

Rainer Hattenhauer

Das Linux-Video-Buch



Alle in diesem Buch enthaltenen Programme, Darstellungen und Informationen wurden nach bestem Wissen erstellt und mit Sorgfalt getestet. Dennoch sind Fehler nicht ganz auszuschließen. Aus diesem Grund ist das in dem vorliegenden Buch enthaltene Programm-Material mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Autoren und die SuSE Linux AG übernehmen infolgedessen keine Verantwortung und werden keine daraus folgende Haftung übernehmen, die auf irgendeine Art aus der Benutzung dieses Programm-Materials, oder Teilen davon, oder durch Rechtsverletzungen Dritter entsteht.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Buch berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann verwendet werden dürften.

Alle Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt und sind möglicherweise eingetragene Warenzeichen. Die SuSE Linux AG richtet sich im Wesentlichen nach den Schreibweisen der Hersteller. Andere hier genannte Produkte können Warenzeichen des jeweiligen Herstellers sein.

Dieses Werk ist urheberrechtlich geschützt.

Alle Rechte, auch die der Übersetzung, des Nachdruckes und der Vervielfältigung des Buches, oder Teilen daraus, vorbehalten. Kein Teil des Werkes darf ohne schriftliche Genehmigung des Verlages in irgendeiner Form (Druck, Fotokopie, Microfilm oder einem anderen Verfahren), auch nicht für Zwecke der Unterrichtsgestaltung, reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Bibliografische Information Der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.
ISBN 3-935922-62-0

© 2003 SuSE Linux AG, Nürnberg (<http://www.suse.de>)

Umschlaggestaltung: Fritz Design GmbH, Erlangen

Gesamtlektorat: Nicolaus Millin

Fachlektorat: Remo Behn, Joerg Dippel, Torsten Duwe, Egbert Eich, Dr. Matthias Nagorni, Dirk Pankonin, Heiko Rommel, Stephan Skrodzki, Sascha Wessels

Satz: \LaTeX

Druck: Kösel, Kempten

Printed in Germany on acid free paper.

Vorwort

Die Faszination, die von bewegten Bildern ausgeht, zieht die Menschen seit der Erfindung des Kinos durch die Gebrüder Lumière in ihren Bann. Galt es in der damaligen Zeit noch, ein mehr oder weniger perfektes Abbild der Realität zu schaffen, so ist man heute bestrebt, eigene (virtuelle) Welten in Szene zu setzen. Die moderne Videodigitaltechnik ermöglicht dabei die Umsetzung von Filmprojekten, die noch vor 10 Jahren als nicht realisierbar galten. Doch nicht nur in den großen Filmstudios, auch in vielen Privathaushalten hat der PC als Videostudio Einzug gehalten, sei es als digitaler Videorecorder oder Schnittwerkzeug für den selbstgedrehten Camcorderfilm. Moderne Kompressionsalgorithmen sorgen dafür, dass ein mehrstündiger Kinofilm auf eine flache Plasticscheibe namens DVD bzw. CD-R gebracht werden kann und somit hochtransportabel wird; in den letzten zwei Jahren hat der Heim-DVD-Player einen ähnlich rasanten Absatz zu verzeichnen wie der PC in den 80er Jahren.

Linux und Video – diese Kombination taucht in den gängigen Computermagazinen und Internetforen in jüngster Zeit verstärkt auf. Ein Artikel über Videokompression unter Linux hier, die Anleitung zum Bau eines PVRs (Personal Video Recorders) dort, aber leider kein zusammenfassender Rundumschlag... und genau an dieser Stelle möchte das vorliegende Buch ansetzen: Aus der Vielzahl der faszinierenden Videosoftware-Projekte, die derzeit im Linux-Umfeld wie Pilze aus dem Boden schießen, soll eine repräsentative Auswahl vorgestellt werden. Die Qualität, welche bei der Umsetzung der verschiedenen Applikationen erreicht wurde, muss dabei kaum den Vergleich mit dem „professionellen Lager“ scheuen. Das vorliegende Buch deckt im Wesentlichen die folgenden Themenkomplexe ab:

- Wiedergabe von Videomaterial
- Erstellung von Videomaterial
- Publikation von Videomaterial

Sowohl der Einsteiger als auch der Fortgeschrittene soll dabei fündig werden; für letzteren bietet Linux bekanntermaßen den Vorteil, die Software eigenen An-

forderungen anzupassen. Aus diesem Grund wird auch besonderes Augenmerk auf die Erstellung der Software „von Hand“ (sprich durch das Kompilieren der Quellen) gelegt, um eine optimale Systemanpassung zu erzielen.

Von der Idee zum Buch – dieser Weg ist steinig und ohne die Hilfe einer Vielzahl von engagierten Menschen nicht zu bewältigen. An erster Stelle sei hier Herr Nicolaus Millin von SuSE PRESS genannt, welcher mir sehr viel Freiheit bei der Realisierung des Buchs gelassen und die wichtigen Kontakte für mich geknüpft hat, ohne die es in unserer heutigen vernetzten Welt nicht geht. Klaus Schmidinger möchte ich für seinen unermüdlichen Einsatz in Bezug auf die Programmierung und Pflege der `vdr`-Software danken, bei der es sich IMHO um ein Projekt handelt, von dem man in Zukunft sicher noch viel hören bzw. lesen wird. Der Firma MainConcept danke ich für die Bereitstellung des hervorragenden Programms *MainActor* als Beigabe zur Begleit-CD und ihre bisherigen Anstrengungen, den Videoschnitt unter Linux hoffähig zu machen. Die Firma Pinnacle Systems hat mir dankenswerterweise Hardware zum Testen zur Verfügung gestellt, so dass der geneigte Leser nunmehr auch erfährt, wie hardwareunterstützte Encodierung unter Linux zu realisieren ist. Meinen „Testlesern“ Remo Behn, Joerg Dippel, Torsten Duwe, Egbert Eich, Dr. Matthias Nagorni, Dirk Pankonin, Heiko Rommel, Stephan Skrodzki und Sascha Wessels danke ich für die kritischen und konstruktiven Kommentare zum Manuskript, die dazu beigetragen haben, das Buch orthografisch, syntaktisch und semantisch zu „debuggen“. Ein Sonderlob gebührt Bernhard Kaindl und Stefan Reinauer für die Zusammenstellung und den finalen Test von Softwarepaketen für die Begleit-CD.

Ohne gewisse Entspannungsinseln lässt sich Kreativität auf die Dauer nicht aufrecht erhalten. Mein besonderer Dank gilt daher der Honda Motor Co., Inc. für die Entwicklung von entspannungsfördernder Hardware in Form des Modells XRV 750, sowie last but not least meiner Sozia Reinhild, die mich durch so manche Schräglage des Lebens begleitet und glücklicherweise ein hohes Maß an Verständnis für meine schriftstellerische Nebentätigkeit aufgebracht hat.

Bad Sachsa, im Februar 2003

Rainer Hattenhauer

Inhaltsverzeichnis

1	Einleitung	1
1.1	Video und Linux – geht das ?	1
1.2	Inhalt des Buches	2
1.3	Anforderungen an den Leser	3
1.4	Schreibkonventionen	4
1.5	Zum Inhalt der Begleit-CD	5
2	Ein Videobasissystem für Linux	9
2.1	Allgemeines: PC und Video	9
2.2	Vorbereitungen zur Installation	11
2.2.1	Partitionierung der Festplatte(n)	11
2.2.1.1	Während der Grundinstallation	12
2.2.1.2	Nachträgliche Partitionierung	13
2.2.1.3	Einbinden weiterer Platten	15
2.3	Integration von Hardware	16
2.3.1	Inbetriebnahme einer Analog-TV-Karte	17
2.3.2	Test der Analog-TV-Karte	19
2.3.3	Anpassen der Standard-Software	21
2.3.3.1	TV-Empfang mit <i>xawtv</i>	21
2.3.3.2	<i>xawtv</i> als Videorecorder	23
2.4	Softwareinstallation: Praktische Tipps	25
2.4.1	RPM-Installation	26
2.4.1.1	Installation mit dem Paketmanager	26

2.4.1.2	Installation über die Konsole	. . .	27
2.4.2	Kompilieren von Quellcode	28
2.4.2.1	Deinstallation mit <code>uninstall</code>	. . .	28
2.4.2.2	Bauen eigener RPM-Pakete	29
2.4.2.3	Überwachung mit <code>checkinstall</code>	. . .	29
2.5	Kernelanpassung	30
2.5.1	Vorarbeiten	31
2.5.2	Konfiguration	32
2.5.3	Kompilierung und Installation	33
2.5.4	Installation des SuSE-Videokernels	34
2.6	Hardwareoptimierung	35
2.6.1	Grafikkarte optimal einbinden	35
2.6.2	DMA-Modus aktivieren	37
3	Linux als Multinorm-Videoplayer		41
3.1	Videokompression	41
3.1.1	Grundlagen	41
3.1.2	Verbreitete Codecs / Standards	43
3.2	Abspielen von Videodateien	45
3.2.1	Videostreaming	45
3.2.1.1	Der Real-Player	46
3.2.1.2	Quicktime und Windows Media	48
3.2.1.3	Open Source Lösung	48
3.2.1.4	Kommerzielle Lösung	49
3.2.2	Videoplayer unter Linux	53
3.2.2.1	Der <i>MPlayer</i>	54
3.2.2.2	Der <i>xine</i> -Player	60
3.2.2.3	Der Rest der Playerwelt	62
3.2.3	Die Win32 / Avifile-Bibliotheken	62
3.3	Linux als DVD-Player	63
3.3.1	Geschichte und Technik der DVD	64
3.3.2	DVDs unter Linux abspielen	66
3.3.2.1	<i>xine</i> und <i>MPlayer</i>	68

3.3.2.2	<i>ogle</i>	69
3.3.2.3	VideoLAN	72
3.3.2.4	Verschlüsselte DVDs	73
3.3.2.5	Ansätze kommerzieller Lösungen	74
3.3.3	DVD-Tuning	75
3.4	Videoausgabe an externe Geräte	77
3.4.1	Manuelle Anpassung der <i>XF86Config</i>	77
3.4.2	Das <i>nv-tv</i> -Tool	80
4	Linux als Analog-zu-Digital Videorecorder	85
4.1	Allgemeines zur Videoaufzeichnung	85
4.2	Videorecording	86
4.2.1	Capturen mit <i>avifile</i> -basierender Software	87
4.2.1.1	Systemvorbereitung	87
4.2.1.2	<i>avicap</i>	88
4.2.1.3	Das <i>vcr</i> -Tool	90
4.2.1.4	Der digitale Videorecorder <i>DVR</i>	92
4.2.2	Capturen mit den <i>MJPEG</i> -Tools	94
4.2.2.1	Kompilieren der Quellen	94
4.2.2.2	Arbeiten mit den <i>MJPEG</i> -Tools	96
4.2.3	Videoaufzeichnung mit integrierter Software	99
4.2.3.1	Capturen mit <i>Cinelerra</i>	99
4.2.3.2	Videorecording mit <i>MainActor</i>	101
4.3	Hardwareunterstützte Videoaufzeichnung	103
4.3.1	Einbinden der Hardware	104
4.3.2	Einsatz der <i>MJPEG</i> -Tools	107
4.4	Einbinden externer Videoquellen	110
4.4.1	Überspielen von Camcordern	110
4.4.2	Digitale Archivierung von VHS-Material	112
4.5	Optimierung des Aufnahmesystems	113
4.5.1	Dateigrößenprobleme	113
4.5.2	Zeitgesteuerte Aufnahmen	115

5	Digitale Videoaufzeichnung unter Linux	119
5.1	Grundlagen der DVB-Technik	120
5.2	Aufbau eines DVB-Systems	122
5.2.1	Hardwareauswahl und -integration	122
5.2.2	Erster Test / Einfache Programme	126
5.3	Linux als Digitalvideorecorder	128
5.3.1	Installation der <i>vdr</i> -Software	129
5.3.2	Bedienung des <i>vdr</i> -Systems	132
5.3.2.1	Besonderheiten der <i>kvdr</i> -Oberfläche	132
5.3.2.2	Menüstruktur von <i>vdr</i>	132
5.3.3	Videoaufzeichnung mit <i>vdr</i>	136
5.3.4	<i>vdr</i> für Fortgeschrittene	140
5.3.4.1	Alternative Ausgabemöglichkeiten	140
5.3.4.2	Integration einer Fernbedienung	141
5.3.4.3	<i>vdr</i> als Multimediacenter	145
5.3.4.4	Timeshifting	148
5.3.4.5	Remoteadministration mit <i>vdradmin</i>	149
5.4	FireWire (IEEE 1394)	152
5.4.1	Technische Spezifikation	152
5.4.2	Hardwareintegration	153
5.4.3	Anwendungssoftware	155
6	Videobearbeitung unter Linux	157
6.1	Grundsätzliches zum Videoschnitt	158
6.2	Videoschnitt mit <i>vdr</i>	159
6.3	Komplexe Schnittprogramme	164
6.3.1	Videoschnitt mit <i>Cinelerra</i>	164
6.3.1.1	Schnitt eines Einzelfilms	165
6.3.1.2	Zusammenfügen mehrerer Clips	167
6.3.2	Videoschnitt mit <i>MainActor</i>	168
6.4	Überblenden und Filtern von Videomaterial	170
6.4.1	Videoeffekte mit <i>MainActor</i>	170
6.4.1.1	Szenenübergänge	170

6.4.1.2	Texte in den Film einfügen	171
6.4.1.3	Einsatz von Filtern	174
6.4.2	Videoeffekte mit <i>Cinelerra</i>	177
6.4.2.1	Szenenübergänge mit <i>Cinelerra</i>	177
6.4.2.2	Einbinden von Texten und Grafiken	178
6.4.2.3	Anwendung der Videofilter	179
6.5	Export des Materials	181
6.5.1	Das <i>MainActor</i> Exportmodul	181
6.5.2	Export aus <i>Cinelerra</i>	183
6.6	Tongestaltung	185
6.6.1	Vertonung mit <i>Cinelerra</i>	185
6.6.1.1	Musikalische Untermalung	185
6.6.1.2	Einbinden von Sprache	186
6.6.2	<i>MainActor</i> als Soundsequencer	187
7	Kompression und Transcodierung von Videos	191
7.1	Historisches und Stand der Dinge	191
7.2	Transcodierung unter Linux	193
7.2.1	Basiswerkzeuge zur Transcodierung	194
7.2.1.1	Installation von Codecs	194
7.2.1.2	Der <i>mencoder</i>	196
7.2.1.3	<i>transcode</i>	200
7.2.1.4	Spezialskripte und Hilfsprogramme	204
7.2.2	Grafische Transcodieroberflächen	205
7.2.2.1	Der <i>kmencoder</i>	206
7.2.2.2	<i>avidemux</i>	207
7.3	Spezielle Transcodierungsprojekte	213
7.3.1	Archivierung von VDR-Streams in DivX	213
7.3.2	Sicherheitskopien von DVDs mit <i>dvd::rip</i>	219
7.3.3	Herstellung von VCD- und SVCD-Material	223
7.3.3.1	Verwendung der <i>MJPEG-Tools</i>	225
7.3.3.2	Verwendung von <i>transcode</i>	227
7.3.3.3	DVDs in SVCDs umwandeln	228

