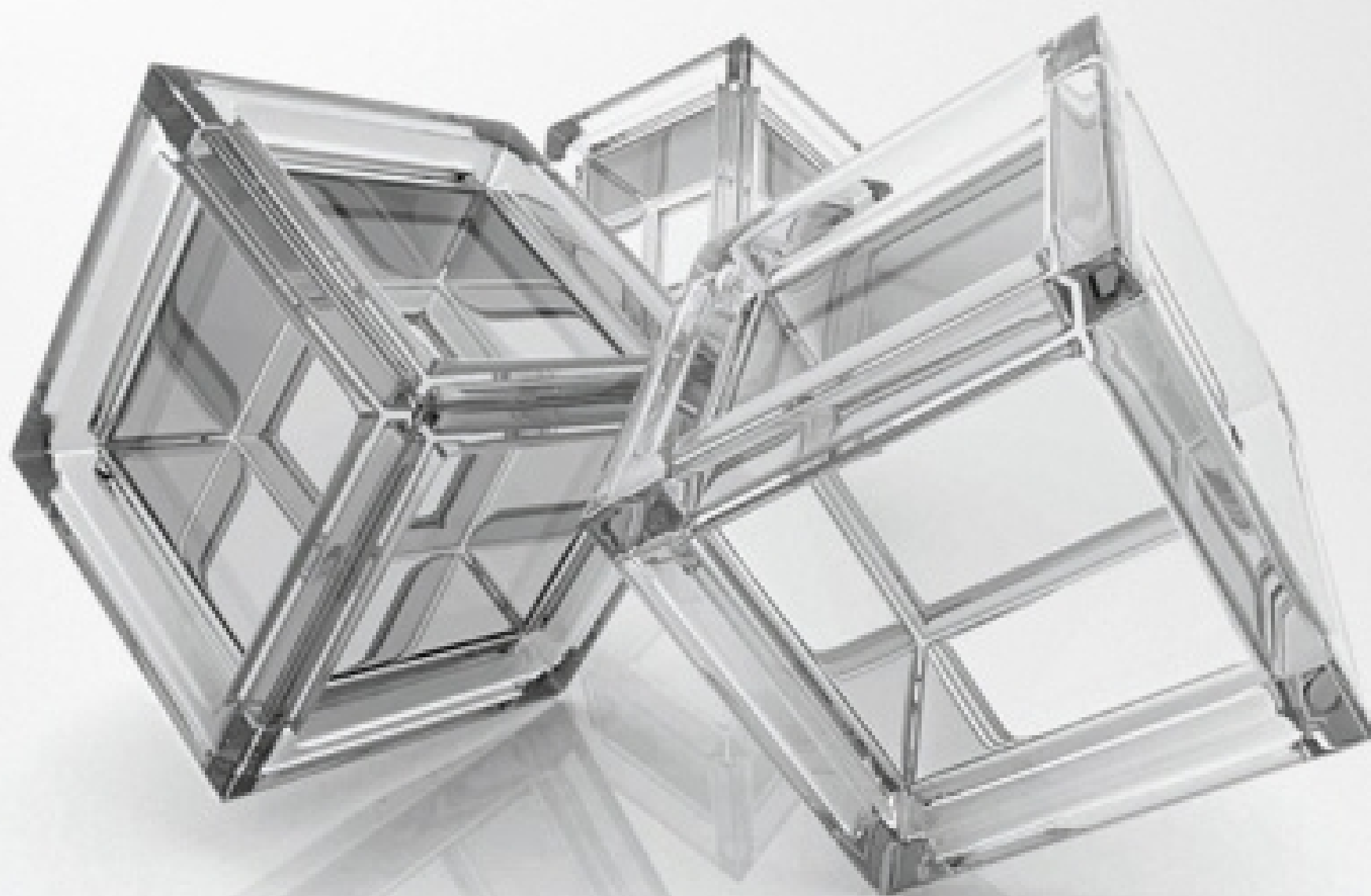


Ab  
Version 3

# VirtualBox

Installation, Anwendung, Praxis



- Von den Grundlagen bis zum praktischen Einsatz
- Troubleshooting, Migration und Kommandozeile
- Für Ein- und Umsteiger geeignet

Dirk Becker

# VirtualBox

Installation, Anwendung, Praxis

# Liebe Leserin, lieber Leser,

IT ohne Virtualisierung ist heute kaum mehr vorstellbar. Und sicher kennen Sie bereits wesentliche Vorteile von Virtualisierungssoftware. Ob Sie die Software privat oder im beruflichen Einsatz verwenden: Mit einer virtuellen Maschine testen Sie schnell neue Software oder ganze Betriebssysteme, nehmen für bestimmte Anwendungsbereiche vorbereitete Maschinen kurzerhand in Betrieb und das alles, ohne Ihr eigentliches Produktivsystem zu verändern oder zu gefährden. Seit die Virtualisierungssoftware VirtualBox im Jahr 2007 in der ersten Version erschien, hat das Open-Source-Tool viele Nutzer überzeugt. Leistungsfähigkeit, Performance, Bedienbarkeit und die große Zahl der unterstützten Systeme sprechen für sich.

Praktisch und nützlich also, was man mit der Software alles erreichen kann. Und obwohl der Autor Dirk Becker natürlich vorrangig das Ziel verfolgt, Ihnen systematisch den nutzbringenden Umgang mit VirtualBox zu erläutern, schafft er es, den Lesern mit seinem Buch auch den Spaß zu vermitteln, den die Arbeit mit virtuellen Maschinen mit sich bringt. Der Experimentierfreude sind bei der Arbeit mit VirtualBox, gerade auch aufgrund der Schnelligkeit und der besonderen grafischen Fähigkeiten des Systems, kaum Grenzen gesetzt.

Gerne überlasse ich Sie nun der langjährigen beruflichen und privaten Erfahrung des Autors mit VirtualBox und Lösungen anderer Hersteller. Dirk Becker zeigt Ihnen gezielt die Vorteile von VirtualBox, gibt aber zugleich auch wertvolle Hinweise auf die Stärken anderer Lösungen wie VMware oder Citrix Xen Server.

Dieses Buch wurde mit großer Sorgfalt geschrieben, lektoriert und produziert. Sollte dennoch etwas nicht so funktionieren, wie Sie es erwarten, dann scheuen Sie sich nicht, sich mit mir in Verbindung zu setzen. Ihre Anregungen und Fragen sind jederzeit willkommen.

Ich wünsche Ihnen viel Vergnügen beim Lesen und viel Spaß beim Arbeiten mit VirtualBox!

**Ihr Jan Watermann**

Lektorat Galileo Computing

jan.watermann@galileo-press.de

www.galileocomputing.de

Galileo Press • Rheinwerkallee 4 • 53227 Bonn

# Auf einen Blick

1	Einführung .....	17
2	Virtualisierung .....	35
3	VirtualBox installieren und einsetzen .....	63
4	Praxisbeispiele .....	135
5	Weitere Funktionen und Tipps .....	199
6	Migration und Import .....	259
7	Probleme und Fehlersuche .....	273
8	Kommandozeilentools .....	283
9	Glossar .....	313



Der Name Galileo Press geht auf den italienischen Mathematiker und Philosophen Galileo Galilei (1564–1642) zurück. Er gilt als Gründungsfigur der neuzeitlichen Wissenschaft und wurde berühmt als Verfechter des modernen, heliozentrischen Weltbilds. Legendär ist sein Ausspruch *Eppur se muove* (Und sie bewegt sich doch). Das Emblem von Galileo Press ist der Jupiter, umkreist von den vier Galileischen Monden. Galilei entdeckte die nach ihm benannten Monde 1610.

**Gerne stehen wir Ihnen mit Rat und Tat zur Seite:**

jan.watermann@galileo-press.de bei Fragen und Anmerkungen zum Inhalt des Buches  
service@galileo-press.de für versandkostenfreie Bestellungen und Reklamationen  
julia.bruch@galileo-press.de für Rezensionen- und Schulungsexemplare

**Lektorat** Jan Watermann, Sebastian Kestel

**Fachgutachten** Ramon Kukla, Niederkassel

**Korrektorat** Heike Jurzik

**Cover** Barbara Thoben, Köln

**Titelbild** Barbara Thoben, Köln

**Typografie und Layout** Vera Brauner

**Herstellung** Iris Warkus

**Satz** Typographie & Computer, Krefeld

**Druck und Bindung** Bercker Graphischer Betrieb, Kevelaer

Dieses Buch wurde gesetzt aus der Linotype Syntax Serif (9,25/13,25 pt) in FrameMaker.  
Gedruckt wurde es auf chlorfrei gebleichtem Offsetpapier.

**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 978-3-8362-1374-5**

© Galileo Press, Bonn 2009

1. Auflage 2009

Das vorliegende Werk ist in all seinen Teilen urheberrechtlich geschützt. Alle Rechte vorbehalten, insbesondere das Recht der Übersetzung, des Vortrags, der Reproduktion, der Vervielfältigung auf fotomechanischem oder anderen Wegen und der Speicherung in elektronischen Medien. Ungeachtet der Sorgfalt, die auf die Erstellung von Text, Abbildungen und Programmen verwendet wurde, können weder Verlag noch Autor, Herausgeber oder Übersetzer für mögliche Fehler und deren Folgen eine juristische Verantwortung oder irgendeine Haftung übernehmen. Die in diesem Werk wiedergegebenen Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. können auch ohne besondere Kennzeichnung Marken sein und als solche den gesetzlichen Bestimmungen unterliegen.

*Dieses Buch widme ich meinem Patenkind Hanna und ihren Eltern.  
Good Luck Pingel Pit.*

# Inhalt

Vorwort ..... 13

**1 Einführung ..... 17**

1.1 Über dieses Buch ..... 17

1.1.1 Ziele und Anforderungen des Buches ..... 17

1.1.2 Hinweise zum Umgang mit dem Buch ..... 21

1.2 Personal Computer ..... 22

1.2.1 Entwicklung ..... 23

1.2.2 Komponenten eines PCs ..... 25

**2 Virtualisierung ..... 35**

2.1 Begriff und Technologie ..... 35

2.1.1 Warum virtualisieren? ..... 35

2.1.2 Pro und Kontra ..... 38

2.1.3 Emulation versus Virtualisierung ..... 39

2.1.4 Emulatoren ..... 40

2.1.5 Virtualisierung ..... 43

2.1.6 Virtualisierungssoftware ..... 47

2.2 Planung ..... 52

2.2.1 Hostsystem ..... 53

2.2.2 Virtuelle Maschine ..... 60

**3 VirtualBox installieren und einsetzen ..... 63**

3.1 Hersteller ..... 63

3.1.1 innotek GmbH ..... 63

3.1.2 Sun und xVM ..... 64

3.1.3 Oracle ..... 65

3.2 Installation und Konfiguration ..... 65

3.2.1 Einsatzgebiet und Funktionen ..... 65

3.2.2 Hostsysteme ..... 66

3.2.3 Gastsysteme ..... 66

3.2.4 Versionen und Lizenzen ..... 67

3.3 Installation ..... 70

3.3.1 Download ..... 71

3.3.2 xVM VirtualBox unter Windows ..... 71

	3.3.3	xVM VirtualBox unter Linux .....	78
	3.3.4	VirtualBox OSE unter Linux .....	80
3.4		Die erste virtuelle Maschine .....	83
	3.4.1	Assistenten aufrufen .....	84
	3.4.2	Name und Gastsystem .....	85
	3.4.3	Hauptspeicher .....	85
	3.4.4	Festplatten-Image erzeugen .....	85
	3.4.5	Dynamisch wachsendes Medium .....	86
	3.4.6	Speicherort und Größe .....	86
	3.4.7	Zusammenfassung Festplatten-Image .....	87
	3.4.8	Zusammenfassung virtuelle Maschine .....	87
	3.4.9	Fertige virtuelle Maschine .....	87
3.5		Grafische Oberfläche .....	88
	3.5.1	Aufbau .....	88
	3.5.2	Globale Einstellungen .....	90
	3.5.3	Manager für virtuelle Medien .....	95
	3.5.4	Konfiguration einer virtuellen Maschine .....	97
	3.5.5	Sicherungspunkte – Snapshots .....	99
	3.5.6	Beschreibung .....	99
	3.5.7	Fenster für virtuelle Maschinen .....	100
	3.5.8	Hilfe .....	101
3.6		Kommandozeilentools .....	102
	3.6.1	VBox/Manage .....	106
	3.6.2	VBoxSDL .....	110
	3.6.3	VBoxHeadless .....	113
	3.6.4	rdesktop-vrdp .....	114
3.7		Aufbau einer virtuellen Maschine .....	115
	3.7.1	Konfigurationsdatei .....	116
	3.7.2	Festplatten-Images .....	118
	3.7.3	CD/DVD-Abbilder .....	118
	3.7.4	Logdateien .....	118
	3.7.5	Sicherungspunkte (Snapshots) .....	119
3.8		Hardware .....	119
	3.8.1	Allgemein .....	119
	3.8.2	Festplatten-Controller .....	122
	3.8.3	Audio-Adapter (Soundkarte) .....	123
	3.8.4	3-D-Beschleunigung .....	123
3.9		Netzwerk .....	123
	3.9.1	Adaptertyp (Netzwerkkarte) .....	124
	3.9.2	Anschlussart .....	125
3.10		Gasterweiterungen .....	132

3.11	Open Virtualization Format (OVF) .....	133
3.12	Fertige virtuelle Maschinen (Virtual Appliances) .....	133

## **4 Praxisbeispiele ..... 135**

4.1	Live-CDs testen .....	135
4.1.1	Quellen .....	136
4.1.2	Virtuelle Maschine einrichten .....	137
4.1.3	Live-System starten .....	142
4.1.4	Beenden der virtuellen Maschine .....	145
4.1.5	Fernsteuerung (Remotezugriff) .....	146
4.2	Windows XP .....	149
4.2.1	Virtuelle Maschine erstellen .....	149
4.2.2	Manager für virtuelle Medien .....	152
4.2.3	Installation von Windows XP .....	154
4.2.4	Gasterweiterungen .....	155
4.2.5	Netzwerk mit NAT .....	158
4.3	Linux-System .....	159
4.3.1	Virtuelle Maschine erstellen .....	160
4.3.2	Installation von Ubuntu .....	161
4.3.3	Gasterweiterungen unter Linux .....	165
4.3.4	Ruhezustand .....	168
4.4	Windows 7 .....	169
4.4.1	Virtuelle Maschine einrichten .....	170
4.4.2	3-D-Beschleunigung .....	171
4.4.3	Audio-Controller .....	172
4.4.4	Installation von Windows 7 .....	174
4.4.5	Gasterweiterungen unter Windows 7 .....	177
4.4.6	Gemeinsame Ordner .....	178
4.4.7	3-D-Beschleunigung testen .....	180
4.4.8	Netzwerkbrücke .....	182
4.5	Internes Netzwerk .....	185
4.5.1	Virtuelle Maschinen einrichten .....	186
4.5.2	Sicherungspunkte (Snapshots) .....	186
4.5.3	Internes Netzwerk einrichten .....	188
4.5.4	Gastsysteme konfigurieren .....	189
4.5.5	Test .....	191
4.5.6	Sicherungspunkt .....	192
4.6	VirtualBox als Server .....	194
4.6.1	Virtuelle Maschinen starten .....	195

4.6.2	Virtuelle Maschine beenden .....	196
4.6.3	Sitzung aufzeichnen .....	196

**5 Weitere Funktionen und Tipps ..... 199**

5.1	Arbeiten mit dem Manager für virtuelle Medien .....	199
5.1.1	Festplatten .....	200
5.1.2	CD/DVD-Abbilder .....	204
5.1.3	Disketten .....	205
5.2	Arbeiten mit Festplatten-Images .....	206
5.2.1	Festplatten-Image vergrößern .....	206
5.2.2	Festplatten-Image verkleinern .....	208
5.2.3	Kopieren (Klonen) .....	209
5.2.4	Einbinden von fremden Festplatten-Images .....	210
5.2.5	Festplatten-Images konvertieren .....	210
5.3	Arbeiten mit Sicherungspunkten (Snapshots) .....	211
5.3.1	Sicherungspunkte erstellen .....	211
5.3.2	Informationen abrufen .....	213
5.3.3	Sicherungspunkt verwerfen .....	214
5.3.4	Zurückkehren zum letzten Sicherungspunkt .....	214
5.3.5	Aktuellen Sicherungspunkt und Zustand verwerfen .....	215
5.4	Arbeiten mit dem Ruhezustand .....	216
5.4.1	Ruhezustand aktivieren .....	216
5.4.2	Virtuelle Maschine wieder starten .....	217
5.4.3	Status verwerfen .....	219
5.4.4	Status übernehmen .....	220
5.5	Gemeinsame Ordner .....	221
5.5.1	Gemeinsame Ordner hinzufügen .....	221
5.5.2	Zugriff auf einen gemeinsamen Ordner .....	223
5.6	Nahtloser Modus (seamless windows) .....	224
5.7	3-D-Beschleunigung .....	226
5.7.1	OpenGL 2.0 .....	226
5.7.2	Direct3D .....	227
5.8	Mehrere Monitore – Multi-Head .....	229
5.9	SMP – mehrere Prozessoren .....	232
5.10	Serielle Schnittstellen .....	235
5.10.1	Serielle Schnittstelle konfigurieren .....	235
5.10.2	Host-Pipe .....	236
5.11	USB .....	239
5.11.1	USB konfigurieren .....	239

- 5.11.2 USB-Stick einbinden ..... 241
  - 5.11.3 USB-Stick freigeben ..... 243
  - 5.11.4 USB-Filter ..... 243
  - 5.11.5 USB über RDP ..... 246
- 5.12 Port-Forwarding mit setextradata ..... 247
- 5.13 RPD-Authentisierung ..... 249
  - 5.13.1 Extern ..... 249
  - 5.13.2 Gast ..... 251
- 5.14 Virtuelle Maschinen an- und abmelden ..... 251
- 5.15 Virtuelle Maschine löschen ..... 252
- 5.16 Automatische Installation der Gasterweiterungen unter einem Windows-Gastsystem ..... 252
- 5.17 Virtuelle Maschine klonen ..... 254
- 5.18 VirtualBox sichern ..... 254
  - 5.18.1 Festplatten-Image sichern ..... 254
  - 5.18.2 Virtuelle Maschine sichern ..... 255
  - 5.18.3 Komplettsicherung ..... 255
- 5.19 Tastenkombinationen ..... 257

**6 Migration und Import ..... 259**

- 6.1 Planung ..... 260
- 6.2 Open Virtualization Format ..... 261
  - 6.2.1 Aufbau von Virtual Appliances ..... 261
  - 6.2.2 Import ..... 263
  - 6.2.3 Erster Start ..... 264
- 6.3 Migration von VMware-Maschinen ..... 265
- 6.4 Migration eines PCs ..... 266
  - 6.4.1 System vorbereiten ..... 266
  - 6.4.2 Festplatten-Image erstellen ..... 267
  - 6.4.3 Neue virtuelle Maschine ..... 269
  - 6.4.4 Erster Start ..... 269
  - 6.4.5 Alternative zu Partimage ..... 270
- 6.5 Migrationsprobleme ..... 270
  - 6.5.1 Gasterweiterungen des Fremdsystems stören ..... 270
  - 6.5.2 Geräte funktionieren unter Windows-Systemen nicht ..... 271
  - 6.5.3 Volle CPU-Auslastung ..... 272

<b>7</b>	<b>Probleme und Fehlersuche .....</b>	<b>273</b>
7.1	Windows-Hostsysteme .....	273
7.2	Linux-Hostsysteme .....	274
7.3	Allgemeine Probleme .....	276
<b>8</b>	<b>Kommandozeilentools .....</b>	<b>283</b>
8.1	VirtualBox .....	284
8.2	VBoxManage .....	285
8.3	VBoxSDL .....	305
8.4	VboxHeadless .....	308
8.5	rdesktop-vrdp .....	310
<b>9</b>	<b>Glossar .....</b>	<b>313</b>
	Index .....	315



# Vorwort

Zunächst möchte ich kurz einige Worte über mich verlieren. Mein Name ist *Dirk Becker*, allerdings kennen mich viele Bekannte wohl noch besser unter meinem »Mädchennamen« *Dirk Bender*. Mein Wohnsitz befindet sich im Saarland. Dort wohne ich in einem Ort direkt an der Landesgrenze zu Frankreich und nur einen Steinwurf von Luxemburg entfernt. Die letzten Jahre fiel es mir zunehmend schwer, dem Saarland treu zu bleiben, da die EDV-Branche hier im Vergleich zu den anderen Bundesländern weit weniger präsent ist und es demzufolge kaum entsprechende Jobangebote gibt. Dennoch habe ich es mit Unterstützung meiner Frau Sybille geschafft.

Ich interessiere mich schon seit 1983 für Computer (die damals eigentlich noch Homecomputer genannt wurden). Ausgelöst wurde mein lebhaftes Interesse (oder besser: meine »Sucht«) durch eine von meinem damaligen Klassenlehrer geleitete Computer-AG. Diese fand in meinem letzten Schuljahr 1983/84 statt. Der Lehrer führte uns damals am sogenannten »Volkscomputer« VC-20 in die Programmiersprache BASIC ein. Kaum aus der Schule, erwarb ich selbst einen VC-20, um erste eigene (meist absolut unnötige) Programme zu schreiben. Dem Volkscomputer folgte kurze Zeit später der allseits bekannte Commodore 64. Mit diesem begann auch das technische Interesse zu wachsen. Die Bauanleitungen und Basteltipps aus den Zeitschriften »Happy Computer« und der »64er« musste ich einfach testen: angefangen vom Profi-Reset-Taster bis hin zum eigenen EPROM-Programmiergerät. Das Einzige, was ich verpasst habe, war ein Akustikkoppler – und das bereue ich bis heute. Parallel zum C64 besaß ich auch noch einen Sinclair ZX Spectrum 48K, den ich ebenfalls für die Programmierung nutzte. Allerdings ersetzte ich nie die nervraubende Gummitastatur durch eine vernünftige, wodurch das auf ihn bezogene (Programmier-)Interesse schnell wieder verflog.

Den Wechsel vom Homecomputer zum PC vollzog ich im Jahre 1989, als ich an einer SINIX- und COBOL-Schulung bei Siemens teilnahm. SINIX war ein von Siemens entwickeltes Unix-Derivat. Ich erwarb damals einen gebrauchten 286er, den ich zum Lernen und Schreiben der Übungsprogramme nutzte. Parallel besaß ich zu dieser Zeit noch immer meinen Commodore 64. Nach der Schulung verkaufte ich den 286er wieder – und vergaß leider die COBOL-Programmierung.

Aber mein Interesse an Unix war geweckt, vor allem deshalb, weil mir das Betriebssystem MS-DOS auf dem 286er dagegen nahezu lächerlich vorkam.

1992 schnupperte ich dann »Amiga-Luft«. Ich fragte mich häufig, warum so viel über diese IBM-kompatiblen PCs geredet wurde – und vor allem, weshalb sie so teuer waren (mir war mein 286er noch in Erinnerung). Sie waren meiner Meinung nach dem Amiga in fast allen Belangen unterlegen. Dennoch kaufte ich mir, bedingt durch den zweijährigen Besuch einer Technikerschule, einen 80486DX mit einer sagenhaften 100-MByte-Festplatte und Windows 3.1. Erstmals konnte ich, ähnlich wie bei Unix, nun auch auf dem PC Programme im Hintergrund ausführen – und das Ganze sogar mit einer grafischen Oberfläche. Allerdings folgte die Ernüchterung dann spätestens nach zehn oder elf Neuinstallationen, die allesamt aufgrund häufiger Abstürze notwendig geworden waren. Im Anschluss daran folgte die Anschaffung verschiedener, nicht weiter erwähnenswerter Standardrechner, die mit Betriebssystemen ausgestattet waren, deren Stand heute als überholt bezeichnet werden kann.

Ich interessierte mich auch gleich von Anfang an für Linux. Mitte der 90er-Jahre stieß ich erstmals selbst auf dieses freie Unix-Derivat. Seit 2001 habe ich unter <http://www.64-bit.de/> eine Homepage, auf der fast ausschließlich deutsche Linux-Dokumentationen zu finden sind. Der Hintergrund war, dass es zu dieser Zeit nur wenige deutsche Anleitungen gab und diese meist schwer im Internet zu finden waren. Weil ich bei Kunden und bei mir privat immer öfter Linux für File- und Mailserver einsetzte, sammelten sich bei mir naturgemäß einige Dokumentationen an, die ich nun, gut sortiert, der Allgemeinheit zur Verfügung stellen wollte. Inzwischen habe ich auch privat auf fast allen Desktoprechnern Linux installiert.

Soviel zu meinem Weg in die Computerbranche. Aber weshalb schreibe ich ein (zweites) Buch? Ich habe mir vor geraumer Zeit angewöhnt, zu Software oder Problemfällen, die ich neu erlerne, immer eine eigene Dokumentation (meist im Linux-Howto-Format) zu schreiben. Einige dieser Anleitungen sind auf meiner Homepage zu finden und manche auch durchaus weit verbreitet (unter anderem ein Mailserver-Howto über Sendmail). Durch meinen Freund Dennis Zimmer kam ich auf die Idee, aus einer dieser Anleitungen ein Buch zu machen. Das Thema meines ersten Buchs war die freie Software OpenVPN, die es ermöglicht, sichere Netzwerkverbindungen über das Internet herzustellen. Das vorliegende Buch folgte gleich im Anschluss. Da ich VirtualBox seit langer Zeit begeistert einsetze und auch der Verlag der Idee positiv gegenüberstand, lag es nahe, über VirtualBox zu schreiben.

Die Software setze ich seit drei Jahren ein. Sie hat bei mir weitgehend die (ebenefalls kostenlose) Software VMware Server ersetzt und kommt dabei auf meinen

Desktops, zum Testen von Software und auch auf meinem kleinen Privatserver zum Einsatz. Auf dem Server laufen meist 4–5 Systeme parallel, unter anderem ein Desktop, eine Entwicklungsumgebung (auf der ich auch dieses Buch schreibe) und ein Linux-Server. Bis heute habe ich den Umstieg nicht bereut.

Beruflich habe ich jedoch auch Erfahrung mit VMware ESX und vor allem Citrix-XenServer-Systemen. Letztere setze ich bei meinem jetzigen Arbeitgeber auf drei Servern ein. Abgesehen von kleineren Mängeln des XenCenters bin ich sehr zufrieden. Wo es im Kontext dieses Buches zu VirtualBox sinnvoll ist, werde ich auch auf die Möglichkeiten der eben genannten Virtualisierungslösungen verweisen.

*Dirk Becker, Überherrn*



*Bevor es in die Praxis geht, sind einige einführende Worte nötig. Sie lesen hier, an wen sich dieses Buch richtet und erfahren einiges an Hardware-Grundlagenwissen.*

# 1 Einführung

Zunächst möchte ich kurz erläutern, welche Kenntnisse vor der Lektüre nützlich sind und natürlich, welche Kenntnisse Sie nach der Lektüre hinzugewonnen haben werden. Außerdem lernen Sie Einzelkomponenten eines PC kennen, denn dieses Wissen vereinfacht das Verständnis und die Konfiguration der Virtualisierungssoftware.

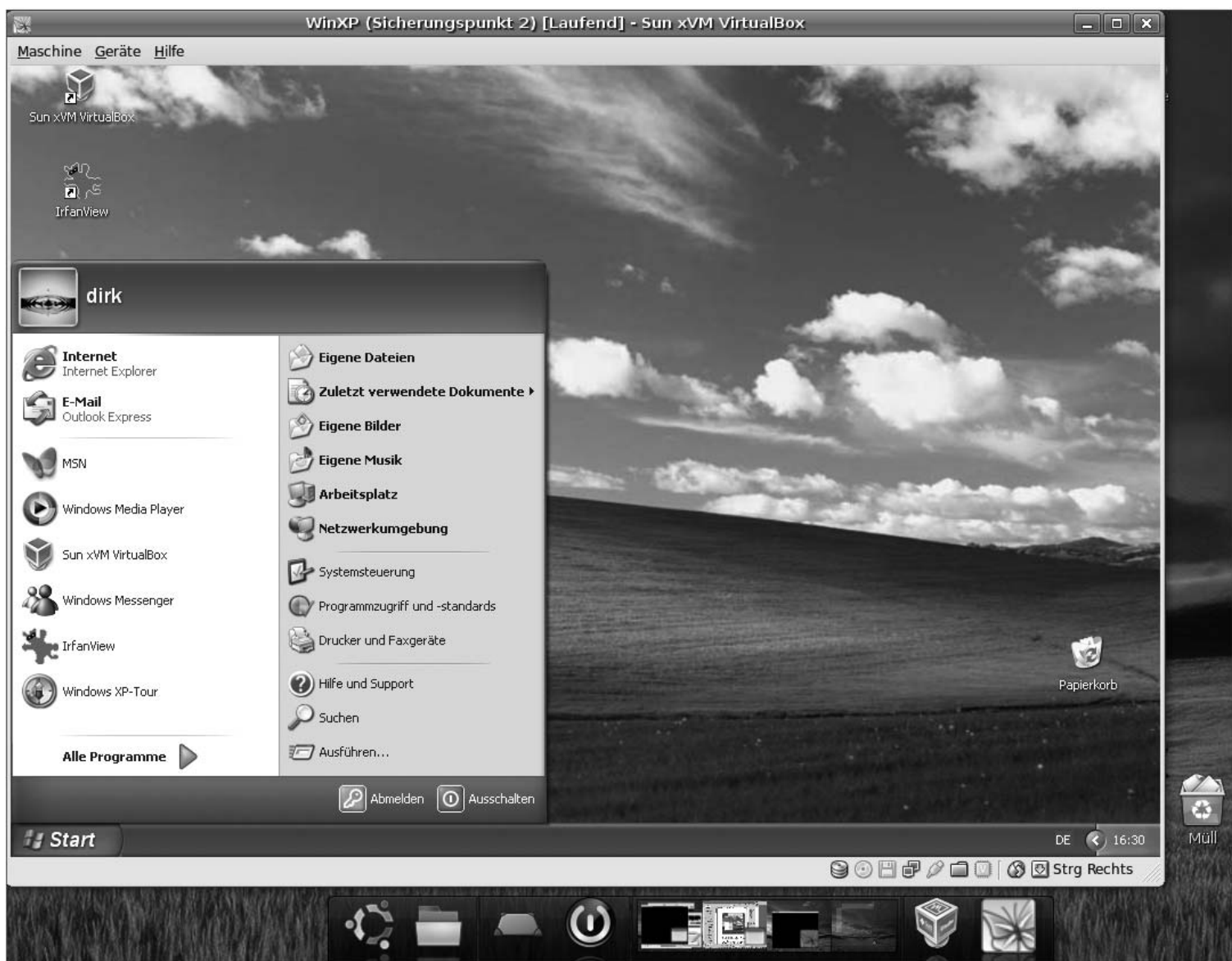
## 1.1 Über dieses Buch

*Für wen ist dieses Buch?* Diese Frage lässt sich – gerade auch mit Verweis auf den Titel des Buches – leicht beantworten: Mein Buch richtet sich an Anwender aller Art, die noch keine oder wenig Erfahrung mit VirtualBox haben. Ich werde Schritt für Schritt die Installation und anhand von Praxisbeispielen die Verwendung von VirtualBox erläutern. Dabei gehe ich auch tiefer auf häufige Probleme und Fehler ein.

### 1.1.1 Ziele und Anforderungen des Buches

Mein Buch ist für Administratoren kleinerer bis mittlerer Unternehmen, Entwickler, die Testsysteme benötigen, und natürlich auch für Privatanwender gedacht, die die Leistung ihrer Systeme ausnutzen oder andere Betriebssysteme testen wollen. Für die Praxisbeispiele sollten Sie möglichst auch eine der aktuellen Windows-Versionen (Windows XP, Windows Vista oder Windows 7) zur Verfügung haben. Am Ende werden Sie sich mit der (Desktop-)Virtualisierung auskennen und VirtualBox schnell und sicher einsetzen. Weiterhin können Sie das Buch später als Nachschlagewerk für die Befehle und Parameter verwenden. Verschiedene, meist kurze Abschnitte ähneln dabei der englischen Dokumentation. Allerdings möchte ich betonen, dass es sich bei diesen Abschnitten nicht lediglich um Übersetzungen der jeweiligen Dokumentation handelt; die betreffenden Passagen wurden vielmehr grundlegend überarbeitet und ergänzt.

Die Leistung von Desktopprozessoren hat sich in den letzten Jahren beträchtlich erhöht. Vor einigen Jahren noch brachten selbst Tabellenkalkulations-Programme eine CPU auf 100 % Auslastung. Bei der Komprimierung eines Videos war an ein Arbeiten mit anderen Programmen nicht mehr zu denken – man musste stundenlang warten. DVDs liefen auf vielen Rechnern nur mit einem zusätzlichen MPEG-Decoder. Hier hat sich jedoch einiges durch immer schnellere Prozessoren geändert. Abgesehen von Rechnern, die speziell für CAD oder 3-D-Spiele genutzt werden, sind heute fast sämtliche PCs nur noch geringfügig ausgelastet, und die CPUs arbeiten über weite Strecken deutlich unterhalb ihrer technischen Leistungsgrenze. Man kann selbst bei der Videobearbeitung noch bequem im Internet surfen. Diese Tatsache hat dafür gesorgt, dass Virtualisierung und Emulatoren nicht nur auf Servern, sondern auch auf Desktops Einzug hielten. Es ist mit modernen PCs kein Problem mehr, eine virtuelle Maschine zu verwenden.



