcopts
vcodec=mpeg4:aspect=16.0/9.0 -vop crop=714:548:0:14 -oac copy -o
ausgabe.avi
```

12 Bekannte Probleme

Spezielle System-/CPU-spezifische Fehler/Probleme:

- * SIGILL (signal 4) auf einem P3 mit 2.2.x Kernel:
Problem: Kernel 2.2.x hat keine richtige SSE-Unterstützung **Lösung:** update den Kernel auf 2.4.x
Abhilfe: `./configure --disable-sse`
- * Allgemein SIGILL (signal 4):
Problem: du hast MPlayer auf einem anderem Computer kompiliert, als auf dem du abspielst (z.B. kompiliert auf einem P3 und Wiedergabe auf einem Celeron)
Lösung: kompiliere MPlayer auf dem Rechner, auf dem du abspielst.
Abhilfe: `./configure --disable-sse` usw.
- * "Internal buffer inconsistency" wird während der Verwendung von MEncoder:
Problem: bekanntes Problem, wenn lame < 3.90 mit gcc 2.96 oder 3.x.x kompiliert wurde.
Lösung: Es sollte lame >=3.90 verwendet werden.
Abhilfe: Kompiliere lame mit gcc 2.95.x und deinstalliere alle installierten lame-Pakete, welche mit gcc 2.96 kompiliert worden sein könnten.

Verschiedene A-V Sync- und andere Audio-Probleme

Allgemein Audio-Verzögerung und ruckartiger Sound (kommt bei allen oder einigen Dateien vor):

- * am häufigsten: fehlerhafte Audio-Treiber! Probiere verschiedene Audio-Treiber aus; probiere ALSA 0.9 OSS-Emulation mit `-ao oss`. Manchmal hilft auch `-ao sdl`. Wenn die Datei mit `-nosound` ohne Probleme wiedergegeben wird, dann liegt höchstwahrscheinlich ein Soundkartentreiber-Problem vor.
- * Audio-Pufferprobleme (Puffergröße schlecht erkannt)
Abhilfe: `mplayer -abs`
- * Samplerate Probleme - vielleicht wird die Samplerate in den benutzten Dateien von der Karte nicht unterstützt. Hier sollte das Resampling-Plugin (`-aop`) getestet werden
- * Langsamer Rechner (CPU oder Grafik)
versuche es mit `-vo null`. Wenn es gut funktioniert, hast du eine langsame VGA-Karte oder einen langsamen Treiber. **Abhilfe:** kaufe eine schneller Karte oder lies die Dokumentationen, wie man die Geschwindigkeiten erhöht.
Versuche auch `-framedrop`

Audio (verschiebt sich verliert die Synchronisation bricht nach einer gewissen Zeit ab) bei einer oder nur einigen wenigen Dateien:

- * defekte Datei
Abhilfe:
 - * `-ni` oder `-nobps` (für nicht-interlaced oder defekte Dateien) und/oder
 - * `-mc 0` (erforderlich bei Dateien mit schlechtem Interleave und VBR-Audio) und/oder
 - * `-delay` Option oder +/- Tasten beim Abspielen

Wenn keine dieser Optionen hilft, kann die Datei auf den Server geladen werden. Wir werden sie untersuchen (und das Problem lösen).
- * deine Soundkarte unterstützt keine 48KHz-Wiedergabe
Abhilfe: kaufe eine bessere Soundkarte... versuche die Bilder/Sekunde um 10% zu vermindern (verwende `-fps 27` bei 30fps-Filmen) oder verwende das Resampler-Plugin.
- * langsamer Rechner
(wenn A-V nicht ungefähr 0 ist und die letzte Nummer in der Status-Leiste steigt)
Abhilfe: `-framedrop`

überhaupt kein Ton:

- * deine Datei verwendet eine nicht unterstützten Audio-Codec
Abhilfe: lese die Dokumentationen und hilf uns, dass wir ihn in Zukunft unterstützen.

überhaupt kein Bild (nur ein graues/grünes Fenster):

- * deine Datei verwendet einen nicht unterstützten Videocodec
Abhilfe: lies die Dokumentation und hilf uns, den Codec zu unterstützen
- * der automatisch gewählte Codec kann die Datei nicht decodieren, versuche, einen anderen mit den `-vc` oder `-vfm` Optionen zu wählen.
- * Es wird versucht eine DivX 3.x Datei mit dem OpenDivX- oder XviD-Decoder (`-vc odivx`) abzuspielen - installiere Divx4Linux und kompiliere MPlayer neu.

Videoausgabe-Probleme:

Erster Hinweis: Die Optionen `-fs`, `-vm` und `-zoom` sind nur Empfehlungen, sie werden (noch) nicht von allen Treiber unterstützt. Also ist dies kein Fehler (Bug), wenn es nicht funktioniert. Nur einige Treiber unterstützen Scaling/Zooming, erwarte das nicht von x11 oder dga.

Zucken des OSD/Untertiteln:

- * x11 Treiber: Sorry, es kann zur Zeit nicht gelöst werden
- * xv Treiber: verwende die `-double`-Option Grünes Bild bei der Verwendung von `mga_vid` (`-vo mga / -vo xmga`):
- * `mga_vid` hat die Grösse des RAMs falsch erkannt. Lade das Modul neu mit der Option `mga_ram_size`.

13 Anhang "Aufschrei der Entwickler"

13.1 Aufschrei der Entwickler