7.4	Einsatz besonderer Werkzeuge	229
7.4.1	Produktion von <i>Real</i> -Streams	230
7.4.2	Einsatz von Windows Software unter Linux	231
7.4.3	Videoencoding im Cluster	234
8	Archivierung von Videomaterial	241
8.1	CD-R: Die günstige Alternative	242
8.1.1	Hardwareintegration	242
8.1.2	Kommandozeilentools	243
8.1.3	Grafische Tools	245
8.1.3.1	X-CD-Roast	245
8.1.3.2	<i>KOnCD</i>	248
8.1.3.3	<i>CD Bake Oven</i>	249
8.1.3.4	<i>k3b</i>	249
8.1.4	CD-R Archivierungsstrategien	251
8.1.5	Brennen von (S)VCDs	253
8.2	DVDs unter Linux erstellen	255
8.2.1	Eine Zwischenlösung: Die miniDVD	255
8.2.2	Der Weg zur DVD	259
8.3	Einrichtung eines LAN-Videospeichers	261
8.3.1	Hardwareauswahl	261
8.3.2	Konfiguration als NFS-Share	262
8.3.3	Konfiguration als Samba-Share	264
9	Soundcheck: Audio-Optimierung	269
9.1	Soundsysteme und Soundkonzepte	269
9.1.1	Geschichtliches und Technisches	269
9.1.2	Soundkarten unter OSS	271
9.1.3	Soundkarten unter ALSA	272
9.1.4	Soundkarten einbinden und steuern	273
9.2	Audiokompression	278
9.2.1	Grundlagen	278
9.2.2	Audiokompression mit <i>Lame</i>	279

9.2.3	<i>Ogg-Vorbis</i> -Kompression	281
9.2.4	Integration in die KDE-Umgebung	283
9.3	Soundeditoren	284
9.3.1	Tonaufzeichnung	285
9.3.2	Tonbearbeitung	286
9.4	Mehrkanal-Sound unter Linux	287
9.4.1	Aufbau der Hardware	288
9.4.2	Mehrkanal-Sound unter ALSA	289
9.4.3	Mehrkanal-Sound unter OSS	292
10	Linux als Videoserver	297
10.1	Grundlagen der Netzwerktechnik	297
10.1.1	Konfiguration der Hardware	297
10.1.2	Konfiguration der Netzwerkdienste	299
10.1.3	Videostreaming: Eine Klassifizierung	301
10.2	Video-on-Demand	302
10.2.1	Technische Voraussetzungen	302
10.2.2	Netzwerkanalyse	303
10.2.3	Lokale Bereitstellung von On-Demand-Material	305
10.3	DVB-Streaming: Vom Satellit ins Netz	308
10.3.1	Das <i>dvbstream</i> -Paket	308
10.3.2	Die <i>ffmpeg</i> -Tools	313
10.4	Das <i>VideoLAN</i> -Projekt	317
10.4.1	Der VideoLAN Client	318
10.4.2	Der VideoLAN Miniserver	319
10.4.3	Der VideoLAN Server	320
10.4.3.1	Kompilierung und Installation	320
10.4.3.2	Konfiguration	321
10.4.3.3	Interaktion und Streaming	324
10.4.3.4	Multicasting	326
10.4.3.5	Der VideoLAN Channel Server	327
10.5	Der drahtlose Videoserver	329
10.5.1	Funkübertragung von Videosignalen	329

10.5.2	Erweiterung des Videonetzes mit WLAN-Technik	331
10.5.2.1	WLAN: Ein kurzer Überblick	331
10.5.2.2	Integration eines Access Points	332
10.5.2.3	Konfiguration der Clients	334
10.5.2.4	Übertragung von Videoinhalten im WLAN	336
11	Tipps, Tricks & Troubleshooting	339
11.1	Hardwareauswahl für Linux Videosysteme	339
11.1.1	Der Multifunktionsplayer fürs Wohnzimmer	340
11.1.2	Videoschnitt- und Transcodiersysteme	342
11.1.3	VideoLAN-Clients	343
11.2	Informationsquellen im Netz	344
11.2.1	Allgemeine Hinweise	345
11.2.2	Themenbezogene Informationsquellen	346
11.2.2.1	Rund um das System und die Hardware	346
11.2.2.2	Videoplayer und Videocodecs	347
11.2.2.3	Spezielle Videohardware bzw. Systeme	347
11.2.2.4	Videoschnitt	348
11.2.2.5	Transcodierung / Encodierung	348
11.2.2.6	Videonetze unter Linux	349
11.2.2.7	Vorkompilierte Softwarepakete	349
11.3	Strategien zur Problemlösung	350
11.3.1	Hardwareprobleme	350
11.3.1.1	Beispiel 1: Ruckelnde DVD-Wiedergabe	350
11.3.1.2	Beispiel 2: Video Out eines Laptops nutzen	351
11.3.2	Softwareprobleme	352
11.3.2.1	Beispiel 1: <i>MPlayer</i> kompilieren	352
11.3.2.2	Beispiel 2: <i>vlc</i> , SuSE und DivX / XviD	354
11.3.2.3	Beispiel 3: <i>vdr</i> -Entwicklerversion kompilieren	355
12	Glossar	357

Kapitel 1

Einleitung

1.1 Video und Linux – geht das ?

Die digitale Videodatenverarbeitung ist ein junger Stern am Himmel der EDV. Nach und nach lösen CD und DVD alteingesessene Speichermedien wie die VHS-Videokassette ab und stellen dem Anwender nie gekannte Bild- und Tonqualität in Aussicht. Durch die Verwendung moderner Kompressionsalgorithmen ist die Erstellung einer eigenen digitalen Videothek mit mehreren hundert Stunden Videomaterial leicht auf einer handelsüblichen Festplatte zu realisieren. Via LAN-Streaming kann ein dedizierter Server einem ganzen Computernetz Videomaterial zur Verfügung stellen und somit als hauseigene Videothek agieren, vorausgesetzt, man verfügt über eine entsprechend leistungsfähige Softwareumgebung.

Zu Unrecht führt das freie Betriebssystem Linux bislang ein Schattendasein im Bereich Video / Multimedia, bringt es doch mit seiner sprichwörtlichen Stabilität und Flexibilität wesentliche Voraussetzungen für professionelle Bilddatenverarbeitung von Haus aus mit. Andererseits dürfte es sich mittlerweile herumgesprochen haben, dass sogar renommierte Hollywood-Regisseure kostengünstige Linux-Cluster zur Realisierung von Spezialeffekten einsetzen.

Das vorliegende Buch möchte dem computerinteressierten Cineasten (oder dem videointeressierten Computeranwender) Wege aufzeigen, eine Linux-Workstation als digitales Multizweck-Videosystem zu nutzen. Insbesondere sollen dabei die besonderen Stärken des freien Betriebssystems ausgespielt werden: Der Einsatz frei erhältlicher Software, die professionellen Lösungen in nichts nachsteht und die Skalierbarkeit, die es gestattet, Lösungen für komplexe Anforderungen aus dem Einfachen heraus zu entwickeln. Der Leser muss zur Umsetzung der vorgestellten Projekte kein Linux-Profi sein, gleichwohl aber über ein funktionierendes Linux-Basissystem auf zumindest einer Partition seines PC verfügen. Selbst für Anfänger sollte das Aufsetzen eines solchen Grundsystems mit moder-

nen Distributionen mittlerweile in unter 30 Minuten erledigt sein. Grundlage des Buches war eine SuSE Linux 8.1-Standarddistribution, welche im Rahmen einer Neuinstallation in weniger als 20 Minuten einsatzbereit ist. Die im Buch beschriebene Konfiguration von Hard- und Software ist darüber hinaus so angelegt, dass Nutzer sämtlicher gängigen Linux-Distributionen keine Probleme haben sollten, die Beispiele nachzuvollziehen.

Zu den Hardwareanforderungen der beschriebenen Anwendungsbereiche ist zu sagen, dass die Anwendungen problemlos auf einem „durchschnittlichen“ System Ihren Dienst verrichten; der Autor selbst hat zum Testen ein Athlon 800 / 512 MByte – System eingesetzt, welches, wie man den Gang der Entwicklung im Computerbereich ja kennt, zum Zeitpunkt der Drucklegung höchstwahrscheinlich schon antiquiert sein wird. Nichtsdestotrotz besteht einer der großen Vorteile des Linux-Betriebssystems darin, dass es vermag, gerade aus älterer Hardware ein Optimum an Leistungsfähigkeit herauszuholen. Wunder vermag allerdings auch Linux nicht zu bewirken: Möchte man ernsthafte Videobearbeitung und Transcodierung betreiben, so führt an einem zeitgemäßen System kein Weg vorbei; insbesondere das Vorhaben, Videos in Echtzeit zu encodieren führt den Anwender in die Region der Multi-Gigahertz-CPUs.

1.2 Inhalt des Buches

Im vorliegenden Buch werden folgende Themenschwerpunkte behandelt:

- Optimale Videowiedergabe (Kapitel 3)
- Analoge Videoaufzeichnung (Kapitel 4)
- Digitale Videotechnik (DVB) und Digitales Videorecording (Kapitel 5)
- Videobearbeitung und -verarbeitung (Kapitel 6 und 7)
- Sicherung von Videomaterial (Kapitel 8)
- Soundoptimierung (Kapitel 9)
- Nutzung von Videomaterial im Netzwerk (Kapitel 10)

Sämtliche genannten Anwendungsbereiche lassen sich auf einem Standardlinuxsystem realisieren. Der Leser möge mir verzeihen, dass der Schwerpunkt der Beschreibungen auf der Konfiguration der SuSE-Distribution liegt. Die vorgestellten Verfahren lassen sich jedoch ohne nennenswerte Komplikationen auf sämtliche anderen im Handel erhältlichen Standarddistributionen übertragen (der Autor hat zum „Crosscheck“ beispielsweise eine per Download aus dem Internet bezogene Mandrake-Distribution verwendet).

Zu den einzelnen Kapiteln:

Nach einer im Kapitel 2 beschriebenen Herstellung eines videotauglichen Basissystems wird im Kapitel 3 gezeigt, wie man die üblichen bekannten Videoformate (DivX, MPEG2, Realmedia, Quicktime ...) unter Linux bändigen kann. Hauptaugenmerk wird unter anderem auf einen Anwendungsbereich gelegt, der unter Linux zum Teil noch im Dornröschenschlaf steckt: Dem Abspielen von DVDs. Das Kapitel schließt mit Tipps zur Videoauskopplung an externe Geräte, wie dem guten alten Fernseher (schließlich möchte man den Filmabend in einer Gruppe ja nicht unbedingt vor dem Computermonitor verbringen ;-)).

Kapitel 4 zeigt, dass man auch mit bescheidenen Mitteln (in diesem Fall unter Verwendung einer günstigen TV-Karte) in das Videobusiness einsteigen kann; die Einrichtung eines einfachen Analogvideorecorders mit der Möglichkeit zur digitalen Archivierung des aufgezeichneten Materials wird hier vorgestellt. Wer's ein wenig luxuriöser mag, kann zur Digitalisierung seiner Urlaubsvideos auch eine hochwertigere Capturekarte einsetzen. . .

Das Linux-VDR Projekt [1] hat sich in letzter Zeit zur Killerapplikation gemauert; die darauf basierende Einrichtung eines digitalen Videorecorders, welcher den im Handel noch spärlich zu findenden kommerziellen Lösungen in nichts nachsteht, ist Thema des Kapitels 5. Interessant ist hier, dass zu diesem Zweck auch ein ausgemusterter PC in Verbindung mit einer DVB-Karte als Digitalvideoeinheit eingesetzt werden kann.

Mit alleinigem Jagen und Sammeln von Videomaterial ist es zumeist nicht getan; dieses muss gesichtet, geschnitten und archiviert (das heißt komprimiert / encodiert) werden. Die Kapitel 6, 7 und 8 zeigen Wege auf, wie das alles unter Linux zu realisieren ist.

Nach einem kleinen Ausflug in die Welt des guten Tons (Kapitel 9) wird zum Abschluss in Kapitel 10 noch gezeigt, wie man Videomaterial aller Art im LAN zugänglich macht und einen eigenen Videosever aufbaut. Kapitel 11 soll Ihnen schließlich helfen, gängige Fehler beim Aufbau eines Videosystems zu vermeiden.

1.3 Anforderungen an den Leser

Sicher geht der Leser mit einer bestimmten Erwartungshaltung an die Lektüre des Buches. Aber auch ihm wird etwas abverlangt, möchte er denn optimalen Nutzen aus dem Werk ziehen: Zunächst wird erwartet, dass der Leser mit grundlegenden Befehlen des Linux-Betriebssystems vertraut ist. Wird die Integration neuer Hardware in das System erforderlich, so sollte der Leser nicht an einer Schraubenzieherallergie leiden. Notfalls gilt immer noch der oftmals im Usenet zu lesende Spruch „contact your local guru“.

Grundlegende Techniken zum Umgang mit dem Internet (und natürlich ein funktionierender Anschluss an Letzteres) werden ebenfalls vorausgesetzt, da es stets ratsam ist, die neueste Version der beschriebenen Software „frisch“ aus dem Netz zu beziehen. Für spezielle Informationen zur Realisierung von eigenen Sonderwünschen bietet das Usenet den besten Umschlagplatz, den man sich vorstellen kann. Aber Vorsicht: Die meisten Fragen sind schon einmal gestellt worden, und vielleicht liefert ja schon `groups.google.com` die gewünschte Antwort.

Zu guter Letzt: Ein wenig Geduld und Durchhaltevermögen bei der Lektüre und dem Ausprobieren der Anwendungen kann auch nicht schaden. Viele Bibliotheken, die zur Kompilierung von Quellpaketen benötigt werden, sind einem steten Wandel unterworfen. Da heißt es oftmals testen, testen und abermals testen, damit der Compilerdurchlauf endlich von Erfolg gekrönt ist. Denn leider stehen nicht zu jedem Programm für die auf dem eigenen PC verwendete Distribution maßgeschneiderte RPMs zur Verfügung.

1.4 Schreibkonventionen

Frei nach dem Motto „learning by doing“ werden im Buch etliche in der Linux-Shell durchzuführende Installations- und Konfigurationsschritte beschrieben, Programme durch Kommandos gestartet und Referenzen angegeben. Folgende Konventionen werden zur Kennzeichnung angewendet:

- Befehle, die der Anwender eingeben soll sowie Systemausgaben werden grau unterlegt im laufenden Text dargestellt, etwa wie folgt:


```
user@linux: > su
Password:
linux: # tail -f /var/log/messages
Apr 23 18:03:34 august su: (to root) hatt on /dev/pts/2
...
%Hier erfolgen weitere Ausgaben
```

Dabei werden Eingaben, die vom Benutzer auf der Konsole durchzuführen sind, im Listing **fett** dargestellt, um diese von den Ausgaben des Programms zu unterscheiden. Kommentare, welche zur näheren Erläuterung des Listings dienen, werden mit vorangestellten %-Zeichen dargestellt.

- Lange Zeilen, die über den Seitenrand hinausgehen würden, sind im Text umbrochen, wie z. B.:

```
user@linux: > lavrec -f a -i P -g 384x288 -q 80 \
> --software-encoding test.avi
```

Sie können bei der Eingabe an Ihrem Computer entweder alles in eine Zeile schreiben, also hinter `-q 80` einfach weiter `--software-encoding test.avi` eingeben. Alternativ können Sie aber auch hinter `-q 80` den

„Backslash“ \ eingeben und  drücken. Der Cursor wechselt daraufhin in eine neue Zeile, wie im Listing abgebildet, und dort geben sie den Rest der Zeile ein. Das ist ganz praktisch, wenn man bei langen Befehlszeilen den Überblick bewahren will.

- Der normale Userprompt wird durch

```
user@linux: >
```

dargestellt, während der root-Modus durch


```
linux: #
```

gekennzeichnet ist.

- Pfadangaben, welche sich relativ auf das Homeverzeichnis des Benutzers beziehen, werden mit vorangestelltem \$HOME angegeben. So entspricht für den Benutzer *tux* die relative Pfadangabe \$HOME/*tux* dem absoluten Pfad /home/*tux*.
- Beim Wechseln in den Superusermodus mit dem `su`-Kommando erwartet das System das root-Passwort:

```
user@linux: > su
Password:
```

Der Password-Prompt wird aus Gründen der Rationalität in den Listings nicht protokolliert.