**Abbildung 1.1** VirtualBox mit Windows XP auf einem Linux-Desktop

Ich werde Ihnen die Installation und den Einsatz von VirtualBox auf Windows-XP- und Linux-Systemen erklären. Benutzer anderer Systeme müssen das Buch jedoch nicht weglegen. Der Einsatz von VirtualBox und seiner Kommandozeilen-

tools ist unter allen Systemen gleich. Sie müssen nur an verschiedene Unterschiede zwischen den Betriebssystemen denken (Laufwerksbuchstaben, Pfadtrennzeichen usw.). Lauffähig ist VirtualBox dabei unter anderem auf den folgenden Betriebssystemen:

- ▶ Windows
- ▶ Linux
- ▶ Mac OS X
- ▶ OpenSolaris

Um den Einsatz zu erklären, verwende ich verschiedene Praxisbeispiele, die viele gängige Anwenderszenarien abbilden, mit denen auch Sie immer wieder konfrontiert sein werden. VirtualBox kann dazu über eine grafische Oberfläche oder mit verschiedenen Programmen in der Eingabeaufforderung bzw. Konsole gesteuert werden (Kommandozeilentools). Ich gehe bei der Beschreibung auf alle diese Programme ein. Sie lernen deren Einsatz und Anwendung kennen und finden darüber hinaus auch immer eine detaillierte Beschreibung der einzelnen Funktionen.

Als Leser dieses Buchs sollten Sie, je nachdem wie tief Sie in VirtualBox einsteigen wollen, zumindest über Grundkenntnisse in den Bereichen »Hardware«, »Betriebssysteme« und gegebenenfalls auch »Netzwerke« verfügen. Letzteres ist wichtig, wenn Sie virtuelle Maschinen mit eigenen Netzwerken oder Vollzugriff auf das Netzwerk einrichten wollen. Begriffe wie »IP-Adressen« oder »Netzwerkadapter« sollten in diesem Fall keine Fremdworte für Sie sein.

VirtualBox wurde im Prinzip nur für Desktopvirtualisierung entwickelt. Ein Ersatz für VMware ESX Server oder Xen-Server ist es nicht und wird es vermutlich auch nie werden. Diese bieten Funktionen, die über den Umfang von VirtualBox hinausgehen wie z. B. VMotion. Für größere Unternehmen, die Server virtualisieren wollen, ist es daher nicht die geeignete Lösung, selbst wenn sich VirtualBox grundsätzlich auch zur Servervirtualisierung einsetzen lässt. In Abbildung 1.2 sehen Sie einen Linux-Server, auf dem vier VirtualBox-Maschinen zur Verwendung als virtuelle Desktops gestartet wurden.



**Abbildung 1.2** VirtualBox als »Server«

Leser, die VirtualBox zunächst kennen lernen und ihre bestehende Virtualisierungssoftware weiter verwenden wollen, können VirtualBox übrigens parallel dazu installieren. In Abbildung 1.3 sehen Sie zwei virtuelle Maschinen: im Hintergrund eine mit Windows XP als Gastsystem unter VirtualBox und im Vordergrund eine mit Linux unter VMware Server. Allerdings warnt Sun vor der parallelen Verwendung von VirtualBox mit anderer Virtualisierungssoftware. Bei mir sind bisher jedoch keine Probleme aufgetreten.



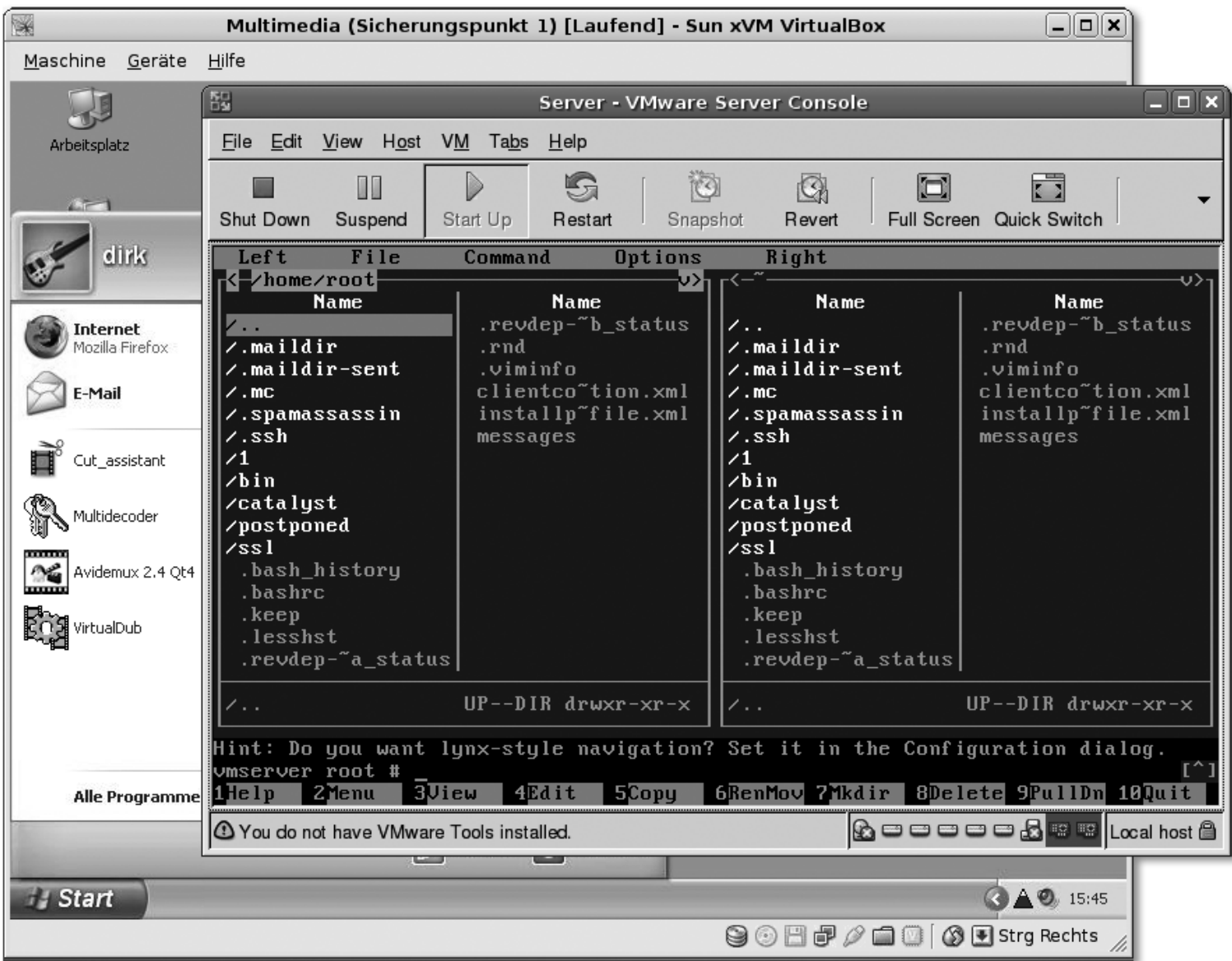


Abbildung 1.3 VirtualBox (im Hintergrund) und VMware Server parallel

### 1.1.2 Hinweise zum Umgang mit dem Buch

Im Folgenden finden Sie einige Bemerkungen, die Ihnen sicherlich helfen werden, sich noch einfacher und schneller in diesem Buch zurechtfinden. Insbesondere die Anmerkungen zu Besonderheiten der Schreibweise bei Befehlen oder Listings in diesem Abschnitt sollten Sie beachten.

#### Hinweise

Besonders wichtige Punkte schreibe ich in sogenannte Hinweiskästen. Diese erkennen Sie am Icon **[+]** am Rand des Kastens, zum Beispiel:

Hinweis

Die Informationen in den Hinweiskästen sollten Sie auf jeden Fall genau durchlesen.

**[+]**

#### Was Sie sich merken sollten

Neben den Hinweiskästen gibt es auch noch am Ende verschiedener Abschnitte Kästen, in denen sich eine Zusammenfassung der wesentlichen Inhalte in Stichworten befindet. Diese Kästen tragen am Rand das Icon **[!]**, zum Beispiel:

**Was Sie sich merken sollten**

- ▶ Diese Kästen beinhalten eine Zusammenfassung eines Abschnitts.
- ▶ Sie befinden sich am Ende eines Abschnitts.

**Backslash**

Gelegentlich verwende ich am Zeilenende einen umgekehrten Schrägstrich (auch »Backslash« genannt). Dabei handelt es sich dann nicht um einen Zeilenwechsel, sondern um eine einzige Eingabezeile!

**Backslash zur Verbesserung der Lesbarkeit**

Der Befehl

```
yast -i \
    VirtualBox-2.1.2_41885_sles10.1-1.i586.rpm
```

ist der Eingabezeile wie folgt einzugeben:

```
yast -i VirtualBox-2.1.2_41885_sles10.1-1.i586.rpm
```

Programmierern und Linux-Anwendern dürfte dies bereits bekannt sein. Diese Schreibweise werde ich verwenden, um Befehle oder Listings übersichtlicher zu gestalten.

**Glossar**

Am Ende des Buchs befindet sich ein Glossar, in dem ich verschiedene Begriffe kurz und bündig beschreibe. Bevor Sie also bei Verständnisproblemen im Internet »googlen«, lohnt sich ein Blick in diesen Abschnitt.

Wenn sich keine Informationen zu einem Begriff im Glossar befinden, sollten Sie zunächst unter Wikipedia nachschauen (<http://de.wikipedia.org/>).

## 1.2 Personal Computer

Bevor wir zum Thema »Virtualisierung« kommen, sollten Sie zunächst noch die Funktionsweise von Personal Computern (PCs) kennen lernen. Dadurch wird die Virtualisierung auch besser verständlich. Ohne Kenntnisse über die Hardware und die verschiedenen Zusammenhänge ist es häufig schwer, Unterschiede zwischen den Einstellungen zu verstehen oder die optimale Konfiguration für eine virtuelle Maschine zu finden. Leser, die sich damit bereits auskennen, können dieses Kapitel natürlich getrost überspringen. Wer jedoch noch nicht weiß, was Intel VT oder AMD-V bedeuten, sollte sich diesen Abschnitt durchlesen.

### 1.2.1 Entwicklung

Die ersten Computer in (größeren) Firmen waren sogenannte Großrechner. An den Arbeitsplätzen standen hierbei keine eigenständigen Computer, sondern nur sogenannte Terminals, die mit dem Großrechner verbunden waren. Diese Terminals bestanden aus einem Monitor und einer Tastatur. Die eigentlichen Rechenaufgaben wurden von dem Großrechner ausgeführt. Die Kosten eines solchen Großrechners waren so hoch, dass es für kleine bis mittlere Unternehmen meistens nicht in Frage kam, sich einen solchen anzuschaffen.

#### Homecomputer

Ende der 70er-Jahre verbreiteten sich immer mehr kleinere Computer mit eigenen Prozessoren (sogenannte Mikrocomputer). In den privaten Haushalten (und meinem Jugendzimmer) dominierten die Heimcomputer (auch Homecomputer genannt). Verschiedene Modelle kamen natürlich auch in Büros zum Einsatz. Diese waren kompakte Geräte mit eigenem Prozessor, Speicher, Tastatur usw. Der Anschluss erfolgte meist an einen Fernseher. Bekannte Heimcomputer waren unter anderem:

- ▶ Commodore VC-20 (»Volkscomputer«)
- ▶ Commodore 64
- ▶ Texas Instruments TI-99 4/A
- ▶ Sinclair ZX-81 (auch als Bausatz erhältlich)
- ▶ Sinclair ZX Spectrum



**Abbildung 1.4** VC-20 (auch VIC 20) (Quelle: <http://commons.wikimedia.org/>)



**Abbildung 1.5** ZX Spectrum (Quelle: <http://commons.wikimedia.org/>)

Ein Nachteil dieser Heimcomputer war, dass sie nicht kompatibel zueinander waren. Es konnten keine Programme und meist auch keine Daten untereinander ausgetauscht werden. Auch die meisten Peripheriegeräte, wie zum Beispiel Drucker, funktionierten nur mit den vorgesehenen Computern.

## Personal Computer

In den Büros verbreiteten sich in dieser Zeit Rechner der Firma Apple. Der Apple II war einer der ersten industriell hergestellten Personal Computer. Anfang der 80er-Jahre wollte die Firma IBM diesen Markt jedoch erobern und entwarf ein Konkurrenzprodukt. Das Ergebnis war der erste IBM Personal Computer: der IBM 5150 Personal Computer. In diesem PC kam bereits ein Prozessor der Firma Intel zum Einsatz, die sich bis heute als weltweit größter Prozessorhersteller etabliert hat.



**Abbildung 1.6** IBM 5150 (Quelle: <http://commons.wikimedia.org/>)

Die Firma Microsoft hat ebenfalls von diesem PC profitiert. Sein Betriebssystem war IBM DOS. Dieses wurde von Microsoft entwickelt und bildete den Grundstein für den Erfolg dieser Firma.

Inspiziert durch den Erfolg des IBM-PCs bauten viele Hersteller diesen Rechner nach und verkauften ihre Modelle als IBM-kompatible PCs. Hierdurch entwickelte sich der PC zum inoffiziellen Standard in der Industrie und in den Büros.

### 1.2.2 Komponenten eines PCs

Das Prinzip des Aufbaus hat sich dabei bis heute fast nicht geändert. Ein Personal Computer besteht unter anderem aus den folgenden Komponenten:

- ▶ Hauptplatine (auch Mainboard)
- ▶ BIOS
- ▶ Prozessor (auch CPU)
- ▶ Hauptspeicher
- ▶ Grafikkarte
- ▶ Eingabegeräte (Tastatur, Maus)
- ▶ Datenspeicher (Diskettenlaufwerke, Festplatten)
- ▶ zusätzliche Peripheriegeräte (Drucker, Scanner usw.)

Einige dieser Komponenten waren in den ersten IBM 5150 Personal Computern noch nicht vorhanden (unter anderem die Festplatte). Das Betriebssystem wurde seinerzeit noch über Disketten gestartet.

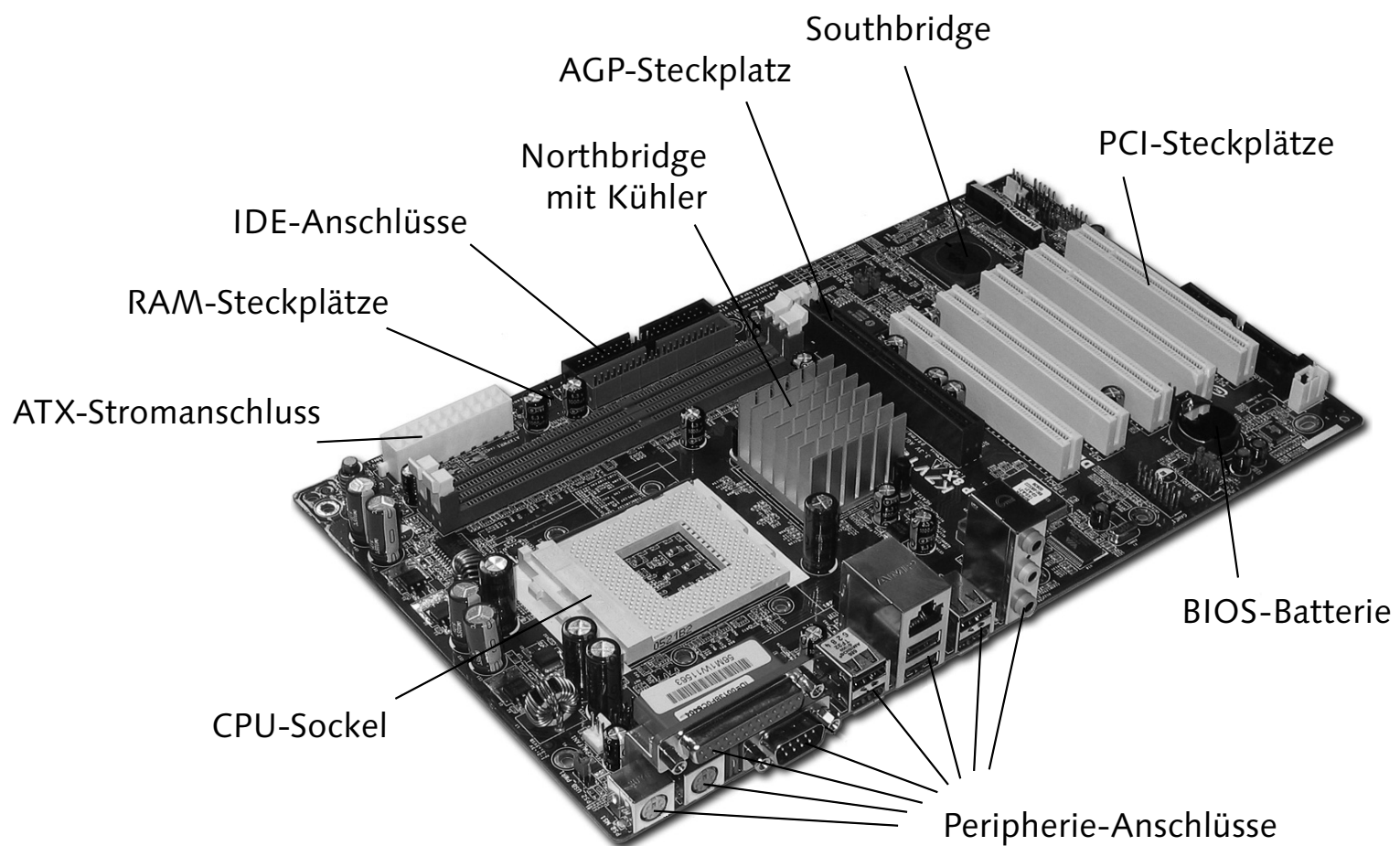
#### Hauptplatine (Mainboard)

Wie ihr Name schon nahelegt, ist die Hauptplatine der wichtigste Baustein eines Rechners. Sie ist eine zentrale Platine, die das Zusammenarbeiten der einzelnen Komponenten ermöglicht.

In Abbildung 1.7 sehen Sie ein (nicht mehr ganz aktuelles) Mainboard. Darauf erkennen Sie unter anderem verschiedene Steckplätze für den Prozessor, die Speicherbausteine und Erweiterungskarten. Weiterhin bietet das Mainboard Steckanschlüsse für zusätzliche Peripheriegeräte.

#### CPU-Sockel

Auf dem CPU-Sockel (Central Processing Unit) wird der Prozessor angebracht. Früher war dies eine knifflige Aufgabe, heute gibt es spezielle Sockel, bei denen die CPUs mit Hebeln befestigt bzw. gelöst werden. Das Wechseln ist dann eine Leichtigkeit.



**Abbildung 1.7** Mainboard (Quelle: <http://commons.wikimedia.org/>)

### RAM-Steckplätze

Die Speicherbausteine befinden sich auf sogenannten *Speicherriegeln*, die in die RAM-Steckplätze (*Random Access Unit*) gesteckt werden. Hier kann man eigentlich nicht viel falsch machen. Die Speicherriegel besitzen eine Richtungskerbe, um zu verhindern, dass sie falsch herum eingebaut werden.

### IDE- (und SATA-)Anschlüsse

IDE-Anschlüsse (*Integrated Device Electronics*) dienen dazu, Festplatten, CD/DVD-Laufwerke und andere Massenspeichergeräte anzuschließen. Hierbei können maximal vier Laufwerke (jeweils zwei am primären und zwei am sekundären Controller) angeschlossen werden. Der Nachfolger von IDE ist die SATA-Schnittstelle (*Serial Advanced Technology Attachment*). An diese können mehr Laufwerke angeschlossen werden und sie ist auch bedeutend flexibler als ihr Vorgänger.

VirtualBox unterstützte anfangs nur IDE-Anschlüsse. Einer der Anschlüsse war hierbei (und ist heute noch) bereits fest für das CD/DVD-Laufwerk vorgesehen. Es konnten also noch maximal drei Festplatten (in Form von Images) an die virtuelle Maschine angeschlossen werden. Erst ab Version 1.6 wurde auch ein SATA-Controller eingeführt und dadurch die Möglichkeit geschaffen, eine größere Anzahl Laufwerke zu verwenden.

Seit Version 2.1 gibt es auch bei VirtualBox die Möglichkeit, *SCSI-Controller* (*Small Computer System Interface*) einzubinden. Diese finden eigentlich haupt-

sächlich Verwendung im Serverbereich. Allerdings existieren viele virtuelle Maschinen von Fremdherstellern, die virtuelle SCSI-Controller eingebunden haben. Da VirtualBox diese nun ebenfalls anbietet, können fremde Maschinen einfacher migriert werden.

### PCI-Steckplätze

Um neue Hardware bzw. Erweiterungskarten einzubauen, dienen die PCI-Steckplätze (*Peripheral Component Interconnect*). Über diese können Sie Ihren PC zum Beispiel mit TV-Karten oder RAID-Controllern erweitern. Der Nachfolger von PCI ist PCI-Express (auch PCIe, *Peripheral Component Interconnect Express*).

In VirtualBox, wie auch in allen mir bekannten Virtualisierern, können Sie jedoch einer virtuellen Maschine diese Karten nicht zuordnen bzw. daraus auf diese zugreifen.

### AGP-Port

Der AGP-Port (*Accelerated Graphics Port*) wurde speziell für Grafikkarten entwickelt. Auch er wurde inzwischen vom PCIe-Steckplatz ersetzt.

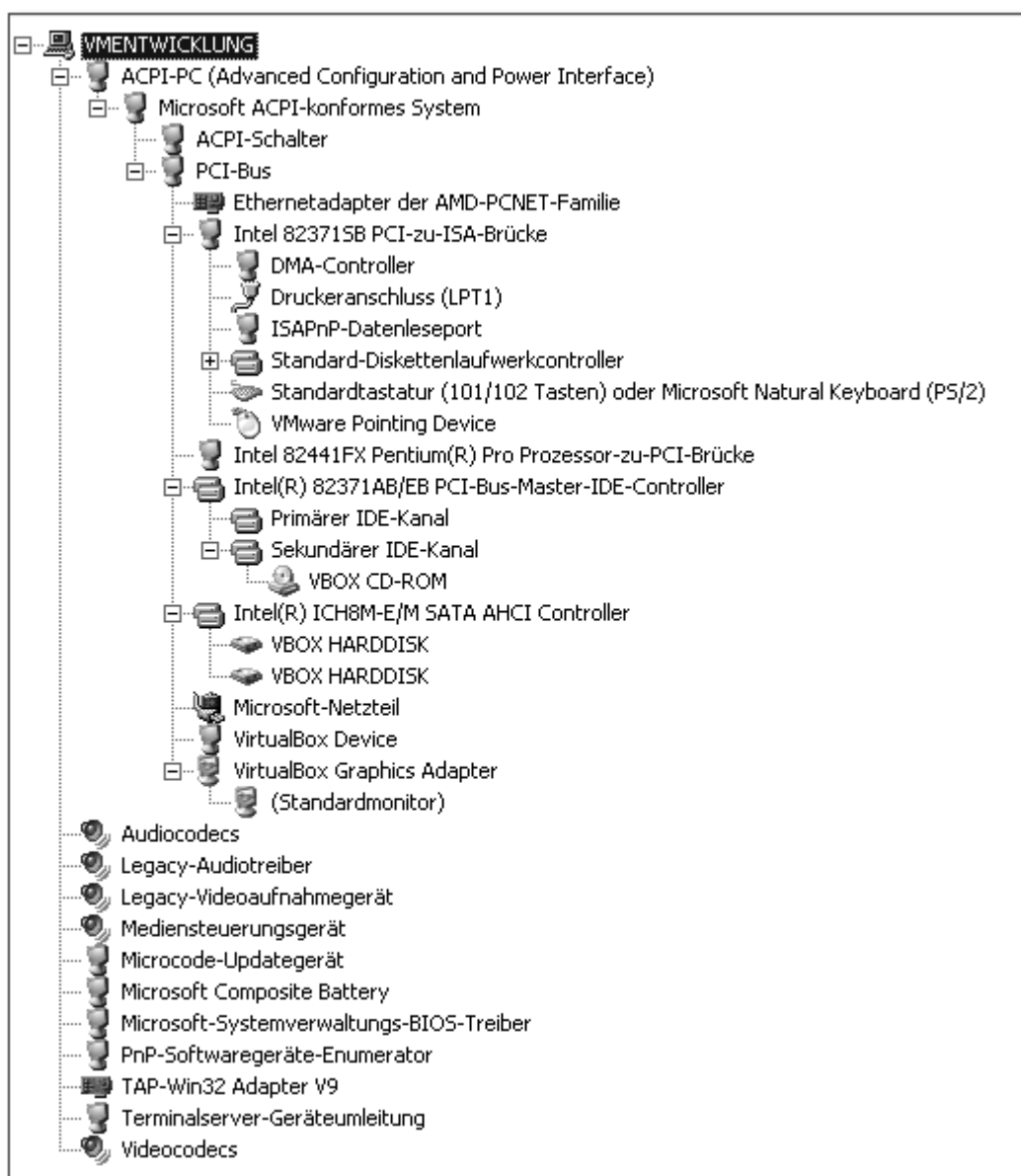
VirtualBox verwendet übrigens »nur« den PCI-Steckplatz für die Grafikkarte.

### Peripherie-Anschlüsse

Über diese Ports können weitere externe Peripheriegeräte angeschlossen werden. Zu diesen Geräte zählen unter anderem Drucker, Joysticks, Tastaturen, Mäuse usw. Zu den älteren Peripherie-Anschlüssen zählen die seriellen und parallelen Schnittstellen. Diese werden inzwischen immer mehr von USB (*Universal Serial Bus*) verdrängt.

VirtualBox ist eine der ersten Virtualisierungslösungen, die auch USB-Geräte »durchreicht« und so grundsätzlich den Zugriff auf diese Geräte ermöglicht. Verschiedene Konkurrenzprodukte können das bis heute noch nicht.

VirtualBox emuliert viele Komponenten des Mainboards. In Abbildung 1.8 sehen Sie die Informationen des Windows-Gerätemanagers in einer virtuellen Maschine. In dieser Maschine schreibe ich auch gerade an diesem Buch. Sie erkennen in der Abbildung unter anderem einen PCI-Bus-Master-IDE-Controller, den SATA AHCI Controller und den VirtualBox Graphics Adapter. Diese befinden sich alle am PCI-Bus.



**Abbildung 1.8** Der Windows-Gerätemanager in einer virtuellen Maschine

## BIOS (Basic Input Output System)

Unmittelbar nach dem Einschalten eines PCs kommt das *Basic Input Output System* – kurz BIOS – zum Einsatz. Dieses ist eine Art »Minibetriebssystem«, das den Rechner initialisiert und den Start des eigentlichen Betriebssystems ermöglicht. Um zu gewährleisten, dass es auch nach dem Ausschalten oder Löschen der Festplatte noch erhalten bleibt, wird es auf einen Chip (siehe Abbildung 1.9) installiert bzw. »gebrannt«.



**Abbildung 1.9** Ein BIOS-Chip (Quelle: <http://commons.wikimedia.org/>)



Auf dem ursprünglichen IBM-PC war nur ein einfaches BIOS im Einsatz. Bei den heutigen Computern hat sich das BIOS inzwischen zu einem umfangreichen System entwickelt. Beim Starten werden dabei unter anderem folgende Punkte abgearbeitet:

- ▶ kurzer Systemtest (Power On Self Test, POST)
- ▶ Passwortabfrage (auf Wunsch)
- ▶ Initialisierung der Hardware (unter anderem der Festplatten, Tastatur, Erweiterungskarten usw.)
- ▶ Starten des Betriebssystems

Die meisten BIOS-Versionen bieten auch umfangreiche Konfigurationsmöglichkeiten an. Sie können hierbei zum Beispiel bestimmte Geräte oder Controller (wie zum Beispiel USB, Parallel-Port) deaktivieren, das Verhalten der Speicherbausteine einstellen oder die Uhrzeit setzen.

Häufig bieten Hersteller auch die Möglichkeit, das BIOS zu erneuern. Hintergrund sind meist Fehlerbehebungen oder die korrekte Erkennung neuer Hardware. Beim Aktualisieren müssen Sie jedoch besonders vorsichtig sein. Wenn der Aktualisierungsvorgang durch einen Stromausfall oder Ähnliches unterbrochen wird, kann dies dazu führen, dass der PC nicht mehr startet.

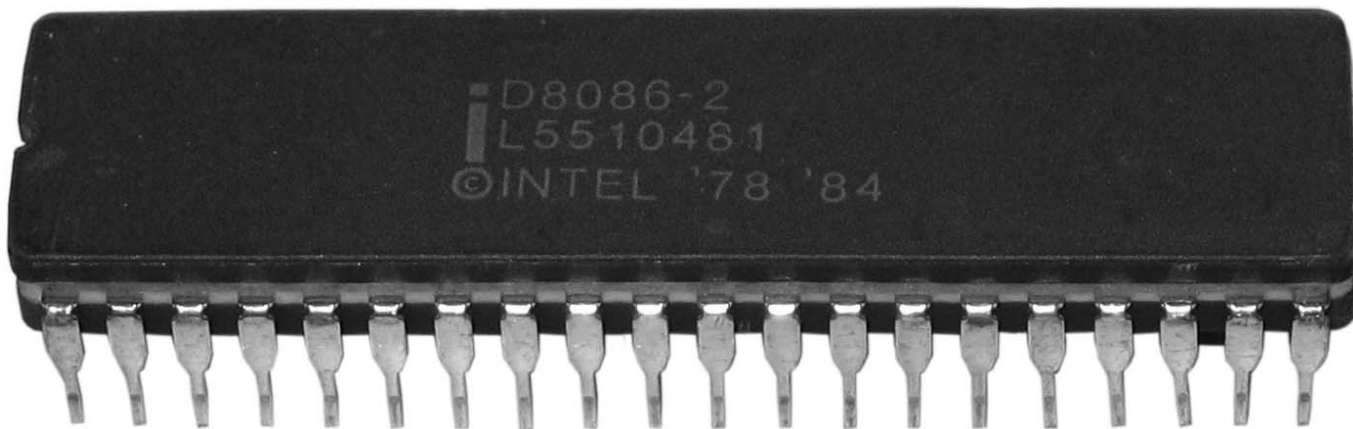
VMware bietet für seine virtuellen Maschinen auch ein simples BIOS an. In VirtualBox ist hingegen keins enthalten. Hier werden die Einstellungen in der grafischen Oberfläche vorgenommen.

## Prozessor

Der Prozessor ist das »Gehirn« eines Computers. Die korrekte Bezeichnung lautet eigentlich Hauptprozessor oder CPU (*Central Processing Unit*). Allerdings hat sich inzwischen umgangssprachlich die Bezeichnung Prozessor durchgesetzt.

Der Prozessor wird über einen Steckplatz (auch Sockel) mit der Hauptplatine verbunden. Obwohl er »Gehirn« genannt wird, kann man ihn dennoch nicht als intelligent bezeichnen – im Gegenteil, er ist relativ »dumm«. Seine Fähigkeiten bestehen darin, dass er mehrere Millionen Instruktionen pro Sekunde durchführen kann. Diese Befehle werden in MIPS (*Million Instructions per Second*) angegeben. Ein Prozessor der ersten Generation (Intel 8080) schaffte zum Beispiel circa 0,6 MIPS (also etwa 600.000 Rechenoperationen pro Sekunde). Heutige Prozessoren schaffen mehrere 1.000 MIPS. Allerdings kann die Einheit MIPS nicht unbedingt dazu verwendet werden, um *unterschiedliche* Prozessoren miteinander zu vergleichen. Dies liegt daran, dass diese auch unterschiedliche Befehlssätze besitzen und

daher bestimmte Instruktionen unter Umständen mit verschiedenen Geschwindigkeiten abarbeiten.



**Abbildung 1.10** 8086-Prozessor (Quelle: <http://commons.wikimedia.org/>)

Ein Prozessor besteht heutzutage aus mehreren Millionen elektronischen Transistoren. Diese bilden hierbei mehrere Einheiten. Dazu zählen unter anderem:

- ▶ **Register**  
Im Register werden die Daten für die Verarbeitung abgelegt.
- ▶ **Adresseinheit**  
Die Adresseinheit ist für den Datenaustausch mit dem Speicher zuständig. Sie liest und schreibt die notwendigen Daten und Befehle.
- ▶ **Befehlsdecoder**  
Der Befehlsdecoder ist, wie der Name schon sagt, dafür zuständig, die Befehle zu verarbeiten.
- ▶ **Recheneinheit**  
Die Recheneinheit wird auch ALU (*Arithmetic Logic Unit*) genannt. In ihr werden die Berechnungen auf binärer Basis vorgenommen.
- ▶ **Cache**  
Der Hauptspeicher ist bedeutend langsamer als der Prozessor. Da Letzterer ständig mit dem Speicher kommuniziert und dabei viel Zeit verloren geht, besitzen fast alle modernen CPUs einen Cache. Dieser ist bedeutend schneller (aber auch bedeutend kleiner) als der Hauptspeicher. In ihm werden Daten aus dem Hauptspeicher zwischengespeichert.