Es gibt zwei Themen, die immer zu großen Streitereien und Beschimpfungen auf der  [mplayer-users](#) Mailingliste führen. Das erste Thema dreht sich um den...

13.1.1 GCC 2.96

Zum Hintergrund: Die Serie **2.95** des GCC ist der offiziell GNU-Release, und Version 2.95.3 ist die stabilste und fehlerfreieste aus dieser Serie. Wir haben niemals Probleme beobachten können, die auf den GCC 2.95.3 zurückzuführen waren. Beginnend mit RedHat Linux 7.0 begann **Red Hat** damit, eine stark veränderte CVS-Version des GCC mitzuliefern. Diese Version nannten sie **2.96**. Red Hat hat diese Version aufgenommen, weil sie einen Compiler brauchten, der auf all ihren unterstützten Plattformen lief (welche auch IA64 und s390 einschloss), und weil der offizielle GCC 3.0 zu diesem Zeitpunkt noch nicht fertiggestellt war. Der Linuxdistributor **Mandrake** folgte bald darauf Red Hats Beispiel und lieferte ab Linux-Mandrake 8.0 ebenfalls den GCC 2.96 aus.

Die Aussagen zu dem Thema: Das GCC-Team hat jegliche Verbindung zu der Version 2.96 bestritten und dazu eine  [offizielle Stellungnahme](#) abgegeben. Viele Entwickler auf der ganzen Welt trafen auf Probleme mit dem GCC 2.96 und empfahlen deswegen andere Compilerversionen. Beispiele dafür sind  [MySQL](#),  [avifile](#) und  [Wine](#). Andere interessante Links sind der  [Linux kernel news flash über den Kernel 2.4.17](#) und das  [Voy-Forum](#). MPlayer war ebenfalls von vorübergehenden Problemen betroffen, die sich alle lösten, sobald eine andere Version des GCC benutzt wurde. Viele Projekte begannen daraufhin damit, um einige der Probleme mit dem GCC 2.96 herumzuarbeiten, aber wir lehnten es ab, die Probleme zu beheben, die andere Leute durch vorschnelles Handeln verursacht hatten. Dazu kommt, dass einige dieser Workarounds zu Performanceeinbußen führten.

GCC 2.96 erlaubt keine |(Pipezeichen) in Assemblerkommentaren, weil er sowohl die Intel- als auch die AT&T-Assemblersyntax unterstützt und das |-Zeichen ein Symbol in der Intelvariante darstellt. Das Problem lag nun darin, dass der GCC **kommentarlos** den kompletten Assemblerblock ignoriert hat. Dieser Fehler wurde inzwischen angeblich behoben. GCC gibt eine Warnung aus, anstatt den kompletten Block einfach unter den Tisch fallen zu lassen.

Die Gegenwart: Red Hat behauptet, dass GCC Version 2.96-85 und neuer keine Fehler mehr enthalten. Das Verhalten dieser Version hat sich tatsächlich deutlich verbessert. Nichts desto trotz werden auf unseren Mailinglisten noch immer Probleme berichtet, die verschwinden, sobald ein anderer Compiler verwendet wird. Sei wie es ist, es ist inzwischen einfach nicht mehr wichtig. Hoffentlich löst eine gereifter GCC 3.x all dieses Problem ein für alle mal. Wenn du wirklich mit dem GCC 2.96 kompilieren möchtest, dann benutze die Option `--disable-gcc-checking` bei `configure`. Denk aber daran, dass du dann auf dich allein gestellt bist. **Schick keine Fehlerberichte!** Solltest du das doch tun, so wirst du nur von der Mailingliste verbannt, weil wir wirklich mehr Flamewars wegen des GCC 2.96 erlebt haben als nötig wär. Lass dieses Thema bitte ruhen.

Wenn du Probleme mit dem GCC 2.96 hast, so kannst du Pakete für die Version 2.96-85 auf [Red Hats FTP-Server](#) finden. Andererseits kannst du auch einfach die Pakete für die Version 3.0.4 benutzen, die Red Hat für Red Hat Linux 7.2 und neuer anbietet. Eine weitere Möglichkeit besteht darin, Pakete für [gcc-3.2-10](#) herunterzuladen (inoffiziell, aber sie funktionieren trotzdem einwandfrei). Sie lassen sich neben dem GCC 2.96 installieren, den du bereits hast. MPlayer wird automatisch Version 3.2-10 finden und diesen GCC anstelle der Version 2.96 benutzen. Wenn du aus irgendeinem Grund die binären Pakete für den GCC nicht benutzen kannst oder willst, dann folgt hier eine kleine Anleitung, wie du den neuesten GCC kompilieren kannst:

1. Lade dir `gcc-core-XXX.tar.gz` von einem der  [GCC-Mirrorseiten](#) herunter, wobei `XXX` die Versionsnummer darstellt. Dieses Paket beinhaltet den kompletten C-Compiler und reicht für MPlayer aus. Wenn du darüber hinaus Unterstützung für C++, Java oder andere Features des GCC benötigst, dann ist `gcc-XXX.tar.gz` besser für dich geeignet.
2. Entpacke das Archiv:

```
tar -xvzf gcc-core-XXX.tar.gz
```
3. Anders als die meisten Programme wird der GCC nicht innerhalb des Quelltextverzeichnisses gebaut, sondern er benötigt dafür ein spezielles Buildverzeichnis außerhalb des Quelltextbaumes. Erstell solch ein Verzeichnis mit

```
mkdir gcc-build
```
4. Jetzt kannst du den GCC im Buildverzeichnis konfigurieren lassen aber das `configure`-Script liegt natürlich im Quelltextverzeichnis:

```
cd gcc-build ../gcc-XXX/configure
```
5. Kompiliere GCC mit dem folgenden Kommando im Buildverzeichnis:

```
make bootstrap
```
6. Jetzt kannst du (wenn du root bist) den GCC mit diesem Kommando installieren:

```
make install
```

13.1.2 Vorkompilierte (binäre) Pakete

Früher enthielt MPlayer Teile des Quelltextes des OpenDivX- Projektes, welches es verbietet, vorkompilierte Pakete zu verteilen. Diese Codeabschnitte wurden aber in Version 0.90pre1 entfernt, und die letzte noch verbleibende Datei `divx_vbr.c`, die noch auf den OpenDivX-Quellen aufbaut, wurden von den Autoren unter die GPL gestellt (Version 0.90pre9). Du darfst jetzt also nach Herzenslust binäre Pakete bauen und verteilen.

Ein weiteres Hindernis für Binärpakete waren die bei der Kompilierung automatisch erkannten Optimierungsmöglichkeiten seitens der CPU-Architektur (MMX, 3DNOW etc.). MPlayer unterstützt inzwischen aber auch die Erkennung der CPU-Features beim Starten von MPlayer, wenn `configure` mit der Option `--enable-runtime-cpudetection` aufgerufen wurde. Diese Option ist standardmäßig deaktiviert, weil sie eine kleine negative Auswirkung auf die Geschwindigkeit mitbringt. Andererseits ist es mit ihr nun möglich, Binärpakete zu erstellen, die auf verschiedenen Mitgliedern der Intel-CPU-Familie beschleunigt laufen.

13.1.3 nVidia-Probleme

Uns missfällt die Tatsache, dass  [nVidia](#) nur binäre Treiber für XFree86 zur Verfügung stellt, die oft genug auch noch einige Fehler enthalten. Auf  [mplayer-users](#) sehen wir viele Fehlermeldungen, die mit diesen Closed-Source-Treibern zusammenhängen: über die allgemein schlechte Qualität der Treiber, über Instabilitäten und über den schlechten Support der Endbenutzer durch nVidia. Einige Beispiele dafür kannst du im  [nVidia-Linux-Forum](#) finden. Viele dieser Fälle sind wiederkehrende Probleme. nVidia hat letzgens Kontakt mit uns aufgenommen und behauptet, dass ihre Treiber keine Fehler enthielten, sondern dass die Instabilitäten von schlechten AGP-Chips verursacht würden, und dass sie keinerlei Fehlerberichte von Nutzern erhalten hätten (wie z.B. die lila Linien). Wenn du also ein Problem mit deiner nVidia-Karte hast, dann solltest du auf jeden Fall die neuesten nVidia- Treiber ausprobieren und/oder ein neues Motherboard kaufen oder aber nVidia darum bitten, dass sie OpenSource-Treiber veröffentlichen. Wie dem auch sei - wenn du die binären nVidia-Treiber benutzt und Treiberprobleme auftreten, dann sei gewarnt, dass du von uns nur sehr wenig Hilfe erhalten wirst, weil wir da einfach nichts tun können, um dir zu helfen.