- Internet-URLs, Pfadangaben, Konsolenprogramme sowie Kommandos jeglicher Art werden im laufenden Text im Teletype-Zeichensatz dargestellt (Beispiel: `www.suse.de`).
- Menüpunkte oder Aktionen, die im grafischen Modus in Fenstern ausgeführt werden sollen, werden *kursiv* dargestellt. Beispiel: „Wählen Sie im Hauptfenster von *xawtv* das Untermenü *Record Movie (avi)*“. Das gleiche gilt für die Namen grafisch orientierter Programme wie beispielsweise *xawtv*.
- Im Text tauchen gelegentlich Fachausdrücke auf, welche wie folgt markiert sind:  *LAN*. Diese können im Glossar am Ende des Buches nachgeschlagen werden.

1.5 Zum Inhalt der Begleit-CD

Eine wesentliche Beigabe zum Buch stellt die Begleit-CD dar. Hier finden Sie unter anderem eine Vollversion der Videoschnitt-Software *MainActor* Version 3.7 für Linux von der Firma *MainConcept*. Zudem befinden sich sämtliche im Buch besprochenen Softwarepakete, sofern Sie nicht ohnehin Bestandteil der gängigen Distributionen sind, als Quellpakete auf der CD.

Aus rechtlichen Gründen konnten einige Softwarepakete (Stichworte: `lame`, `libdvdcss`) nicht beigefügt werden, im Buch wird an den entsprechenden Stel-

len darauf hingewiesen. In diesem Zusammenhang haben wir uns auch dafür entschieden, keine „all in one“-RPMs zu bauen, da dann aus den genannten Gründen einige wesentliche Features in den Programmen ausgelassen werden müßten. Die Selbstkompilierung der entsprechenden Software stellt zudem eine optimale Anpassung an Ihr spezielles System sicher. Wer dennoch aus Gründen der Bequemlichkeit auf RPMs zurückgreifen möchte, wird in Bezug auf die im Buch besprochenen Programmen im Internet fündig, siehe hierzu [2], [3], [4] und [5].

Im Einzelnen finden Sie, nach den zugehörigen Kapiteln geordnet, die folgenden Quellen und Programme auf der CD:

- ❑ **Kapitel 2** (Ein Videobasissystem für Linux)
 - Angepasster Kernel für Videoanwendungen / Basissystem SuSE 8.1
 - ATI-Gatos-Treiber
- ❑ **Kapitel 3** (Linux als Multinorm-Videooplayer)
 - *MPlayer*
 - *xine*
 - *ogle*
 - *VideoLAN-Client*
 - *nvtv-Tool*
- ❑ **Kapitel 4** (Linux als Analog-zu-Digital Videorecorder)
 - *avifile*
 - *vcr*
 - *dvr*
 - *mjpegtools* mit *Zoran-Treiber*
- ❑ **Kapitel 5** (Digitale Videoaufzeichnung unter Linux)
 - *DVB-Treiber*
 - *vdr*
 - *AIO: All-in-One-Patches* für *vdr*
 - *kvdr* Frontend
 - *LIRC* Fernsteuerungssoftware
 - *vdradmin*
 - *FireWire-Treiber*
 - *kino* und *dvgrab*
- ❑ **Kapitel 6** (Videobearbeitung unter Linux)
 - *MainActor* Vollversion für SuSE Linux (als RPM)
 - *Cinelerra* (als RPM, inklusive benötigter Libraries)
- ❑ **Kapitel 7** (Kompression und Transcodierung von Videos)
 - *xvid-Codec*

- `ffmpeg`
- `transcode`
- `2divx`
- `x2divx`
- `gv4l`
- `kmencoder`
- `dvd:rip`
- **Kapitel 8** (Archivierung von Videomaterial)
 - `xcdroast`
 - `KonCD`
 - `cdbakeoven`
 - `divxcalc`
 - `vcdimager`
 - `cdrdao`
 - `dvdrttools`
- **Kapitel 9** (Soundcheck: Audiooptimierung)
 - Ogg-Vorbis
 - `audacity`
 - `liba52`
 - `emu10k1`
- **Kapitel 10** (Linux als Videoserver)
 - `dvbstream`
 - Die kompletten *VideoLAN*-Softwarepakete (ausgenommen `libdvdcss`)
 - WLAN-Treiber
- **Kapitel 11** (Tipps, Tricks & Troubleshooting)
 - `vdr`-Entwicklerversion, neue Convergence DVB-Treiber
 - `atitvout`
- **Videos**

In diesem Unterverzeichnis auf der Begleit-CD finden Sie einige Testvideos, die für erste Experimente z. B. mit Videoschnittprogrammen verwendet werden können.

Das Kompilieren der Software setzt in jedem Fall die Installation des GNU-C-Compilers `gcc` voraus, optimalerweise in der aktuellen Version 3.2. Bei den Paketen handelt es sich um Snapshots aus den entsprechenden Projekten (Stand Februar 2003). Teilweise wurden mehrere Versionen (bzw. Releases) der gleichen Software in die entsprechenden Unterordner gepackt. Wer die aktuellsten Software-releases verwenden möchte, sei auf die im Buch genannten Links verwiesen. Und schließlich noch der obligatorische Hinweis: Es kann keine Garantie

für die Funktionsfähigkeit der Programme auf Ihrem speziellen System gegeben werden. Fertigen Sie vor dem Installieren bzw. Testen der Programme eine Sicherheitskopie Ihrer Daten an, da SuSE PRESS und der Autor keine Haftung im Falle etwaiger Datenverluste übernehmen.

Kapitel 2

Ein Videobasissystem für Linux

Voraussetzung für die Umsetzung der in diesem Buch beschriebenen Anwendungen ist ein stabiles Linuxbasissystem, welches heutzutage mit nahezu jeder der handelsüblichen Distribution in kurzer Zeit aufgesetzt werden kann. Hier sei der geneigte Leser auf die distributionsspezifische Lektüre verwiesen. Wesentlich interessanter (und oftmals auch trickreich) ist die Aktivierung von Videohard- und Software unter dem Pinguinsystem. Im vorliegenden Kapitel erfolgt daher zunächst eine Einführung, wie auch Ottonormaltux in den Genuss kommt, Videomaterial unter Linux in Augen- und Ohrenschein zu nehmen.

Die Vorgehensweise lautet hierbei „vom Einfachen zum Schwierigen“: Über die Integration einfacher Hard- und Softwarekomponenten mit den Bordmitteln, die moderne Distributionen von Hause aus zur Verfügung stellen wird die prinzipielle Vorgehensweise zur Integration externer Softwarekomponenten sowie die Kernelmodifikation im Falle exotischer Hardware besprochen. Es empfiehlt sich auch für den fortgeschrittenen Leser, die vorgestellten handwerklichen Hinweise zumindest kurz zu überfliegen, da spätere Kapitel von einigen der beschriebenen Vorgehensweisen Gebrauch machen werden.

2.1 Allgemeines: PC und Video

Wohl kaum eine Mediengattung hat die gesellschaftliche Entwicklung im ausgehenden 20. Jahrhundert derart geprägt wie Funk, Film und das Fernsehen, und nun zu Beginn des 21. Jahrhunderts ist die „Multimedialisierung“ des Internet ein wichtiges Thema. Der Begriff „Video“ umfasst längst nicht mehr unser altbekanntes Fernsehen im ursprünglichen Sinn.

Die Abbildung 2.1 zeigt ein Spektrum audiovisueller Anwendungen, welche Thema des vorliegenden Buches sein werden. Im Zentrum steht der PC, welcher längst zur Allzweck-Multimediakonsole mutiert ist. Das Wort „video“ – la-

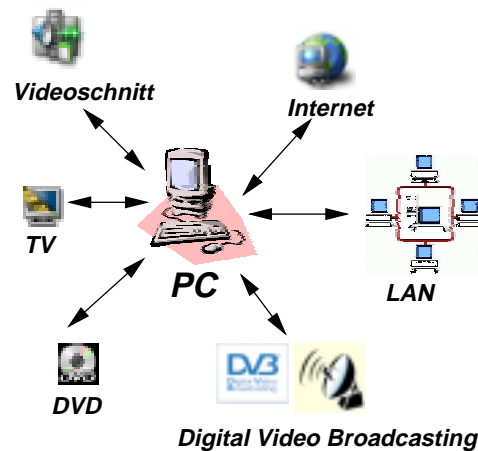


Abbildung 2.1: Struktur eines modernen PC-Videosystems

teinisch „ich sehe“ – trifft somit nur unzureichend die ganzheitliche Erfahrung, die heutzutage aufgrund der technischen Möglichkeiten dem Betrachter geboten werden.

Dem Computeranwender bieten sich folgende Hauptnutzungsmöglichkeiten:

- Einsatz des PC als TV- / oder DVB-Satellitenempfänger
- Einsatz des PC als Analog- oder Digitalvideorecorder
- Abspielen von (Fremd-)Videomaterial auf dem PC
- Herstellen von eigenem Videomaterial; Encodierung und Videoschnitt mit dem PC
- Nutzung des PC als zentraler Multimedia / -Videoserver in einem Computernetz

Sämtliche der oben aufgeführten Anwendungsbereiche werden im vorliegenden Buch behandelt. Um die grundsätzliche Videotauglichkeit des Basissystems herzustellen, empfiehlt sich als Einstieg zunächst die Inbetriebnahme einer handelsüblichen Analog-TV-Karte sowie der Test derselben mit einigen Standardapplikationen. In den folgenden Abschnitten wird dies zunächst am Beispiel der Installation unter einer typischen paketorientierten Distribution (in diesem Fall SuSE Linux, zur Zeit der Drucklegung Version 8.1) demonstriert. Spezielle Unterkapitel gehen dabei ein wenig ins „Eingemachte“, um das Knowhow zu entwickeln, welches die Einrichtung eines komplexeren Videosystems, wie z. B. das in Abbildung 2.1 beschriebene, auf beliebigen Linuxrechnern erfordert. Die für spezielle Hardware (wie beispielsweise DVB-Karten und Videoschnittkarten) zu erfolgen-

den Konfigurationsschritte werden dann später in den zugehörigen Spezialkapiteln besprochen.

2.2 Vorbereitungen zur Installation

Die Installation moderner Linuxdistributionen steht mittlerweile der oftmals gepriesenen Benutzerfreundlichkeit gewisser bekannter Produkte aus Redmond um nichts nach; ausgestattet mit einer exzellenten Hardwareerkennung und angepassten Standardkernen schickt sich Linux an, auch den Desktop von durchschnittlich begabten Computernutzern zu erobern. Dennoch gilt es, verschiedene Aspekte bereits bei der Basisinstallation des Systems zu beachten, um eine optimale Performance für Videoanwendungen sicherzustellen.

2.2.1 Partitionierung der Festplatte(n)

Grundvoraussetzung für optimale Ergebnisse in der Videoverarbeitung am PC ist ein schneller Schreib-Lesezugriff auf die Videodaten. Das Hauptproblem z. B. der Aufzeichnung eines Videodatenstroms besteht in der großen Menge der anfallenden Daten für den Fall, dass diese unkomprimiert auf der Platte gespeichert werden. Die Tabelle 2.1 zeigt den Platzbedarf für einminütige Filmsequenzen (Auflösung 384x288 Pixel), welche mit unterschiedlichen Kompressionsalgorithmen (siehe hierzu Kapitel 7) auf der Festplatte gespeichert wurden.

Bei den auftretenden Datenmengen ist es offensichtlich, dass selbst moderne Festplatten erheblich belastet werden können, wenn Videodaten unkomprimiert verarbeitet werden sollen. Setzt man Kompressionsverfahren zum Abspeichern ein, so verschiebt sich die zu bewältigende Arbeitslast in Richtung Prozessor; mehr dazu ebenfalls in Kapitel 7.

Als Konsequenz ergibt sich, dass das Festplattensystem bezüglich des Datentransferverhaltens optimal eingerichtet werden muss. Der erste Schritt besteht in einer intelligenten Partitionierung möglichst *vor* der eigentlichen Systeminstallation. Unter dem Begriff *Partitionierung* versteht man die Aufteilung der Festplatte

Tabelle 2.1: Platzbedarf unterschiedlich stark komprimierter Videodateien

Kompressionsverfahren	Dateigröße 1 min	Dateigröße 1 h
unkomprimiert	328 MByte	19,2 GByte
huffyuv v2.1.1	244 MByte	14,3 GByte
MJPEG	60 MByte	3,5 GByte
DivX 4.12 / 1500 kbit/s	11,7 MByte	702 MByte

verschiedene Bereiche, welchen z. B. unter Linux auch verschiedene *Mountpoints* zugewiesen werden können. In der DOS/Windows-Welt sind verschiedene Partitionen beispielsweise durch verschiedene Laufwerksbuchstaben ansprechbar („C:“, „E:“, ...“). Moderne Distributionen partitionieren die zur Verfügung stehende Festplatte für die Linux-Installation bei der Standardinstallation zumeist automatisch in die drei Bereiche

- /boot,
- /swap und
- /root

mit dem Standard ext2fs-Dateisystem. Für Videozwecke empfiehlt es sich allerdings, eine eigene Partition mit dem Mountpoint `/video` einzurichten. Des Weiteren ist die Verwendung eines sogenannten Journaling Filesystems (wie z. B. dem ReiserFS oder ext3fs) ratsam. Letzteres hat den Vorteil, dass die vom System periodisch durchgeführte und bei großen Platten recht langwierige Überprüfung des Dateisystems in sehr kurzer Zeit erledigt ist. Im Folgenden werden die hierfür notwendigen Schritte im Rahmen der Grundinstallation eines SuSE Linux-Systems erläutert. Sollten Sie bereits über ein vorkonfiguriertes System verfügen, so wird im Anschluss erläutert, welche Möglichkeiten bestehen, die vorgegebene Partitionierung noch nachträglich zu ändern. Wohlgemerkt: Die Umpartitionierung ist nicht zwingend erforderlich; Sie können auch alles beim Alten lassen. Für eine optimale Systemperformance empfiehlt sich allerdings die beschriebene Vorgehensweise. Und noch ein letzter Hinweis: Mit Hilfe des LVM (Logical Volume Manager) Systems lassen sich Festplattenpartitionen sogar im Nachhinein unter Linux dynamisch verwalten. Da es sich hierbei allerdings um durchaus systemkritische Operationen handelt, soll an dieser Stelle nicht weiter darauf eingegangen werden. Dem geneigten Leser sei vor der Benutzung des LVM-Systems die Lektüre der LVM-Howto angeraten, welche mittlerweile im Dokumentationsumfang der gängigen Distributionen enthalten ist. Anderenfalls finden Sie die Howto im Internet unter [6].

2.2.1.1 Während der Grundinstallation

Booten Sie von der Installations-CD und geben Sie am Bootprompt `manual` bzw. wählen Sie den Bootmenüeintrag `Manual Installation` aus. Die weiteren Schritte werden standardmäßig bis zum Start der YaST 2-Oberfläche abgearbeitet. Hier kann nun im Untermenü `Partitionieren` die Partitionierung der bestehenden Festplatte(n) vorgenommen werden (siehe Abbildung 2.2). Überaus praktisch ist hierbei zunächst der Button `FAT Größe ändern`. Er gestattet die Verkleinerung einer bestehenden FAT-Windows-Partition, um beispielsweise Platz für die Linux-Installation zu schaffen. Leider funktioniert eine derartige Partitionsreduktion nicht für das bei Windows XP und Windows 2000 üblicher-

Kapitel 4

Linux als Analog-zu-Digital Videorecorder

4.1 Allgemeines zur Videoaufzeichnung

Im Jahr 1995 feierte das Kino seinen 100. Geburtstag. Die Erfindung, die den Gebrüdern Louis und Auguste Lumière zugeschrieben wird, war die Quintessenz einer Reihe von wichtigen Erfindungen und Entdeckungen, in deren Zusammenhang so gewichtige Personen wie Thomas Alva Eddison eine bedeutende Rolle spielten. Die Tatsache, dass das menschliche Auge und Gehirn die Fähigkeit besitzen, eine schnelle Abfolge von stehenden Bewegungsphasen zusammenhängend wahrzunehmen, war ein wesentlicher Schlüssel zur „Suggestion“ des bewegten Bildes.