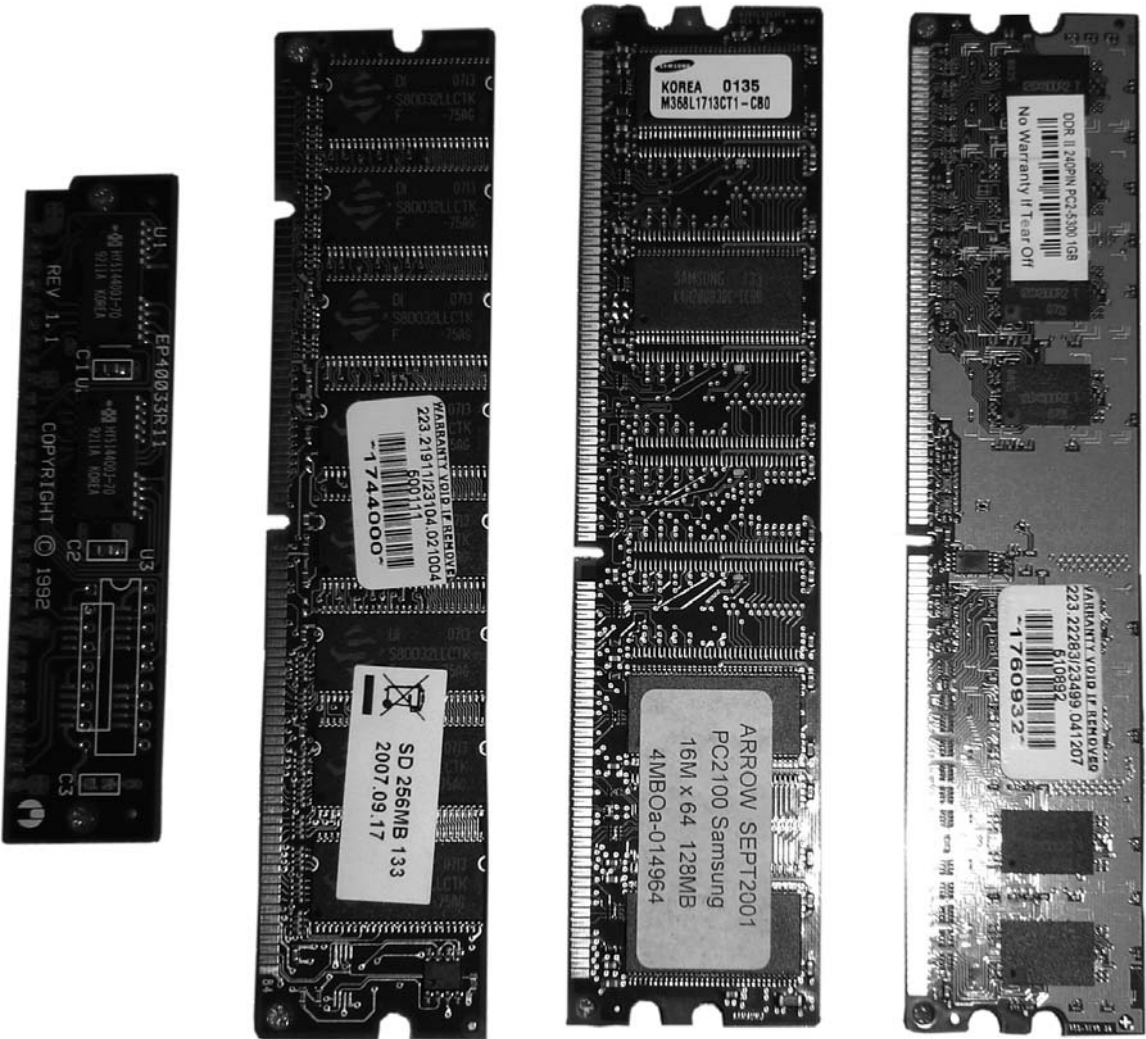
Bei einem heutigen Prozessor befinden sich all diese Einheiten auf einem einzigen Chip. Viele Hersteller erweitern ihre CPUs inzwischen auch mit zusätzlichen Funktionen und Befehlssätzen. Zwei wichtige Funktionen für die Virtualisierung sind hierbei:

- ▶ Intel VT (*Intel Virtualization Technology*, Codename »Vanderpool«)
- ▶ AMD-V (*AMD Virtualization*, Codename »Pacifica«)

Diese beiden Funktionen sind spezielle Technologien, welche die virtuellen Maschinen auf einem Rechner beschleunigen und spezielle Arten der Virtualisierung überhaupt erst ermöglichen. Dies wird dadurch erreicht, dass bestimmte Code-Teile, die bisher durch Software emuliert wurden, nun direkt über die Hardware (bzw. vom Prozessor) ausgeführt werden. Weiterhin erkennt ein Prozessor, der diese Technologie unterstützt, eine virtuelle Maschine und kann so dabei helfen, diese unter anderem durch Abgrenzung des Arbeitsspeichers sicherer zu machen. Verschiedene Hypervisor (Xen) sind sogar auf diese Funktionen angewiesen, um virtuelle Maschinen mit proprietären Betriebssystemen wie zum Beispiel Microsoft Windows laufen zu lassen. Auch VirtualBox unterstützt und benötigt diese Funktion in verschiedenen Fällen. SMP (*Symmetrisches Multiprozessorsystem*) kann nur auf Hostsystemen verwendet werden, die Intel VT oder AMD-V unterstützen. Und einige wenige proprietäre Systeme wie zum Beispiel OS/2 benötigen sie ebenfalls – allerdings laufen Windows-Systeme ohne diese.

**Hauptspeicher**

Die Programme, die der Prozessor bearbeiten soll, werden im Hauptspeicher (auch Arbeitsspeicher genannt) abgelegt. Dieser Speicher ist ein flüchtiger Speicher. Das bedeutet, dass er im Gegensatz zu Festplatten seine Daten beim Ausschalten des Rechners verliert. Allerdings kann auf seine Daten bedeutend schneller zugegriffen werden als auf die einer Festplatte.



**Abbildung 1.11** Verschiedene Speicherriegel (Quelle: <http://commons.wikimedia.org/>)

Die Speichergrößen sind in den letzten Jahren nahezu explodiert, aber leider benötigen auch die meisten Betriebssysteme bedeutend mehr Arbeitsspeicher als noch vor ein paar Jahren. Gängige Größen sind zurzeit 1024-MByte- und 2048-MByte-Bausteine, die meist paarweise eingebaut werden und dadurch 2 oder 4 GByte Hauptspeicher ergeben. Für die Virtualisierung können Sie gar nicht genug Speicher einbauen, wobei 32-Bit-Systeme normalerweise auf 4 GByte beschränkt sind. Jede virtuelle Maschine beansprucht in etwa so viel Speicher für sich, wie Sie ihr zuweisen. Und das Betriebssystem, auf dem VirtualBox läuft, benötigt ebenfalls noch seinen eigenen Speicher. Wenn Sie allerdings nur gelegentlich eine virtuelle Maschine starten wollen, reichen je nach Betriebssystem 1–2 GByte Hauptspeicher normalerweise aus.

### **Datenspeicher**

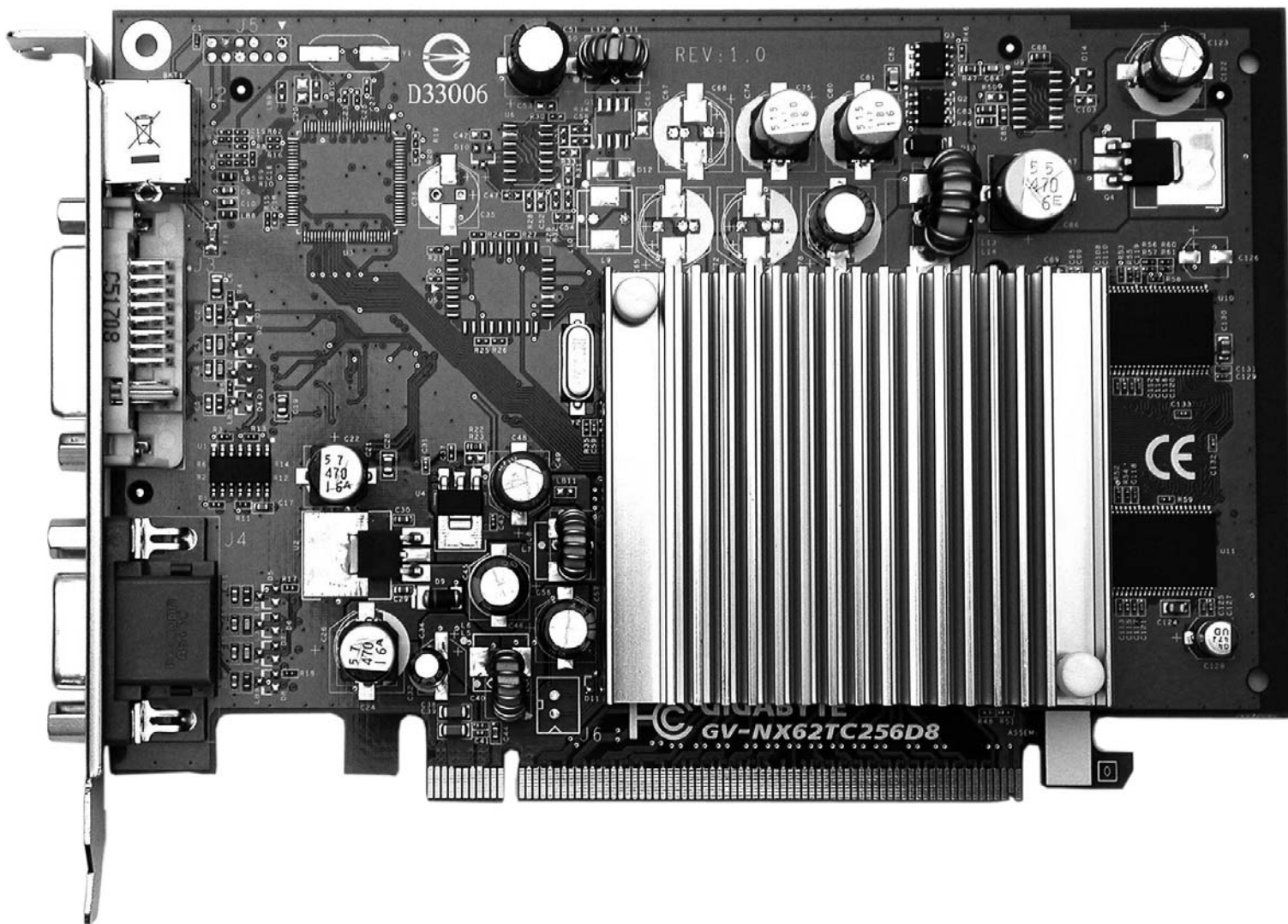
Auf einem Datenspeicher werden Informationen, ähnlich wie beim Hauptspeicher, abgelegt. Allerdings sollen diese Speichermedien ihre Daten auch beim Ausschalten des PCs noch erhalten.

Die Heimcomputer wurden meist noch ohne Datenspeicher ausgeliefert. Mein erster VC-20 lief gelegentlich mehrere Tage, um das mühsam eingetippte Programm nicht zu verlieren. Nach kurzer Zeit investierte ich daher sofort in meinen ersten Datenspeicher – die Datasette. Diese ermöglichte es, Programme und Daten auf handelsübliche Musikkassetten (Magnetbänder) zu speichern. Viele andere Modelle boten ebenfalls die Speicherung auf Magnetbändern an. Ein großer Nachteil dieser Speichermethode war das mühselige Suchen nach Programmen, wenn mehrere hintereinander auf einem Band abgelegt wurden.

Die Nachfolger der Kassetten waren die Disketten (Magnetscheiben). Diese ermöglichten das Laden der Daten, ohne zu spulen. Der erste IBM-PC verwendete auch 5¼-Zoll-Diskettenlaufwerke. 3½-Zoll-Disketten werden auch heute gelegentlich noch eingesetzt.

### **Grafikkarte**

Eine Grafikkarte ist dafür zuständig, die Informationen auf einem Bildschirm anzuzeigen. Erste Modelle waren dabei nicht wirklich grafikfähig und konnten nur Zeichen bzw. Buchstaben darstellen. Auch Farben waren nicht möglich, sondern nur eine einfarbige (monochrome) Anzeige. Der IBM 5150 hatte zum Beispiel eine MDA-Karte (*Monochrome Display Adapter*), die nur den Text in einer Farbe darstellen konnte. Mit heutigen Grafikkarten gehört das Abspielen von Videos und DVDs ohne Ruckeln zum Standard. Höherwertige Karten ermöglichen sogar Video- und Grafikbearbeitungen und den Einsatz aufwändiger 3-D-Software.



**Abbildung 1.12** Eine PCIe-Karte (Quelle: <http://commons.wikimedia.org/>)

### Netzwerkkarte

Fast alle aktuellen Personal Computer haben inzwischen eine Netzwerkkarte (bzw. einen Netzwerkadapter) »on Board«. Das bedeutet, dass diese bereits fest auf der Hauptplatine integriert sind. Eine Netzwerkkarte dient dazu, mit anderen Computern oder Teilnehmern Daten auszutauschen. Die gängigsten Methoden sind hierbei Kabel- oder Funknetzwerke.

Das Thema »Datenübertragung« ist ein komplexes und weitreichendes Thema, über das es zahlreiche Bücher gibt. Für Leser, welche die Netzwerkfunktionen von VirtualBox anwenden wollen, setze ich in diesem Buch voraus, dass sie bereits wissen, wie man ein einfaches Netzwerk einrichtet (IP-Adressen vergeben, Freigaben einrichten usw.).

VirtualBox bietet übrigens die Möglichkeit, einer virtuellen Maschine bis zu acht virtuelle Netzwerkadapter zur Verfügung zu stellen.

### Eingabegeräte

Eingabegeräte steuern einen PC und ermöglichen es, Eingaben zu tätigen. Die gängigsten Eingabegeräte sind Tastaturen und Mäuse, aber auch Joysticks oder

Scanner gehören dazu. In der Anfangszeit der Computer hat man auch Lochstreifen zur Eingabe verwendet.

### **(Externe) Peripheriegeräte**

Die bisher aufgeführten Geräte gehören fast alle zur Kategorie der »internen Peripheriegeräte«. Ein Peripheriegerät ist eine Hardware, die sich außerhalb der Zentraleinheit befindet und dem PC weitere Funktionen bereitstellt. Meist meint man mit dem allgemeinen Begriff »Peripheriegerät« jedoch Hardware, die sich nicht nur nicht in der Zentraleinheit befindet, sondern auch außerhalb des Computers. Dies sind streng genommen jedoch externe Peripheriegeräte.

Die Peripheriegeräte erweitern den PC ähnlich wie die Erweiterungskarten um weitere Funktionen. Zu den gängigsten zählen:

- ▶ Monitore
- ▶ Drucker
- ▶ Scanner
- ▶ Tastaturen und Mäuse
- ▶ Modems
- ▶ Joysticks
- ▶ Kartenleser
- ▶ USB-Sticks und -Festplatten

In VirtualBox ist es möglich, bestimmte Peripheriegeräte »durchzureichen« (über USB oder die serielle Schnittstelle). Dadurch kann der virtuellen Maschine der Zugriff auch auf vorhandene Hardware ermöglicht werden.

*Was bedeutet »Virtualisierung«? Welche Vorteile bietet sie Ihnen und welche Arten gibt es? Wo liegen die Stärken der verschiedenen Lösungen und wo die Schwächen? In diesem Kapitel finden Sie die Antworten und erhalten außerdem Tipps zur richtigen Konfiguration Ihres PC für den Einsatz von VirtualBox.*

## 2 Virtualisierung

Wenn ich den Begriff »Virtualisierung« höre, denke ich immer gleich an virtuelle Maschinen. Dies ist eigentlich nicht ganz korrekt. Virtualisierung ist in der Informatik ein viel weitreichenderes Thema.

### 2.1 Begriff und Technologie

Es gibt keine klare Definition des Begriffs. Selbst in der Enzyklopädie Wikipedia lautet der erste Satz:

*»In der Informatik ist die eindeutige Definition des Begriffs Virtualisierung nicht möglich, da der Begriff in vielen unterschiedlichen Anwendungsfällen anders ausgeprägt ist.«*

Ich werde dennoch versuchen, Ihnen in den nächsten Abschnitten das Thema Virtualisierung detaillierter zu erläutern.

#### 2.1.1 Warum virtualisieren?

Zunächst stellt sich die Frage, warum man überhaupt virtualisieren soll? Es gibt hierzu verschiedene Gründe und Ansätze. Bei mir hat die Virtualisierung mit VMware Workstation 1.0 im Jahr 2000 begonnen. Zu dieser Zeit war ich Programmierer, und mein Desktop-PC war nie voll ausgelastet. Zur schnellen Wiederherstellung meines Systems verwendete ich diverse Image-Programme. Dennoch dauerte es immer eine gewisse Zeit, bis ein abgestürztes, defektes System wieder lauffähig gemacht werden konnte. Als ich von VMware Workstation las, überzeugte ich meinen damaligen Arbeitgeber gleich davon, dieses auch für meine Entwicklungs- und Testumgebungen einzusetzen. Anfangs sicherte ich die virtuellen Maschinen in ZIP-Archiven und konnte dadurch schnell eine defekte

Maschine wiederherstellen und neutrale Testumgebungen bereitstellen. Den Snapshot-Funktionen traute ich zu dieser Zeit noch nicht so richtig. Aber nach einigen Einsätzen bei Softwaretests oder Ähnlichem lernte ich auch diese schnell zu schätzen. Auch das Zurücksetzen meiner Entwicklungsumgebung innerhalb von Sekunden überzeugte mich von dem Einsatz der Snapshots.

Beim Arbeiten traf ich aber auch gleich auf die ersten Nachteile der virtuellen Maschinen: Lizenzprobleme. Eines unserer Entwicklungstools sendete Broadcast-Anfragen durch das Netzwerk. Wenn ich nun einem Entwickler eine Kopie meiner Entwicklungsumgebung bereitstellen wollte, ging dies nicht so einfach, da das Tool behauptete, wir verwendeten Raubkopien und deshalb seinen Dienst versagte. Dies war jedoch nicht korrekt – wir hatten nur in jeder virtuellen Maschine die gleiche Seriennummer. Die Seriennummer konnte übrigens nicht geändert werden, sondern das Tool musste entfernt und wieder neu installiert werden. Auch ein weiteres Tool, das einen Dongle an der seriellen Schnittstelle benötigte, konnte überhaupt nicht verwendet werden – obwohl VMware Workstation dieses Dongle grundsätzlich richtig emulierte.

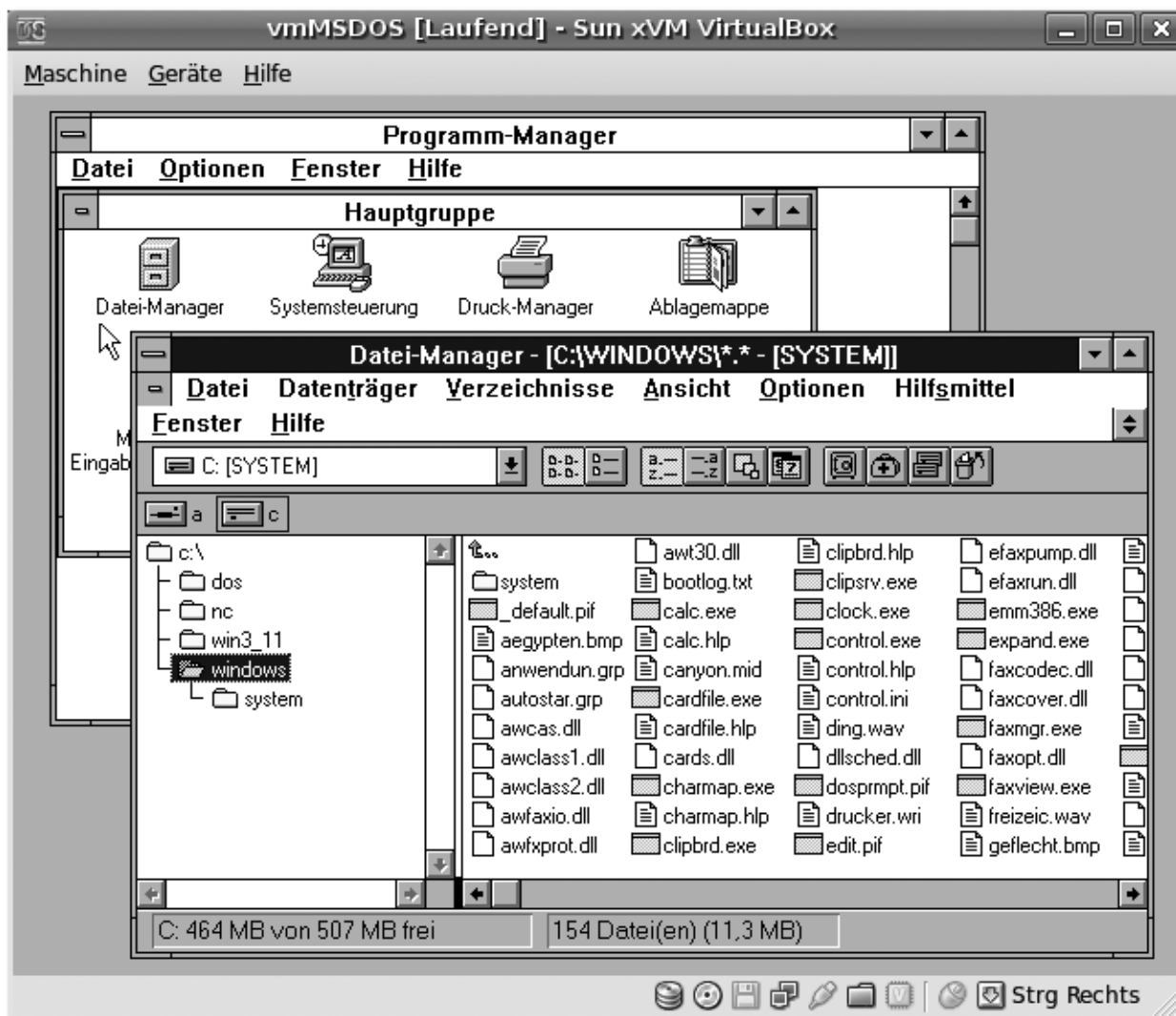
Im Privatbereich oder kleineren Unternehmen kann man mit der Virtualisierung viel Zeit und Arbeit sparen. Ein Beispiel ist die virtuelle Arbeitsplatzumgebung. Man erstellt eine virtuelle Maschine und installiert und konfiguriert dort alle notwendigen Anwendungen. Auf dem Arbeitsplatzrechner kann eine (kostenlose) Linux-Distribution als Hostsystem installiert werden. Gearbeitet wird dann ständig in der virtuellen Maschine. Wenn es nun wegen Systemausfällen, Fehlkonfigurationen oder Viren zu Problemen kommt, kann diese auf einfache Art wiederhergestellt werden. Dazu können dann Snapshots oder simple Kopien der virtuellen Maschine verwendet werden. Diese Art der Umgebungen könnte auch nur zum Surfen im Internet verwendet werden. Es ist möglich, virtuelle Maschinen so einzurichten, dass beim Beenden alle Änderungen verworfen werden. Dadurch hätten eventuelle Viren oder Trojaner keine Chance. Ein Einsatz dieser Maschinen ist auch im Schulungsbereich sinnvoll. Die (meist »scherzhaften«) Änderungen von Schülern werden nach dem Beenden der virtuellen Maschine wieder verworfen, und der Lehrer braucht sich nicht mehr um defekte Systeme zu kümmern.

Ein weiteres Einsatzgebiet für die Virtualisierung sind Serversysteme. Bis vor ein paar Jahren war es noch üblich, für verschiedene Dienste bzw. Services jeweils einen physikalischen Server bereitzustellen. Diese Server waren meist nicht voll ausgelastet, da es oft mehr Ruhe- bzw. Leerlaufphasen gab als Spitzenzeiten. Kleinere Unternehmen hatten auch mehrere Dienste auf einem Server laufen. Der Nachteil daran war, dass ein Neustart oder Systemabsturz alle Dienste betraf. Diese Methoden haben sich in den letzten Jahren grundlegend geändert. Inzwischen ist man dazu übergegangen, Dienste in virtuelle Maschinen auszulagern und diese, meist parallel zu anderen, auf Hostsystemen laufen zu lassen.



Im privaten Bereich gibt es noch weitere Einsatzgebiete. Häufig will man neue Software testen, ohne das Risiko einzugehen, das laufende System zu beeinträchtigen. Dies ist in einer virtuellen Maschine ohne Probleme möglich. Ich habe hierzu verschiedene Systeme (Windows und Linux) eingerichtet, die ich bei Bedarf verwende. Vor der Installation eines Programms erstelle ich von dem entsprechenden System noch einen Snapshot. Dadurch kann ich nach erfolgreichem Test das System wieder auf den Urzustand zurückstellen.

Auch »alte« Software, die zum Beispiel nur unter MS-DOS läuft, kann mit Hilfe einer virtuellen Maschine »reanimiert« werden. MS-DOS (oder FreeDOS) wird zwar von mancher Virtualisierungssoftware offiziell nicht mehr unterstützt, kann jedoch fast immer ohne Probleme installiert werden. (VirtualBox unterstützt es übrigens immer noch.) Ein gutes Beispiel hierzu ist Windows 3.x. Es läuft in einer VirtualBox-Maschine stabil. Die Einschränkungen sind jedoch, dass manche Hardware nicht unterstützt wird.



**Abbildung 2.1** Windows 3.11 in VirtualBox

### Virtualisierung im Privatbereich

- neue Programme testen, ohne das Originalsystem zu verändern
- sicher im Internet surfen
- alte Software oder Spiele in einer virtualisierten Umgebung laufen lassen

### Virtualisierung in Unternehmen

- ▶ Hardwareressourcen effektiver nutzen
- ▶ mehrere Serverdienste parallel auf einem Rechner anbieten
- ▶ Stabilität und Ausfallsicherheit verbessern

#### 2.1.2 Pro und Kontra

Die Virtualisierung hat viele Vor- und Nachteile. Bei den im vorherigen Abschnitt genannten Einsatzbeispielen habe ich hauptsächlich die Vorteile angesprochen:

### Vorteile der Virtualisierung

- ▶ effizientere Auslastung der Hardware
- ▶ eine eigene Umgebung für jeden Service (zum Beispiel Mailserver, SQL-Server usw.) möglich
- ▶ Kosten-, Energie- und Platzeinsparungen durch Reduzierung der physikalischen Hardware
- ▶ einfaches Verwerfen von Änderungen (Stichwort »Snapshots«)
- ▶ schnelles Bereitstellen neuer Maschinen (Stichwort »Cloning«)
- ▶ schnelles Wiederherstellen »defekter« Systeme
- ▶ nicht änderbare Maschinen (für Schulungen oder Internet)

Aber wie immer ist nicht alles Gold was glänzt. Zu den Vorteilen kommen auch einige Nachteile bzw. Probleme. Allerdings kann ich Sie gleich vorab beruhigen: Diesen Nachteile können Sie meistens »aus dem Weg gehen«, und einige betreffen Sie bei der Desktopvirtualisierung überhaupt nicht, da sie sich auf die Servervirtualisierung beziehen.

### Nachteile der Virtualisierung

- ▶ Bei Ausfall des Hostsystems fallen *alle* darauf laufenden virtuellen Maschinen aus.
- ▶ Die Uhrzeit von virtuellen Maschinen ist nicht mit der des Hostsystems synchron. (Dieses Manko wird meist durch »Gasterweiterungen« behoben).
- ▶ Bei mehreren virtuellen Maschinen kann die hohe Auslastung einer einzelnen die anderen ausbremsen.
- ▶ Eventuell treten Lizenzprobleme beim Klonen auf.
- ▶ Software, die spezielle Hardware erfordert, kann meist nicht verwendet werden (TV- oder ISDN-Karten usw.).

### 2.1.3 Emulation versus Virtualisierung

Neben der Virtualisierung hört man auch ständig den Begriff »Emulation«. Was jedoch ist Emulation? Und vor allem: Was ist der Unterschied zwischen den beiden?

Die Emulation ist der Virtualisierung sehr ähnlich – wobei man eigentlich sagen müsste, dass Letztere der Emulation sehr ähnlich ist. Die Emulation kann man eigentlich in einem Satz beschreiben: »*Ein Emulator bildet ein anderes System nach.*«

Im Gegensatz zur Virtualisierung wird dabei meist das komplette System nachgebildet bzw. emuliert. In der Informatik kann man zwei Arten von Emulatoren unterscheiden:

- ▶ Hardwareemulator
- ▶ Softwareemulator

Ein *Hardwareemulator* ist kein Programm, sondern ein physikalisches Gerät, das die gleichen Schnittstellen und Eigenschaften wie das Original hat. Der Hardwareemulator soll dabei einem anderen System oder Gerät den Eindruck vermitteln, es würde mit dem Original verbunden sein bzw. mit diesem arbeiten. Der Emulator muss jedoch nicht unbedingt die gleichen Funktionen erfüllen. Eine Steckkarte, die zum Beispiel über LEDs bestimmte Funktionen anzeigt, die das Original ausführen würde, wäre also auch ein Hardwareemulator.

Ein Programm, das ein komplettes System oder eine Umgebung nachbildet, ist ein *Softwareemulator*. Mit diesem kann man zum Beispiel Anwendungen, die für das emulierte System entwickelt wurden, auf einem komplett fremden System mit absolut inkompatibler Hardware verwenden. Ein Einsatzgebiet für Softwareemulatoren ist die Anwendungsentwicklung für PDAs oder Handys. Für diese gibt es für die Entwickler spezielle Emulatoren, damit sie die Software testen können, ohne ständig die echten Geräte zu verwenden.

VirtualBox ist also grob gesehen ebenfalls ein Emulator, da es einen PC und dessen Hardware nachbildet. Der Unterschied zwischen einem Emulator und VirtualBox ist jedoch, dass Letzteres nur einen Teilbereich emuliert. Dazu gehören unter anderem der Chipsatz und die Netzwerkkarte. Die wichtigsten Komponenten, die nicht emuliert werden, sind der Prozessor und der Hauptspeicher. Das bedeutet, dass unter VirtualBox kein Betriebssystem, das für einen Motorola-Prozessor geschrieben wurde, installiert und keine Commodore-64-Anwendung gestartet werden kann. Das System und die Anwendung müssen für den gleichen Prozessortyp entwickelt worden sein. Dies ist der (Haupt-)Unterschied zwischen Emulation und Virtualisierung.

### 2.1.4 Emulatoren

Meine ersten Erfahrungen mit Emulatoren machte ich mit einem Commodore-64-Emulator unter MS-DOS. Dieser ermöglichte es, Software, die für den C64 entwickelt wurde, auch auf einem PC zu verwenden. Dabei handelte es sich natürlich meist um Spiele. Über einen Adapter konnte ich auch mein altes 1541-Laufwerk anschließen und die Disketten direkt einlesen. Viele Leser werden diese Art der Emulatoren kennen und vermutlich auch gelegentlich einsetzen. Sehr bekannt im Privatbereich sind unter anderem Emulatoren für:

- ▶ Arcade-Spiele (M.A.M.E., *Multiple Arcade Machine Emulator*)
- ▶ Commodore 64
- ▶ ZX Spectrum
- ▶ Nintendo
- ▶ Super Nintendo
- ▶ Gameboy
- ▶ Sony Playstation

#### VICE

Der Emulator VICE (*Versatile Commodore Emulator*) ist ein Softwareemulator, der verschiedene Homecomputer der Firma Commodore und dazugehörige Hardware nachbildet. VICE ist unter Windows, Linux und vielen weiteren Betriebssystemen lauffähig. Es gibt nur wenig Software, die in diesem Emulator nicht läuft. Meist handelt es sich bei den nicht lauffähigen Programmen um (sehr) systemnahe Software. In Abbildung 2.2 sehen Sie den Startbildschirm eines Commodore 64 in VICE.

VICE emuliert fast die komplette Commodore-Hardware. Dazu gehören unter anderem:

- ▶ VC-20
- ▶ C64
- ▶ C128
- ▶ Plus/4
- ▶ Floppy 1541
- ▶ Datasette 1530

Bei einem Zugriff auf eine Diskette wird der Software nur »vorgegaukelt«, dass ein 1541-Floppylaufwerk angeschlossen ist. Auch die Grafikkarte (bzw. der Grafikchip) und die Soundkarte werden emuliert.

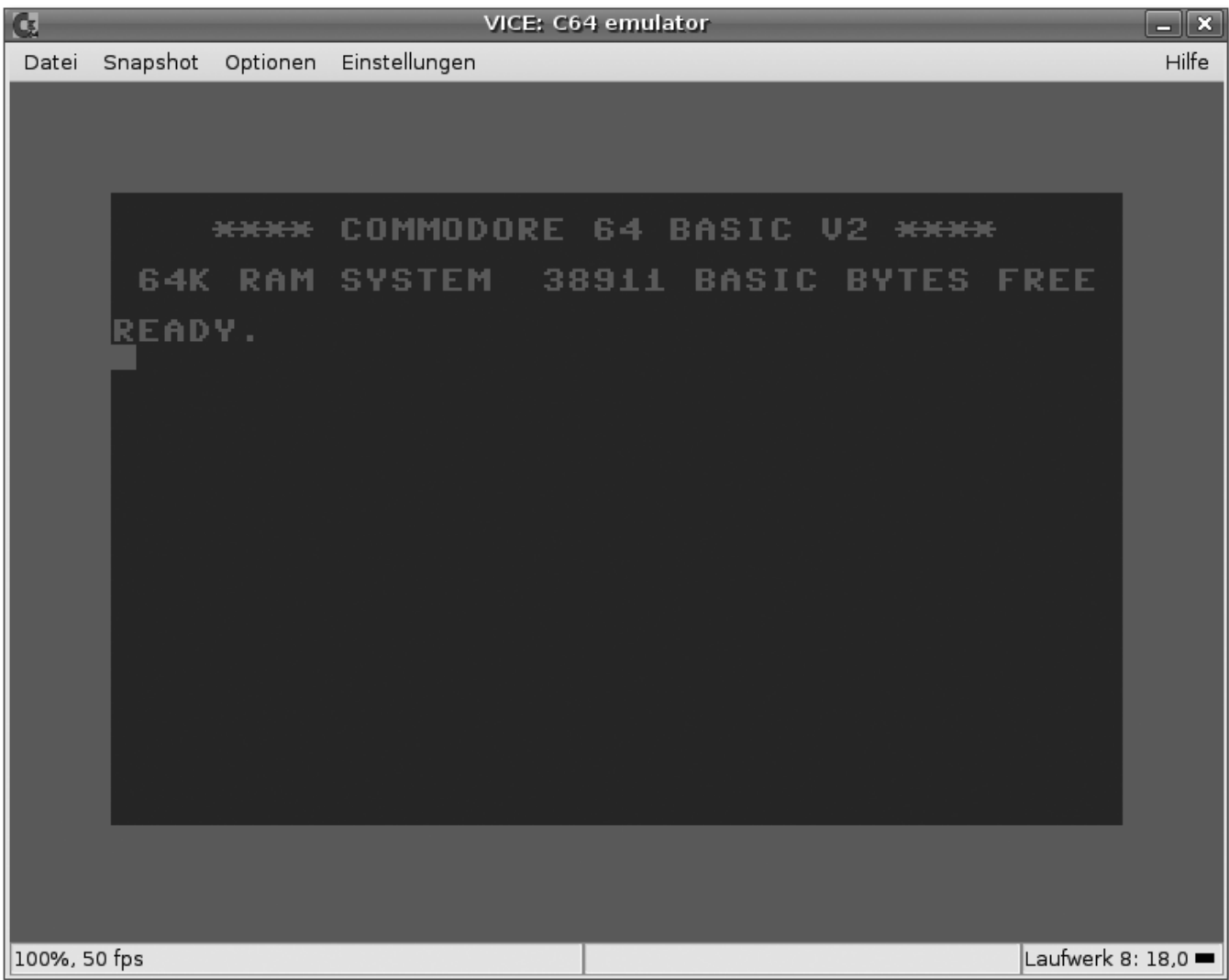


Abbildung 2.2 VICE unter Linux

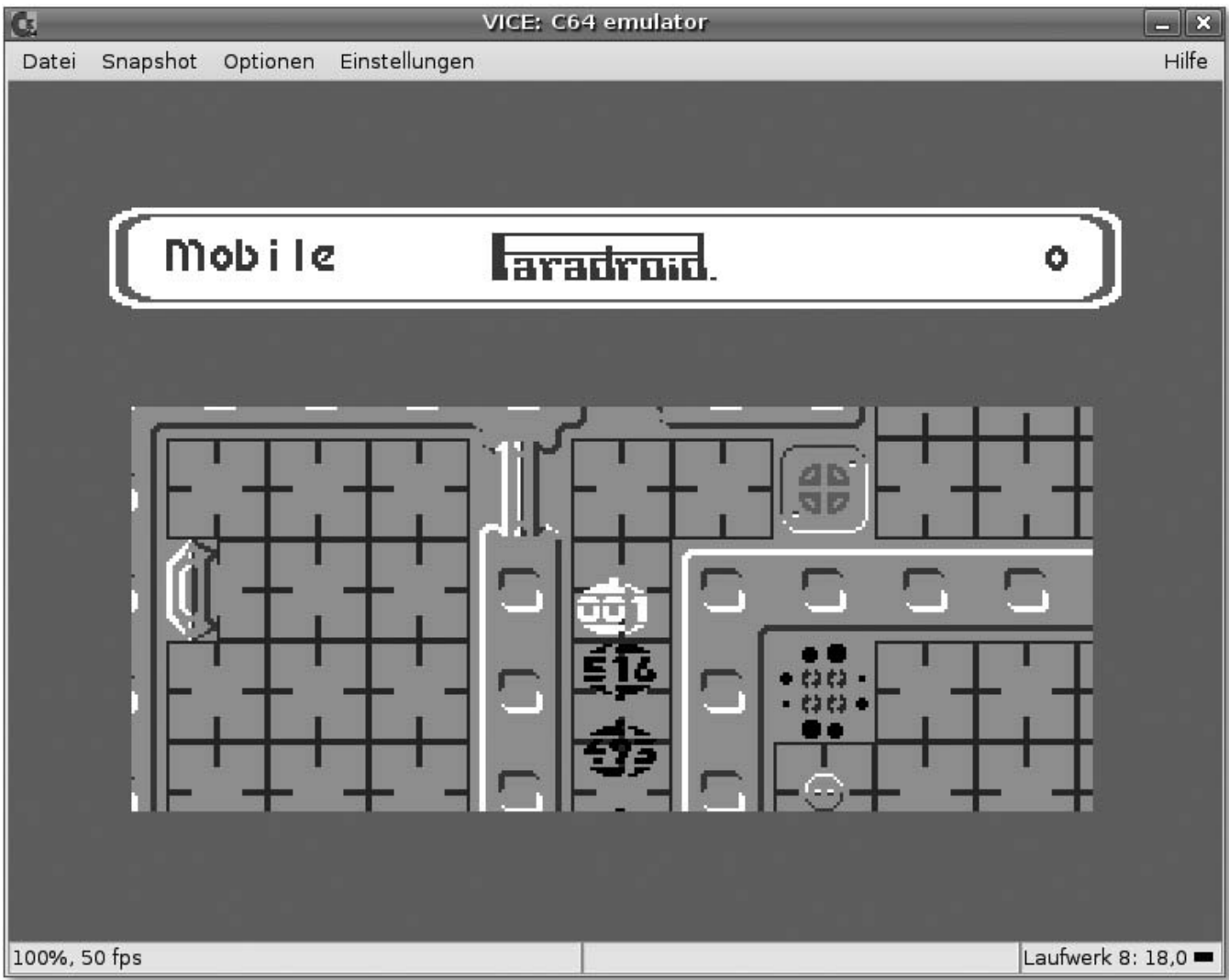


Abbildung 2.3 Paradroid unter VICE

In Abbildung 2.3 habe ich das bekannte Spiel Paradroid gestartet. Gerade Spiele reizten die Hardware des C64 komplett aus und verlangen daher auch vom Emulator einiges. Es gab sogar Spiele und Programme, die zusätzlich den Prozessor des Floppylaufwerks verwendeten – auch diese laufen unter VICE.

Diese Virtualisierung funktionierte bereits unter einem Intel 80486 mit 33 MHz Taktfrequenz ohne Geschwindigkeitseinbußen. Das liegt daran, dass die Homecomputer im Vergleich zu diesem Prozessor sehr langsam waren. Der C64 hatte einen Prozessor mit ca. 1 MHz der Firma MOS, die von Commodore aufgekauft wurde. Auch wenn die Taktfrequenz allein nicht ausschlaggebend ist, so kann man sich dennoch vorstellen, dass der Unterschied zwischen 1 MHz und 33 MHz enorm ist. Auf Prozessoren der heutigen Generation kann man ohne Einbußen mehrere VICE-Emulationen parallel laufen lassen.

## Android

Android ist ein (teilweise) freies Handy-Betriebssystem, das von Google, T-Mobile, Motorola und anderen Anbietern entwickelt wurde. Es basiert auf einem Linux-System. Zum Entwickeln der Software wird vorwiegend die Programmiersprache Java eingesetzt. In Abbildung 2.4 sehen Sie den Emulator der Android-Plattform.



Abbildung 2.4 Googles Android-Emulator

Auf diesem Emulator kann ein Programmierer seine Software testen, ohne dass er ein Android-Handy einsetzen muss. Der Emulator ist natürlich auch für Unentschlossene interessant, die sich die Funktionen der Plattform gerne einmal anschauen möchten.

## Bochs

Häufig wird Bochs mit Virtualisierungssoftware gleichgesetzt. Im Gegensatz zu VirtualBox emuliert Bochs jedoch die gesamte Hardwareumgebung – auch den Prozessor. Daher ist man damit in der Lage, auch virtuelle Maschinen zum Beispiel mit Windows als Betriebssystem unter »nicht-x86-CPU« zu nutzen.

Bochs kann inzwischen unter anderem in den folgenden Umgebungen eingesetzt werden:

- ▶ Apple Macintosh
- ▶ Amiga
- ▶ PlayStation Portable
- ▶ GamePark GP2X

Allerdings ist Bochs, gerade durch die Emulation der kompletten Hardware, im Vergleich zu VirtualBox oder VMware Workstation sehr langsam. Aufgrund der Einsatzmöglichkeiten von Bochs darf man jedoch keinen direkten Vergleich ziehen.

### 2.1.5 Virtualisierung

Nachdem ich Ihnen nun den Unterschied zwischen Emulation und Virtualisierung anhand von Beispielen erläutert habe, kommen wir zum Thema »Virtualisierung«.

In Abbildung 2.5 sehen Sie eine herkömmliche PC-Umgebung. Auf der Ebene Hardware läuft ein Betriebssystem, wie zum Beispiel MS Windows oder eine Linux-Distribution. Dieses System vermittelt zwischen den laufenden Anwendungen und der Hardware. Im Normalfall haben Anwendungen keinen direkten Zugriff mehr auf die Hardware. Dadurch laufen Systeme, im Gegensatz zu früheren Ansätzen wie zum Beispiel MS-DOS bedeutend stabiler. Wenn eine der Anwendungen aufgrund einer Fehlfunktion abstürzt, sind die anderen Programme meistens nicht betroffen und können weiter arbeiten. Wenn allerdings das Betriebssystem abstürzt (oder »einfriert«), sind alle Anwendungen davon betroffen und sämtliche nicht gespeicherten Daten sind in den meisten Fällen verloren.

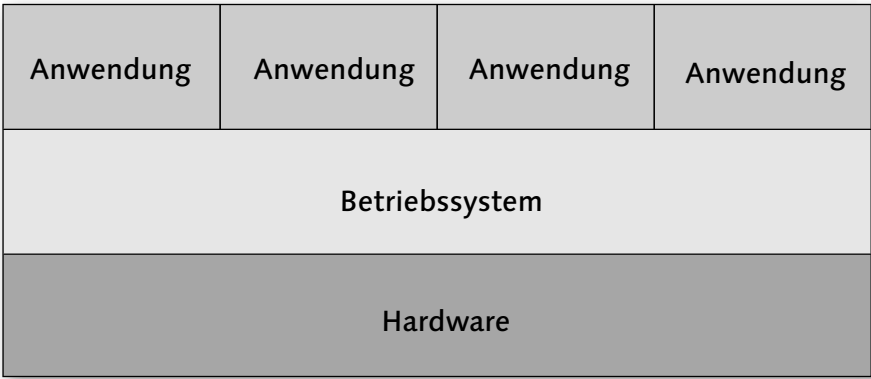


Abbildung 2.5    Herkömmliche PC-Umgebung

**Virtuelle Maschinen**

In der folgenden Abbildung sehen Sie nun mehrere virtuelle Maschinen, die parallel auf einem System laufen. In einer dieser virtuellen Maschinen können Sie sich nun die in Abbildung 2.6 gezeigte PC-Umgebung vorstellen.

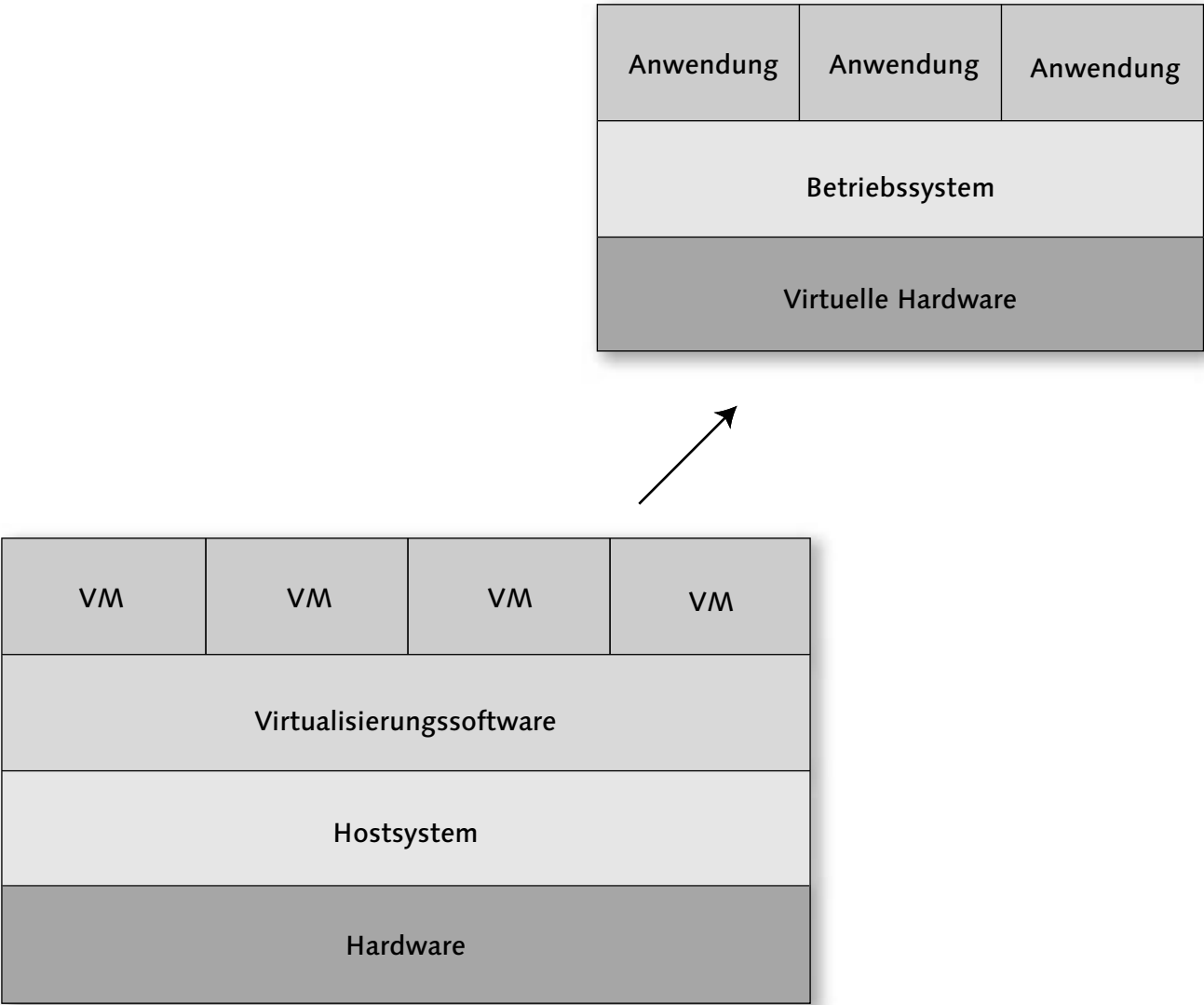


Abbildung 2.6    Virtuelle Maschinen

Verschiedene Vorteile der Virtualisierung habe ich Ihnen bereits in Abschnitt 2.1.1, »Warum virtualisieren?«, erläutert. Hier können Sie nun erkennen, dass der Absturz des Betriebssystems in einer virtuellen Maschine die anderen Maschinen nicht betrifft. Wenn Sie sich nun vorstellen, dass es sich um virtualisierte



Serversysteme handelt, auf denen unterschiedliche Dienste laufen (wie zum Beispiel Datenbankserver, Groupwareserver usw.), können Sie schnell die weiteren Vor- und Nachteile nachvollziehen:

- ▶ Die Ressourcen der Hardware werden effizienter genutzt.
- ▶ Der Absturz einer virtuellen Maschine hat keinen Einfluss auf die anderen.
- ▶ Der Absturz der Virtualisierungssoftware hat den Ausfall aller virtuellen Maschinen zur Folge.

Die Virtualisierungssoftware, welche die virtuellen Maschinen verwaltet, nennt man *Virtual Machine Monitor* (VMM). Sie stellt für die Maschinen virtuelle Laufzeitumgebungen zur Verfügung und sorgt dafür, dass die Ressourcen der physikalischen Hardware korrekt verteilt werden. Häufig werden VMMs auch *Hypervisor* genannt. Meist sind damit dann jedoch immer die »großen« Brüder von VirtualBox, wie zum Beispiel VMware ESX oder Xen, gemeint. Es gibt zwei Arten von Virtual Machine Monitors:

- ▶ Typ1: Der VMM läuft direkt auf der Hardware.
- ▶ Typ2: Der VMM benötigt ein Betriebssystem, auf dem er installiert wird.

VirtualBox ist ein Virtual Machine Monitor vom Typ 2, da es ja nur unter Windows, Linux oder einem der anderen Betriebssysteme läuft. Das System, auf dem die Virtualisierungssoftware installiert wird, ist das *Hostsystem*. Ein Betriebssystem, das in einer virtuellen Maschine läuft, ist das sogenannte *Gastsystem*.

Seit geraumer Zeit wird die Virtualisierung auch durch die neuen Funktionen Intel VT und AMD-V unterstützt. Diese Funktionen sind *Befehlssatzerweiterungen*, die in viele aktuelle Desktop- und vor allem Serverprozessoren integriert wurden. Durch sie erhalten die Betriebssysteme (Gastsysteme) in den virtuellen Maschinen vollen Zugriff auf die CPUs, ohne dass sie sich gegenseitig stören. Ohne diese Erweiterung muss der Virtual Machine Monitor diese Zugriffe selbst überwachen, was meist zu Einbußen in der Funktionalität und Geschwindigkeit führt.

#### Wichtige Begriffe rund um die Virtualisierung

- ▶ Die Virtualisierungssoftware ist ein Virtual Machine Monitor (VMM) und wird auch häufig Hypervisor genannt.
- ▶ Das System, auf dem die Virtualisierungssoftware läuft, ist das Hostsystem.
- ▶ Eine virtuelle Maschine ist eine virtuelle Laufzeitumgebung.
- ▶ Das System, das in einer virtuellen Maschine läuft, ist das Gastsystem.

[!]

## Weitere Arten der Virtualisierung

Neben dem Virtual Machine Monitor gibt es noch weitere Arten der Virtualisierung. Zu diesen zählen unter anderem:

### ► Paravirtualisierung

Bei der Paravirtualisierung muss der Kernel des entsprechenden Gastsystems so angepasst werden, dass dieser direkt mit der von der Virtualisierungssoftware bereitgestellten Hardware kommuniziert. Dadurch weiß das Gastsystem, dass es sich in einer virtuellen Hardware befindet. Da jedoch der Kernel aktualisiert werden muss, ist dies mit proprietären Systemen wie zum Beispiel MS Windows nicht möglich. Xen ist einer der Hauptvertreter der Paravirtualisierung. Anfangs konnten unter Xen aufgrund von Kernel-Problemen keine Microsoft-Systeme verwendet werden. Seitdem die Hardwareerweiterungen Intel VT und AMD-V verwendet werden, ist dies jedoch möglich.

### ► Anwendungsvirtualisierung

Die Anwendungsvirtualisierung wird immer beliebter und erlaubt es, Programme auf Arbeitsplätzen zu verwenden, ohne diese über Setup-Routinen oder Ähnliches zu installieren. Ein bekannter Vertreter dieser Art ist das Tool Sandboxie. Bei der Anwendungsvirtualisierung wird keine Hardware virtualisiert, sondern zwischen das Betriebssystem und die Anwendung eine Art »Zwischenschicht« geschoben, welche die Anwendung vom System abschottet. Notwendige Registrierungseinträge und Bibliotheken sind darin enthalten. Ein Vorteil ist, dass die Programme so vom System und anderen Anwendungen getrennt sind und eventuelle Probleme wegen Inkompatibilität gar nicht erst auftreten. Weiterhin entfällt die meist aufwändige Installation der Software auf den Arbeitsplätzen.

### ► Netzwerkvirtualisierung

Mit dem Begriff Netzwerkvirtualisierung können viele Leute nicht direkt etwas anfangen, obwohl die meisten bereits damit zu tun hatten. Ein VPN (*Virtuelles Privates Netzwerk*) fällt zum Beispiel darunter. Über dieses Thema habe ich übrigens mein erstes Buch geschrieben (*OpenVPN. Das Praxisbuch*. 1. Aufl. Bonn: Galileo Press 2008.)

## Lizenzen

Wenn Sie damit beginnen, Ihre Anwendungen in virtuelle Maschinen auszulagern, denken Sie daran, die Lizenzbestimmungen zu beachten. Die besten Beispiele sind hierbei Microsoft-Produkte wie zum Beispiel Windows XP, Vista oder das MS-Office-Paket. In den meisten Fällen dürften die Leser dieses Buchs Einzelplatz- oder sogar nur OEM-Lizenzen von den Programmen besitzen. Fall Sie MS Office erworben haben, ist dies auch nur eine Einzelplatzversion. Wenn Sie nun

parallel mehrere Maschinen mit Windows XP und/oder sogar MS Office laufen lassen, begehen Sie also eine Straftat, und Sie sollten sich beim Einsatz von Virtualisierungssoftware darüber im Klaren sein.

Denken Sie auch daran, dass die meiste Software heutzutage so »frech« ist und – teilweise sogar ungefragt – Verbindung mit ihrem Hersteller aufnimmt oder im Netzwerk kommuniziert (siehe auch das Beispiel in Abschnitt 2.1.1, »Warum virtualisieren?«). Ich habe schon erlebt, dass verschiedene Programme, vermutlich durch Einträge in der Windows-Registrierungsdatenbank, überhaupt nicht mehr lauffähig waren. Nur eine Neuinstallation des kompletten Systems oder ein Snapshot halfen hier noch. In Unternehmen sollten Sie also nicht mehrere Desktops oder sogar Server virtualisieren, wenn Sie nur eine Lizenz erworben haben.

Mit freier Software wie einer Linux-Distribution oder OpenOffice treten diese Probleme natürlich nicht auf. Viele kostenpflichtige Anwendungen können ja bekanntlich durch freie Software (wie zum Beispiel VMware Workstation durch VirtualBox) ersetzt werden. Beispiele und Informationen hierzu finden Sie auf den Internetseiten <http://www.linuxalt.com/> und <http://www.osalt.com/>.

**2.1.6 Virtualisierungssoftware**

Neben VirtualBox existieren noch viele weitere Programme zur Virtualisierung. Dabei gibt es direkte Konkurrenten und Hypervisoren, die für andere Bereiche gedacht sind. Verschiedene dieser Lösungen habe ich bereits kurz in den vorherigen Abschnitten erwähnt. Hier möchte ich Ihnen diese Alternativen kurz vorstellen. Wenn Sie ein Ziel haben, das mit VirtualBox nicht erreicht werden kann, finden Sie hier eventuell eine andere Lösung.

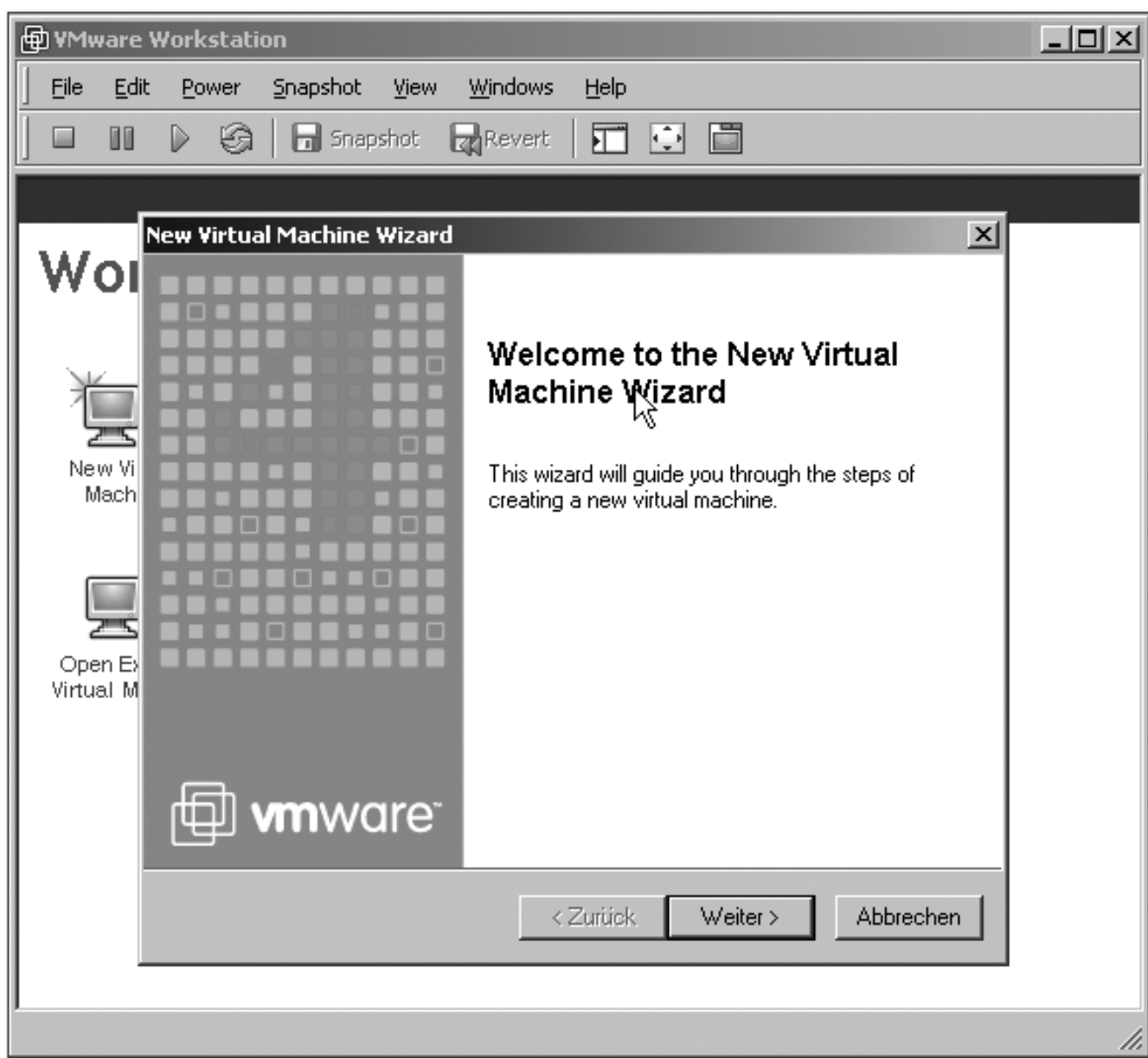
**VMware Workstation**

Die Firma VMware ist so etwas wie der Vorreiter in der Virtualisierung. Ihr Produkt VMware Workstation war meine erste Virtualisierungssoftware. Ich habe das Programm bis Version 4.0 verwendet (siehe Abbildung 2.7).

Inzwischen liegt VMware Workstation in Version 6.5 vor. Es ist im Gegensatz zu VirtualBox kostenpflichtig. Von den Funktionen her ist es ein direkter Konkurrent zu VirtualBox.

Für Apple-Nutzer gibt es eine ähnliche Version mit dem Namen VMware Fusion.

Einsatz von VMware Workstation
▶ vorwiegend für Desktopvirtualisierung
▶ für Privatanutzer und Unternehmen



**Abbildung 2.7** VMware Workstation 4.0

### VMware Server

VMware Server ist eine »abgespeckte« Version des VMware ESX Servers, die kostenlos von VMware angeboten wird. Die Software liegt inzwischen in Version 2.0 vor. Ich verwendete den VMware Server als Nachfolger von VMware Workstation.

Im Gegensatz zum VMware ESX Server benötigt VMware Server Windows oder Linux als Hostsystem.

Seit Version 2.0 erfolgt die Verwaltung über eine Weboberfläche. Ich setze das Programm auch heute noch parallel zu VirtualBox ein. Der Hauptgrund dafür ist, dass im Internet viele fertig eingerichtete Maschinen mit Testsystemen (neue Linux-Versionen, Groupwareserver, laufzeitlimitierte Windows-Server usw.) angeboten werden. Dabei handelt es sich jedoch in den meisten Fällen nur um VMware-Maschinen.

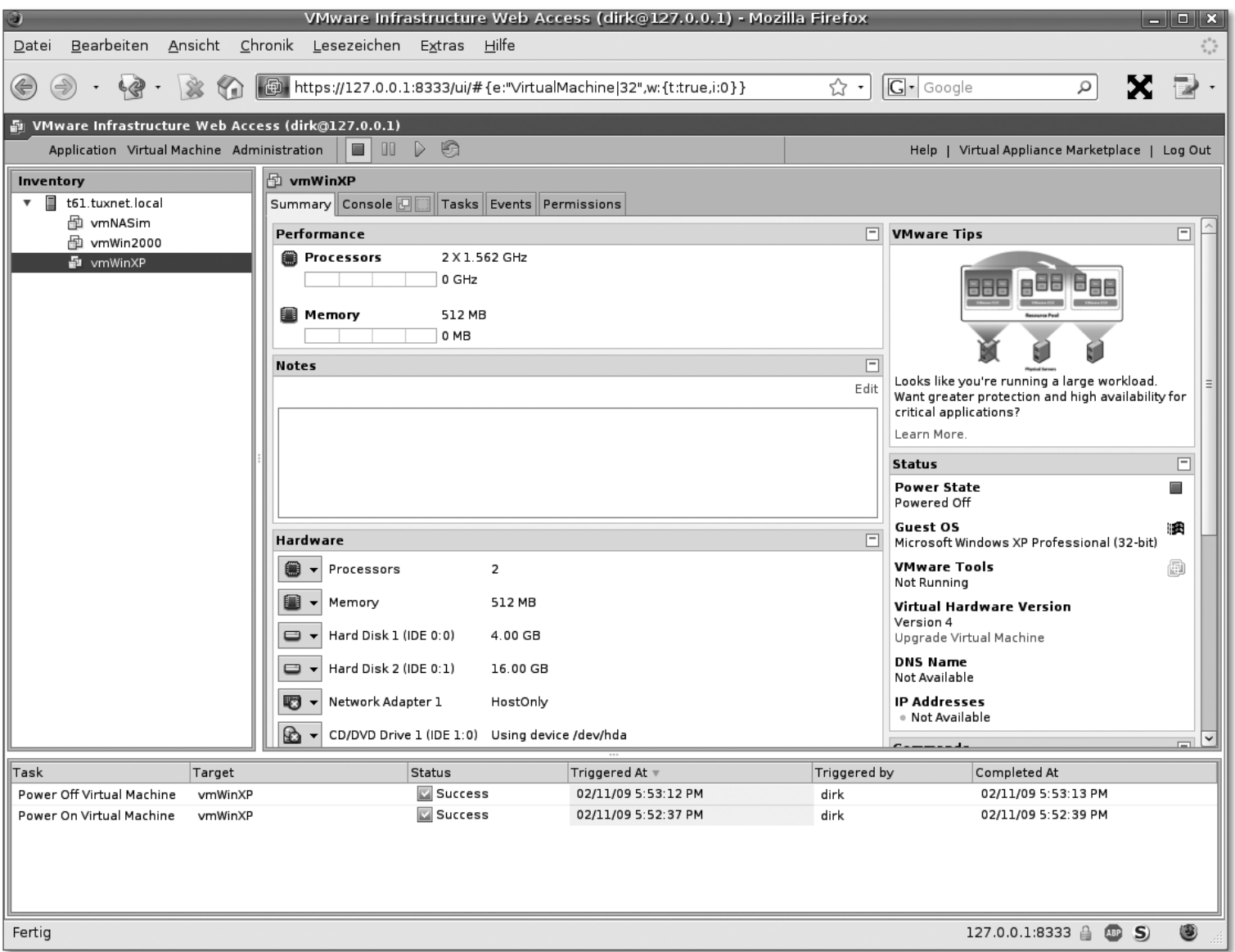


Abbildung 2.8 VMware Server 2.0 mit Infrastructure Web Access

Einsatz von VMware Server

- ▶ für Desktop- und Servervirtualisierung
- ▶ für Privatanutzer und kleine bis mittlere Unternehmen
- ▶ kostenlos

VMware ESX Server

Der VMware ESX Server ist das Flaggschiff von VMware. Er ist der zurzeit am weitesten verbreitete Hypervisor. Da er für Servervirtualisierungen ausgelegt ist, beinhaltet er auch spezielle Funktionen und Erweiterungen wie zum Beispiel VMotion, High Availability (Hochverfügbarkeit) und Load Balancing (Lastverteilung). Manche dieser Erweiterungen sind jedoch nur in Verbindung mit dem Virtual Center einsetzbar. Das Virtual Center ist für die Verwaltung mehrerer VMware ESX Server oder auch VMware Server zuständig.

VMotion ermöglicht das »Umziehen« einer virtuellen Maschine im laufenden Betrieb auf einen anderen Server ohne wirklichen Ausfall. Beim Anpingen der Maschine während eines solchen Umzugs geht meist nicht mehr als ein Ping-Paket verloren.

Durch High Availability wird erreicht, dass virtuelle Maschinen immer zur Verfügung stehen. Um es nutzen zu können, ist mehr als ein Server notwendig. Beim Ausfall eines physikalischen Servers werden dann die davon betroffenen virtuellen Maschinen automatisch auf einem anderen Server gestartet. Dadurch reduzieren sich Ausfallzeiten.

Mit Load Balancing wird die Auslastung der virtuellen Maschinen überwacht. Wenn nun eine viel zu arbeiten hat und dabei andere, die sich auf dem gleichen physikalischen Server befinden, blockiert, werden die betroffenen Maschinen automatisch auf einen freien Server »umgezogen«.

Einsatz von VMware ESX Server

- ▶ Servervirtualisierung
- ▶ vorwiegend für Unternehmen geeignet
- ▶ spezielle Funktionen (High Availability, Load Balancing usw.)
- ▶ erweiterbar

Citrix XenServer

Beim Citrix XenServer handelt es sich um eine verbesserte Ausführung der Virtualisierungssoftware Xen und einen Konkurrenten zum VMware ESX Server. Citrix XenServer beherrscht ebenfalls ähnliche Funktionen wie VMotion und High Availability.