An dem Prinzip, eine Bewegung durch eine Folge von Einzelbildern darzustellen, hat sich selbst heute in der Welt der Bits und Bytes nichts geändert. Genügte in damaligen Zeiten allerdings noch Bildraten von 16 Bildern pro Sekunde, so findet man heutzutage beim \Rightarrow PAL-Verfahren 25 fps, beim \Rightarrow NTSC-Verfahren gar 29,97 fps vor (dies allerdings bei gegenüber dem PAL-Verfahren verringerter Auflösung). Eine weitere Qualitätssteigerung wird mit der Verbreitung des neuen \Rightarrow HDTV-Standards zu erwarten sein.

Grundsätzlich unterscheidet man zwei Speicher- bzw. Übertragungsmethoden der (Einzel-)Bildfolgen:

- Analoge Signalaufbereitung: Die Bildfolgen und darin enthaltenen Bildinformationen werden durch elektrische Spannungsfolgen unterschiedlicher Stärken repräsentiert.

- Digitale Signalaufbereitung: Die Bilder werden hier durch Signalfolgen, die nur aus zwei Basissignalen (Spannung / keine Spannung) bestehen, übertragen bzw. verarbeitet.

Grundsätzlich erfolgt die computerinterne Behandlung von Bild- oder Tonsignalen digital. Um nun ein analoges Video- oder Audiosignal, welches von einer externen Quelle dem Computer zugeführt wird, verarbeiten zu können, muss diese von einem sogenannten A/D-Wandler in eine digitale Form überführt werden. Dies ist beispielsweise dann der Fall, wenn TV-Signale mittels einer handelsüblichen TV-Karte dem PC zugeführt werden sollen. Für die Ausgabe eines digitalen Signals vom PC zum TV-Gerät ist der umgekehrte Weg zu beschreiben, hier werden D/A-Bausteine eingesetzt. In logischer Konsequenz sind derartige Wandelprozeduren meist mit Qualitätsverlusten behaftet.

Einfacher ist die Situation natürlich, wenn das Video bzw. Audiosignal bereits in digitaler Form vorliegt, wie z. B. bei DVDs, DVB-Streams oder DV-Camcordermaterial. Die Bequemlichkeit hat in diesem Fall allerdings ihren Preis, da entsprechende Hardware meist kostspieliger als entsprechende Analoglösungen ist.

Das vorliegende Kapitel beschäftigt sich zunächst mit einfachen Möglichkeiten der „Analog-zu-Digital“-Videoaufzeichnung, um auch Lesern mit schmalere Budget einen Einstieg in die Videowelt unter Linux zu ermöglichen. Voraussetzung ist die Integration einer einfachen TV-Karte, wie bereits in Kapitel 2.3.1 näher ausgeführt wurde.

Wer semiprofessionell mit analogem Videomaterial arbeiten möchte, wird im Teilkapitel 4.3 fündig; hier wird der Einsatz einer speziellen Videoschnittkarte unter Linux beschrieben.

In Kapitel 5 wird schließlich der Königsweg, die rein digitale Videoaufzeichnung besprochen, welche allerdings den Einsatz zusätzlicher digitaler Hardware bedingt.

4.2 Videorecording

Nachdem bereits in Kapitel 2.3.3.2 testweise eine einfache Bildsequenz mit *xawtv* registriert wurde, welche hinsichtlich Bildqualität und Kompression nicht optimiert war, werden im Folgenden Werkzeuge vorgestellt, die die Recording-Fähigkeiten der Linux-Box optimal ausnutzen. Dabei kann *xawtv* zwar nach wie vor Aufgaben, wie z. B. das Auffinden von TV-Sendern, übernehmen, den eigentlichen Aufnahmevorgang überlassen wir aber besser spezialisierter Software, wie z. B. *vcr* [54] oder *dvr* [55].

Speicherplatzbedarfs mit einem modernen Codec wie beispielsweise DivX „on the fly“, d. h. direkt während der Aufnahme komprimiert, so sind Prozessoren der 2 GHz-Klasse von Nöten, um die gefürchteten Framedrops (also Informationsverluste durch das Auslassen von Bildern) zu vermeiden.

In der „Szene“ wird zumeist ein anderer Weg beschritten: Das Ausgangsmaterial wird zunächst mit einem weniger verlustbehafteten, allerdings auch Festplattenplatz-intensiven Codec aufgenommen und nach erfolgter Aufnahme nachbearbeitet (geschnitten, transcodiert). Hier würde sich theoretisch der *huffyuv*-Codec anbieten [60], welcher in den *avifile*-Binaries sowie in experimenteller Form im *ffmpeg*-Paket [98] enthalten ist, z. Zt. allerdings nur unzureichend unter Linux unterstützt wird. Der Codec erfordert zudem große Festplattenkapazitäten. Den goldenen Mittelweg unter Linux findet man im *MJPEG*-Codec, welcher im Folgenden besprochen wird.

4.2.2 Capturen mit den *MJPEG*-Tools

Während unter Windows zum großen Teil nur kostenpflichtige Versionen des *MJPEG*-Codecs existieren, hat der Linux-Anwender Glück: In Form der *MJPEG*-Tools [61] liegt ein hervorragendes Open Source-Softwarepaket vor. Die Tools gestatten einerseits die softwaregestützte Kompression im *MJPEG*-Format, darüber hinaus können aber auch Hardwareencoder mit *Zoran*-Chip, wie das *Pinnacle DC10+*-PCI-Board (vormals *Miro*), eingesetzt werden, siehe hierzu Kapitel 4.3. Zunächst soll die nicht ganz triviale Installation der *MJPEG*-Tools gezeigt werden. Auch hier wird wieder davon ausgegangen, dass Sie „tapfer genug“ sind, die Sourcen selbst zu kompilieren. Nur so ist sichergestellt, dass die Software optimal an Ihr System angepasst ist. Von der Installation über RPM-Pakete möchte ich an dieser Stelle abraten, da gerade eine softwaregestützte Kompression auf ein optimal eingerichtetes System angewiesen ist.

4.2.2.1 Kompilieren der Quellen

Besorgen Sie sich zunächst die neuesten Quellen von [62] (erkennbar an der höchsten Versionsnummer und der Endung `*.tar.gz`). Sie benötigen die folgenden Dateien:

```
jpeg-mmx-0.1.4.tar.gz
libmovtar-0.1.3.tar.gz
quicktime4linux-1.4-patched-2.tar.gz
mjpegtools-1.6.0.tar.gz
```

(Ersetzen Sie in Gedanken die Versionsnummern durch die jeweils aktuellen Zahlen). Die Kompilierung von `libmovtar` setzt zudem das Vorhandensein des Standardpakets `SDL-devel` voraus.

Zusätzlich werden zum Kompilieren noch die *jpeg*-Quellen

`jpegsrc.Version.tar.gz`

von der *Independent JPE Group* [63] benötigt. Packen Sie nun alle `*.tar.gz`-Dateien in ein gemeinsames Verzeichnis. Die Installation erfolgt aus diesem Verzeichnis heraus über die folgenden Schritte:

1. *jpeg*-Quellen kompilieren und installieren:

```
user@linux: > tar xzf jpegsrc.Version.tar.gz
user@linux: > cd jpeg-Version
user@linux: > ./configure
user@linux: > make
linux: # make install
```

2. *jpeg-mmx* kompilieren, aber *nicht* installieren:

```
user@linux: > tar xzf jpeg-mmx-0.1.4.tar.gz
user@linux: > cd jpeg-mmx-0.1.4
user@linux: > ./configure
user@linux: > make
% kein make install !!
```

3. *libmovtar* mit speziellen Optionen konfigurieren, kompilieren und installieren:

```
user@linux: > tar xzf libmovtar-0.1.3.tar.gz
user@linux: > cd libmovtar-0.1.3
user@linux: > ./configure --with-jpeg-mmx=/Pfad/nach/jpeg-mmx-0.1.4
libmovtar 0.1.3 build configuration :
...
- Build movtar_play      : yes
- JPEG input in         : yes
- RTJPEG support        : yes

user@linux: > make
linux: # make install
```

Bei der `./configure`-Option muss der *absolute* Pfad zur oben kompilierten *jpeg-mmx*-Library angegeben werden.

4. Schließlich wird die speziell gepatchte *Quicktime4Linux*-Library entpackt und kompiliert, aber *nicht* installiert:

```
user@linux: > tar xzf quicktime4linux-1.4-patched-2.tar.gz
user@linux: > cd quicktime4linux-1.4-patch
user@linux: > ./configure
user@linux: > make
% kein make install !!
```

5. Letztendlich können die *MJPEG*-Tools kompiliert und installiert werden, wobei wieder besonderes Augenmerk auf die Parameter der Konfigurationsprozedur gelegt werden muss:

```
user@linux: > tar xzf mjpegtools-1.6.0.tar.gz
user@linux: > cd mjpegtools-1.6.0
user@linux: > ./configure \
> --with-jpeg-mmx=/Pfad/nach/jpeg-mmx-0.1.4 \
> --with-quicktime=/Pfad/nach/mjpeg/quicktime4linux-1.4-patch \
> --enable-large-file
MJPEG tools 1.6.0 build configuration :

- X86 Optimizations:
- MMX/3DNow!/SSE enabled      : true
- cmov support enabled       : true
- video4linux recording/playback: true
- software MJPEG playback    : true
- movtar playback/recording  : true
- Quicktime playback/recording : true
- Quicktime source in       :
    /packages/mjpeg/quicktime4linux-1.4-patch
- AVI MJPEG playback/recording : true (always)
- libDV (digital video) support : false
- AVIFILE divx encoding support : true
- Gtk+ support for glav       : true
- glibc support for >2GB files : true

user@linux: > make
linux: # make install
```

Insbesondere wurde hier auch die Unterstützung für Dateien, deren Größe die „magische“ 2 GByte-Grenze übersteigt, miteinkompiliert.

Konsultieren Sie für aktuelle Informationen bezüglich des Installationsprozesses der *MJPEG*-Tools bitte stets auch die im Paket enthaltene Datei `INSTALL`.

Damit wären die optimalen Voraussetzungen für ein Videocapturing mit dem *MJPEG*-Codec geschaffen.

4.2.2.2 Arbeiten mit den *MJPEG*-Tools

Das *MJPEG*-Paket besteht aus den folgenden Hauptprogrammen:

- `lavrec`: Tool zur Aufnahme von Videodatenströmen.
Die Encodierung kann sowohl softwarebasiert als auch Hardware-unterstützt erfolgen.

- `lavplay`: Player für mit `lavrec` aufgenommene Videodateien.
- `glav`: GUI-Tool zum Abspielen und Schneiden von Videodateien.

Zusätzlich steht eine Fülle von Hilfswerkzeugen zur Transcodierung des *MJPEG*-Materials in andere Formate, wie z. B. DivX, VCD oder SVCD zur Verfügung, was in Kapitel 7 näher besprochen wird.

Eine einfache Aufnahme, welche eine softwarebasierte *MJPEG*-Encodierung des Videostreams vornimmt, erfolgt über den Kommandozeilenaufruf

```
user@linux: > lavrec -f a -i P -g 384x288 -q 80 \
> --software-encoding test.avi
0.00.31:06 int:000 lst:0 ins: 23 del:0 ae:0 td1=0.012 td2=0.073
```

Gestoppt wird die Aufnahme durch $(\text{Strg})+(\text{C})$. Beim Programmstart auftretende Warnmeldungen der Art

```
++ WARN: [lavrec] Unable to set negative priority for main thread
++ WARN: [lavrec] Pthread Real-time scheduling for main thread
           could not be enabled
++ WARN: [lavrec] Unable to set negative priority for audio thread.
++ WARN: [lavrec] Pthread Real-time scheduling for audio thread
           could not be enabled.
```

können getrost ignoriert werden. Sie beruhen darauf, dass das Programm versucht, den beteiligten Prozessen maximale Priorität zu verschaffen und lassen sich dadurch abstellen, dass man das SUID-Bit beim `lavrec`-Programm setzt, was allerdings ein Sicherheitsrisiko im System darstellt, da das Programm dann mit root-Rechten läuft.

Die Bedeutung der einzelnen Optionen des `lavrec`-Tools entnehmen Sie bitte der Tabelle 4.2. Wichtig ist, dass man im Falle der softwarebasierten Encodierung die Option

```
-software-encoding
```

angibt. Der obige Aufruf erzeugt dann einen *MJPEG*-encodierten Stream. Möchte man das *Quicktime4Linux*-Format einsetzen, so ist der Aufnahmebefehl wie folgt abzuändern:

```
user@linux: > lavrec -f q -i P -g 384x288 -q 80 \
> --software-encoding test.mov
```

Der große Vorteil des *Quicktime*-Videoformates besteht darin, dass damit aufgenommene Streams im Videobearbeitungsprogramm *Xinelerra* (siehe Kapitel 6) geöffnet und bearbeitet werden können.

Für weitere Informationen bezüglich der Kommandozeilensyntax steht die Man-Page `man lavrec` zur Verfügung.

Kapitel 5

Digitale Videoaufzeichnung unter Linux

Moderne Digitalvideorecorder liegen heute im Preissegment zwischen 500 € und 1000 €, d. h., der Endanwender zahlt einen hohen Preis für eine sehr spezielle Hardwarefunktionalität. Für den Computerkundigen ist es wesentlich interessanter, sein bestehendes PC-System zu einer derartigen Maschine aufzurüsten, bekommt er doch so neben der Digital-TV-Funktionalität noch die Möglichkeit, mit der bestehenden Hardware Videoschnitt und Codierung zu betreiben. Die Kosten halten sich für einfache Projekte in Grenzen; eine Low-Budget-PCI-Karte für digitalen Satellitenempfang ist derzeit ab 129 € im Fachhandel erhältlich, d. h. für einen Bruchteil des Preises einer digitalen Videorecorderanlage. Lediglich für eine \Rightarrow DVB-s-Karte mit dediziertem MPEG2-Chip, welchen man z. B. für den Einsatz mit der *vdr*-Software benötigt, sind ca. 200 € anzulegen. Hinzu kommen, falls noch nicht vorhanden, ca. 70 € für den Erwerb einer Satellitenschüssel nebst Universal-LNB. Noch einfacher gestaltet sich die Situation, wenn im örtlichen Breitbandkabelnetz bereits Digital-TV eingespeist wird: In diesem Fall muss lediglich eine kabeltaugliche DVB-Karte erworben werden.

Das folgende Kapitel erklärt zunächst die Grundlagen der DVB-Technik und erläutert Schritt für Schritt den Aufbau und die Konfiguration einer solchen PC-gestützten Digital-Satelliten-Anlage. Deren Möglichkeiten sind dann auch faszinierend: Die Aufnahme von Filmmaterial in DVD-naher Qualität inklusive Dolby Digital Ton lassen das Herz eines jeden Kinofreundes höher schlagen. Bei solch hervorragendem Ausgangsmaterial steht dem Mastering einer DivX-CD oder gar DVD nichts mehr im Weg, dazu aber mehr in den Kapiteln 7 und 8.

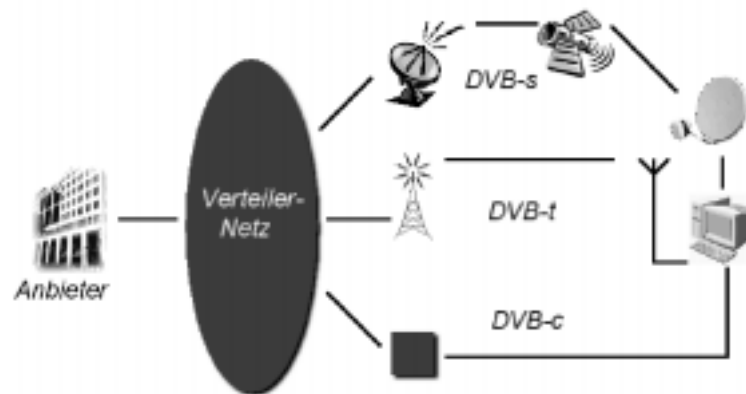


Abbildung 5.1: Varianten des DVB-Verfahrens

5.1 Grundlagen der DVB-Technik

Mit dem *Digital Video Broadcasting*-Verfahren können Multimediainhalte (Video, Audio, Internet) auf einfache Weise in Haushalte transportiert werden. Voraussetzung für die Nutzung des Verfahrens ist zunächst das Vorhandensein einer entsprechenden Infrastruktur zur Übertragung der „Contents“ (engl.: Inhalte).

Die Abbildung 5.1 zeigt die prinzipiell möglichen Übertragungswege: Man unterscheidet zwischen der Übertragung via Satellit (DVB-s), über Kabel (DVB-c) sowie dem Ausstrahlen des Signals mittels Funkrelais (DVB-t). Mindestens eine der drei genannten Varianten sollte Ihnen zugänglich sein, im vorliegenden Buch beschränken wir uns bei der Beschreibung auf die weit verbreitete DVB-s Methode.

Welche Möglichkeiten bietet nun aber DVB dem Endverbraucher, und welche Technik wird dort zur Übertragung multimedialer Inhalte eingesetzt? Nachdem

Tabelle 5.1: Vergleich der DVB-Varianten, Quelle: [73]

Parameter	DVB-s	DVB-t	DVB-c
Frequenzbereich	10,7 - 12,75 GHz	174 - 230 MHz	47 - 470 MHz
	digital im Highband: 11,7 - 12,75 GHz	Band III, Band V: 582 - 790	
Bandbreite	33 - 36 MHz	7 - 8 MHz	7 - 8 MHz
max. Kapazität	55 Mbps	39 Mbps	27 - 41 Mbps
Nutzkapazität	38 Mbps	14 - 24 Mbps	38 Mbps

das Verfahren zunächst hauptsächlich von Pay-TV-Anbietern eingesetzt wurde, welche darin eine vermeintlich sichere (d. h. nicht illegal dechiffrierbare) Plattform zur kostenpflichtigen Vermittlung von Videoinhalten fanden, zogen schließlich die öffentlich-rechtlichen Sendeanstalten sowie die Mehrzahl der werbefinanzierten Privatsender nach und bieten nun digitale Inhalte über sogenannte Satellitentransponder an. Ein Transponder ist ein relativ breitbandiger Frequenzbereich, in dem mehrere analoge und digitale Kanäle untergebracht werden können. Die Trägerfrequenz bewegt sich zwischen 10 GHz und 12 GHz. Pro Transponderbereich, von denen mehrere auf einem Satelliten zur Verfügung stehen, können mindestens 5 bis 6 digitale Fernsehprogramme zuzüglich einiger digitaler Radioprogramme, aber auch breitbandige Datendienste wie Internet bzw. DSL via Satellit übertragen werden.

Im Vergleich zu DVB-s fällt die Bandbreite bei den Verfahren DVB-t bzw. DVB-c schmal aus: Hier stehen Bereiche von lediglich 7 MHz - 8 MHz zur Verfügung. Die Tabelle 5.1 stellt die Charakteristika der jeweiligen Techniken gegenüber. Es ist ersichtlich, dass z. Zt. die DVB-s-Variante Vorteile gegenüber den anderen beiden Methoden besitzt, wovon auch das breite Spektrum der über Satellit erhältlichen digitalen Sender zeugt.

Im Folgenden soll die Übertragung von Bilddatenströmen mittels DVB-s im Mittelpunkt stehen: Die wesentliche Idee beim *Digitalen Fernsehen* ist die Konversion der analogen Signale in eine Bitfolge bei gleichzeitiger Kompression. Hierbei findet derzeit das MPEG2-Verfahren Anwendung, welches bereits im Zusammenhang mit der Videodaten Speicherung im Kapitel 3.1.1 eingehend besprochen wurde. Die Fernsehdaten kommen auch beim Endnutzer bereits in komprimierter Form an; findet man also einen Weg, diese zu speichern, so hätte man den wesentlichen Schritt zum digitalen Videorecorder vollzogen. Nichts anderes passiert übrigens in den oben erwähnten kommerziellen Digitalvideorecordern: auch hier werden die Daten in komprimierter Form auf einem DV-Tape abgelegt, z. T. sogar noch in Form von MPEG1-Streams. Neue Wege beschreiten diesbezüglich DVD-Recorder, wengleich hier die Problematik besteht, dass die Hersteller offensichtlich Schwierigkeiten haben, sich auf einen gemeinsamen Standard zu einigen.

Der Linux-Anwender kann derzeit zum Einsatz der DVB-Technik ein großzügiges Treiber- und Softwaresortiment nutzen, in dessen Mittelpunkt das VDR-Paket steht [1]. In diesem Multimediasegment hat das Linux-System gegenüber Windows sogar eindeutig die Nase vorn, können mit der Software doch mehrere DVB-Karten in einem Rechner betrieben werden. Entsprechende Multi-LNBs an der Satellitenempfangsanlage vorausgesetzt, können mit mehreren Karten auch mehrere Programme gleichzeitig aufgezeichnet werden, zusätzlich kann die begehrte Timeshifting-Funktionalität realisiert werden. Das heißt, noch während der Aufnahme kann diese bereits betrachtet werden; ein typischer Anwendungsfall wäre das Fußballspiel, zu dem man erst später zuschaltet. Man kann parallel zur weiterlaufenden Aufzeichnung schon den Anfang des Spieles betrachten.

Im Folgenden wird zunächst die Installation und Konfiguration eines DVB-Grundsystems beschrieben. Anschließend wird mit der *VDR*-Software ein „Rundum-Sorglos-Paket“ für Digitales Fernsehen vorgestellt, welches sich insbesondere bei der Aufzeichnung von Satellitenstreams vielfach bewährt hat.

5.2 Aufbau eines DVB-Systems

Herzstück eines als Digitalsatellitenempfänger agierenden PCs stellt die DVB-PCI-Karte dar. Wie im vorangegangenen Abschnitt dargelegt, unterscheidet man hier die Varianten DVB-s, DVB-c und DVB-t. Im Computerbereich werden z. Zt. die ersteren beiden Verfahren in Form entsprechender PCI-Karten unterstützt. Davon ist wiederum die DVB-s-Karte am meisten verbreitet, so dass im Folgenden deren Installation beschrieben wird. Sollten Sie planen, eine DVB-c- oder DVB-t-Karte einzusetzen, so informieren Sie sich zunächst, ob diese vom unten beschriebenen Linux-Treiber unterstützt wird. Die prinzipiellen Installationschritte ändern sich jedoch nicht.

5.2.1 Hardwareauswahl und -integration

Stehen Sie vor der Entscheidung, eine DVB-Karte zu erwerben, so sollten Sie ihr Augenmerk zunächst auf die folgenden grundsätzlichen Kartentypen richten:

- ❑ DVB-Karte mit MPEG2-Decoder-Chip: Hierbei haben Sie die Möglichkeit, einen alten (vielleicht ausgemusterten) PC einzusetzen, da die zum Abspielen von DVB-Streams erforderlichen Rechenleistungen vom auf der Karte befindlichen MPEG-Chip bereitgestellt werden. Darüber hinaus kann der Chip in Verbindung mit dem *VDR*-Programm DVDs hardwarebeschleunigt wiedergeben. Der Preis der Karten beläuft sich derzeit auf ca. 220 €.
- ❑ „Budget“-DVB-Karte: Auf dieser ist der oben genannte Chip nicht enthalten, so dass man als PC-Basis mindestens über ein 500 MHz-System verfügen sollte, um aufgezeichnete MPEG2-Sequenzen entsprechend flüssig wieder abspielen zu können. Eine derartige Karte ist bereits ab ca. 130 € erhältlich.

Wohlgemerkt: Mit beiden Kartenvarianten können Sie digitale Satellitenströme ansehen und aufzeichnen, beide werden mittlerweile von den für Linux erhältlichen Treibern unterstützt. Zum jetzigen Zeitpunkt ist es aber ratsam, auf eine vollausgestattete DVB-s Karte mit integriertem MPEG2-Encoder zurückzugreifen, da nur diese sämtliche Funktionen des in Kapitel 5.3 besprochenen *vdr*-Videosystems unterstützt.

Tabelle 5.2: Vergleich von gängigen DVB-s-Karten, Stand: 08/02

Hersteller	Bezeichnung	MPEG Chip	Preis (ca.)	Linux-Treiber
Hauppauge	WinTV Nexus	+	220 €	+
Hauppauge	WinTV Nova	-	150 €	+
Siemens	DVB-PCI-Card	+	199 €	+
Pinnacle	PCTV-Sat	-	149 €	-

Die Tabelle 5.2 gibt einen groben Überblick der derzeit im Handel befindlichen DVB-s-Karten, ohne den Anspruch auf Vollständigkeit zu erheben. Die meisten der Karten basieren auf dem Design des Herstellers TechnoTrend, wobei insbesondere die Varianten TechnoTrend Premium 2.1 / 1.3 (Karten mit MPEG-Decoder) und TechnoTrend Budget verbreitet sind. Die TechnoTrend-Karten werden allesamt von dem im Folgenden beschriebenen *Convergence*-Treiber unterstützt, siehe hierzu [74] zum Download der Originaltreiber und [75] für weitergehende Informationen.

Abbildung 5.2 zeigt eine typische DVB-s-Karte mit den üblichen Schnittstellen. In CI-Slot kann (falls vorhanden) ein *Common Interface Modul* eingesetzt werden, welches es gestattet, verschlüsselte Privatsender gegen Gebühr zu empfangen. Des Weiteren können über die Kabelpeitsche Bild- und Tonsignale an externe Peripherie (TV-Gerät, HiFi-Anlage) weitergegeben werden. Auch eine Infrarot-Fernsteuereinheit findet man heute im Lieferumfang der DVB-s-Oberklasse. Der Einbau der Standard-PCI-Karte sollte keine Probleme bereiten. Interessanter ist die Installation der notwendigen Treiber unter Linux.

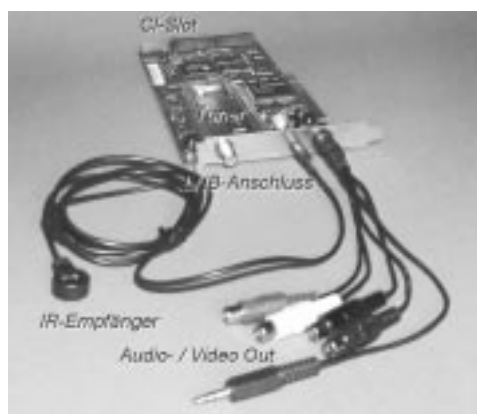


Abbildung 5.2: DVB-s-Karte mit Peripherie

Kapitel 6

Videobearbeitung unter Linux

Nachdem das Videomaterial mit einer der in den vorangegangenen Kapiteln beschriebenen Methoden auf die Festplatte des Rechners gebannt wurde, gilt es nun, dieses zu sichten und zu bearbeiten.

In den letzten Jahren hat Linux bei der Realisierung vieler großer Filmprojekte einen festen Platz als Videoschnitt- und Rendersystem eingenommen; Hollywoodblockbuster wie „Titanic“ oder „Der Herr der Ringe“ wurden mit Hilfe von Linux-Clustern mit atemberaubenden Spezialeffekten ausgestattet. Die dort eingesetzte Software hatte allerdings den Nachteil, zu speziell auf das entsprechende Filmprojekt zugeschnitten zu sein; sie war darüber hinaus für den Privatanwender aufgrund von Closed Source-Lizenzbedingungen kaum zugänglich. Die mangelnde Benutzerfreundlichkeit tat dann ihr übriges, um den Hobbycinemasten von der Linux-Software fernzuhalten. Erfreulicherweise etablierten sich gerade in letzter Zeit verschiedene benutzerfreundliche Videoschnittlösungen unter Linux.

Das aktuelle Kapitel gibt sowohl dem Nutzer digitaler Bildaufzeichnungssysteme á la *vdr* als auch dem Hobbyfilmer einen Überblick, welche Möglichkeiten der nachträglichen Filmbearbeitung unter Linux derzeit existieren.

Nach einer kurzen Einführung in die Grundlagen des PC-gestützten Videoschnitts wird eine einfache Schnittsequenz am Beispiel der *vdr*-Software erläutert. Im Abschnitt 6.3 und 6.4 werden anspruchsvolle Schnittprogramme unter Linux vorgestellt und eingesetzt, welche insbesondere auch in der Lage sind, das Bildmaterial nachträglich mit einer eigenen Tonspur zu versehen (Kapitel 6.6).

6.1 Grundsätzliches zum Videoschnitt

Der Begriff „*Videoschnitt*“ entstammt der Zeit des guten alten Zelluloidfilms: Unerwünschte Sequenzen wurden aus dem Filmmaterial einfach per Schere entfernt. Die Erstellung eines kompletten Spielfilms konnte aufgrund des so entstandenen Filmschnipsel-Sammelsuriums rasch in eine Geduldsprobe ausarten; um den Überblick zu bewahren, befestigte der Hobbyfilmer die Filmstreifen mit Wäscheklammern auf einer Leine, zumeist mit Zetteln versehen, welche eine Kurzbeschreibung der entsprechenden Szene enthielten.

Die Welt des digitalen Filmfreundes ist um ein Vielfaches einfacher geworden. Das Filmmaterial wird heutzutage bequem am PC gesichtet, geschnitten und neu zusammengesetzt. Zudem bieten moderne Videoschnittprogramme (in Fachkreisen oft auch „*Videosequencer*“ genannt) eine Vielzahl von Bearbeitungsmöglichkeiten: Überblendungseffekte, Bildretuschierfilter und Transcodieralgorithmen seien hier als Beispiele genannt.

Im Bereich des Videoschnitts am Computer werden zwei grundlegende Methoden unterschieden:

- ❑ *Linearer Videoschnitt*: In diesem Fall agiert der Computer lediglich als Steuergerät, welches die Szenen in der gewünschten Reihenfolge vom Quellband auf ein Zielband hintereinander (d. h. linear) aufspielt.
- ❑ *Nichtlinearer Videoschnitt*: Dieser bedingt die Digitalisierung der Bilddaten vor Schnitt und Nachbearbeitung, so dass dem Anwender das ganze Spektrum digitaler Bildbearbeitung zur Verfügung steht.

Im Folgenden werden wir uns mit dem *nichtlinearen* Videoschnitt befassen, und zwar aus dem einfachen Grund, dass sämtliche der bislang vorgestellten Captureverfahren Bildrohdaten in digitalisierter Form liefern. Als Ausgangsmaterial dienen Filme, welche im MJPEG-, DV- oder auch *.vdr-Format vorliegen. Natürlich kann auch noch stärker komprimiertes Material wie z. B. DivX geschnitten werden; aus Qualitätsgründen empfiehlt es sich allerdings, die Transcodierung in das angestrebte Endformat erst nach Schnitt und Vertonung vorzunehmen. Für die Umwandlung des heimischen PCs in ein Videoschnittstudio werden folgende Softwaremodule benötigt (siehe auch Abbildung 6.1):

- ❑ Ein Capture-Modul zum Importieren des Filmmaterials in den Rechner. Dieses erübrigt sich natürlich, wenn das Material digital in den Rechner eingespielt wurde, also z. B. als FireWire-Import, DVB-Stream oder DVD-Rip vorliegt.
- ❑ Ein Schnittprogramm zum Schneiden und qualitativen Nachbearbeiten des Materials.
- ❑ Optional: Ein Audioprogramm zum Kommentieren oder auch Nachvertönen des Materials.