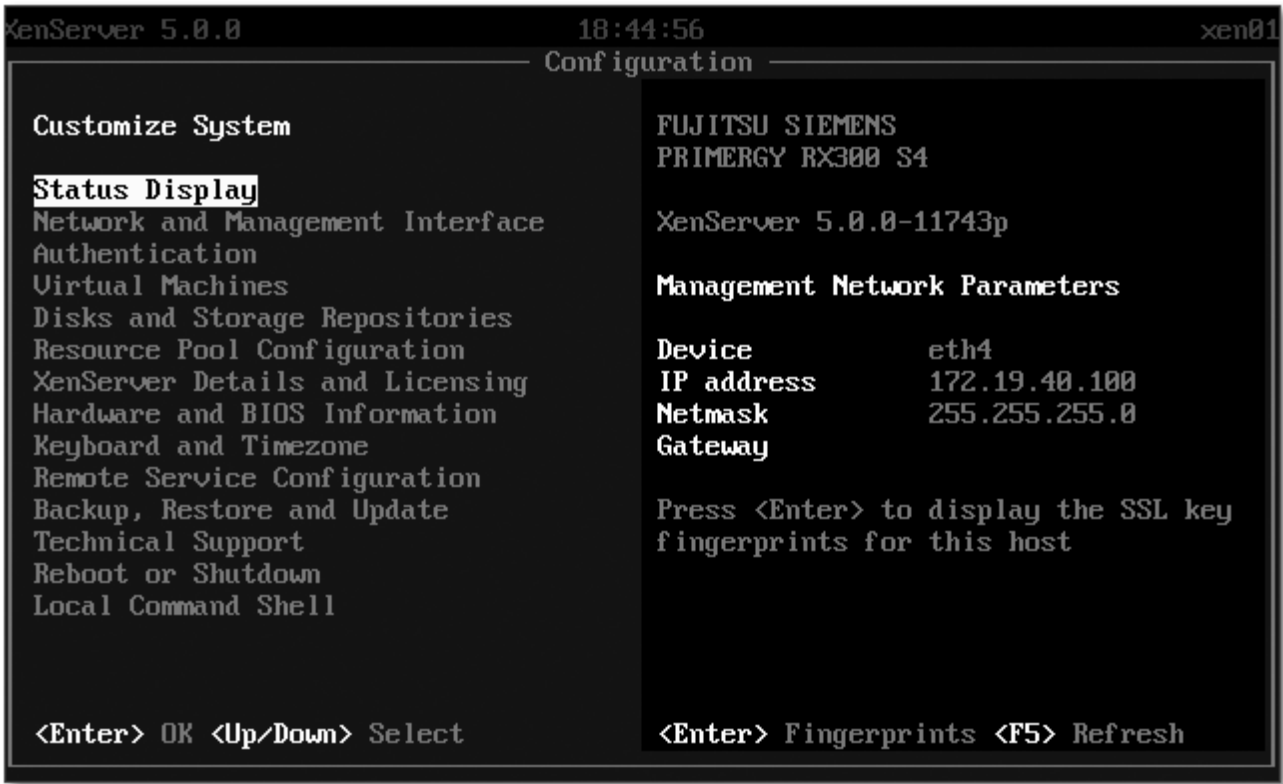


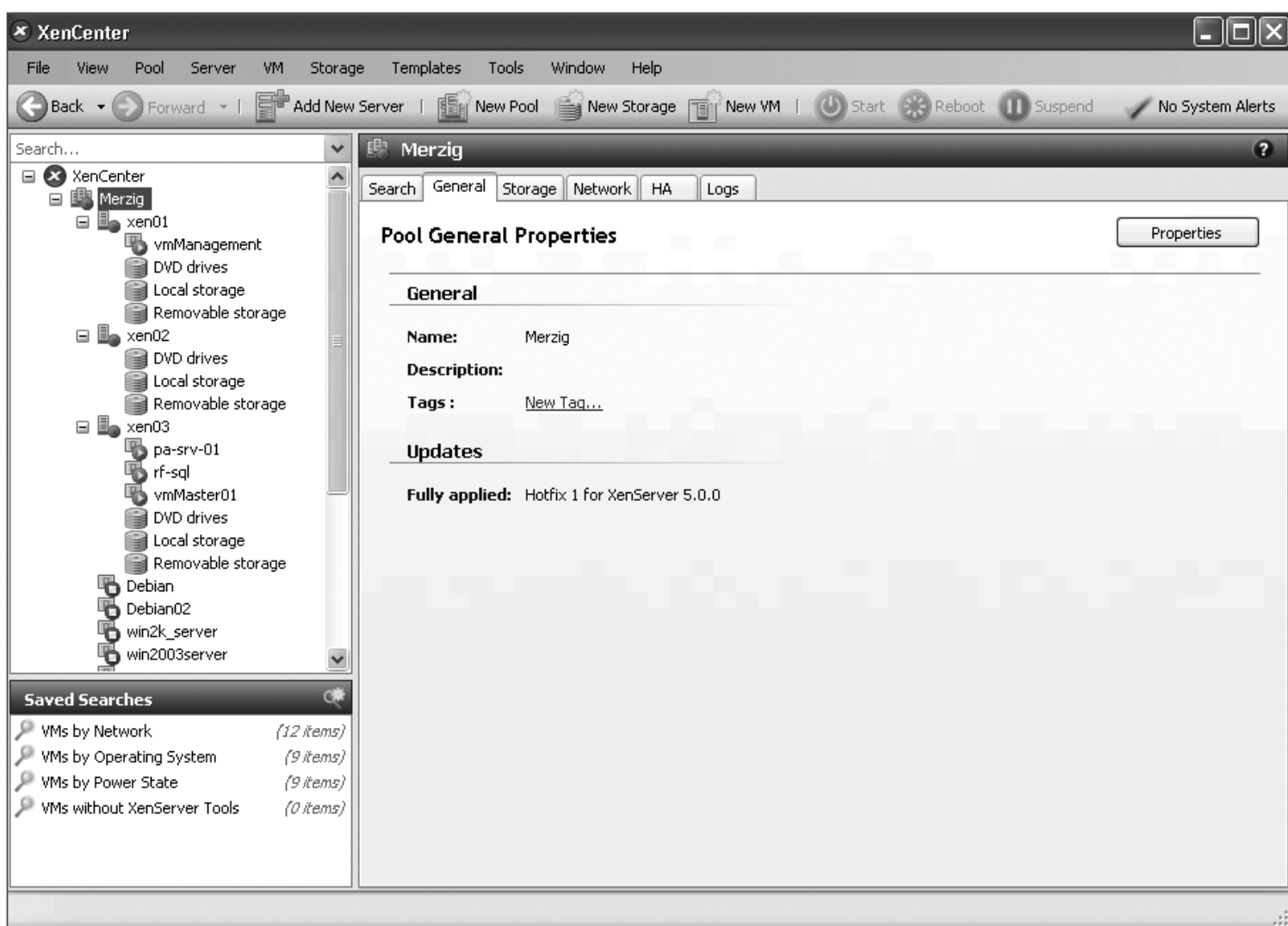
Abbildung 2.9 Konfiguration des Citrix XenServer 5.0

Die Steuerung des Citrix XenServers erfolgt zunächst über eine etwas unkomfortable, textbasierte Oberfläche (Abbildung 2.9). Mit ihr kann man die grundlegenden Konfigurationen (Netzwerk, Uhrzeit, Storage usw.) vornehmen.

Die etwas anwenderfreundlichere Verwaltung erfolgt mit dem XenCenter (Abbildung 2.10). Dieses benötigt jedoch eine .NET-Laufzeitumgebung und steht daher nur für MS-Windows-Systeme zur Verfügung (was etwas eigenartig ist, da Xen seine Wurzeln in der Linux-Welt hat). Allerdings sind noch nicht alle Funktionen in das XenCenter integriert, und man muss gelegentlich auf die Konsole zurückgreifen. Dies soll sich jedoch in den nächsten Versionen ändern.

Citrix XenServer ist kostenlos in mehreren Ausführungen erhältlich:

- ▶ Express Edition
- ▶ Standard Edition
- ▶ Enterprise Edition
- ▶ Platinum Edition



**Abbildung 2.10** Citrix XenCenter

Die Enterprise-Version soll auch bald kostenlos verfügbar sein. Ob dann weiterhin die Express und Standard Edition angeboten werden, ist zurzeit noch unklar.

**Einsatz von Citrix Xen Server**

- ▶ Servervirtualisierung
- ▶ vorwiegend für kleine bis mittlere Unternehmen geeignet
- ▶ spezielle Funktionen (High Availability, Load Balancing usw.)
- ▶ erweiterbar
- ▶ auch kostenlos erhältlich

**Microsoft Virtual PC**

Virtual PC wurde anfangs von der Firma Connectix und der innotek GmbH entwickelt. Als Microsoft im Jahr 2003 Connectix aufkaufte, begann innotek mit der Entwicklung von VirtualBox.

Virtual PC ist ein direkter Konkurrent zu VirtualBox. Die Software wurde nicht nur für Windows- sondern auch für Apple-Systeme entwickelt. Daher wird auch ein x86-Prozessor emuliert. Microsoft wollte so ermöglichen, dass Mac-OS-X-Benutzer auch Windows in einer virtuellen Maschine verwenden können. Die Windows-Version ist im Gegensatz zu der für Mac OS X seit einigen Jahren ebenfalls kostenlos erhältlich.

**Einsatz von Microsoft Virtual PC**

- ▶ vorwiegend für Desktopvirtualisierung
- ▶ für Privatanutzer und Unternehmen

## 2.2 Planung

In den nächsten Abschnitten werden Sie VirtualBox kennen lernen und erfahren, wie Sie Ihre eigenen virtuellen Maschinen erstellen. Zuvor ist jedoch ein wenig Planung notwendig. Sie suchen sich zunächst ein Hostsystem aus, auf dem VirtualBox laufen soll. Dann entscheiden Sie, welches Betriebssystem (also Gastsystem) und welche Software Sie installieren wollen und überlegen, welche Hardwareressourcen die virtuelle Maschine benötigt.

Natürlich ist zur Verwendung einer virtuellen Maschine auf einem Desktopsystem bedeutend weniger Planung notwendig, als bei einer umfangreichen Servervirtualisierung mit eventuell mehreren Hostsystemen; aus diesem Grund ist dieser Abschnitt auch sehr kurz gehalten. Nichtsdestotrotz sind einige Punkte zu beachten.



### 2.2.1 Hostsystem

Im Privatbereich werden Sie vermutlich nur ein Hostsystem zur Verfügung haben: Ihren Desktop-PC. In kleinen bis mittleren Unternehmen sind jedoch meist mehrere Rechner bzw. Server vorhanden. Vielleicht wird sogar ein neues System speziell für die Virtualisierung angeschafft. Ich will an dieser Stelle nicht auf irgendwelche Mindestanforderungen eingehen; diese ändern sich aufgrund der schnellen Entwicklung sowieso jeden Monat. Wichtig ist nur, dass Ihr System über genügend Ressourcen verfügt, um das Hostsystem und die gewünschte Anzahl von Gastsystemen ausreichend zu »versorgen«.

#### Prozessor

Früher waren Mehrprozessorsysteme hauptsächlich den Serversystemen vorenthalten. Durch Dual- und Quadprozessoren hat sich dieses Privileg jedoch auch auf Desktopsysteme ausgeweitet. Für die Virtualisierung ist dies ein großer Vorteil. VirtualBox (seit Version 3.0) und verschiedene Hypervisoren können dem Gastsystem sogar mehrere CPUs zuweisen. Eine Maschine unter VirtualBox kann bis zu 32 CPUs verwenden. Dies ist jedoch meist unnötig – ein Prozessor für ein Gastsystem ist fast immer völlig ausreichend.

Der PC des Hostsystems sollte mindestens über einen Dualprozessor verfügen. Dadurch wird die CPU-Auslastung besser verteilt – wobei diese Verteilung nicht von VirtualBox, sondern vom Hostsystem vorgenommen wird.

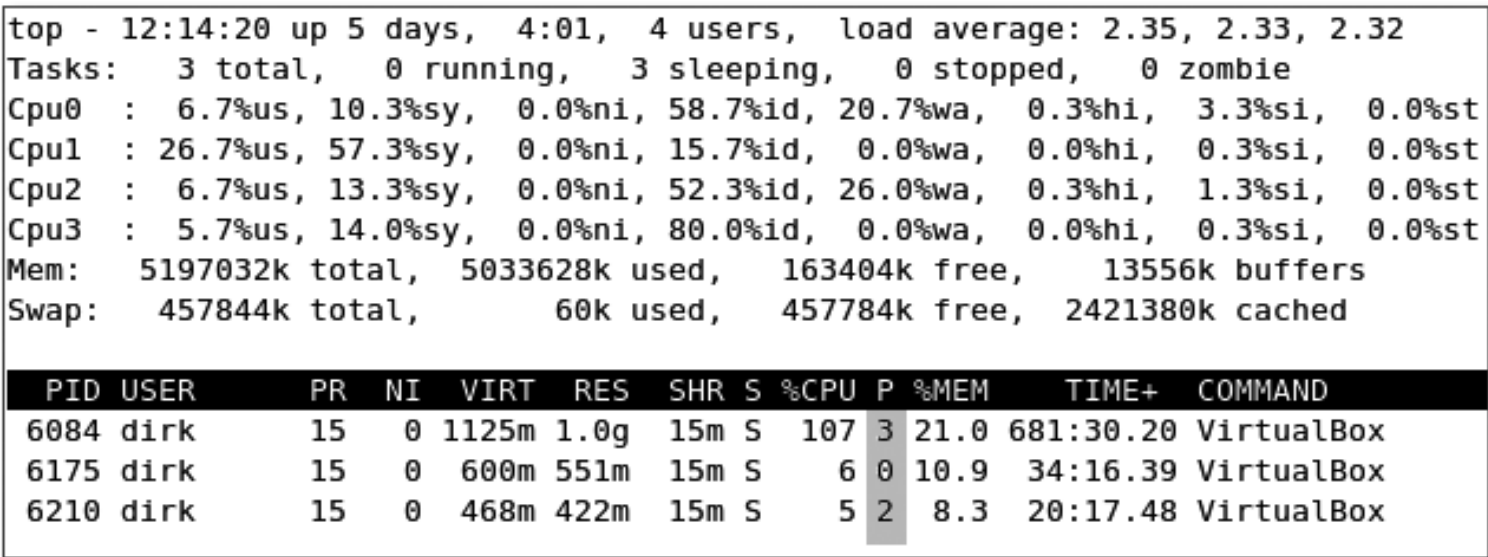


Abbildung 2.11 CPU-Auslastung unter Linux

In Abbildung 2.11 erkennen Sie drei laufende VirtualBox-Maschinen auf einem Linux-Host mit vier CPUs. Die Maschinen sind den Prozessoren 3, 0 und 2 zugeordnet; die Nummerierung beginnt hier bei 0 für die erste CPU. Dabei kann es auch vorkommen, dass die Systeme auf der gleichen CPU laufen – also wundern Sie sich nicht, wenn bei Ihnen eine andere Verteilung angezeigt wird.

Ein weiterer Punkt sind die Hardwareerweiterungen Intel V und AMD-V. Wenn Sie unter VirtualBox die Möglichkeit nutzen wollen, einer virtuellen Maschine mehrere Prozessoren zuzuweisen (SMP), muss der Prozessor des Hostsystems diese Erweiterungen unterstützen. Auch für verschiedene »exotische« Betriebssysteme (wie zum Beispiel OS/2) sind sie notwendig. Für einen normalen Einsatz von VirtualBox benötigen Sie diese allerdings nicht unbedingt. Mein Xeon-Server unterstützt sie ebenfalls noch nicht.

Speicher

Der Hauptspeicher ist bei der Virtualisierung ein sehr wichtiger Punkt. Man kann hiervon fast nicht genug haben. Eine virtuelle Maschine belegt zum Beispiel mit einem Windows-Gastsystem unter VirtualBox meist etwas mehr Speicherplatz als ihr zugewiesen wurde. In der Spalte *RES* in Abbildung 2.12 erkennen Sie den Verbrauch der Maschinen. Der ersten wurde 1 GByte, der zweiten 512 MByte und der dritten 392 MByte zugewiesen. Alle verwenden tatsächlich ein paar MByte mehr als in der Konfiguration angegeben wurde.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	P	%MEM	TIME+	COMMAND
6084	dirk	15	0	1125m	1.0g	15m	S	107	3	21.0	681:30.20	VirtualBox
6175	dirk	15	0	600m	551m	15m	S	6	0	10.9	34:16.39	VirtualBox
6210	dirk	15	0	468m	422m	15m	S	5	2	8.3	20:17.48	VirtualBox

Abbildung 2.12 Speicherverbrauch der drei virtuellen Maschinen

Trotz dieses »Mehrverbrauchs« können Sie grob rechnen:

Speicher Hostsystem  
+ Speicher VM01  
+ Speicher VM02  
usw.  
-----  
= Gesamtspeicher

Wenn auf Ihrem Desktop nun Windows XP oder Linux (zum Beispiel Ubuntu) installiert ist, reichen diesen Betriebssystemen circa 512 MByte Speicher, um damit zu arbeiten. Windows Vista und Windows 7 benötigen bekanntermaßen mindestens 1 GByte. Mit 2 GByte Hauptspeicher können Sie bereits mit mehreren Maschinen parallel arbeiten.

Einschränkungen ergeben sich jedoch gegebenenfalls durch Ihr Hostsystem. Verschiedene (meist 32-Bit-)Systeme können nur bis zu 4 GByte Hauptspeicher adressieren. Überprüfen Sie also unbedingt vor einem Kauf, ob die gewünschte Größe überhaupt verwendet werden kann.

## Festplattenspeicher

Beim Testen und Arbeiten von und mit virtuellen Maschinen werden schnell über 100 GByte an Festplattenspeicher benötigt. Ich verwende daher immer eine eigene Festplatte oder zumindest eine eigene Partition mit ausreichender Kapazität. Wenn das Hostsystem und die virtuellen Maschinen auf verschiedenen Festplatten liegen, bringt dies außerdem noch einen merklichen Geschwindigkeitsschub.

Standardmäßig verwendet VirtualBox zum Abspeichern der Maschinen und deren Festplatten-Images das Heimatverzeichnis des Benutzers die folgenden Verzeichnisse:

Heimatverzeichnis unter Linux:

*/home/BENUTZER/.VirtualBox*

Heimatverzeichnis unter Windows:

*C:\Dokumente und Einstellungen\BENUTZER\VirtualBox*

Die Größen der virtuellen Maschinen können je nach Gastsystem schnell mehrere GByte betragen. Daher sind meiner Meinung nach diese Verzeichnisse nicht geeignet dazu. Am sinnvollsten ist es, für die Maschinen eine eigene Partition bereitzustellen. Die meisten Festplatten von »Fertig-PCs« sind aufgrund der enormen Größen bereits ab Werk partitioniert.

Wenn Ihre Festplatte nur eine Partition besitzt und keine weitere Festplatte eingebaut werden kann bzw. soll, können Sie auf die freie Linux-Distribution Parted Magic zurückgreifen. Mit dieser kann die Partition einer Festplatte ohne Datenverlust geändert werden. Allerdings sollten Sie zumindest Grundkenntnisse über die Partitionierung besitzen und nicht auf eine vorherige Datensicherung verzichten.

### Parted Magic

Normalerweise wäre ich vermutlich nicht weiter auf das Thema »Partitionierung« und auf *Parted Magic* eingegangen. Aber bei der Verwaltung von virtuellen Maschinen und der Migration von physikalischen Rechnern ist *Parted Magic* in Zusammenarbeit mit *Partimage* ein sehr gutes, kostenloses Hilfsmittel. Die aktuelle Version dieser Distribution finden Sie unter <http://partedmagic.com/>.

[+]

Parted Magic steht dort als ISO- oder USB-Image zum Download zur Verfügung. Laden Sie die gewünschte Datei herunter und erstellen Sie dann die Boot-CD bzw. den Boot-USB-Stick. Falls Sie mit USB-Sticks noch wenig Erfahrung haben, empfehle ich Ihnen, das ISO-Image zu verwenden. Dieses kann mit fast jeder gängigen

Brennsoftware auf eine CD gebrannt werden. Außerdem booten alle aktuellen und auch fast alle älteren PCs von CD. Das Booten von USB-Medien wird hingegen nicht von allen Computern unterstützt.

Beim Starten von Parted Magic erscheint zunächst ein Menü, in dem Sie verschiedenen Startkonfigurationen wählen können.



**Abbildung 2.13** Das Bootmenü von Parted Magic

Versuchen Sie es zunächst mit dem ersten Punkt (DEFAULT SETTINGS). Gelegentlich kommt es hierbei jedoch zu Problemen mit der Grafikkarte. In einem solchen Fall verwende ich immer Punkt 6 (SAFE GRAPHICS SETTINGS). Für Emulatoren bzw. virtuelle Systeme bietet Parted Magic hier noch den Punkt 4 an (DO NOT EJECT CD). Mit dieser Einstellung wird unter anderem auf das Auswerfen der CD verzichtet.

Um nach dem Start eine deutsche Tastatur zu verwenden, öffnen Sie die Konsole (zum Beispiel über das Symbol ROXTERM auf dem Desktop) und geben dort `setxkbmap de` ein.

Zum eigentlichen Partitionieren dient das Programm GParted (Abbildung 2.14). Kenner von PartitionMagic werden Ähnlichkeiten entdecken.

## [+]

### Erst testen, dann ausführen

Sie können GParted übrigens in aller Ruhe testen. Erst durch einen Klick auf APPLY werden die Änderungen an den Partitionen wirklich übernommen.

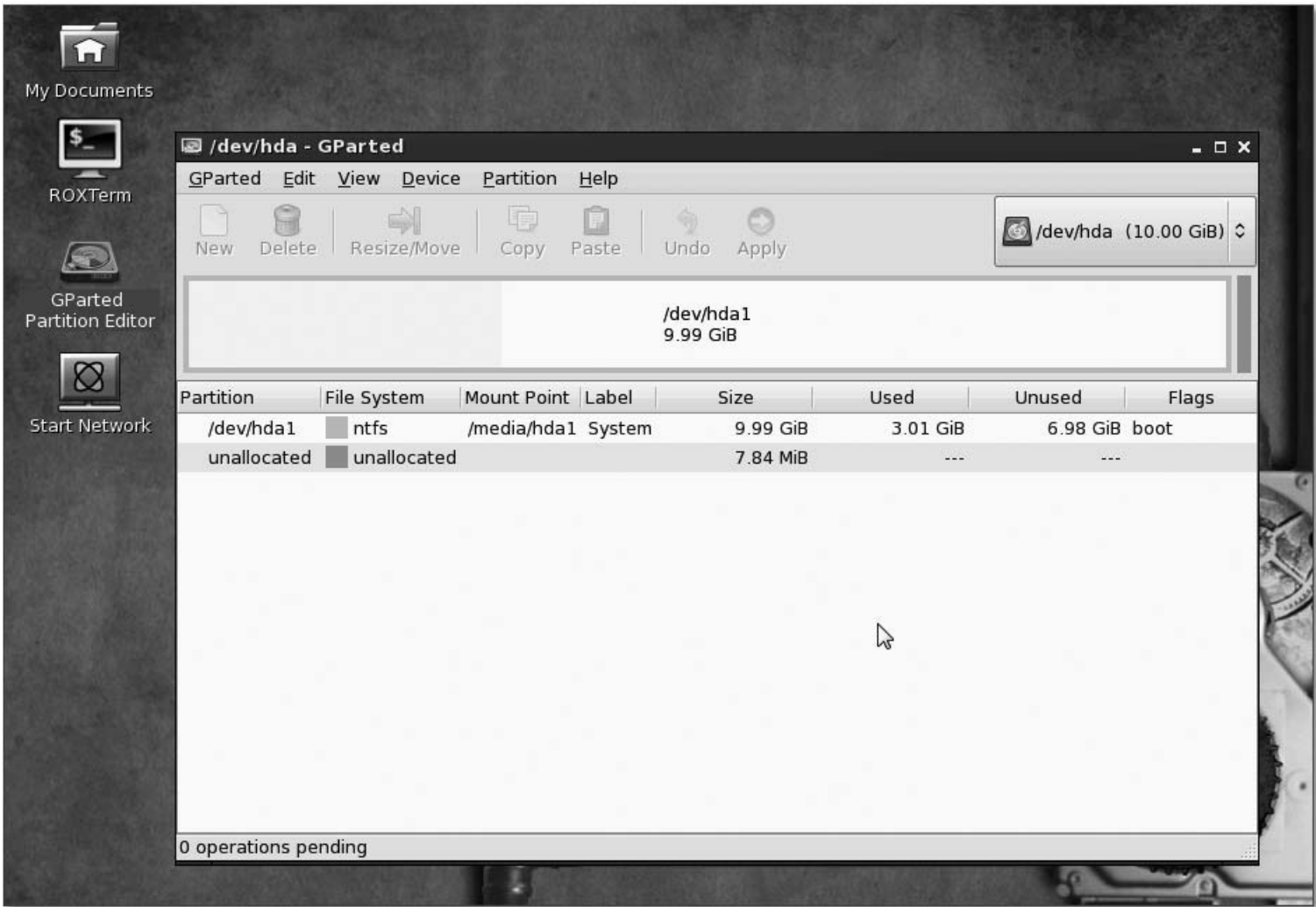


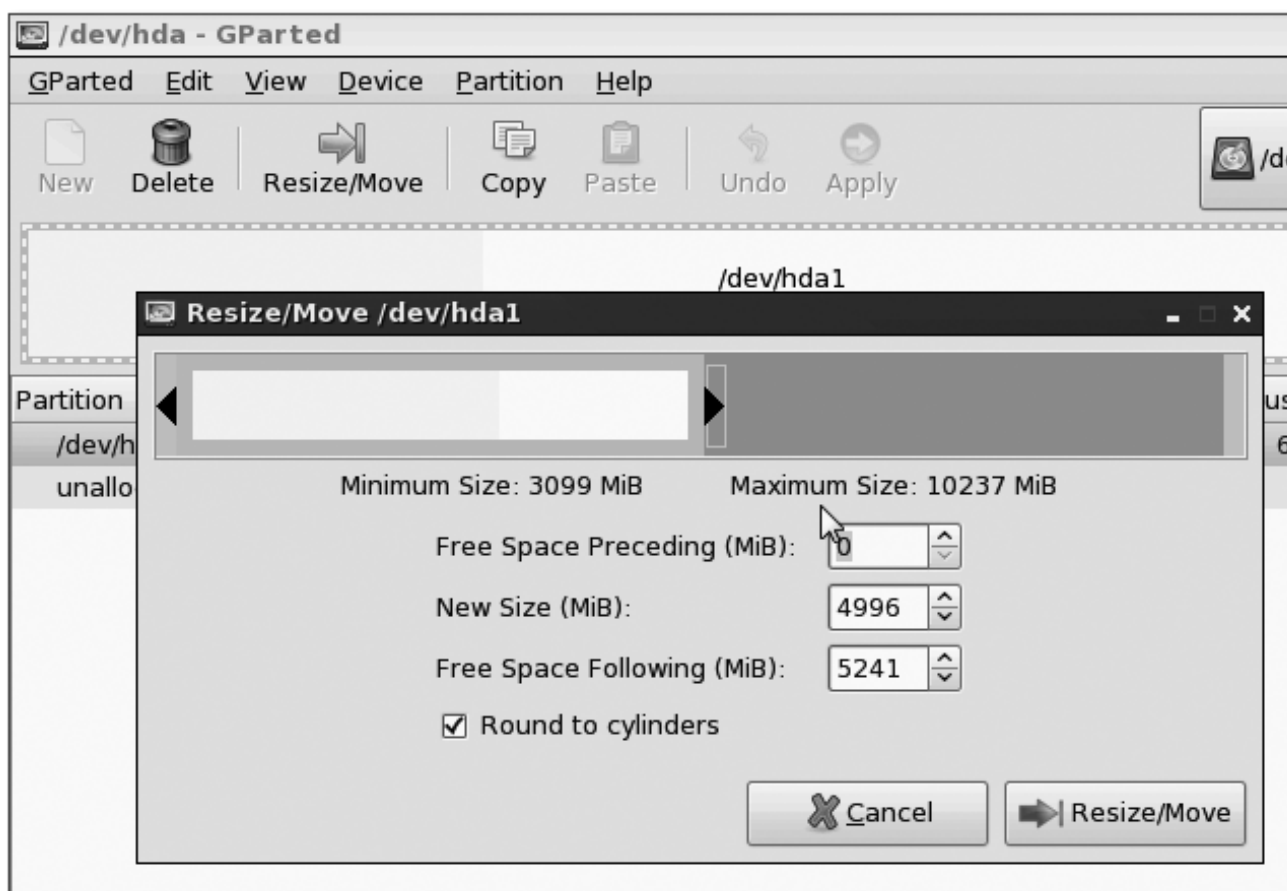
Abbildung 2.14 GParted unter Parted Magic

Da Parted Magic ein Linux-System ist, werden die Festplatten nicht mit den von Windows her bekannten Laufwerksbuchstaben angezeigt, sondern mit sogenannten Device-Namen. Diese werden dabei wie folgt vergeben: `/dev/hda` bezeichnet die erste IDE-Festplatte, `/dev/hdb` die zweite IDE-Festplatte, `/dev/sda` die erste SCSI- oder SATA-Platte usw.

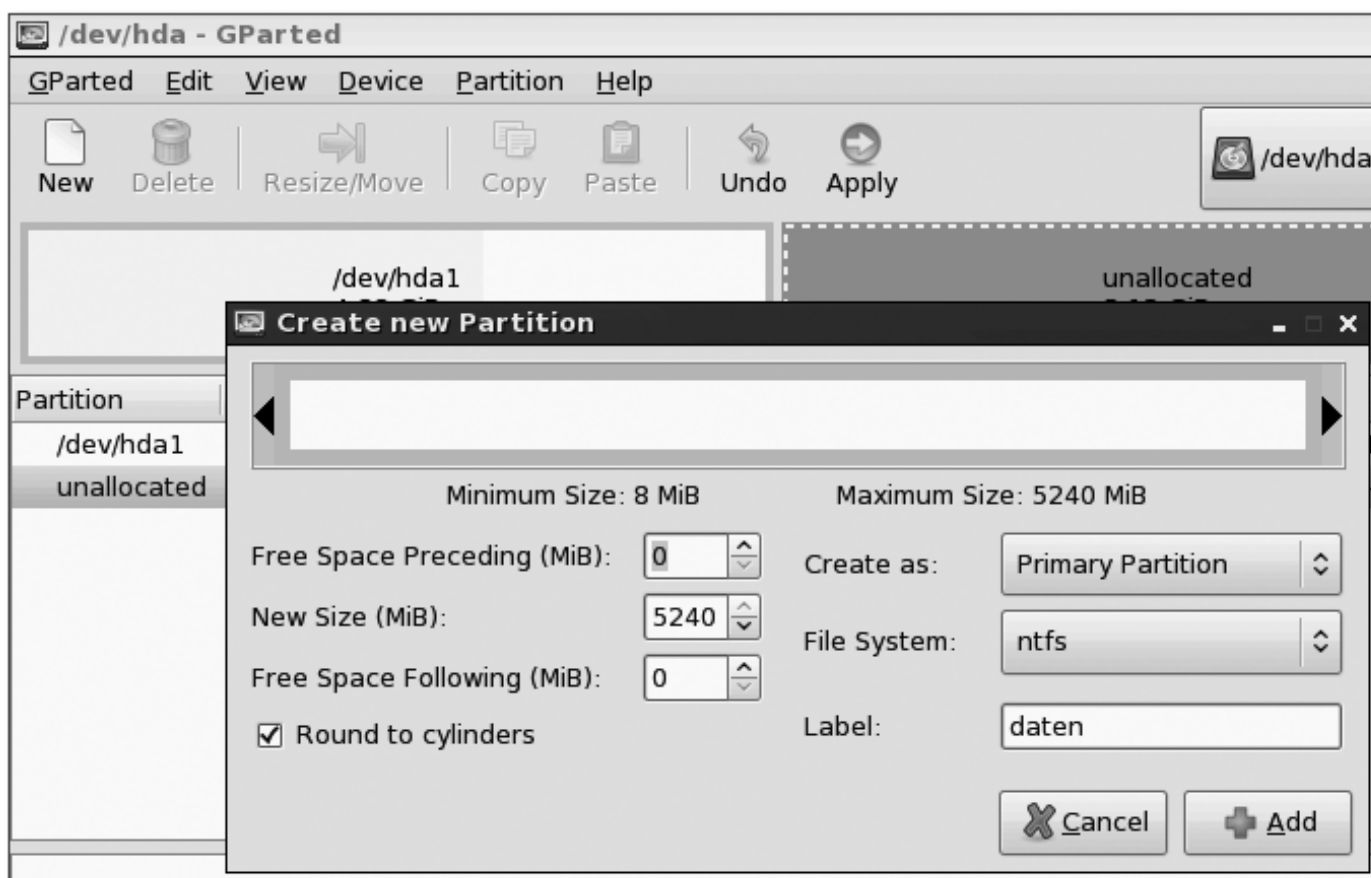
Hinter den Device-Namen steht außerdem eine Ziffer. Diese Zahlen geben die Nummer der Partitionen an: 1–4 steht für die primären und erweiterten Partitionen, und ab 5 sind die logischen Laufwerke gemeint. So bezeichnet `/dev/hda1` zum Beispiel die primäre Partition der ersten IDE-Festplatte (C:) und `/dev/hdb5` das erste logische Laufwerk auf der zweiten IDE-Festplatte.

Um eine neue Partition für die virtuellen Maschinen zu erstellen, muss die vorhandene zunächst verkleinert werden. Dazu markieren Sie diese und klicken auf den Button `RESIZE/MOVE` in der Symbolleiste (Abbildung 2.15).

Danach können Sie in dem neuen freien Bereich (UNALLOCATED) eine Partition erstellen. Achten Sie hierbei vor allem auf den Typ des Dateisystems (FILESYSTEM). Ab Windows 2000 sollte hier NTFS und unter Linux EXT3 oder REISERFS verwendet werden.



**Abbildung 2.15** Größenänderung einer Partition in GParted



**Abbildung 2.16** Anlegen einer neuen Partition

Wenn Sie alles angepasst haben, übernehmen Sie durch einen Klick auf APPLY die Änderungen (Abbildung 2.17). Das Ganze kann je nach Größe der vorhandenen Partition einige Zeit dauern, da diese zuerst »aufgeräumt« werden muss. Dabei werden Dateien, die sich am Ende der Partition befinden, in den vorderen Bereich verschoben.