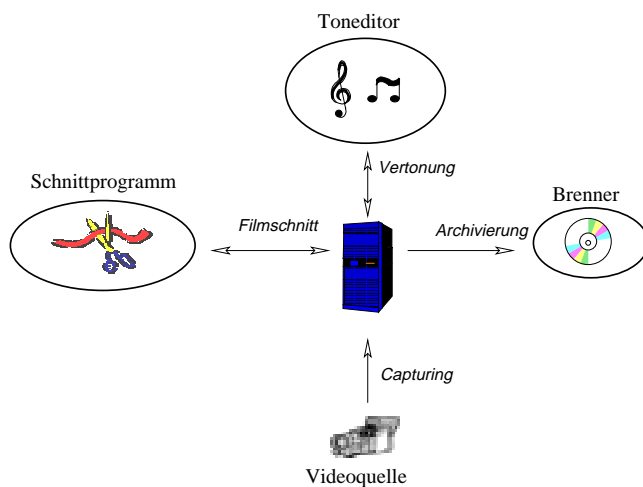


Abbildung 6.1: Flussdiagramm der Videobearbeitung

Der erste Softwarebaustein wurde in diversen Varianten in den vorangegangenen Kapiteln besprochen. Der nächste Abschnitt soll anhand eines einfachen Schnittsystems (nämlich dem *vdr*-System) zunächst die Grundlagen des Videoschnitts erläutern, bevor komplexe Programme vorgestellt werden, die keine Wünsche offen lassen.

6.2 Videoschnitt mit *vdr*

Ein Großteil der sogenannten „Blockbuster“ wird uns heutzutage von der nahezu unüberschaubaren Vielzahl der Privatsender dargeboten. Das größte Ärgernis einer Videoaufzeichnung besteht bei diesen Sendern in dem großen Anteil an (für die meisten Filmfreunde überflüssigen) Werbepausen im laufenden Film. Die Datenmenge der Werbepausen kann hierbei schon einmal ein Viertel des Speicherplatzes für den gesamten Film betragen. Solange noch kein zuverlässiges System existiert, welches in der Lage ist, diese Werbeblöcke schon während der Aufnahme herauszufiltern (z. B. anhand des Senderlogos), bleibt dem Anwender nurmehr die nachträgliche Entfernung dieser unerwünschten Unterbrechungen. Dies soll an einem mit *vdr* aufgenommenen Film demonstriert werden.

Wechseln Sie zunächst innerhalb der *vdr*-Umgebung zum Menüpunkt *Aufzeichnungen* und wählen Sie den zu schneidenden Film für die Wiedergabe aus. Diese wird im Allgemeinen durch die Eingabetaste gestartet. Durch erneute Betätigung der Eingabetaste erscheint ein Bildlaufbalken, der die aktuelle Position im Film wiedergibt (Abbildung 6.2). Möchte man den Film später erneut wiedergeben, so

Abbildung 6.8: Komplexes *Copy and Paste* mit *Cinelerra*

6.3.2 Videoschnitt mit *MainActor*

Das kommerzielle Videobearbeitungs- und Schnittsystem *MainActor* kann als Vollversion 3.7 für SuSE Linux 8.0 und 8.1 von der Begleit-CD des Buches installiert werden. Auf der Homepage von *MainActor* [65] finden sich weitere Informationen über aktuelle Entwicklungen rund um dieses Schnittsystem.

Das Programm selbst besticht durch einfache Handhabung, zudem verschafft ein auf der Begleit-CD im Verzeichnis `/kapitel6/MainActor` befindliches Tutorial im HTML-Format dem Einsteiger einen hervorragenden Überblick. Nach Entpacken der Datei `MainActor_Doku-D_HTML.tar.gz` und Wechseln in das neu entstandene Verzeichnis `MainActor_Dokumentation-D_HTML` beginnt der Rundgang durch das Programm durch Aufruf der Datei `start.htm`.

Installation und der erste Programmstart des Sequenzers wurden bereits auf Seite 101 beschrieben, so dass nun noch die speziellen Videoschnittfeatures zu besprechen sind. In erster Linie soll dabei auf das *Sequencer*-Modul eingegangen werden. Das Programmpaket beinhaltet zwar auch ein einfaches *Videoeditor*-Modul, der aber bei weitem nicht an den Funktionsumfang und die Möglichkeiten des Sequenzers heranreicht.

Neben Clips, die mit dem Capturemodul des Programms erstellt wurden, akzeptiert der *MainActor*-Sequencer auch *avi*-Dateien, die mit den *MJPEG*-Tools aufgenommen wurden, sowie MPEG-Filme. Besonders interessant ist für den Hobbyfilmer natürlich die Möglichkeit des Im- und Exports von DV-Material.

Die folgenden Ausführungen beziehen sich auf die englische Menüführung von *MainActor*, welche durch *Edit* -> *Preferences* -> *Language* ausgewählt werden kann. Prinzipiell kann hier natürlich auch die deutsche Sprache eingestellt werden, aus Gründen der Sprachkompatibilität mit anderen Linux-Videoprogrammen wie beispielsweise *Cinelerra* wurde hier bewusst die Erläuterung der englischen Begriffe gewählt, um einen einfachen Transfer der Methoden zu gewährleisten.

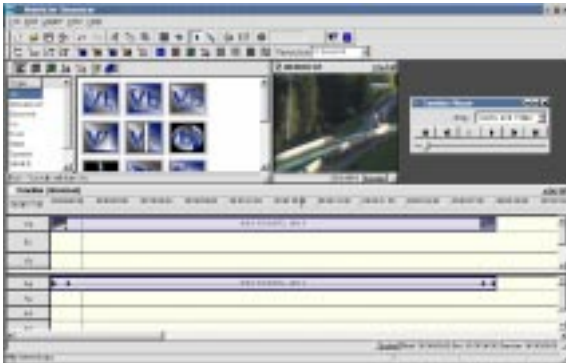


Abbildung 6.9: Videoschnitt mit dem Sequenzer-Modul von *MainActor*

Ein bereits vorliegender Videoclip wird über die *Insert multimedia*-Option eingefügt; diese ist entweder über den gleichnamigen Button zu erreichen, oder aber über das Kontextmenü der rechten Maustaste mittels Klicken über der entsprechenden Videospur, in welche das Material eingefügt werden soll. Darauf öffnet sich ein Auswahlfenster, in welchem die zu bearbeitende *avi*- oder *mpeg*-Datei selektiert werden muss. Diese erscheint erst dann in der sogenannten Timeline des Sequenzers, wenn sie durch Drücken der linken Maustaste in der Videospur platziert wurde. Die Videospuren sind allesamt mit einem *V* bezeichnet, und es empfiehlt sich für erste Experimente die Auswahl der Videospur *Va*. Die entsprechenden Audiospuren werden durch den Einfügevorgang simultan übernommen. Abbildung 6.9 zeigt das Sequenzerhauptfenster mit den Teilfenstern *Browser*, *Preview*, *Timeline Player* sowie der eigentlichen *Timeline*, in welcher sämtliche Schnitt-, Filter- und Überblendaktionen stattfinden. Der eigentliche Videoschnitt erfolgt direkt in der Timeline im *Cut*-Modus (auswählbar mittels Rasiermesserknopf auf der Buttonleiste):

1. Markieren Sie im *Cut*-Modus die gewünschten Schnittstellen durch Anklicken auf der Timeline.
2. Die Filmsequenz zwischen zwei Schnittstellen (bzw. zwischen Filmbeginn und erster Schnittstelle oder letzter Schnittstelle und Filmende) können im *Edit*-Modus (gedrückter Mauspfel-Button) mit der Maus durch Anklicken ausgewählt und mittels *Cut*-, *Copy*- und *Paste*-Funktion weiterbehandelt werden.
3. Bei mehreren Teilfilmstreifen, die zu einem Gesamtfilm zusammengefügt werden sollen, ist unter Zuhilfenahme weiterer Videospuren analog zu dem im Kapitel 6.3.1.2 Gesagten zu verfahren.

Der elementare Videoschnitt ist nur ein Teilaspekt des *MainActor*-Programmpaketes. Die Stärke der Software liegt vielmehr in der schier unerschöpflichen Viel-

falt von Übergangs- und Filtereffekten, die im folgenden Teilkapitel vorgestellt werden.

6.4 Überblenden und Filtern von Videomaterial

Seit den Anfängen der Filmtechnik besteht ein wesentlicher Reiz des Mediums in der schlüssigen Aneinanderreihung der Filmsequenzen und deren geschickter Überblendung. Nicht umsonst erscheint der Begriff bzw. das Team „Schnitt“ in einer gesonderten Würdigung am Ende des Films. Längst vorbei sind heute die Zeiten, in denen Schnitte scharf erfolgten. Der anspruchsvolle Videokonsument bzw. -produzent verlangt nach optisch ansprechenden Übergängen zwischen den einzelnen Filmsequenzen. Zudem besteht oft der Wunsch nach „optischem Tuning“ des vorliegenden Videomaterials, was sich durch die Verwendung von digitalen Videofiltern einfach bewerkstelligen lässt.

6.4.1 Videoeffekte mit *MainActor*

Folgende Teilaspekte sollen anhand einiger Beispiele behandelt werden:

- Erstellen von Szenenübergängen
- Einbinden von Text in das Video
- Optimierung des Ausgangsmaterials durch Filter

6.4.1.1 Szenenübergänge

Das Einfügen von Szenenübergängen sollte nach dem Schnitt des Filmes erfolgen. Günstig ist es hierbei, den geschnittenen Film nicht sofort zu exportieren, sondern vielmehr als *MainActor*-Projektdatei abzuspeichern. Mehrfaches Rendern führt bekanntermaßen zu Qualitätsverlust des Endmaterials. Nach Behandlung mit dem in Teilkapitel 6.3.2 vorgestellten Schnittverfahren werden nun fließende Übergänge zwischen den einzelnen Filmabschnitten hergestellt. Einen Eindruck über die Vielfältigkeit der Übergänge liefert das Browserfenster (siehe auch Abbildung 6.9). Die Überblendungen selbst werden wie folgt in der Timeline verankert:

1. Laden Sie den zu bearbeitenden Videofilm oder die Projektdatei in *MainActor*. Ermitteln Sie die zu überblendenden Filmteile mit Hilfe des Preview-Fensters und setzen Sie dort Schnittmarken, falls diese nicht ohnehin schon mit der Projektdatei abgespeichert worden sind.
2. Mittels „Cut and Paste“ werden die auf die Schnittstellen folgenden Filmteile in eine zweite Videospur kopiert (also z. B. *Vb*). Achten Sie dabei auf ein Überlappen der einzelnen Sequenzen (siehe auch Abbildung 6.10).

Kapitel 10

Linux als Videosever

Kein anderes Betriebssystem ist derzeit so konsequent mit netzwerkrelevanten Funktionen ausgestattet wie Linux. Das ist nicht weiter verwunderlich, da das System ein echtes Kind des Internet ist: Durch die Initiative von Linus Torvalds wurde Linux im Netz für das Netz entwickelt. Ein eindrucksvolles Beispiel für die Dominanz im Serverbereich ist die Linux-basierte Apache-Software¹, welche mit einem Anteil von über 60% im Webserversegment quasi das Rückgrat des Internet darstellt.

Weniger bekannt ist die Tatsache, dass sich Linux vorzüglich als dedizierter Server für multimediale Inhalte eignet. Projekte wie *VideoLAN* [45] und *dvbstream* [159] schicken sich an, die Welt der multimedialen Netzwerke zu erobern. Das vorliegende Kapitel greift zunächst noch einmal einige Grundlagen der Netzwerktechnik auf, um ein funktionsfähiges Gerüst für ein Videonetzwerk zu errichten, auf welchem dann spezielle, videobasierte Applikationen aufgesetzt werden können. Hierbei ist das Augenmerk auf die einfache Handhabung gerichtet. Zudem werden grundsätzliche Überlegungen zum Streaming von Videoinhalten angestellt, bevor einige Musterlösungen in Sachen Videoübertragung im Netz präsentiert werden. Das Kapitel wird durch Anleitungen zum Thema „Drahtlose Übertragung von Videomaterial“ abgerundet.

10.1 Grundlagen der Netzwerktechnik

10.1.1 Konfiguration der Hardware

Mittlerweile ist die für den Netzbetrieb zwingend notwendige LAN-Schnittstelle auf aktuellen Motherboards bzw. Laptops im Lieferumfang enthalten. Mit

¹Der Apache-Webserver wurde auch auf Microsoft Windows portiert, wenngleich diese Kombination im professionellen Umfeld eher selten zu finden ist

einer Übertragungsleistung von weniger als 100 MBit/s muss sich heute niemand mehr begnügen, wie sich zeigen wird, lässt sich mit einer solchen Bandbreite schon ein kleines Netzwerk ausreichend versorgen. Sollen größere Netze bedient werden, so empfiehlt sich eine Aufrüstung auf GBit-LAN-Technik, welche allerdings die Existenz einer glasfaserbasierten Infrastruktur voraussetzt.

Bei nachträglicher Aufrüstung des Rechners mit einer Netzwerkkarte wird diese nach dem Einbau vom Hardware-Daemon in der Regel problemlos erkannt. Die Einbindung als Ressource in das System erfordert zumeist noch etwas manuelle Nacharbeit, unter SuSE Linux erledigt man die entsprechenden Handgriffe mit YaST 2, welches die folgenden Konfigurationsmodule zur Verfügung stellt:

- *Netzwerk Basis* (Abbildung 10.1): Hier wird zunächst über den Menüpunkt *Konfiguration der Netzwerkkarte* die IP-Adresse des Rechners im LAN definiert, z. B. 192.168.0.1. Sollte im Netzwerk ein DHCP-Server laufen, so ist der Punkt „Automatische Adressvergabe“ auszuwählen, mehr dazu später. In einem Untermenü kann ebenfalls ein Rechner- sowie ein Domänenname definiert werden.
- *Netzwerk / Erweitert*: In diesem Modul können erweiterte Netzwerkfunktionen wie die Einrichtung von NFS / NIS Servern und Clients vorgenommen werden, dazu mehr im nächsten Unterkapitel.

Der so eingerichtete Rechner kann nun an ein bestehendes Netzwerk angeschlossen werden. Zu diesem Zweck hat sich der Einsatz von Hubs bzw. Switches bewährt. Im Gegensatz zu einem Hub ist ein Switch in der Lage, 1:1 Verbindungen zwischen zwei miteinander kommunizierenden Rechnern zu routen und somit die zur Verfügung stehende Bandbreite optimal auszunutzen. In letzter Zeit



Abbildung 10.1: Einrichten der Netzwerkkarte mit YaST 2

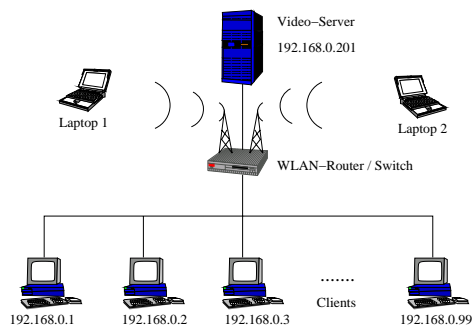


Abbildung 10.2: Topologie eines Videonetzes

kommen zudem Geräte auf den Markt, die multiple Funktionalitäten anbieten und gleichermaßen als LAN-Switch, DSL-Router und WirelessLAN (WLAN)-Accesspoint agieren können. Abbildung 10.2 zeigt eine mögliche Netzinfrastruktur auf der Basis eines solchen Switches.

10.1.2 Konfiguration der Netzwerkdienste

Zur Administration kleinerer Netzwerke genügt es, eine begrenzte Anzahl fester IP-Adressen zu verwenden, welche in der Datei `/etc/hosts` gepflegt wird. Spätestens dann, wenn die Anzahl der im Netz befindlichen Rechner die Zahl 50 übersteigt bzw. wenn eine größere Anzahl mobiler Geräte einzubinden ist, lohnt sich die Einrichtung eines DHCP-Servers, der die entsprechenden IP-Adressen zentral vergibt. Zu diesem Zweck muss das DHCP-Serverpaket der entsprechenden Distribution installiert sowie die im Paket enthaltenen Beispielkonfigurationsdatei `dhcp.conf` angepasst werden. Das folgende Listing zeigt die Minimalversion der Datei:

```
# /etc/dhcpd.conf - Beispiel
ddns-update-style none; ddns-updates off;
authoritative;
default-lease-time      600;
max-lease-time          7200;
option routers           192.168.0.1;
option domain-name-servers 192.168.0.1;
option broadcast-address 192.168.0.255;
# Ausgabe folgender 10 Adressen
subnet 192.168.0.0 netmask 255.255.255.0 {
range 192.168.0.11 192.168.0.20;}
```

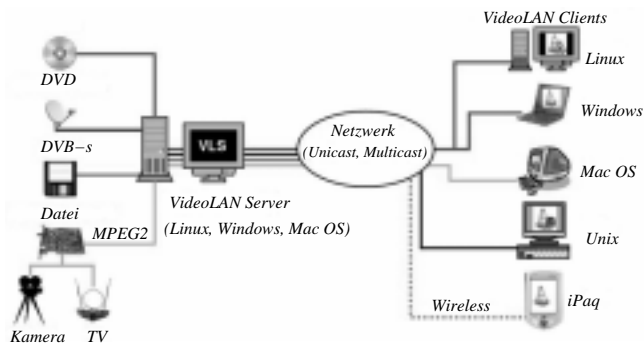


Abbildung 10.8: Topologie eines heterogenen VideoLAN (nach [45])

10.4 Das VideoLAN-Projekt

Mit dem VideoLAN-Projekt [45] ist der École Centrale Paris ein (mittlerweile im Rahmen des IBM Linux Challenge prämiertes) Meisterstück in punkto Video-streaming in heterogenen Netzen gelungen. Dabei handelt es sich um ein Client-/Serversystem zur Ausstrahlung von Videoinhalten in ein lokales Netz. Während die Serversoftware `vls` vorrangig Linux-basiert entwickelt wurde, existieren mittlerweile für sämtliche gängigen Betriebssysteme (als da wären Linux, Windows, Mac OS X, BeOS, BSD, Solaris, QNX, iPaq) `vlc`-Clients. Abbildung 10.8 zeigt den prinzipiellen Aufbau einer VideoLAN-Infrastruktur.

Im Wesentlichen wurde die VideoLAN-Software auf das Streaming von MPEG2-Inhalten (also DVD- und DVB-Material sowie VOB-Dateien) optimiert. Dabei werden folgende Softwarekombinationen angeboten:

- ❑ VideoLAN-Client `vlc` und VideoLAN miniServer (`v1ms`): Hiermit ist die einfachste Lösung zur Errichtung eines VideoLANs gegeben; man kommt dabei ohne komplexe Konfiguration der Serversoftware aus.
- ❑ VideoLAN-Client `vlc` und VideoLAN-Server (`v1c`): Hierbei handelt es sich um eine äußerst flexible Lösung, welche allerdings einen Mehraufwand in punkto Konfiguration erfordert.
- ❑ VideoLAN-Client `vlc` und VideoLAN ChannelServer (`v1cs`): Diese Lösung gestattet die einfache Auswahl des gesendeten Videocontents am Client über eine GUI.

Die Installation und Konfiguration der drei Ansätze soll im Folgenden vorgestellt werden. Dabei beschränken wir uns auf das Streaming von lokal abgelegten VOB- bzw. MPEG2-Dateien, da dies im Gegensatz zum DVB-s-Streaming mit der Software schon sehr stabil zu realisieren ist; man sollte sich diesbezüglich immer vor Augen halten, dass die VideoLAN-Software offiziell das Alpha-Stadium noch

nicht verlassen hat. Ziel unserer Bemühungen ist die Einrichtung einer digitalen lokalen Videothek für DVD-Material. Wir beginnen zunächst mit der optimalen Anpassung der Client-Software.

10.4.1 Der VideoLAN Client

Sämtliche Varianten bedingen die Installation des VideoLAN-Clients `vlc` auf dem Clientrechner (und zu Testzwecken möglichst auch auf dem Server). Die Software ist als Binärpaket bzw. RPM-Datei für Linux auf [45] erhältlich.

Wer eine optimale angepasste Version des Clients haben möchte, kann auch die Quellen herunterladen und anschließend mit dem gewohnten Linux-Dreisatz `./configure; make; make install` kompilieren und installieren, siehe hierzu auch Kapitel 3.3.2.3. Mögliche Optionen liefert `./configure --help`. Zu beachten ist, dass zum erfolgreichen Kompilieren der Clientsoftware die Pakete `libdvpsi` und `libdvcss` installiert werden müssen, beide sind ebenfalls auf [45] zu finden. Für den Betrieb als DVD-Player ist zudem die Installation des `a52dec`-Pakets erforderlich, wobei die `a52`-Bibliothek als Shared Library konfiguriert sein muss (`./configure --enable-shared`). Ein erster Test kann mit einer VOB-Datei erfolgen, die Sie entweder ebenfalls von [45] herunterladen können oder auf der dem Buch beiliegenden CD finden. Dazu ist die Clientsoftware von der Konsole über `vlc` zu starten und im Menü *File* der Pfad zu der entsprechenden VOB-Datei anzugeben (Abbildung 10.9).

Für andere Betriebssysteme, wie z.B. Windows 2000/XP, gibt es vorkompilierte Programme sowohl für die Client- als auch die Serversoftware auf [45], siehe Abbildung 10.10. Aufgrund der erzielbaren optimalen Systemanpassung wird



Abbildung 10.9: Abspielen einer VOB-Datei im Linux-`vlc`

Kapitel 11

Tipps, Tricks & Troubleshooting

Nachdem Sie sich nun anhand der vorangegangenen Kapitel durch die Höhen und Tiefen der Videoerstellung, -verarbeitung und -bearbeitung unter Linux gekämpft haben, soll in den folgenden kurzen Abhandlungen das „Danach“ angesprochen werden. Zunächst wird in Ihnen nach den ersten Experimenten mit der vorhandenen Hardware sicher der Wunsch reifen, ein speziell auf die Bedürfnisse der digitalen Videotechnik angepasstes System zusammenzustellen. Hier soll das Unterkapitel 11.1 als Leitfaden dienen. Das vorliegende Buch bietet einen Überblick über die Videoszene unter Linux; wer aber „am Puls der Zeit“ bleiben möchte, greift auf die in Kapitel 11.2 vorgestellten Informationsquellen zurück. Und wenn es im Videosystem mal „klemmt“, erfahren Sie in Teilkapitel 11.3 Strategien zum Lösen der digitalen Knoten.

11.1 Hardwareauswahl für Linux Videosysteme

Gilt schon für Standard-Linuxsysteme die Devise „erst schauen, dann kaufen“, so ist dieser Leitsatz in verstärktem Maße auf Linux-Videosysteme anzuwenden. Zunächst sollte vom potenziellen Käufer eine Feindifferenzierung bezüglich der Frage erfolgen, welchem Hauptzweck das System dienen soll. Folgende Video-PC-Typen sind zu unterscheiden:

- Der Multifunktionswohnzimmerplayer
- Das Videoschnittsystem
- Das Transcodiersystem
- Der VideoLAN-Client

Sicher lassen sich alle Funktionen auch in einer einzigen, entsprechend leistungsfähigen Maschine, die mit sämtlichen Ausstattungsmerkmalen der oben genannten Systeme versehen ist, erledigen. Die meisten Anwender werden sich aber mit

einem mehr oder weniger eng begrenzten Einsatzgebiet begnügen, so dass im Folgenden eine Aufstellung von Hardwaremerkmalen für die obigen Spezialsysteme gegeben werden soll.

11.1.1 Der Multifunktionsplayer fürs Wohnzimmer

Grundvoraussetzung für den Betrieb eines PCs in der guten Stube ist die Tatsache, dass dessen Lärmpegel so gering wie möglich gehalten wird. In letzter Zeit drängen zunehmend lautlose Komponenten auf den Markt, einen guten Übersichtsartikel zum Aufbau eines flüsterleisen Systems findet man z.B. in [167]. Die Leistungsfähigkeit des Systems selbst kann sich für den genannten Zweck in Grenzen halten, eine flüssige Wiedergabe von DVDs leistet die heute schon antik zu nennende Geräteklasse des Intel Pentium III bzw. AMD Athlon ab 800 MHz. So ist es durchaus denkbar, dass ein ausgemusterter PC die Aufgabe übernehmen kann, sofern die Lüftersysteme entsprechend leise gestaltet werden.

Die vorhandene Videoinfrastruktur (DVB-Anschluss, Analog-Satellitenanschluss, Kabelanschluss...) bestimmt die weitere Hardwareausrichtung des Systems; idealerweise findet man im Wohnzimmer den Anschluss an eine Satellitenanlage mit Universal-LNB vor, so dass sich für diesen Fall die Einrichtung eines *vdr*-Systems nach der Beschreibung von Kapitel 5.3 anbietet. Dies schwebte auch dem Autor der *vdr*-Software, Klaus Schmidinger, in erster Linie bei der Entwicklung seines Programms vor, siehe Abbildung 11.1. Eine weitere Übersicht derartiger wohnzimmer-tauglicher Lösungen finden Sie auf [168].

Tabelle 11.1 zeigt eine mögliche Hardwarekombination für ein wohnzimmer-taugliches System. Bei den angegebenen Preisen handelt es sich um Straßenpreise, die zur Zeit der Drucklegung galten und somit nur als grobe Anhaltspunkte anzusehen sind. Die Empfehlungen der angegebenen Komponenten sind durch



Abbildung 11.1: Klaus Schmidingers *vdr*-System (Bildquelle: [1])

Index

Symbole

/boot 13
/etc/fstab 15
/swap 13
/video 13, 130, 140
2 GByte Limit ... 109, 113
50-Hz-Brummen 17

AC3-Samples 289

A

A/D-Wandler 86, 278
AC3 . 65, 69, 214, 288, 357
Accesspoint 299
 Integration in LAN 332
 Webinterface 332
aktion 62
ALSA 270, 357
 /dev/snd/controlCn
 272
 /dev/snd/pcmCnDm .
 272
 /proc/asound ... 272
 Module 272
 OSS Emulation 272
 Sequenzler 272
 Soundkarte einbinden .
 272
 Tonaufzeichnung .. 285
alsactl 276
alsamixer 275
 AC3 Konfiguration 291

Aussteuerung 275
 GUI 277
 Mic Boost 285
 Mute 275
 Tastatursteuerung . 276
Anamorphie 216
Apache 297
API 269, 357
Ardour 284
aRts 274
Astra-Satellit 126
atitvout 352
Audacity 284
 Aufnahme 285
 Hüllkurve 286
 Tonüberblendung . 286
Audio-PID 310
aumix 294
AVI 357
avi-Format 53
avicap 88
 Konfiguration 88
avidemux 207
 Codecauswahl 209
 Cropping 211
 Deinterlacing 211
 Encodierbeispiel .. 210
 Filter 211
 Menü 209
 Synchronisierung . 210
avifile
 Bibliothek 62
 Installation 63

Kompilieren 87
avimerge 236
aviplay 90
avisplit 218, 252

B

Backbone 303
Backup 14
Bandbreite 303
BeOS 317
BetaMax 191
BIOS 15
Bitrate 44
Bitrate-Kalkulator ... 214,
 251
Block Device 270
Broadcast 301
Broadcastadresse 320
BSD 317

C

Camcorder 110
 Anschluss 110
CD
 Image erstellen 244
 Rippen 280
CD Bake Oven 249
CD-Brenner
 Einbau 242
CD-Image 224
CD-Qualität 279
CD-R 241
 überlange Rohlinge 247
CD-R-Brenner 242

- cdda2wav 245
CDDDB 283
cdparanoia 280
cdrdao 253
cdrecord 243, 248
Centerlautsprecher .. 288
Character Device 270
checkinstall 29
Chunk 236, 259
CI-Slot 123
CICAM 135
Cinelerra 99, 164
 Überblendung 178
 Übersicht Filter 180
 Audioüberblendung ..
 188
 Audiokanal 187
 Aufzeichnung 101
 Clips zusammenfügen .
 167
 Compositor ... 100, 166
 Effekte 177
 Film exportieren .. 166,
 183
 Installation 99
 Konfiguration 100
 Material exportieren ...
 181
 Material importieren ..
 164, 177
 Projektfenster 165
 Renderfarm Modus 183
 Resources 100, 179
 Schnitt Einzelfilm . 165
 Sprache einbinden . 186
 Ton aufnehmen ... 187
 Ton einblenden 186
 Tonbearbeitung ... 185
 Viewer 100
Closed Source 157
Closed Source Treiber 341
Cluster 234
 Knoten 235
Cluster Control Daemon .
 237
cmd/etc/hosts 264
Codec 53, 357
Codecs
 Überblick 44
 Installation 195
Common Interface ... 123
Composite 112
Convergence-Treiber 123,
 355
CPU-Last 304
Cray 234
Cron Daemon 115
crontab 116
Crossover-Plugin 48
 Installation 50
 Konfiguration 51
 Quicktime 51
 Windows Media 52
CrossOver-Plugin ... 231
crypt 322
CSS .. 63, 65, 74, 219, 306,
 357
CVS 357
 auschecken 105
D
Daemon 81, 358
DAO 224
Decodierung 53
DeCSS 65, 73
Developer-Paket 352
DGA-Modus 77
DHCP 298
DHCP-Server 299
 Konfiguration 300
DiSeq 358
Disk-at-Once-Modus 253
DivX ... 43, 192, 251, 281,
 305, 359
 libdivxencore.so .
 202
 5.x pro 192
 Archivierungsstrategie
 215
DivX-Pro-Codec 233
divx4linux 63
 Installation 195
DivX;-) 44
divxcalc 252
DMA 350, 358
DMA-Modus .. 35, 37, 75,
 243, 246
DNS-Server 264
Dolby Digital 65
Dolby 5.1 72, 277, 287
Dolby Digital ... 119, 214,
 269, 273
 Center 287
 Farbcodierung 289
 Lautsprecherposition ..
 289
 Soundkarte anschließen
 288
Dolby Surround 287
Dolby Trailer 289
Domänenrechner 264
doom9
 DivX Guides 200
Downstream 303
Drive Image 14
DSL 45, 121, 192, 358
DSL-Router 299
DTS 287
Dual-Head-Betrieb .. 343
Dualview 81
DV 110, 153, 358
 Camcorder 86
DVB 72, 119, 358
DVB-Budget-Karte .. 122
DVB-c 120
DVB-PCI-Karte
 Installation 122
 Marktübersicht 123
DVB-s 120
 Content 120
 Karte 119
DVB-t 120
DVB-Treiber
 Installation 124
 Modul laden 124
dvbstream 308
 Installation 309
 Interaktion über
 telnet 312