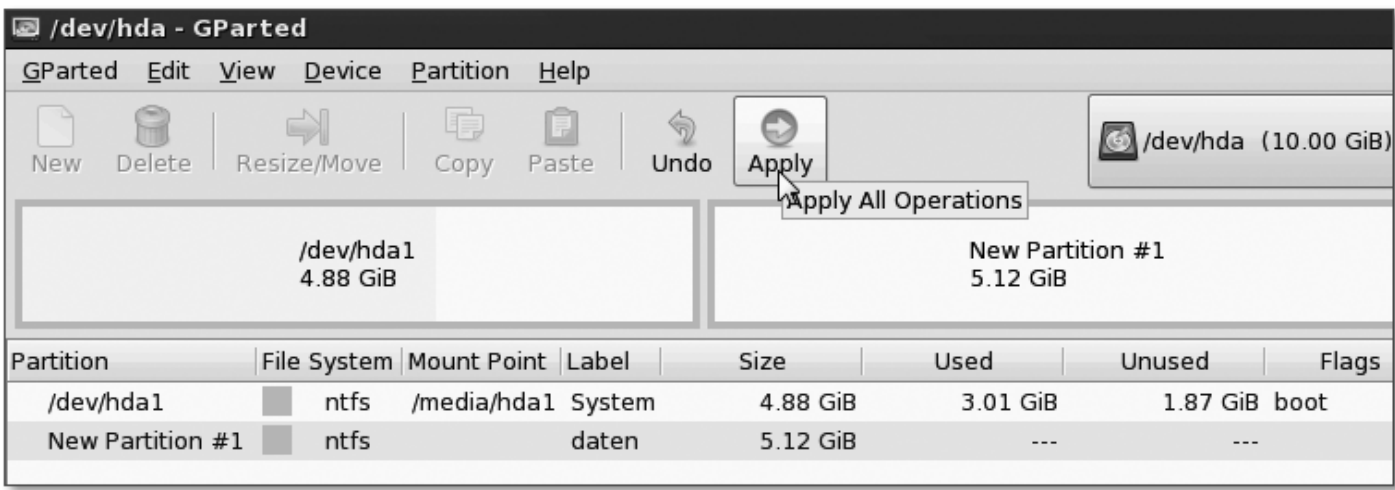


Abbildung 2.17 Die Änderungen übernehmen

Defragmentieren vorm Partitionieren

Um den Vorgang unter einem Windows-Hostsystem zu beschleunigen, können Sie vor der Partitionierung das System durch eine Defragmentierung aufräumen.

[+]

Netzwerk

Das Hostsystem sollte Zugriff auf Ihr Netzwerk und auch das Internet haben. Dabei sind mehrere Netzwerkkarten von Vorteil. Dadurch kann später der Netzwerkverkehr des Hostsystems und der Gastsysteme verteilt werden. Falls eine virtuelle Maschine als Gateway fungieren soll, sind mindestens zwei Netzwerkkarten im PC Pflicht. (Eine Ausnahme von dieser Regel wäre dann gegeben, wenn zum Beispiel ein Internetrouter über USB angeschlossen wird. Dieser ersetzt dann den zweiten Netzwerkadapter.)

USB

Einer der großen Vorzüge von VirtualBox ist der USB-Zugriff. Virtuelle Maschinen können dabei fast problemlos auf USB-Endgeräte zugreifen bzw. diese »durchgereicht« bekommen. Falls Sie USB einsetzen wollen, so sollte das Hostsystem USB 2.0 verwenden. Dies ist bei aktuellen Systemen in der Regel der Standard.

Serielle Schnittstellen

Wenn Sie planen, ein altes System zu virtualisieren, das Geräte über die serielle Schnittstelle steuert bzw. benötigt, muss Ihr Hostsystem natürlich ebenfalls über solche Anschlüsse verfügen. Viele (meist billige) Neugeräte und Notebooks besitzen diese Schnittstellen nicht mehr. Ein USB-Seriell-Adapter ist dabei leider häufig auch keine Lösung, da nicht alle seriellen Geräte damit angesprochen werden können.



### Was Sie für das Host-System beachten sollten

- ▶ Ihr Hostsystem sollte über einen Dualprozessor verfügen.
- ▶ Denken Sie immer daran, dem Hostsystem genügend Speicher frei zu halten.
- ▶ Durch Auslagerung der virtuellen Maschinen auf eine zweite Festplatte erhöhen Sie die Performance.

## 2.2.2 Virtuelle Maschine

Bei der Planung einer virtuellen Maschine müssen Sie sich als Erstes Gedanken über deren Einsatzzweck machen. Zunächst gilt es zu überlegen, was die virtuelle Maschine leisten soll:

- ▶ virtueller Server
- ▶ Desktopersatz
- ▶ Testumgebung für neue Software
- ▶ Testumgebung für Systeme

Je nachdem, welche Aufgaben sie übernehmen soll, entscheiden Sie, welches Betriebssystem und welche Software Sie benötigen, und nach diesen Kriterien richten sich wiederum die notwendigen Ressourcen.

### Ressourcen

Wenn es nur um einen kurzen Test geht und die Maschine danach nicht mehr benötigt wird, ist eine Ressourcenplanung natürlich egal. Wenn Sie jedoch Maschinen erstellen, die vielleicht sogar parallel laufen sollen, überlegen Sie genau, was die Maschine benötigt. Ich stelle einem neuen Gastsystem am Anfang meist weniger Speicher zur Verfügung und beobachte dessen Verhalten. Wenn die Maschine als Desktopumgebung mit Windows XP laufen soll, auf der intensiv Office- und Grafikprogramme verwendet werden, ist mehr Speicher notwendig als für ein Linux-System, das nur zum sicheren Surfen im Internet eingesetzt wird.



### Ressourcen begrenzen

Stellen Sie, vor allem beim Parallelbetrieb, einer virtuellen Maschine nicht mehr Ressourcen zur Verfügung als sie benötigt! Dies würde sich negativ auf die Leistung anderer Maschinen und des Hostsystems auswirken.

### Hardware

Denken Sie daran, dass VirtualBox zwar Zugriff auf USB ermöglicht, aber spezielle Hardware wie zum Beispiel eine ISDN-Karte nicht vom Gastsystem verwendet werden kann. Ein virtueller Faxserver, der eine solche Karte benötigt, kann also



überhaupt nicht virtualisiert werden. Denken Sie also genau darüber nach, ob Ihre Maschine spezielle Hardware benötigt, und wenn das der Fall ist, prüfen Sie, ob diese Hardware auch »umgangen« werden kann. Manche Faxsoftware kann zum Beispiel über eine Netzwerkkarte mit einer ISDN-Box kommunizieren. In diesem Fall kann der Faxserver also dennoch virtualisiert werden.

**Was Sie bei der virtuellen Maschine beachten sollten****[!]**

- ▶ Prüfen Sie, welche Hardware die virtuelle Maschine benötigt.
- ▶ Vergeben Sie nur so viel Speicher wie mindestens notwendig ist.



*Sie installieren VirtualBox nach Ihren Vorstellungen, nehmen alle wichtigen »Hardware-« und Netzwerkeinstellungen vor und erlernen systematisch den Umgang mit VirtualBox.*

## **3 VirtualBox installieren und einsetzen**

Nun kommen wir zum eigentlichen Kern des Buchs: zu VirtualBox. Zunächst werde ich in diesem Abschnitt die Entwicklungsgeschichte und die verschiedenen Versionen von VirtualBox ansprechen. Danach beginnen wir mit der Installation und der Erstellung unserer ersten virtuellen Maschine.

Verschiedene Programme und Funktionen werde ich hier nur kurz und oberflächlich vorstellen. Deren detaillierter Einsatz und Anwendung werden Sie später in den Praxisbeispielen und in Kapitel 5, »Weitere Funktionen und Tipps«, kennen lernen. Der Grund ist, dass es einfach leichter ist, unbekannte Funktionen zu erlernen, wenn Sie parallel auch deren Einsatz sehen.

### **3.1 Hersteller**

#### **3.1.1 innotek GmbH**

Ursprünglich wurde VirtualBox von der Firma innotek GmbH entwickelt. innotek hatte auch zuvor schon Erfahrungen in der Entwicklung von Virtualisierungssoftware. Die Firma arbeitete zusammen mit Connectix an Virtual PC. Connectix wurde jedoch 2003 von Microsoft aufgekauft, und Virtual PC wird heute von der Softwareschmiede aus Redmond weiterentwickelt.

Ich selbst hatte im Jahr 2007 Kontakt mit verschiedenen Mitarbeitern der Firma innotek. Hintergrund war die Entwicklung einer Art »Archivierungssoftware« für Systeme. Diese archivierten Systeme konnten dann in VirtualBox-Umgebungen gestartet werden. Dadurch kam ich zum ersten Mal mit VirtualBox in Kontakt, und kurze Zeit später migrierte ich die ersten meiner virtuellen Maschinen, die bis dato unter VMware Server liefen.

### 3.1.2 Sun und xVM

Sun übernahm im Februar 2008 die Firma innotek und gliederte VirtualBox in die xVM-Plattform mit ein. xVM steht für *Intersection of Virtualization and Management* und die Plattform besteht aus den folgenden Komponenten:

- ▶ xVM Server
- ▶ xVM Ops Center
- ▶ VDI Software
- ▶ xVM VirtualBox

Die einzelnen Produkte stehen dabei größtenteils unter der GPL und sind also OpenSource-Programme; nur verschiedene Module sind nicht quelloffen. Man kann die freien Komponenten also beliebig einsetzen und den Quellcode einsehen. Sun hat dazu die OpenxVM-Community (<http://www.openxvm.org/>) gegründet und bietet auch verschiedene Supportverträge an.

Die xVM-Plattform befindet sich jedoch teilweise noch im Betastadium (zumindest zur Drucklegung des Buches).

#### xVM Server

Der xVM Server ist ein auf Xen basierender Hypervisor. Er ist also unter anderem ein Konkurrent zum VMware ESX Server und Citrix XenServer. Weitere Infos finden Sie unter <http://www.xvmserver.org/>.

#### xVM Ops Center

Das Ops Center ist eine Management-Plattform zum Verwalten von physikalischen und virtuellen Systemen. Als Administrator eines Rechenzentrums hat man hierbei den Überblick über die Systeme und kann automatisch Aktualisierungen einspielen.

#### VDI Software

Die Virtual Desktop Infrastructure Software (VDI) ermöglicht die Bereitstellung von virtuellen Desktopumgebungen über das lokale Netzwerk (LAN, *Local Area Network*), das WAN (*Wide Area Network*) oder sogar über das Internet.



#### Namensgleichheit von Dateiendungen

Die Disk-Images von VirtualBox haben ebenfalls die Dateiendung *.VDI*. Dieses Kürzel steht hier für *Virtual Disk Image*. Verwechseln Sie diese Images daher nicht mit der Virtual Desktop Infrastructure.

### 3.1.3 Oracle

Im April 2009 hat Oracle, der zurzeit weltweit drittgrößte Softwareentwickler, angekündigt, Sun zu übernehmen. Wie es um die Zukunft der freien Software steht, die von Sun sehr unterstützt wurde, ist noch offen. Dies betrifft unter anderem OpenOffice, als Ableger von Suns StarOffice, MySQL und natürlich auch VirtualBox.

## 3.2 Installation und Konfiguration

In den folgenden Abschnitten erkläre ich Ihnen unter anderem die Installation und Konfiguration von VirtualBox. Einige dieser Punkte finden Sie auch in Kapitel 4, »Praxisbeispiele«, wieder. Allerdings werden sie dort anhand von Beispielen erläutert und sind dann, gerade für Anfänger, besser nachvollziehbar. Wenn Sie also noch Verständnisprobleme beim Lesen der folgenden Abschnitte haben, bringen die Praxisbeispiele bestimmt Klarheit.

### 3.2.1 Einsatzgebiet und Funktionen

In Abschnitt 2.1.5, »Virtualisierung«, wurde bereits angesprochen, was VirtualBox eigentlich ist: VirtualBox ist ein Virtual Machine Monitor für Desktopsysteme und damit auch ein direkter Konkurrent von VMware Workstation und Microsoft Virtual PC. VirtualBox wurde nicht entwickelt, um Serversysteme zu virtualisieren. Dazu fehlen viele wichtige Funktionen, die bei den Servervirtualisierern vorhanden sind. Das Einsatzgebiet von VirtualBox ist daher im Privatbereich oder auch bei Unternehmen, unter anderem, um Betriebssysteme oder Software zu testen. In kleinerem Umfang können natürlich dennoch virtuelle Desktops oder Serverumgebungen zur Verfügung gestellt werden.

VirtualBox bietet virtuelle Laufzeitumgebungen, in denen verschiedene Hardwarekomponenten emuliert werden. Weiterhin stellt es viele Funktionen zum Arbeiten mit den virtuellen Maschinen zur Verfügung. Zu den wichtigsten Funktionen zählen:

- ▶ Virtuelle Festplatten-Images (VDI, *Virtual Disk Image*)
- ▶ Verwenden von ISO-Images (CD/DVD-Abbildern)
- ▶ Verwenden von Floppy-images
- ▶ Zugriff auf USB-Geräte des Hostsystems
- ▶ Konfigurationsdateien im XML-Format
- ▶ Gasterweiterungen für die Gastsysteme
- ▶ Gemeinsame Ordner zum Datenaustausch

- ▶ Fernsteuerung der virtuellen Maschinen per RDP (*Remote Desktop Protocol*)
- ▶ USB über RDP
- ▶ Sicherungspunkte (Snapshots)
- ▶ Assistent zum Erstellen neuer Maschinen
- ▶ SMP (*Symmetrisches Multiprozessorsystem*)
- ▶ Größenänderung des virtuellen Grafikkartenspeichers
- ▶ OpenGL 2.0 und Direct3D
- ▶ Intel-VT und AMD-V-Unterstützung
- ▶ Im- und Export von Maschinen im Open Virtualization Format

#### 3.2.2 Hostsysteme

Wie bereits mehrfach angesprochen, läuft VirtualBox auf fast allen gängigen Betriebssystemen. Es ist daher fast gleichgültig, für welches System Sie sich als Hostsystem entscheiden. Zu den unterstützten Systemen gehören:

- ▶ Windows (auf x86 und AMD-64-CPU's)
- ▶ Mac OS X (mit Intel-CPU's)
- ▶ Linux
- ▶ Solaris und OpenSolaris

Der Begriff »Linux« in dieser Liste muss noch ein wenig spezifiziert werden, da es inzwischen unzählige verschiedene Linux-Distributionen gibt. Sun bietet für viele dieser Distributionen spezielle Pakete und ein allgemeingültiges Paket für alle an. Die wichtigsten der aktuell unterstützten Linux-Versionen sind:

- ▶ Ubuntu, Debian, openSUSE
- ▶ SUSE Linux Enterprise Server
- ▶ Fedora
- ▶ Mandriva
- ▶ Red Hat Enterprise Linux, CentOS

#### 3.2.3 Gastsysteme

VirtualBox unterstützt eine große Anzahl von Gastsystemen – sowohl offiziell, als auch inoffiziell. Wenn Ihr »Wunschsystem« nicht in der folgenden Liste aufgeführt ist, bedeutet dies nicht zwangsläufig, dass Sie es nicht installieren können. Solange es keine spezielle Hardware benötigt, führt ein Versuch in den meisten Fällen auch zum Erfolg. Fast alle gängigen Betriebssysteme, die auf einem x86-

Prozessor laufen, können unter VirtualBox als Gastsystem installiert werden. Dazu zählen auch Systeme wie IBM OS/2, das mittlerweile eigentlich schon zu den »Exoten« zählt. Zu den unterstützten Microsoft-Gastsystemen gehören:

- ▶ MS-DOS/Windows 3.1
- ▶ Windows 95, 98 und ME
- ▶ Windows NT, 2000, XP und Vista
- ▶ Windows 2003 und 2008
- ▶ Windows 7
- ▶ alle Systeme auch als 64-Bit-Version

Weitere unterstützte Gastsysteme sind:

- ▶ Linux mit Kernel 2.4 (bedingt), Linux mit Kernel 2.6
- ▶ Debian, Ubuntu, openSUSE
- ▶ Gentoo
- ▶ Xandros
- ▶ IBM OS/2 Warp 4.5
- ▶ Solaris, OpenSolaris
- ▶ FreeBSD, OpenBSD (ohne Gasterweiterungen!)
- ▶ Novell NetWare (ohne Gasterweiterungen!)
- ▶ die meisten dieser Systeme auch als 64-Bit-Versionen

Mac OS X unter VirtualBox als Gastsystem zu installieren ist zurzeit nicht möglich und laut den Lizenzbestimmungen von Apple auch nicht zulässig. Man kann also sagen, dass unter VirtualBox alle gängigen Betriebssysteme installiert werden können – bis auf Mac OS X.

#### Übersicht über die offiziell unterstützten Systeme

[+]

In den vorangehenden Listen habe ich offiziell und inoffiziell unterstützte Gastsysteme aufgelistet. Eine Übersicht über die offiziell unterstützten Systemen finden Sie auch auf der VirtualBox-Homepage oder im aktuellen Handbuch (in englischer Sprache). Diese Liste kann sich von Version zu Version ändern!

### 3.2.4 Versionen und Lizenzen

Häufig wird in Artikeln geschrieben, VirtualBox sei frei und stehe unter der GPL (*GNU General Public License*). Dies stimmt jedoch nur zum Teil, denn es stehen zwei Versionen von VirtualBox zur Verfügung:

- ▶ VirtualBox OSE (Open Source Edition mit Quellcode)
- ▶ xVM VirtualBox (die Voll- bzw. die Binärversion)

Für die erste der beiden ist der komplette Quellcode einsehbar. Die zweite Version ist nur im Binärformat verfügbar. Sie enthält spezielle Erweiterungen, die nur in kompilierter Form vorliegen.

[+]

Versionen

Aktuelle Infos zu den verschiedenen Versionen finden Sie auf der Homepage von VirtualBox (<http://www.virtualbox.org/wiki/Editions>.)

**VirtualBox Open Source Edition**

Die VirtualBox Open Source Edition steht unter der bekannten GNU General Public License (kurz GPL) und hat somit die folgenden Vorteile:

- ▶ Sie darf ohne Einschränkungen eingesetzt werden – auch von Firmen.
- ▶ Der Quellcode kann eingesehen und verändert (und so von jedem an eigene Bedürfnisse angepasst) werden. Diese Änderungen müssen aber weiter unter der GPL stehen.
- ▶ Die Software darf weiterverbreitet werden.

Viele bekannte Programme stehen ebenfalls unter der GPL. Dabei handelt es sich nicht immer nur um Linux-, sondern auch um Windows-Anwendungen:

- ▶ der Linux-Kernel
- ▶ KDE, GNOME
- ▶ OpenOffice
- ▶ MySQL (ein Datenbankserver)
- ▶ OpenVPN (eine VPN-Software)

Die Open Source Edition bringt jedoch leider auch ein paar Nachteile mit. In dieser Version fehlen einige, teilweise wichtige Funktionen. Zu den Nachteilen zählen unter anderem:

- ▶ VirtualBox OSE ist nur als Quellcode verfügbar und muss von jedem selbst übersetzt bzw. kompiliert werden, was für Anfänger gegebenenfalls sehr kompliziert ist. Ausnahmen bestätigen jedoch die Regel – aber dazu später mehr.
- ▶ Es gibt keinen USB-Support.
- ▶ Es ist kein RDP-Zugriff auf die virtuellen Maschinen möglich (nicht zu verwechseln mit dem RDP-Zugriff auf ein Windows-Gastsystem).



- Der SATA-Controller wird nicht unterstützt.
- Anstelle des Gigabit-Ethernet-Controllers steht nur der 100-MBit-Controller zur Verfügung.

Trotz dieser Nachteile ist die freie Version jedoch für einfache virtuelle Maschinen völlig ausreichend. Die grafische Oberfläche unterscheidet sich auf den ersten Blick nicht von der der Vollversion. Nur im Titel steht statt »xVM VirtualBox« die Bezeichnung »VirtualBox OSE« (siehe Abbildung 3.1). In den Konfigurationseinstellungen einer virtuellen Maschine werden Sie jedoch schnell feststellen, dass die genannten Funktionen fehlen.



**Abbildung 3.1** Die grafische Oberfläche von VirtualBox OSE

### xVM VirtualBox

Die vorher genannten Nachteile der Open Source Edition sind gleichzeitig die Vorteile der Binärversion und umgekehrt. Die Optionen bzw. Funktionen, die in der Quellcode-Version nicht vorhanden sind, stehen nur binär (als closed source) zur Verfügung. Aus diesem Grund steht diese Version von VirtualBox auch unter einer eigenen, proprietären Lizenz – der *VirtualBox Personal Use and Evaluation License* (PUEL). Zwei wichtige Punkte der PUEL-Lizenz sind:

- ▶ xVM VirtualBox kann von Privatleuten und Schulen kostenlos eingesetzt werden.
- ▶ xVM VirtualBox darf für eine »angemessene« Dauer (ein »paar« Wochen) auch von Unternehmen getestet werden.

Der zweite Punkt ist ein wenig schwammig. Man fragt sich, was eine »angemessene« Dauer von ein »paar« Wochen ist. Danach soll man sich laut Lizenz mit Sun in Verbindung setzen, um über einen Support-Vertrag zu verhandeln. Details dazu stehen in §2 der PUEL-Lizenz. Die Lizenz selbst finden Sie, in englischer Sprache, entweder bei der Installation oder unter: [http://www.virtualbox.org/wiki/VirtualBox\\_PUEL](http://www.virtualbox.org/wiki/VirtualBox_PUEL).

Bevor Sie VirtualBox installieren und anwenden, sollten Sie also genau überlegen, welche Funktionen Sie benötigen. Wenn Sie auf die Sonderfunktionen der Binärversion angewiesen sind, müssen Sie auch die PUEL-Lizenz akzeptieren – vor allem, wenn VirtualBox in einem Unternehmen eingesetzt werden soll. In diesem Buch beschreibe ich die Vollversion xVM VirtualBox inklusive der »Sonderfunktionen«.



#### Die wichtigsten Fakten zu VirtualBox

- ▶ VirtualBox ist ein Desktopvirtualisierer und Konkurrent zu VMware Workstation und MS Virtual PC.
- ▶ VirtualBox kann unter den aktuellen Windows- und Linux-Versionen, unter Mac OS X und Solaris installiert werden.
- ▶ Es gibt eine freie (OSE) und eine Closed-Source-Version.
- ▶ Die Closed-Source-Version bietet mehr Funktionen als die Open-Source-Variante.
- ▶ Die PUEL-Lizenz erlaubt die Nutzung der Vollversion in Firmen nur für einen »angemessenen« Zeitraum.

## 3.3 Installation

Die Installation von xVM VirtualBox ist recht einfach. In den folgenden Abschnitten erkläre ich Ihnen die notwendigen Vorbereitungen und zuerst die Installation der Vollversion. Die Installation der GPL-Version folgt dann in Abschnitt 3.3.4, »VirtualBox OSE unter Linux«.

Bevor Sie VirtualBox installieren, sollten Sie noch ein paar Vorbereitungen für das Hostsystem treffen. Das Wichtigste ist, dass Sie sich überlegen, an welchem Ort die virtuellen Maschinen und deren Festplatten-Images gespeichert werden sollen. Weitere Informationen hierzu haben Sie ja bereits in Abschnitt 2.2, »Planung«, erhalten.

### 3.3.1 Download

Nachdem das Hostsystem eingerichtet ist, können Sie die aktuellste VirtualBox-Version von der Homepage herunterladen. Sämtliche Versionen finden Sie unter <http://www.virtualbox.org/wiki/Downloads>. Im oberen Abschnitt der Seite werden die Binärversionen angeboten.

Für Windows-, Mac-OS-X- und Solaris-Systeme steht ein entsprechender Installer zur Verfügung. Linux-Anwender finden auf einer weiteren Seite fertige Pakete für viele gängige Distributionen (siehe Abbildung 3.2).

Download VirtualBox for Linux Hosts

VirtualBox for Linux

**Note:** The package architecture has to match the Linux kernel architecture, that is, if you are running a 64-bit kernel, install the appropriate AMD64 package (it does not matter if you have an Intel or an AMD CPU). Mixed installations (e.g. Debian/Lenny ships an AMD64 kernel with 32-bit packages) are not supported. To install VirtualBox anyway you need to setup a 64-bit chroot environment.

**Note:** The VirtualBox 2.1.0 deb/rpm packages do not properly unregister from dkms before the package is removed / upgraded. Please have a look at our FAQ for further details.

Please choose the appropriate package for your Linux distribution:

- Ubuntu 9.04 ("Jaunty Jackalope") ⇄ i386 | ⇄ AMD64
- Ubuntu 8.10 ("Intrepid Ibex") ⇄ i386 | ⇄ AMD64
- Ubuntu 8.04 LTS ("Hardy Heron") ⇄ i386 | ⇄ AMD64
- Ubuntu 7.10 ("Gutsy Gibbon") ⇄ i386 | ⇄ AMD64
- Ubuntu 6.06 LTS ("Dapper Drake") ⇄ i386 | ⇄ AMD64
- Debian 5.0 ("Lenny") ⇄ i386 | ⇄ AMD64
- Debian 4.0 ("Etch") ⇄ i386 | ⇄ AMD64
- Debian 3.1 ("Sarge") ⇄ i386
- openSUSE 11.1 ⇄ i386 | ⇄ AMD64
- openSUSE 11.0 ⇄ i386 | ⇄ AMD64
- openSUSE 10.3 ⇄ i386 | ⇄ AMD64
- SUSE Linux Enterprise Server 10 (SLES10) ⇄ i386 | ⇄ AMD64
- Fedora 11 ("Leonidas") ⇄ i386 | ⇄ AMD64
- Fedora 9 ("Sulphur") / 10 ("Cambridge") ⇄ i386 | ⇄ AMD64
- Fedora 8 ("Werewolf") ⇄ i386 | ⇄ AMD64
- Fedora 7 ("Moonshine") ⇄ i386 | ⇄ AMD64
- Mandriva 2009.1 ⇄ i386 | ⇄ AMD64
- Mandriva 2008.0 ⇄ i386
- Mandriva 2007.1 ⇄ i386
- Red Hat Enterprise Linux 5 ("RHEL5") / CentOS 5 ⇄ i386 | ⇄ AMD64
- Red Hat Enterprise Linux 4 ("RHEL4") ⇄ i386
- Turbolinux 11 ⇄ i386 | ⇄ AMD64
- PCLinuxOS 2007 ⇄ i386
- Xandros Desktop 4.1 ⇄ i386
- All distributions ⇄ i386 | ⇄ AMD64

Abbildung 3.2 Linux-Pakete für die verschiedenen Distributionen

### 3.3.2 xVM VirtualBox unter Windows

Die Installation unter Windows erfolgt durch eine Setup-Routine oder ein MSI-Paket (Microsoft Installer). Seit Version 1.3.2 läuft VirtualBox auch unter Vista und inzwischen auch auf 64-Bit-Systemen.

#### Für alle Benutzer

Sie starten die Installation durch einen Doppelklick auf die heruntergeladene .exe-Datei. Standardmäßig wird VirtualBox nun für alle Benutzer installiert.

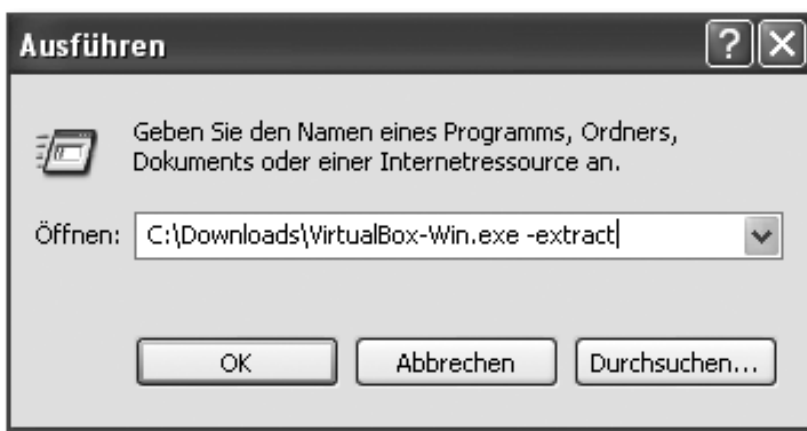
### Nur für den aktuellen Benutzer

Wenn Sie VirtualBox nicht für alle Benutzer zur Verfügung stellen wollen, nehmen Sie die Installation mit dem MSI-Paket vor. Dieses Paket ist in der Setup-Routine enthalten und muss zunächst entpackt werden. Dazu öffnen Sie die Windows-Eingabeaufforderung bzw. öffnen den AUSFÜHREN-Dialog (START • AUSFÜHREN).

Das MSI-Paket entpacken Sie durch Eingabe von

```
PFAD\VirtualBox.exe -extract
```

Ersetzen Sie im Beispiel `PFAD\VirtualBox.exe` durch den Pfad und Dateinamen Ihrer Version (Abbildung 3.3).



**Abbildung 3.3** Das MSI-Paket entpacken

Nach dem Beenden dieses Vorgangs erscheint ein Hinweis mit dem Pfad, in den das Paket extrahiert wurde. Dabei handelt es sich normalerweise um den Ordner für temporäre Dateien. Dieser befindet sich unter Windows XP in Ihrem Heimatverzeichnis unter *Lokale Einstellungen\Temp*. Dabei handelt es sich um einen versteckten Ordner, der in der Standardansicht des Explorers nicht zu sehen ist. Um ihn sichtbar zu machen, aktivieren Sie unter EXTRAS • ORDNEROPTIONEN im Register ANSICHT den Punkt ALLE DATEIEN UND ORDNER ANZEIGEN (siehe Abbildung 3.4).

In dem temporären Ordner befindet sich ein Verzeichnis mit dem Namen *VirtualBox*, in dem mehrere Dateien enthalten sind. Verwenden Sie zur Installation das MSI-Paket, das zu Ihrem Hostsystem passt (entweder Intel x86 oder AMD 64).

Installation nur für den aktuellen Benutzer:

```
msiexec /i PFAD\VirtualBox.msi ALLUSERS=2
```



Abbildung 3.4 Alle Dateien und Ordner im Windows-Explorer anzeigen

Ersetzen Sie im Beispiel PFAD durch das Verzeichnis, in dem sich das MSI-Paket befindet. Wenn der Pfad Leerzeichen enthält, setzen Sie diesen in Anführungszeichen:

Installation *mit Pfadangabe*:

```
msiexec /i C:\Downloads\VirtualBox.msi ALLUSERS=2
```

oder *mit Leerzeichen in der Pfadangabe* (Pfad in Anführungszeichen!):

```
msiexec /i "C:\Dokumente und Einstellungen\dirk\Lokale  
Einstellungen\Temp\VirtualBox.msi" ALLUSERS=2
```

Drag & Drop

Sie können das MSI-Paket per Drag & Drop ins AUSFÜHREN-Fenster ziehen. Dabei wird der komplette Pfad inklusive der Dateinamen übernommen und bei Bedarf in Anführungszeichen gesetzt.

[+]

Installationsverlauf

Als Erstes erscheint nun der übliche Willkommensbildschirm (siehe Abbildung 3.5) – allerdings in englischer Sprache (Stand Version 3.0).



Abbildung 3.5 Start der Installation

Durch einen Klick auf den Button NEXT gelangen Sie zur Anzeige der Lizenzinformationen. Dabei handelt es sich um die bereits erwähnte VirtualBox Personal Use and Evaluation License (PUEL). Abbildung 3.6 zeigt den vorher erwähnten Abschnitt mit der ungenauen Zeitangabe.

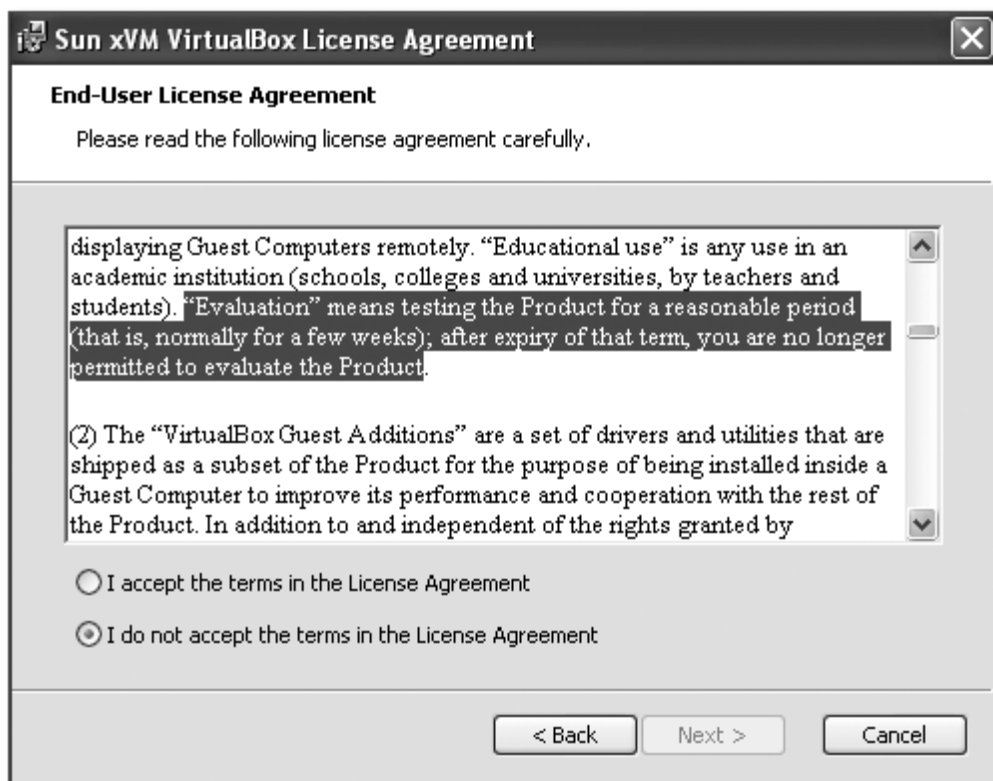


Abbildung 3.6 End-User License Agreement (PUEL)

Wie üblich müssen Sie diese bestätigen bzw. akzeptieren, um die Installation fortzuführen. Ob Sie das gesamte Lizenzabkommen durchlesen, bleibt Ihnen überlassen.

In der Auswahl der Komponenten (Abbildung 3.7) entscheiden Sie, ob Sie die Unterstützung für USB und Netzwerk ebenfalls installieren. Standardmäßig sind diese Funktionen aktiviert, und es gibt meist keine zwingenden Gründe, sie nicht zu installieren. Im Gegenteil – diese sind ja mit ein Grund, warum wir hier nicht VirtualBox OSE verwenden. Der Punkt VIRTUALBOX PYTHON SUPPORT installiert eine Python-API, auf die ich in diesem Buch jedoch nicht weiter eingehe.

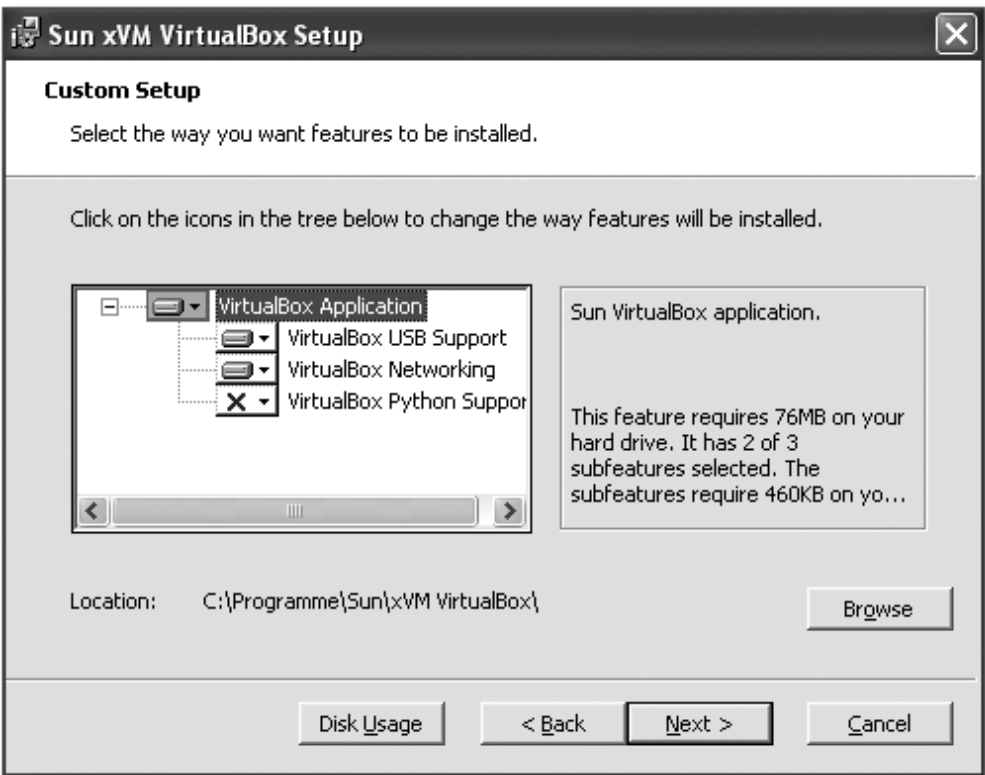


Abbildung 3.7 Auswahl der Programm-Komponenten

Danach legen Sie fest, ob außer im Startmenü auch Verknüpfungen auf dem Desktop oder in der Schnellstartleiste angelegt werden sollen. Diese Verknüpfungen beziehen sich nur auf die grafische Oberfläche.

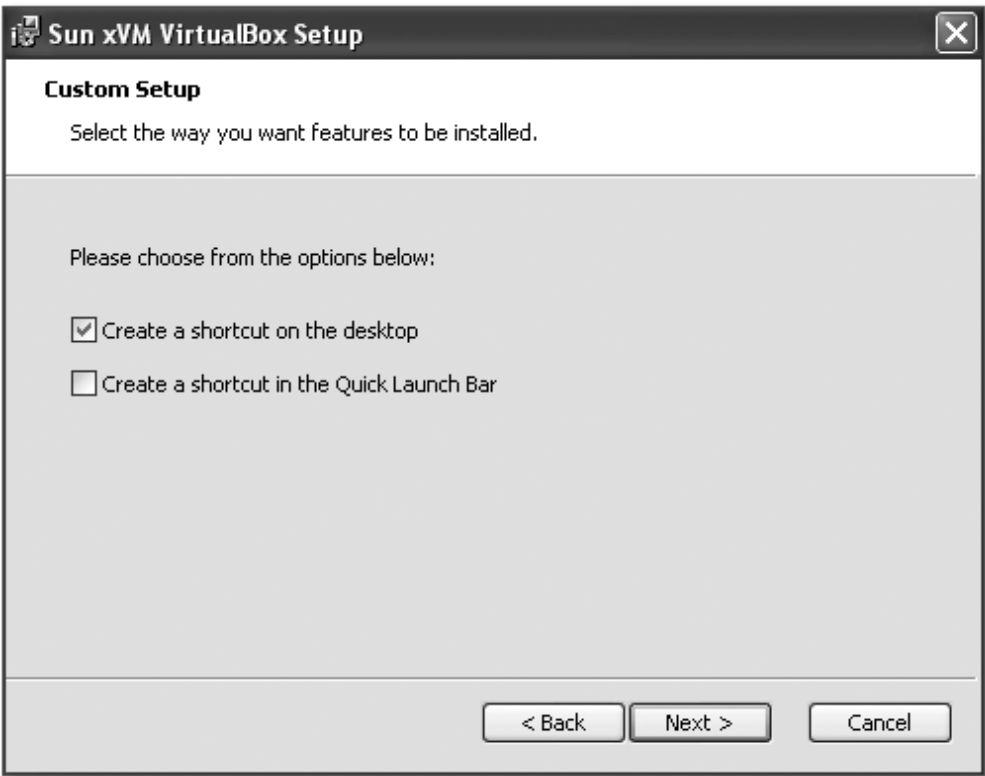


Abbildung 3.8 Verknüpfungen für den Desktop und die Schnellstartleiste erstellen

Im nächsten Schritt wird nun die eigentliche Installation gestartet. Während der Installation werden Sie unter Umständen mehrmals darauf hingewiesen, dass die Software und die Treiber den Windows-Logo-Test nicht bestanden haben (Abbildung 3.9 und Abbildung 3.10). Dies besagt jedoch nur, dass keine Zertifikate von Microsoft vorhanden sind. Privatanwender können diese Meldung getrost ignorieren. Ein Administrator wird VirtualBox sicherlich auf einem Standalone-Rechner einrichten und dürfte sich daher auch keine Sorgen machen. Eine Parallelinstallation neben einem SQL-Server oder Ähnlichem kann ich in einer Firma auf jeden Fall nicht empfehlen, da eine Desktopvirtualisierung nicht auf einen produktiven Server gehört.

[+] Netzwerkprogramme bei Installation stoppen

Bei der Installation wird die Netzwerkverbindung kurz neu gestartet. Achten Sie daher darauf, dass im Hintergrund keine aktiven Netzwerkprogramme laufen.



Abbildung 3.9 Windows-Logo-Test der Software



Abbildung 3.10 Windows-Logo-Test des Netzwerktreibers



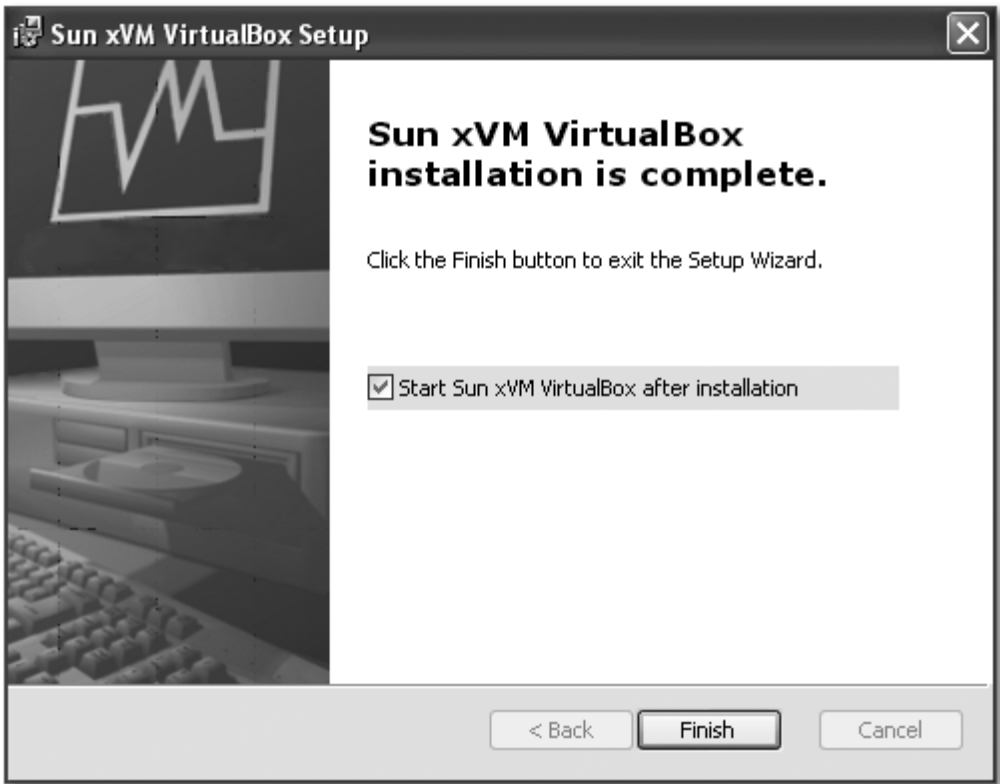


Abbildung 3.11    Ende der Installation

Nach der Installation haben Sie die Möglichkeit, die grafische Oberfläche von VirtualBox direkt zu starten.

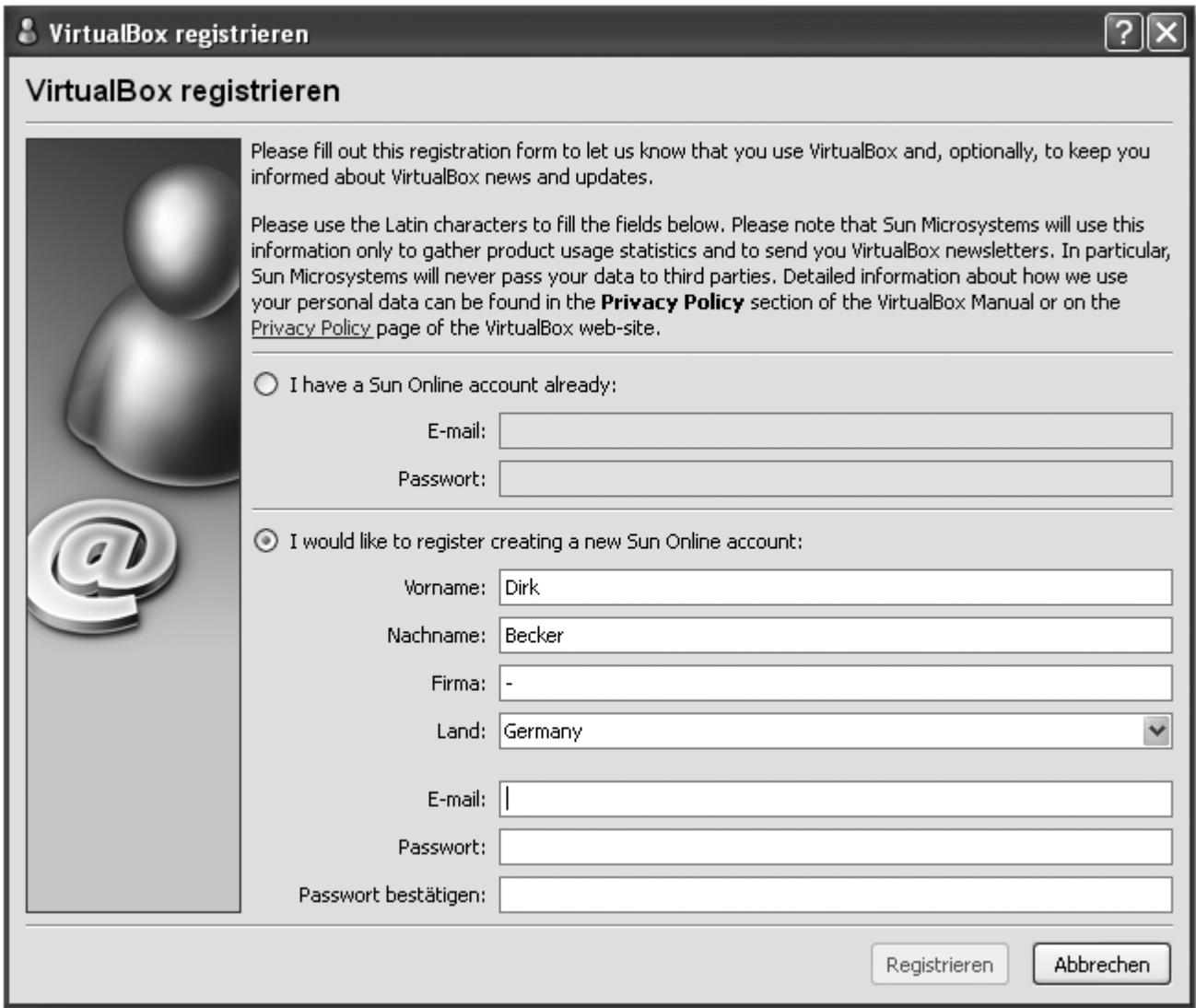


Abbildung 3.12    VirtualBox-Registrierung

Beim ersten Start fragt VirtualBox Sie nach Ihren Namen, Ihrer E-Mail-Adresse usw. und fordert Sie auf, sich bei Sun zu registrieren. Wenn Sie ABBRECHEN wählen, sehen Sie den Dialog zur Registrierung bei späteren Starts wieder.



#### Beachten Sie bei der Installation

- ▶ Standardmäßig wird VirtualBox für alle Benutzer installiert.
- ▶ Zur Installation für einzelne Benutzer dient das MSI-Paket.
- ▶ Alle Treiber von VirtualBox bestehen den Windows-Logo-Test nicht.

### 3.3.3 xVM VirtualBox unter Linux

Zunächst möchte ich Ihnen nahelegen, VirtualBox auf einer aktuellen Distribution, für die auch ein fertiges Paket existiert, zu installieren. Bei älteren Distributionen und Kernel-Versionen kann es später zu verschiedenen Problemen kommen (vorwiegend USB-Geräte und Netzwerk). Auf diese veralteten Probleme werde ich auch nicht weiter eingehen.

Die Installation unter Linux fällt abhängig von der Distribution unterschiedlich aus. VirtualBox setzt dabei unter anderem folgende Bibliotheken voraus, die jedoch bei fast allen Distributionen vorhanden sind:

- ▶ Qt 4.3.0 (oder neuer)
- ▶ SDL 1.2.7 (oder neuer)

Bei einigen Distributionen ist VirtualBox bereits im entsprechenden Paketarchiv enthalten und kann ohne Probleme über den Paketmanager installiert werden. Achtung: Hierbei handelt es sich häufig um die Version VirtualBox OSE! Die eben genannten Bibliotheken werden dann meist, wenn nicht bereits geschehen, durch Abhängigkeitsregeln automatisch mit installiert. Der (kleine) Nachteil an den Distributionsversionen ist jedoch, dass Sie meist 1–2 Versionen »hinterherhinken«. Der Grund ist, dass neue Programmversionen immer zuerst auf Fehler und Ähnliches getestet werden, bevor sie eine Freigabe erhalten und im Paketarchiv zur Verfügung stehen. Auf die Installationen über einen Paketmanager gehe ich nicht weiter ein, da ich davon ausgehe, dass Linux-Benutzer, die dieses Buch lesen, mit Ihrem System vertraut sind.



#### Versionsunterschiede von VirtualBox

Aus lizenzrechtlichen Gründen handelt es sich bei den Versionen in den Linux-Paketquellen häufig um VirtualBox OSE. Wenn Sie xVM VirtualBox einsetzen wollen, verwenden Sie das Paket von der Downloadseite!

Wenn Sie ein Linux-System verwenden, das auf der Distribution Debian basiert, können Sie die Paketquellen für xVM VirtualBox in Ihren Paketmanager eintragen und dadurch immer bequem die aktuelle Version installieren. Dazu tragen Sie die folgende Zeile in die Datei `/etc/apt/sources.list` ein:

```
deb http://download.virtualbox.org/virtualbox/debian X non-free
```

Dabei ersetzen Sie `X` durch den Namen Ihrer Distribution ersetzt werden. Für Ubuntu 8.10 (Intrepid Ibex) lautet der Eintrag zum Beispiel:

```
deb http://download.virtualbox.org/virtualbox/debian\
    intrepid non-free
```

Ab nun steht Ihnen die aktuelle Version über den Paketmanager zur Verfügung. Aus Sicherheitsgründen ist es sinnvoll, den GnuPG-Schlüssel der Paketquelle zu importieren:

```
wget -q http://download.virtualbox.org/virtualbox\
    /debian/sun_vbox.asc -O- | sudo apt-key add -
```

Wenn Sie die aktuelle Version von der Webseite verwenden wollen, laden Sie diese wie in Abschnitt 3.3.1, »Download«, beschrieben herunter. Sollten Sie kein passendes Paket für Ihre Distribution finden, wählen Sie den Link *All distributions*. Ich gehe im Folgenden davon aus, dass Sie die Installation mit dem Tool *sudo* (*su »do«*, <http://www.courtesan.com/sudo/>) vornehmen; alternativ können Sie das Programm auch als Benutzer Root einspielen:

- ▶ **SUSE Linux Enterprise:** `sudo yast -i VirtualBox-VERSION.rpm`
- ▶ **openSUSE:** `sudo yast -i VirtualBox-VERSION.rpm`
- ▶ **Debian:** `sudo apt-get install VirtualBox-VERSION.deb`
- ▶ **Ubuntu:** `sudo apt-get install VirtualBox-VERSION.deb`
- ▶ **Fedora:** `sudo yum install VirtualBox-VERSION.rpm`
- ▶ **All distributions (alle Distributionen):** `sudo ./VirtualBox-VERSION-Linux_x86.run`

Das zuletzt gezeigte Kommando startet ein Installationsskript, das die binäre Version einspielt. Wenn Sie dieses installieren, »weiß« Ihr Paketmanager nichts davon. Dadurch kann es später bei der Installation anderer Pakete eventuell zu Komplikationen oder Abhängigkeitsproblemen kommen, die der Paketmanager normalerweise selbst erkennen würde.

Falls es bei der Installation zu Problemen kommt, liegt das meist an fehlenden Abhängigkeiten. Lesen Sie in dem Fall die Fehlermeldung genauestens durch und in-

stallieren Sie dann die notwendigen Pakete. Häufig benötigt das System die folgenden zusätzlichen Komponenten:

- ▶ die *GNU Compiler Collection* (GCC)
- ▶ die Kernel-Header
- ▶ das Programm `make`

Bei der Installation wird (normalerweise) eine neue Gruppe mit dem Namen *vboxusers* angelegt. In diese Gruppe nehmen Sie alle Benutzer auf, die VirtualBox verwenden. Dies kann entweder über Ihren Benutzermanager oder wie folgt in der Konsole geschehen:

### Gruppe *vboxusers*

- ▶ Die Gruppe erstellen (einmalig – falls sie bereits existiert, erscheint eine entsprechende Meldung):  
`sudo groupadd vboxusers`
- ▶ Den Benutzer der Gruppe zuordnen:  
`sudo usermod -G vboxusers dirk`



### Was Sie sich für die Installation unter Linux merken sollten

- ▶ Die Vollversion von VirtualBox steht meist nicht in den Paketarchiven der Distributionen zur Verfügung.
- ▶ Verwenden Sie möglichst das Paket, das zu Ihrer Distribution passt.
- ▶ Bei der Installation der *All-distributions*-Version kann es später zu Abhängigkeitsproblemen kommen.
- ▶ Sämtliche VirtualBox-Benutzer müssen der Gruppe *vboxusers* angehören.

### 3.3.4 VirtualBox OSE unter Linux

Wie bereits erwähnt, bieten viele Linux-Distributionen standardmäßig die Open Source Edition in den Archiven. In Abbildung 3.13 sehen Sie den Paketmanager Synaptic, der unter anderem bei Debian und Ubuntu zum Einsatz kommt. Die Abbildung zeigt, dass bereits eine xVM-VirtualBox-2.1-Version händisch installiert wurde (zu erkennen an dem farbig markierten Kontrollkästchen). Die markierte Zeile im Screenshot zeigt außerdem das installierbare Paket für VirtualBox OSE. In Abschnitt 3.2.4, »Versionen und Lizenzen«, erwähnte ich, dass man den Quellcode selbst übersetzen muss, und schrieb auch: »Ausnahmen bestätigen die Regel.« Hier haben wir eine solche Ausnahme. Synaptic installiert die OSE-Version, ohne dass Sie sich um die Kompilierung des Quellcodes kümmern müssen.



Abbildung 3.13 xVM VirtualBox und VirtualBox OSE unter Ubuntu

Wenn Ihre Distribution kein OSE-Paket anbietet und/oder Sie die aktuellste Version verwenden wollen, müssen Sie den Quellcode selbst kompilieren. Das funktioniert unter Windows, Linux, Mac OS X und unter Solaris. (Die Kompilierung ist auch unter OS/2 und FreeBSD möglich.) Ich werde im Folgenden die Kompilierung unter Linux erklären.

Für weiterführende Informationen lesen Sie unbedingt auch die *Build instructions* unter [http://www.virtualbox.org/wiki/Build\\_instructions](http://www.virtualbox.org/wiki/Build_instructions). Dort wird in wenigen Schritten erklärt, wie Sie die Kompilierung vornehmen – diese Beschreibung ist immer aktuell. Den Quellcode finden Sie auf der in Abschnitt 3.3.1, »Download«, angegebenen Downloadseite von VirtualBox (siehe Abbildung 3.14).

### VirtualBox Open Source Edition (OSE)

For a detailed explanation about the differences between **VirtualBox** and **VirtualBox OSE**, please take a look at the Editions page. The **VirtualBox OSE** sources are available free of charge under the terms and conditions of the GNU General Public License, Version 2. By downloading from the below links, you agree to these terms and conditions.

- **Open Source Edition (OSE) Source code**  
This is a snapshot of the OSE sources at the time of the stable release.
- Browse the source code repository  
This is the current development code, which is not necessarily stable.
- View the latest source code changes
- Checking out from our Subversion server.**

```
svn co http://www.virtualbox.org/svn/vbox/trunk vbox
```

This is the current development code, which is not necessarily stable.

**Note:** The repository was recently moved. If svn update aborts with an error message regarding a moved repository, then do

```
svn switch --relocate \  
http://virtualbox.org/svn/vbox/trunk \  
http://www.virtualbox.org/svn/vbox/trunk
```

After getting the sources in one of the ways listed above, you should have a look at the build instructions.

Please note that the Open Source Edition does not include an installer or setup utilities, as it is mainly aimed at developers and Linux distributors. We do request that anyone intending to redistribute the Open Source Edition ➤ contact us first. Please also take a look at our licensing FAQ, in particular regarding the use of the name VirtualBox.

Abbildung 3.14 VirtualBox OSE (Open Source Edition) selbst kompilieren



#### »Build Instructions« lesen

Lesen Sie unbedingt auch die »Build instructions«, da sich die folgenden Angaben bei jeder neuen Version ändern können!

Nun prüfen Sie, ob alle für die Kompilierung notwendigen Pakete installiert sind.

- ▶ GCC 3.3.3 oder höher
- ▶ as86
- ▶ bcc
- ▶ IASL (Intel ACPI compiler)
- ▶ xsltproc (libxslt)
- ▶ libxml2
- ▶ libxslt1
- ▶ libXcursor
- ▶ Qt 4.4.3 oder höher
- ▶ libIDL
- ▶ libSDL
- ▶ ALSA
- ▶ PulseAudio
- ▶ libhal
- ▶ libcap
- ▶ Python 2.5

Für 64-Bit-Systeme werden zusätzlich folgende Pakete benötigt:

- ▶ ia32-libs
- ▶ libc6-dev-i386 (libc6)
- ▶ lib32gcc1
- ▶ gcc-multilib
- ▶ lib32stdc++6 (libstdc++)
- ▶ g++-multilib

Kompilieren Sie wie nun den Code, wie in den folgenden Schritten beschrieben:

- ▶ Entpacken Sie den Quellcode in ein beliebiges Verzeichnis.
- ▶ Öffnen Sie die Konsole.
- ▶ Wechseln Sie in der Konsole in das entsprechende Verzeichnis.

- ▶ Bereiten Sie mit folgendem Skript die Kompilierung vor:  
`./configure --disable-hardening`
- ▶ Setzen Sie die Umgebungsvariablen:  
`source ./env.sh`
- ▶ Stoßen Sie den Kompiliervorgang an:  
`kmk all`
- ▶ Die Binärdateien finden Sie danach im Unterverzeichnis  
`out/linux.x86/release/bin/`.
- ▶ Zum Installieren wechseln Sie nun in dieses Unterverzeichnis und starten dort die Installation der notwendigen Kernel-Module:  
`cd out/linux.x86/release/bin/src`  
`make`  
`sudo make install`

Wenn es beim Befehl `configure` oder der kompilierten Version zu Problemen kommt, können Sie mit dem Befehl `./configure --help` weitere Informationen aufrufen. Geläufige Parameter für `configure` sind unter anderem:

- ▶ `--disable-alsa`: ohne ALSA-Treiber
- ▶ `--disable-pulse`: ohne PulseAudio-Treiber

Nach dem Kompilieren und der Installation steht Ihnen die OSE-Version zur Verfügung. Eventuell müssen Sie von Hand Einträge in den KDE- und GNOME-Startmenüs erstellen. Beachten Sie, dass alle VirtualBox-Benutzer Mitglied der Gruppe `vboxusers` sein müssen (siehe Abschnitt 3.3.3, »xVM VirtualBox unter Linux«).

#### Bei der Installation von VirtualBox OSE sollten Sie diese Punkte beachten

- ▶ In den Paketarchiven der Linux-Distributionen wird meist »nur« die OSE-Version angeboten.
- ▶ Setzen Sie die OSE-Version ein, können Sie viele der späteren Praxisbeispiele nicht nachvollziehen.
- ▶ Sämtliche VirtualBox-Benutzer müssen der Gruppe `vboxusers` angehören.

[!]

## 3.4 Die erste virtuelle Maschine

Nun, da die Installation von VirtualBox abgeschlossen ist, werden wir die erste virtuelle Maschine erstellen. Bei dieser handelt es sich jedoch nur um eine Testumgebung, mit der Sie die nächsten Abschnitte besser nachvollziehen können. Auf die Installation eines Gastsystems werden wir daher an dieser Stelle verzichten.

[+]

**Hinweis zur Verwendung der grafischen Benutzeroberfläche in diesem Abschnitt**

Sie werden in den folgenden Schritten die grafische Oberfläche verwenden. Eine detaillierte Beschreibung dieser GUI erfolgt jedoch erst in Abschnitt 3.5, »Grafische Oberfläche«.

3.4.1 Assistenten aufrufen

Starten Sie die grafische Oberfläche von VirtualBox, und wählen Sie als Erstes den Menüpunkt MASCHINE • NEU, oder klicken Sie auf den entsprechenden Button in der Symbolleiste (Abbildung 3.15).

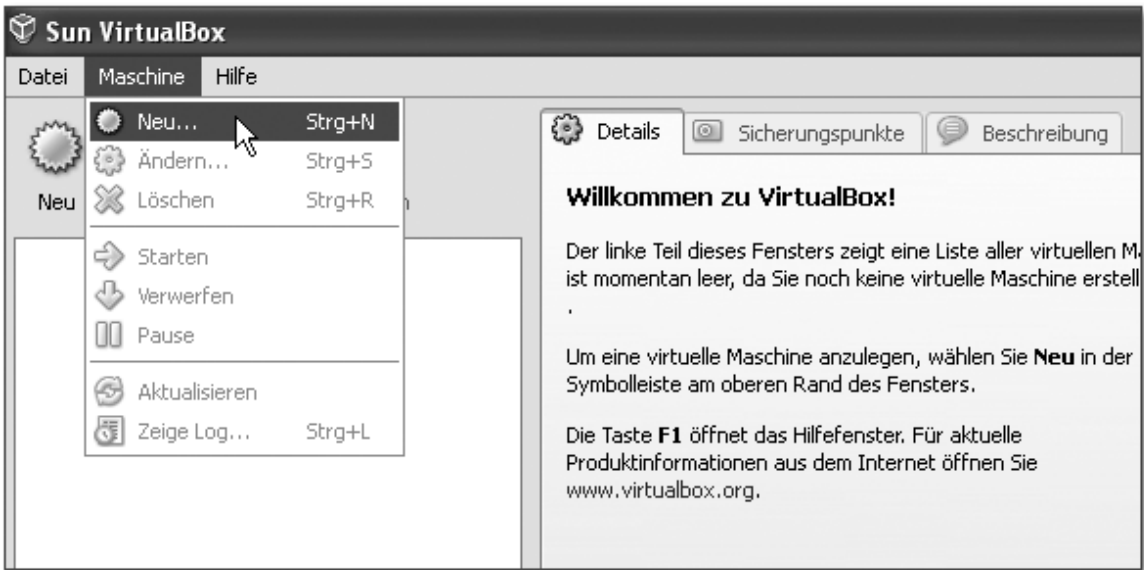


Abbildung 3.15 Eine neue Maschine erstellen

Danach öffnet sich der Assistent, der Sie durch die einzelnen Schritte führt (Abbildung 3.16).

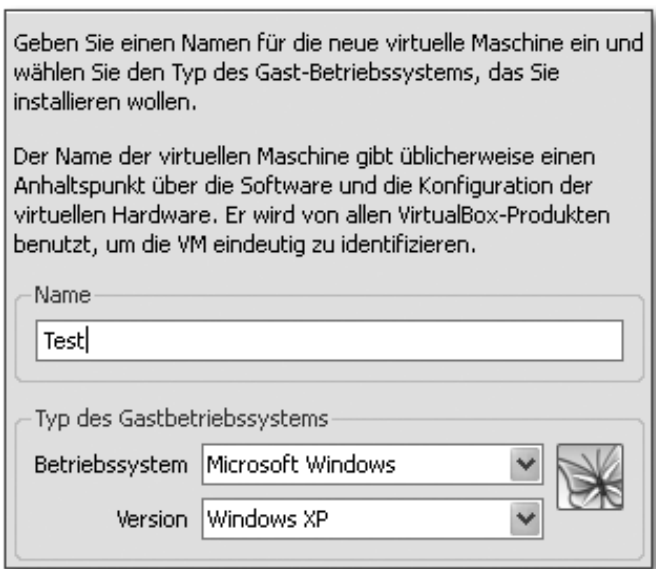


Abbildung 3.16 Der Assistent zum Erstellen einer virtuellen Maschine



### 3.4.2 Name und Gastsystem

Im zweiten Schritt legen Sie den Namen der virtuellen Maschine fest und wählen das Gastsystem aus, das installiert werden soll. Da wir in diesem Beispiel noch kein System installieren, ist diese Auswahl für den Moment nicht wichtig.



**Abbildung 3.17** Den Namen und Typ des Gastsystems der virtuellen Maschine wählen.

Nennen Sie die Maschine wie im Beispiel »Test«, und wählen Sie als Gastsystem WINDOWS XP.

### 3.4.3 Hauptspeicher

Als Nächstes vergeben Sie den zu verwendenden Hauptspeicher. Auf meinem Beispielsystem schlägt VirtualBox 192 MByte vor. Dieser Vorschlag kann, je nach Hostsystem, anders aussehen. Übernehmen Sie für dieses Beispiel die vorgeschlagenen 192 MByte.



**Abbildung 3.18** Die Speichergröße festlegen

### 3.4.4 Festplatten-Image erzeugen

Nun erzeugen wir eine virtuelle Festplatte in Form eines Festplatten-Images. Wählen Sie, wie in Abbildung 3.19 gezeigt, den Punkt FESTPLATTE ERZEUGEN.

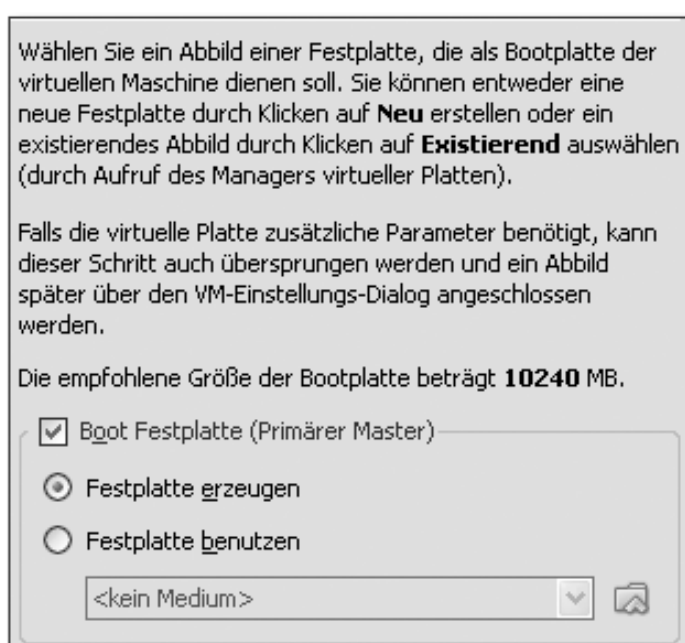


Abbildung 3.19 Ein Festplatten-Image erzeugen

### 3.4.5 Dynamisch wachsendes Medium

Bei der Erstellung eines Festplatten-Images geben Sie an, ob es sich um ein dynamisch wachsendes Medium oder um eines mit fester Größe handeln soll. Aktivieren Sie hier **DYNAMISCH WACHSENDES MEDIUM**. Dadurch wird nur eine kleine Datei erstellt, die sich erst bei Bedarf vergrößert.

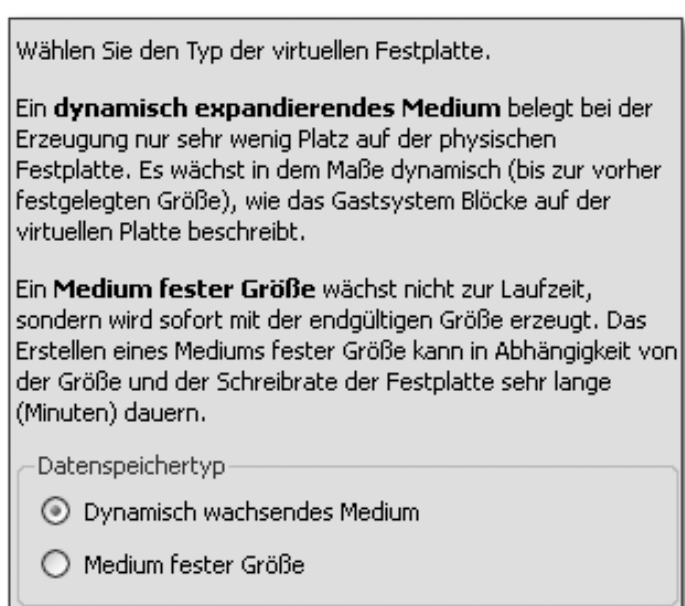


Abbildung 3.20 Den Typ der virtuellen Festplatte wählen

### 3.4.6 Speicherort und Größe

Im sechsten Schritt legen Sie den Speicherort und die Größe des Festplatten-Images fest. Einen besonderen Speicherort brauchen Sie nicht zu vergeben. Ändern Sie den von VirtualBox vorgeschlagenen Namen »Test« in »test\_system«. Die Größe spielt keine Rolle, da wir keine Daten darauf ablegen.