- Kanal umschalten . 312
 PID ummappen ... 310
 RTP Übertragung . 312
 DVD 42, 63, 358
 Brennen 255, 259
 Formate 193
 Kapazität 64
 Sicherheitskopie ... 219
 Transcodierung in
 S-VCD 228
 Verschlüsselung 73
 DVD+R 255
 DVD+RW 241, 255
 DVD-Kopien 219
 DVD-Mastering 260
 DVD-R 241, 255
 DVD-Recorder 121
 DVD-RW 241, 255
 dvd::rip 206, 213, 219
 Cluster Control Modus
 238
 Encodierung 222
 Installation 220
 Rippen eines Titels 221
 Schneiden und
 Skalieren 221
 Zoom Calculator .. 222
 dvdauthor 229, 256
 dvdrecord 259
 dvdrttools 259
 dvgrab 155
 dvr 86, 92, 114
 Installation 92
 Konfiguration 93
 DynIP-Server 333
- E**
 Eierkopfeffekt
 (Anamorphie) ... 216
 emu-tools 292
 emu10k1-Chip 288
 Encodierbitrate 303
 Encodierrate 203
 Encodierung 53
 im Cluster 234
 EPG 133, 359
 ESD 274
- Ethernet Device
 Deaktivierung 337
 EX 287
 ext2fs 12
 ext3fs 12, 114
- F**
 FAQ 359
 FAT 12
 Festplatte
 Lebensdauer 241
 ffmpeg 194, 213, 308
 Encodierung 314
 Formatwahl 315
 Optionen 315
 ffmpeg-Projekt 196
 ffmpeg-Tools 313
 Installation 314
 ffserver 315
 Streaming 316
 FireWire 110, 152
 FLTK 277
 Four Character Code
 Übersicht 193
 fping 237
 fps 359
 Framebuffermodus .. 342
 Funktransmitter 329
 Anschlussdiagramm ..
 330
- G**
 GBit-LAN 298
 gcc 28
 GIF 42, 359
 glav 97
 GLX 35
 GNOME 274
 GPL 125, 193, 279
 GPSYCHO 279
 Groves 242
 GRUB 31
 gscanbus 155
 GUI 359
 gv4l 205
- H**
 Halloween-Dokumente ..
 265
 hdparm 37
 HDTV 85, 359
 Header 352
 Helix Producer 229
 Hi8 110, 360
 Hot Spot 331
 Hotplug 261
 Hub 298
 huffyuv 94
- I**
 I2C-Bus 20
 IDE 360
 IDE-Platte 261
 IEEE1394 .. 110, 152, 261,
 360
 Technische
 Spezifikation 152
 Hardware 153
 Module laden 154
 IEEE802.11b 330
 IFO 229, 250
 Infrarotschnittstelle .. 141
 Interlace 109
 Iomega Buz 103
 IP-Adresse 298, 360
 IP-Adressraum 300
 iPaq 317
 IS09660 247
 ISDN 192
 isoinfo 258
 ITU 289
- J**
 Joliet 247
 JPEG 42, 360
 jpeg-mmx 95
- K**
 k3b 249
 DVD rippen 251
 Konfiguration 250
 Kammeffekt 109
 KDE 26

- Develop Paket 30
Systemüberwachung ..
 304
KDE-Kontrollzentrum ...
 275
KDE3 130, 249
Kernel
 cloneconfig 32
 kompilieren 33
 Konfiguration 32
 Module 31
 Module installieren . 33
 Quellen 30, 32, 124
 xconfig 32
kFIR-Chip 104
kino 155
kmenccoder 205, 206
 Codecauswahl 207
 Untermenüs 207
kmix 24, 275, 276, 285
 Soundprofil erstellen ..
 277
KMix 106
Knoppix 13
Kompressionsverfahren .
 11
KOnCD 248
kpackage 19, 27
kvdr 130, 314
 Hotkeys 132
 Installation 130
 Oberfläche 132
- L**
Lüftersystem 340
lame 87, 90
Lame 279, 280
 Encodierung 280
lamp 62
LAN 241, 360
LAN-Switch 299
LAN-Videospeichers 261
Laptop 351
Laserdisk 191
Latenzzeit 274
lav2wav 225
lav2yuv 225
lavplay 97, 108
lavrec 96
 Software Encoding . 97
LFE 288, 360
LGPL 125
liba52 69
libavcodec 193
libavdcss 74
libavdread 66, 69
libmovtar 95
libraw 155
libtool 87
lilo 31
LinDVD 74
Line In 17, 24
Line Out 17, 78
LIRC 141, 361
 Module laden 143
 Tasten anlernen ... 143
Livesystem 14
LML33 103
LNB 340, 360
Log-File 203
Loopbackdevice 244
lsmod 20
LVM 12
lynx 345
 Navigation 345
- M**
Mac OS 317
MAC-Adresse .. 334, 336
MacOS 247
Macrovision 83
Mailingliste 347
MainActor 101
 2D-Textobjekt 172
 Ausblendung 172
 Cut Modus 169
 Deinterlacing 174
 Effektspur 171
 Export Modul 182
 Exportformate 182
 Filter 170, 174
 Gammaprofil justieren
 174
 Helligkeit justieren 174
Installation 101
Kontrast justieren . 174
macap 102
madvcap 102
maseq 102
Material exportieren ...
 181
matool 102
Menüsprache wählen .
 168
MPEG Export 168
MPEG Import 168
Musik einbinden .. 187
Ränder beschneiden ...
 175
Sequenzener 168
Szenenübergänge . 170
Texte einbinden ... 171
Videoeditor 168
Videopfad definieren ..
 176
Videospuren 169
makedvd 257
Mantelstromfilter 18
mccrypt 322
mencoder 194
 lavcopts 199
 libavcodec 199
 Codecauswahl 197
 Deinterlacefilter ... 198
 DivX Optionen 198
 Encodierparameter 200
 Encodierung DivX . 196
 Postprocessing 198
 Zweipassencodierung .
 199
MIME 361
miniDVD 255, 258
miniVLCS 327
 Konfiguration 327
MJPEG ... 42, 94, 103, 361
MJPEG-Tools 94, 196
 Hardware Capturing ..
 107
 Kompilieren 94
 S-VCD erstellen ... 108

- Synchronisation ... 108
 VCD erstellen 108
 mkisofs 243
 mkpasswd 322
 Mountpoint 12, 250
 mp2enc 225
 MP3 .. 192, 278, 279, 284, 362
 MPEG 42, 361
 MPEG-LayerIII 278
 MPEG1 361
 MPEG2 .. 43, 65, 121, 278, 361
 MPEG21 43
 mpeg2divx 205
 MPEG3 361
 MPEG4 192, 213, 250, 251, 361
 MPlayer 54
 Cache Modus . 311, 337
 DGA Modus 59
 DVD Navigation ... 69
 DVDs abspielen ... 69
 Framebuffer 58
 Installation 54
 Konfiguration 56
 Screenshot 58
 Skins 55
 Steuerung 57
 mplex-Tool 226
 MRL 362
 MTRR 76
 Multi-Angle 65
 Multi-LNB 121
 Multicast 301, 311
 Multicast-IP 327
 Multiplexen 224
 Multitasking 274
 Muxen 224
- N**
- nasm 195
 ncurses 275
 Netzlast 303
 Netzwerkkollisionen 303
 Netzwerkperformance
 Analyse 303
- Newsgroup 345
 NFS ... 235, 237, 262, 298, 362
 kernelbasiert 263
 NFS-Server 235
 NFS-Share 262
 nice 218
 NIS 298, 362
 NIS-Client 300
 NIS-Server 262, 300
 Noatun 62
 NTFS 13
 ntop 303
 NTSC 85, 362
 ntuzzap 127
 NVidia-Chip 77
 nvtv-Tool 80
- O**
- Ogg-Projekt 281
 Ogg-Vorbis 193, 201, 218, 278, 281
 Integration in KDE 283
 Kompressionsverluste . 284
 qualitätsbasiertes
 Encoding 282
 ogg123 283
 oggdec 283
 ogle 69
 GUI 70
 Steuerung 70
 OHCI 153
 On-Demand-Server .. 332
 on-the-fly 247
 Open Quicktime 49
 Open Source Software 25
 Open Source-Software ... 362
 OpenDivX 192
 OSD 128, 141, 362
 OSS 269, 362
 /dev/audio 271
 /dev/dsp 271
 /dev/mixer 271
 /dev/sndstat ... 271
 AC3 Konfiguration 292
- API 272
 Soundkarte einbinden . 271
- P**
- Packman 87
 PAL ... 23, 41, 82, 85, 103, 256, 363
 Partition Magic 13
 Partitionierung 11
 Passwort
 verschlüsseltes 323
 Pay-TV 121
 PCM ... 65, 272, 278, 287, 363
 PCMCIA 363
 Karte 334
 Kartenmanager 335
 Treiber 334
 Perl 237
 Perl-Gtk-Modul 220
 PHP4 92
 Pinnacle DC 10+ . 94, 103
 Einbau 104
 Treiber einbinden .. 105
 Pits 242
 Plug-and-Play 273
 Plugger 48
 Pocket PC 343
 Polarisationsrichtung 310
 PowerDVD 74
 PSU 235
- Q**
- QNX 317
 QT3 130, 249
 Quantisierung 226
 Quarterpixelnorm ... 213
 quicktime4linux 95
- R**
- RAID 363
 RealMedia 44, 229
 Realplayer 44
 Konfiguration 46
 MIME-Zuordnung . 47
 Plugin 46

- ReiserFS 12, 113, 306
Rettungssystem 34
RGB 363
Ripper 219
Rock Ridge 247
Router
 Administrationstools ..
 333
RPC
 Mount-Daemon ... 263
 NFS-Daemon 263
 Portmapper 263
RPM 363
 installieren 26
 Kommandozeile 27
 Paket bauen 29
 Source 29
 vorkompliert 25
RSA-Fingerprint 301
RTP 312
Runlevel 80
Runlevel-Editor . 125, 300
- S**
S-VCD 43, 108, 364
 Herstellung ... 224, 253
 Test 254
S-VHS ... 22, 83, 112, 191,
 364
S-Video 84
S/PDIF 288, 364
Samba 262, 363
 nmbd 265
 smb.conf 266
 smbd 265
 Passwortliste 267
 Share 264
Sampling Rate 278
Satellitenanlage
 Justierung 126
Satellitenspiegel 126
SCART .. 17, 77, 112, 330,
 364
SCSI 15, 261, 364
SCSI-Emulation 242
SCSI-Platte 261
SDB 350, 351
- SI 133
Slimline-Gehäuse 341
SMB 364
SMP 364
SMP-Kernel 343
Soft-DVD-Player 259
Solaris 317
Soundblaster 273
Soundblaster Live ... 277
Soundkarte
 DAC 271
Soundstudio 188
Specfile 29
Spilling System 114
ssh 300
 Keyerzeugung 238
SSH 237, 364
Standalone-DVD-Player .
 226
Subwoofer 288
SUID-Bit 245, 248
SVCD-Profil 225
SVDR-Protokoll 313
Sweep 284
Switch 298
Symbolrate 310
- T**
tarballs 28
tccat 307
TCP/IP 365
tcpdump 303
tcprobe 307
Telefoniequalität 279
telnet 149, 237
THX 287
Timeshifting 121, 148
Torvalds, Linus 297
tosvcd 228
Trägerfrequenz 121
transcode .. 194, 200, 305
 Asynchronitäten
 beheben 228
 Audiocodecs 217
 Clustermodus 235
 Encodierung mit XviD .
 202
- Installation 201
Kompilierung 201
Korrektur
 Seitenverhältnis . 216
Optionen 204
PostProcessing-Filter ..
 202
Randbeschneidung 216
resample-Parameter ...
 227
Skripte 204
Zweipassencodierung .
 218
Transcodiersystem ... 339
Transcodierung 158
Transponder 121, 310
ts2ps 311
TTL-Wert 327
tuxzap 127
 .dvbrc 128
 Konfiguration 127
TV-Karte
 Ton aufnehmen ... 277
TV-Out 77
TwinView 78
- U**
UDF 66, 365
UDP-Port 319
ULTRA-ATA 15
Unicast 301
Universal-LNB 119
Universalrouter 333
Urheberrecht 219
USB 2.0 152
Usenet 345
- V**
V4L 365
v4l-conf 131
V4L-Quelle 206
V4L2 16, 365
VCD 43, 108, 365
 Herstellung 224
vcdimager 253
vcr 86, 90

- Installation 90
 Konfiguration 90
 Optionen 91
 WebVCR 92
 Zeitgesteuerte
 Aufnahme 115
vdr
 keys-pc.conf ... 140
 setup.conf 138
 all-in-one-Patch ... 145
 Bildlaufanzeige 159
 Daemon Modus .. 130,
 137
 DivX Wiedergabe . 145
 DVD Wiedergabe .. 145
 Einstellungen 135
 Entwicklerversion . 355
 Feinschnitt 162
 Fernbedienung 141
 Fernseher anschließen .
 140
 Grobschnitt 161
 Installation Software ..
 129
 Material archivieren ...
 252
 Max. Dateigröße .. 138
 Menüaufbau 134
 mp3 Wiedergabe .. 145
 MPlayer einbinden 145
 Multispeedmodus . 160
 Navigation 160
 Schnittmarken 162
 Sofortaufnahme ... 136
 Streams
 zusammenfügen 215
 Timer 137
 Timeshifting 148
 Umwandlung in DivX .
 213
 Videoaufzeichnung 136
 Videoschnitt 159
 VDR-Projekt . 75, 124, 128
 vdr2divx 205
 vdradmin 129, 149
 Browseranbindung 151
 Kanalwahl 149
 Kommandoübersicht ..
 150
 VFAT 113
 VHS 110, 191, 365
 VHS-Videokassette .. 112
 Video on Demand ... 302
 Video Overlay 76
 Video-PC 339
 Video-PID 310
 Video2000 191
 Video8 110, 365
 Videobitrate
 optimale 214
 VideoLAN 72, 297
 VideoLAN Channelserver
 327
 VideoLAN Miniserver ...
 319
 VideoLAN Server ... 320
 VideoLAN-Client 317
 VideoLAN-Projekt ... 317
 Videoschnitt
 Definition 158
 Flussdiagramm 158
 linear 158
 nichtlinear 158
 Videoschnittsystem .. 339
 Videsequencer 158
 Videosequencer 158
 Videosequencer 158
 Videostreaming 45
 Aktiv 301
 Broadcasting 301
 Channel 301
 Passiv 301
 Videowebserver 316
 VirtualDub 203, 206, 211,
 230
 Betrieb unter Linux 231
 vlc 62, 72, 317
 GTK-Oberfläche 73
 Installation 318
 vlcs 317
 vlms 319
 Broadcasting 319
 Streaming 319
 vls 320
 telnet
 Befehlsübersicht . 326
 vls.config 321
 Daemon Modus ... 324
 Installation 320
 Konfiguration 321, 323
 Multicasting 326
 Programme starten 325
 Streaming 324
 Unicaststreaming .. 326
 Zugriff via telnet 323
 vmlinuz 31
 VOB 229, 250, 306
 VOB-Datei 67
 VTS-Datei 67
W
 WEP 333, 336, 365
 Windows 247, 317
 wine 49, 231, 260
 Installation 231
 Konfiguration 231
 WinTV-PVR 104
 WirelessLAN-Technik ...
 331
 WLAN 299, 329, 366
 Ad hoc Modus 332
 Infrastructure Modus ...
 332
 NFS nutzen 337
 Samba nutzen 337
 WLAN-Karte
 Integration 335
 WLAN-NG-Projekt .. 334
 WLAN-Standards
 Übersicht 331
X
 X11R6 36
 x2divx 205
 xawtv 19, 86
 .xawtv 23
 DVB Empfang 126
 Frequenztabelle 21
 Kanaleditor 22
 Konfiguration 21

Index

- Recordmodus 23
- TV-Empfang 21
- Vollbildmodus 23
- xcdroast 245
 - CD mastern 246
 - Konfiguration 245
- XFree86Config 77
- XFree86 35
- xine 60
 - d5d Plugin 69
 - DVDs abspielen 68
 - Installation 60
 - Konfiguration 61
 - Konfiguration von AC3 290
 - Playlist 115
 - xine-ui 60
- XML 71, 366
- xmms 62
- xmovie 62
- XSVCD 224
- Xv 36, 37
- Xv-Extensions 76, 78
- XviD 250, 251, 305
- XviD Codec
 - libxvidcore.so 195
 - CVS-Snapshot 195
- XviD-Codec 203
- XviD-Projekt 192
- Y**
- YaST 2
 - Backupmodul 13
 - Netzwerkkarte
 - einbinden 298
 - Softwaremodul 19
- Soundkarte einbinden . 274
- Soundmixer 274
- TV-Modul 19
- Yopy 344
- YUV 36, 366
- yuvscaler 225
- YV12 202
- Z**
- Zaurus 344
- Zielauflösung 214
- Zoran-Chip 94, 104
- Zoran-Module
 - Einbinden 107
 - Laden 106
- Zweipassencodierung ... 198