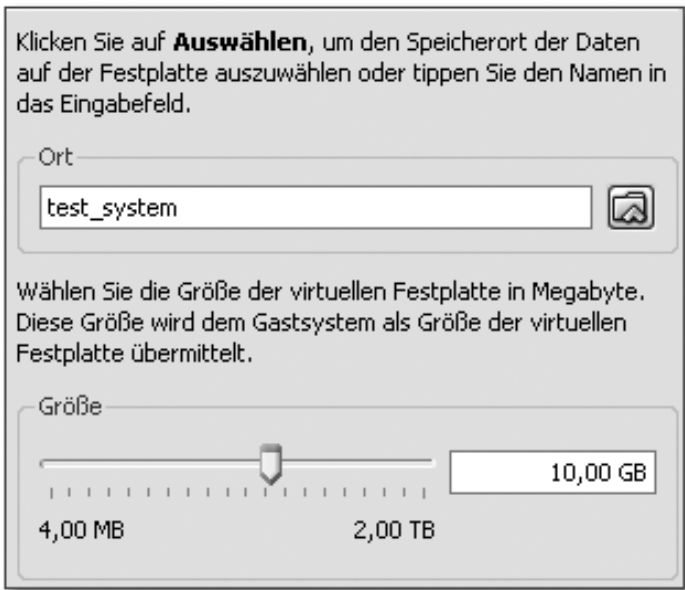


Abbildung 3.21 Den Ablageort und die Größe des Festplatten-Images wählen

3.4.7 Zusammenfassung Festplatten-Image

Nun erscheint die Zusammenfassung zur Erstellung des Festplatten-Images. Klicken Sie auf den Button FERTIG, und der Assistent erstellt das Image.

3.4.8 Zusammenfassung virtuelle Maschine

Vor der Fertigstellung sehen Sie die Zusammenfassung der Parameter für die virtuelle Maschine.

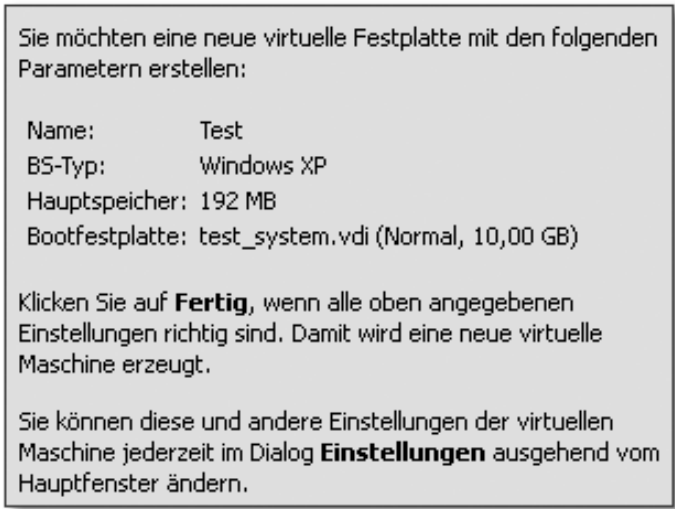


Abbildung 3.22 Zusammenfassung der Einstellungen für die neue virtuelle Maschine

3.4.9 Fertige virtuelle Maschine

Anschließend erscheint die Maschine *Test* in der Auflistung der grafischen Oberfläche.

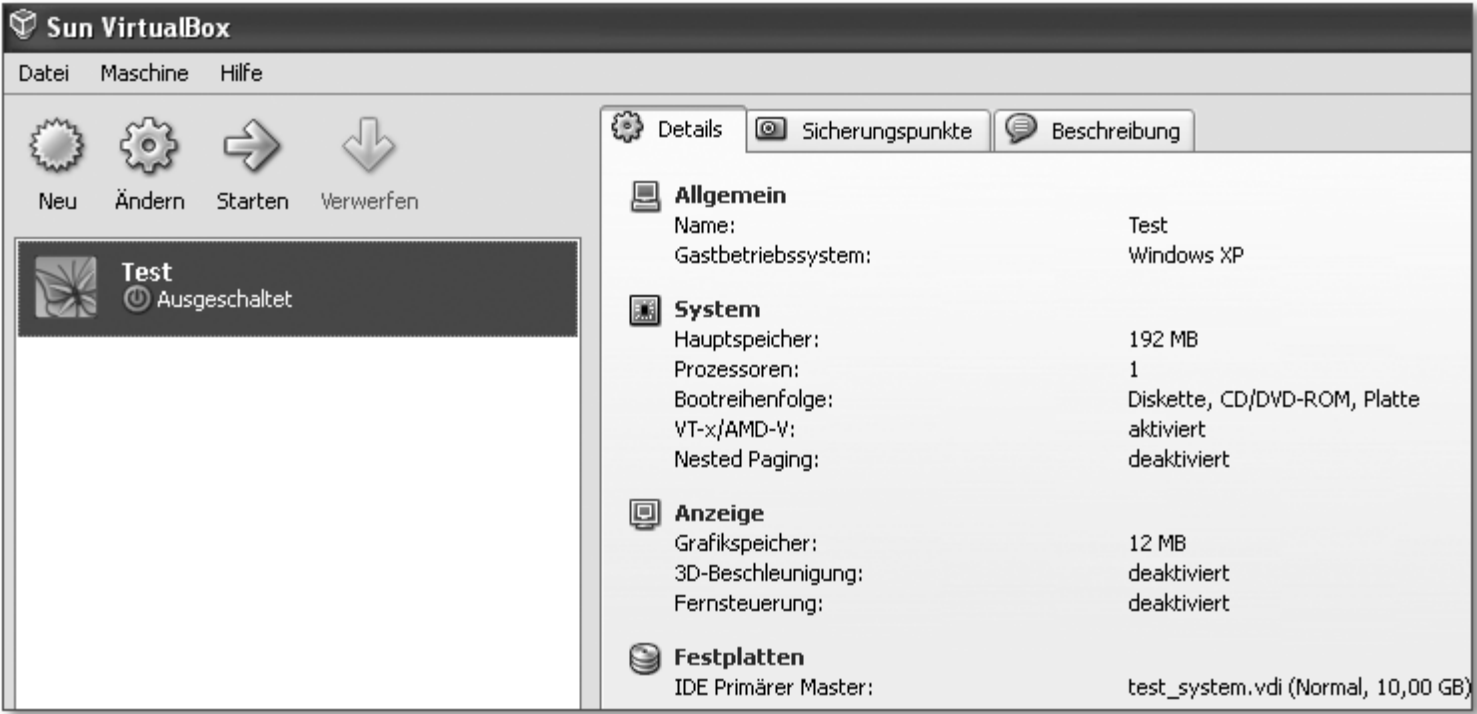


Abbildung 3.23 Die Maschine »Test« in der grafischen Oberfläche

### 3.5 Grafische Oberfläche

Wie die meisten Virtualisierungsprogramme bietet auch VirtualBox eine grafische Oberfläche (GUI, *Graphical User Interface*), mit der Sie die virtuellen Maschinen bequem verwalten. Diese haben Sie kurz beim Erstellen der Maschine »Test« kennengelernt. In diesem Abschnitt werde ich Ihnen die Oberfläche nun im Detail erläutern.

#### 3.5.1 Aufbau

Die grafischen Oberflächen der verschiedenen möglichen Hostsysteme (Windows, Linux, Solaris usw.) sind fast identisch. Die Unterschiede sind nur durch die Betriebssysteme bedingt und betreffen unter anderem die Laufwerksbuchstaben und Pfade.

In Abbildung 3.24 sehen Sie die grafische Oberfläche unter einem Windows-XP-Hostsystem mit einer ersten virtuellen Maschine. Abbildung 3.25 zeigt zum Vergleich eine Installation unter Ubuntu.

Leider können Sie mit der grafischen Oberfläche nicht alle Funktionen von VirtualBox verwenden und müssen daher gelegentlich auf die Kommandozeilentools (siehe Abschnitt 3.6, »Kommandozeilentools«) ausweichen.

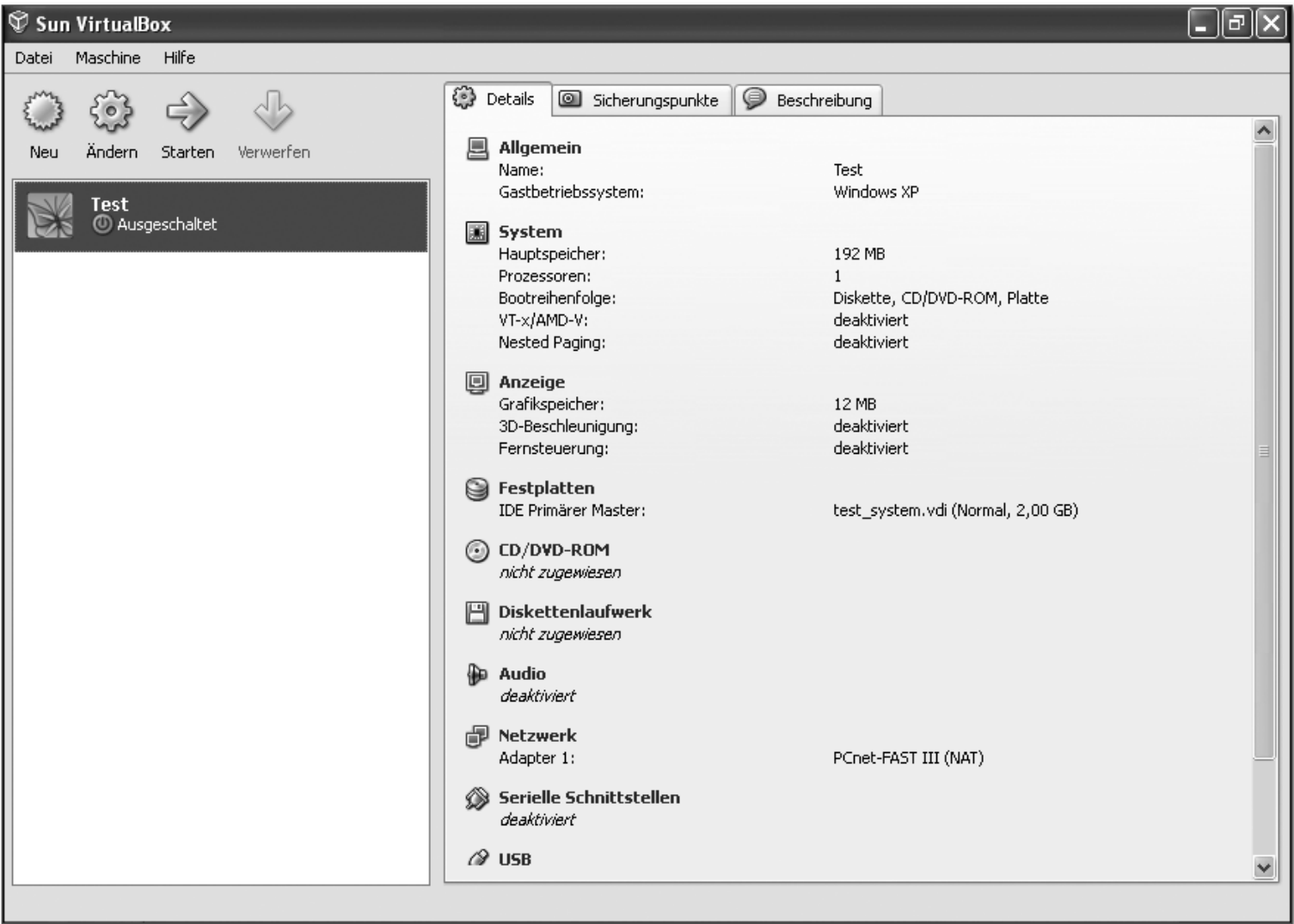


Abbildung 3.24 Grafische Oberfläche unter Windows XP



Abbildung 3.25 Die grafische Oberfläche von VirtualBox unter Ubuntu

Das Fenster der grafischen Oberfläche besteht aus vier Bereichen; es enthält

- ▶ das Benutzermenü,
- ▶ die Symbolleiste zum Erstellen, Starten und Stoppen der virtuellen Maschinen,
- ▶ eine Auflistung der registrierten Maschinen und
- ▶ einen Infobereich für die Detailansichten, Sicherungspunkte und Beschreibungen.

3.5.2 Globale Einstellungen

Vor der Verwendung von VirtualBox sollten Sie noch ein paar Änderungen an den Einstellungen vornehmen. Diese finden Sie im Menü DATEI • GLOBALE EINSTELLUNGEN der grafischen Oberfläche.

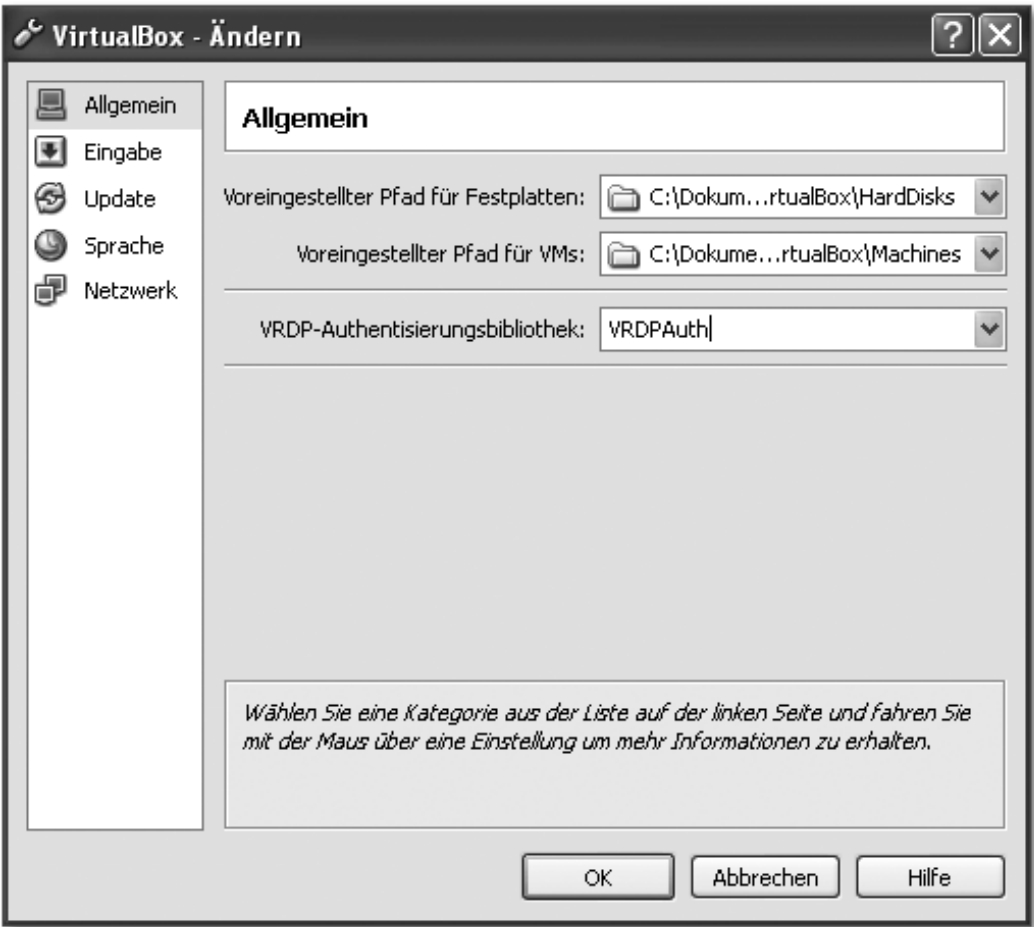


Abbildung 3.26 Das Konfigurationsfenster unter Windows XP



Hilfe durch Tooltips

Im unteren Teil des Fensters wird immer eine Kurzbeschreibung der Einstellung eingeblendet, über der sich der Mauszeiger gerade befindet. Dies ist besonders für Einsteiger sehr hilfreich, da manche Menüpunkte nicht unbedingt selbsterklärend sind.

Allgemein

Im Register ALLGEMEIN legen Sie den Speicherort der virtuellen Festplatten (Festplatten-Images) und der virtuellen Maschinen (VMs) fest. Wenn Sie meinen Emp-

fehlungen in Abschnitt 2.2.1, »Hostsystem«, gefolgt sind und für die virtuellen Maschinen eine eigene Partition bzw. einen eigenen Speicherort bereitstellen, tragen Sie diesen hier ein. Dazu öffnen Sie das Kombinationsfeld und gehen zum Punkt ÄNDERN. Danach wählen Sie den neuen Speicherort aus.

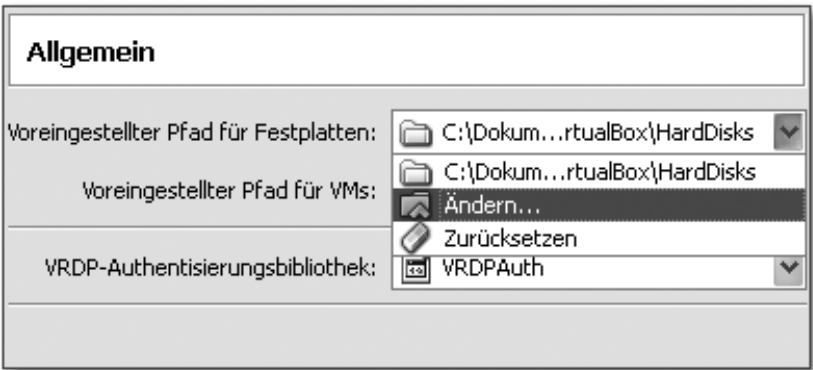


Abbildung 3.27 Den Speicherort ändern

Speicherort von Festplatten-Images

Ein Festplatten-Image wird (anders als bei VMware) *nicht* in dem Verzeichnis der virtuellen Maschine gespeichert. VirtualBox trennt die Festplatten-Images von den Maschinendaten. Im Ordner für die virtuellen Maschinen sind nur die Konfigurationsdateien, Logbücher usw. abgelegt.

[+]

Eingabe und Host-Taste

VirtualBox verwendet eine sogenannte Host-Taste, um spezielle Befehle an die virtuellen Maschinen zu senden (Abbildung 3.28). Standardmäßig verwendet das Programm dazu die rechte `Strg`-Taste. Es gibt eigentlich nur wenige Gründe, diese Einstellung zu ändern.

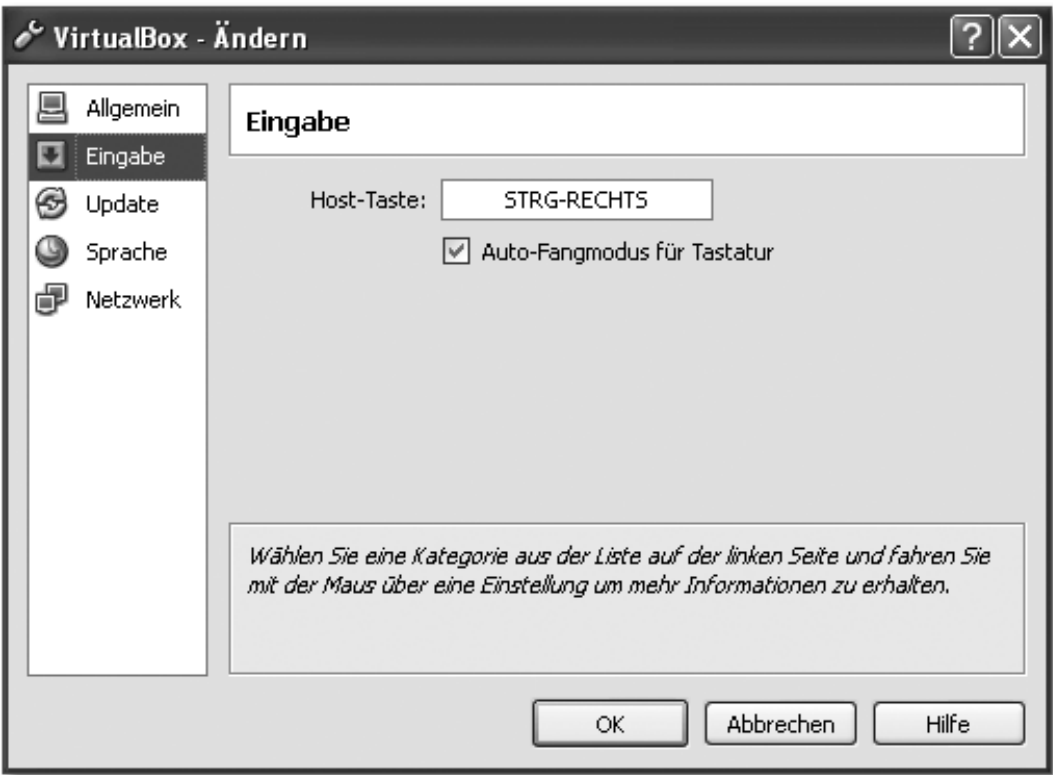


Abbildung 3.28 Die Host-Taste einstellen

Ein wichtiges Einsatzgebiet der Host-Taste ist die Freigabe der Tastatur und der Maus. Wenn eine gestartete virtuelle Maschine den Fokus erhält, werden die Maus- und Tastatureingaben an diese umgeleitet. Ohne installierte Gasterweiterungen (siehe Abschnitt 3.10, »Gasterweiterungen«) sind der Mauszeiger und die Tastatur »gefangen«, und Sie befreien die Eingabegeräte durch Betätigen der Host-Taste. Weitere Tastenkombinationen in Verbindung mit der Host-Taste sind:

- ▶ **[F]**: wechselt zwischen Vollbild und Fensteransicht
- ▶ **[H]**: ACPI-Power-Button
- ▶ **[N]**: Sicherungspunkt im Hintergrund erstellen
- ▶ **[P]**: Pause/Weiter (Maschine anhalten und wieder fortführen)
- ▶ **[Q]**: Power off
- ▶ **[R]**: Reset
- ▶ **[S]**: Ruhezustand (Zustand speichern und Maschine beenden)
- ▶ **[Entf]**: sendet die Tastenkombination **[Strg] + [Alt] + [Entf]**
- ▶ **[F1] – [F12]**: sendet die Tastenkombination **[Strg] + [Alt] + [Fx]**

Normalerweise fängt VirtualBox die Tastatur erst, wenn Sie mit der Maus ins Programmfenster klicken. Durch Aktivieren der Option AUTO-FANGMODUS FÜR DIE TASTATUR werden die Eingaben bereits umgeleitet, wenn das Fenster den Fokus erhält – ohne dass die Tastatur gefangen wird. Allerdings funktionieren manche Tasten und Tastenkombinationen erst, wenn die Tastatur gefangen ist.

### Update und Sprache

Die beiden Register UPDATE und SPRACHE sind eigentlich selbsterklärend. Mit UPDATE können Sie festlegen, ob und in welchen Intervallen VirtualBox nach neuen Versionen sucht, und in der Abteilung SPRACHE stellen Sie die verwendete Menüsprache ein.

### Netzwerk

Das Register NETZWERK dient dazu, Host-only Netzwerke zu verwalten. Details hierzu folgen in Abschnitt 3.9, »Netzwerk«.



#### Einstellungen, Images und Host-Taste

- ▶ In den globalen Einstellungen geben Sie den Speicherort der virtuellen Maschinen an.
- ▶ Die Festplatten-Images und die Konfiguration legt VirtualBox in getrennten Verzeichnissen ab.
- ▶ Die Host-Taste ist wichtig, um verschiedene Funktionen aufzurufen.
- ▶ Bei Gastsystemen, die noch keine Gasterweiterungen installiert haben, können die Tastatur und die Maus nur mit der Host-Taste freigegeben werden.





**Abbildung 3.29** Host-only Netzwerkeinstellungen

### Konfigurationsdatei

Die globalen Einstellungen speichert VirtualBox in einer Konfigurationsdatei. Es handelt sich um eine Textdatei im XML-Format, und Sie können sie beispielsweise mit einem Browser betrachten. Zum Bearbeiten verwenden Sie einen Texteditor.

#### Empfehlung: Notepad++

[+]

Der Editor muss in der Lage sein, die Zeilenwechselzeichen von Linux korrekt zu interpretieren. Der Windows-Editor Notepad ist dazu nicht in der Lage und zeigt die komplette Datei in einer einzigen Zeile an. Eine gute und kostenlose Alternative für Windows-Benutzer ist Notepad++ (<http://notepad-plus.sourceforge.net/de/site.htm>).

Die Einrichtungsdatei befindet sich nach der Installation jeweils im folgenden Verzeichnis:

- ▶ Linux: */home/BENUTZER/.VirtualBox*
- ▶ Windows XP: *C:\Dokumente und Einstellungen\BENUTZER\.VirtualBox*
- ▶ Windows Vista: *C:\Users\BENUTZER\.VirtualBox*

Auf allen Systemen trägt sie den Namen *VirtualBox.xml*. In der Konfigurationsdatei finden Sie die globalen Einstellungen, sämtliche registrierte Maschinen, die Festplatten-Images und CD/DVD-Abbilder.

Im folgenden Listing sehen Sie eine *VirtualBox.xml*-Datei eines Linux-Hostsystems mit zwei der in den Praxisbeispielen folgenden, virtuellen Maschinen und verschiedenen CD/DVD-Abbildern:

```
<?xml version="1.0"?>
<!-- Automatically converted from version '1.5-linux' -->
<!-- Automatically converted from version '1.4-linux' -->
<!-- Automatically converted from version '1.3-linux' -->
<!-- Sun xVM VirtualBox Global Configuration -->
<VirtualBox xmlns="http://www.innotek.de/VirtualBox-settings"
  version="1.6-linux">
  <Global>
    <ExtraData>
      <ExtraDataItem name="GUI/LicenseAgreed" value="7"/>
      <ExtraDataItem name="GUI/RegistrationData"
        value="triesLeft=0"/>
    </ExtraData>
    [...]
    <MachineRegistry>
      <MachineEntry uuid="{07a4350c-2285-478a-489b-4949ce3d970}"
        src="/media/vms/vbox/VM/LiveCD/LiveCD.xml" />
      <MachineEntry uuid="{626160c1-58f6-4d49-b1a9-909ab5c023d}"
        src="/media/vms/vbox/VM/WinXP/WinXP.xml" />
    </MachineRegistry>
    <MediaRegistry>
      <HardDisks>
        <HardDisk uuid="{a394a5c3-1970-401a-31b6-6baae6b8fc5}"
          location="/media/vms/vbox/VM/WinXP/WinXP_system.vdi"
          format="VDI" type="Normal">
        </HardDisk>
      </HardDisks>
      <DVDImages>
        <Image uuid="{4100527e-5d65-4d5b-e5b3-db8309de0ec4}"
          location="/usr/share/virtualbox/VBoxGuestAdditions.iso" />
        <Image uuid="{c41316d4-0bbd-4b0b-8612-8799a14c6437}"
          location="/media/daten/Parted Magic.iso" />
        <Image uuid="{17e24564-a6f8-41db-8f54-72e400b944d7}"
          location="/media/daten/ubuntu-8.10-desktop-i386.iso" />
      </DVDImages>
      <FloppyImages>
        <Image uuid="{9960685f-f529-4234-850d-604a6126b793}"
          location="/media/vms/vbox/VDI/floppy.img" />
      </FloppyImages>
    </MediaRegistry>
    <USBDeviceFilters/>
    <SystemProperties
```

```
defaultMachineFolder="/media/vms/vbox/VM"
defaultHardDiskFolder="/media/vms/vbox/VDI"
defaultHardDiskFormat="VDI"
remoteDisplayAuthLibrary="VRDPAuth"
webServiceAuthLibrary="VRDPAuth"
HWVirtExEnabled="true"
LogHistoryCount="3"/>
</Global>
</VirtualBox>
```

**Listing 3.1** VirtualBox.xml

Diese Konfigurationsdatei ist sehr wichtig. Sie sollten sie daher regelmäßig, zumindest aber vor einer Neuinstallation, sichern (siehe Abschnitt 5.18, »VirtualBox sichern«. Wenn es Probleme mit VirtualBox gibt, können Sie die Datei auch löschen. VirtualBox legt dann beim nächsten Start eine neue an. Dies ist jedoch die absolute Notlösung, denn dadurch gehen alle Informationen über die virtuellen Maschinen, Festplatten-Images und ISO-Abbilder verloren. Die danach folgende Neuregistrierung ist sehr mühselig.

Konfigurationsdatei

Die Konfigurationsdatei trägt den Namen *VirtualBox.xml* und befindet sich im Unterordner *.VirtualBox* im Heimatverzeichnis der Benutzer.

[!]

### 3.5.3 Manager für virtuelle Medien

Der Manager für virtuelle Medien verwaltet sämtliche registrierten Festplatten-Images, CD/DVD-Abbilder (ISO-Images) und Diskettenabbilder. Sie öffnen diesen Manager über das Benutzermenü DATEI • MANAGER FÜR VIRTUELLE MEDIEN.

In Abbildung 3.30 sehen Sie unser Festplatten-Image mit dem Namen *test\_system.vdi*. Im unteren Teil des Fensters blendet VirtualBox eine Kurzinformation zum Image ein. Diese verrät, wo der Ablageort ist, welchen Typ das Image hat und welcher virtuellen Maschine es zurzeit zugewiesen ist. Der Typ gibt an, dass es sich um eine »normale« Festplatte handelt. Einen anderen Typ können Sie hier nicht einstellen.

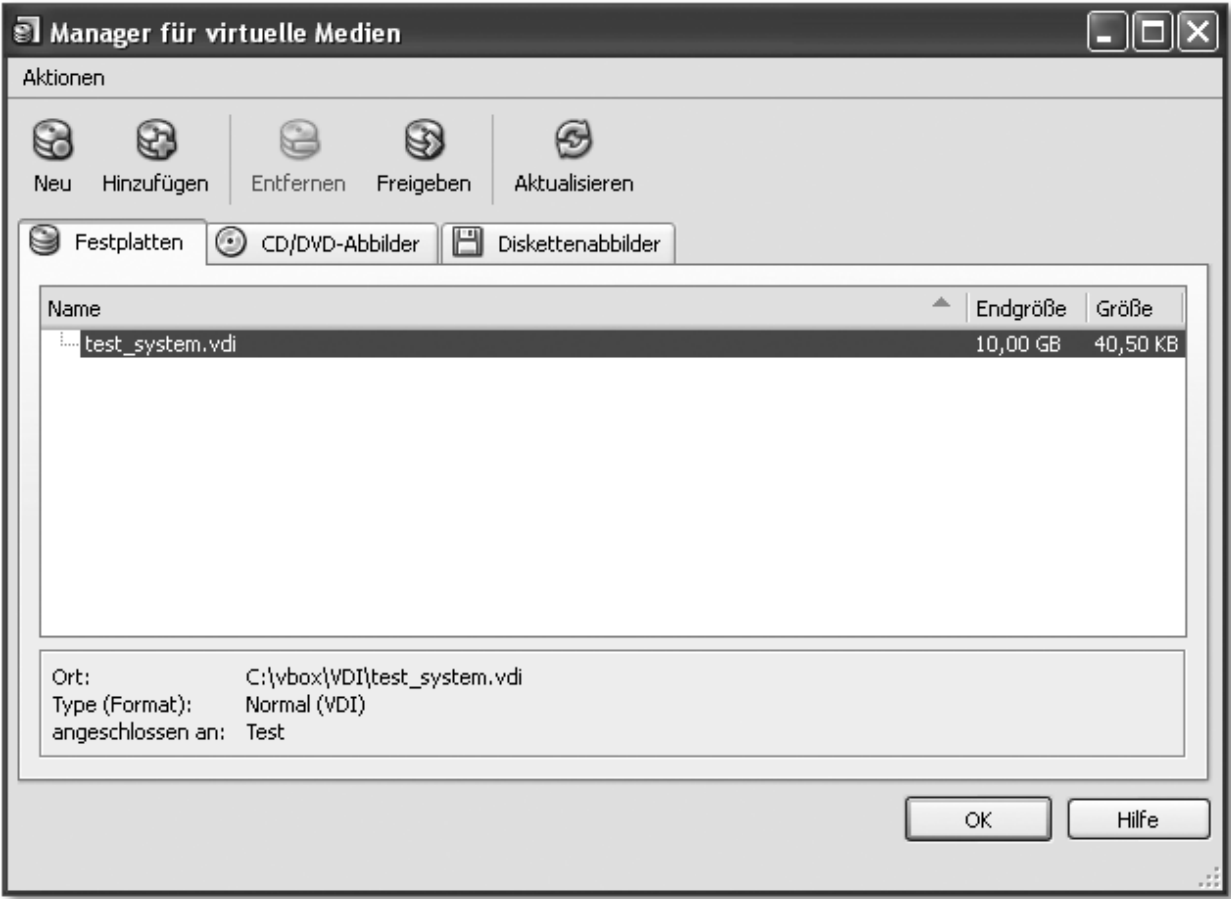


Abbildung 3.30 Der Manager für virtuelle Medien

CD/DVD-Abbilder

Im Register CD/DVD-ABBILDER verwalten Sie ISO-Images von CDs und DVDs (Abbildung 3.31). Dies verhält sich ähnlich wie die Verwaltung der Festplatten – es können nur keine neuen CD/DVD-Abbilder erstellt werden. Dafür greifen Sie auf andere Hilfsmittel zurück.

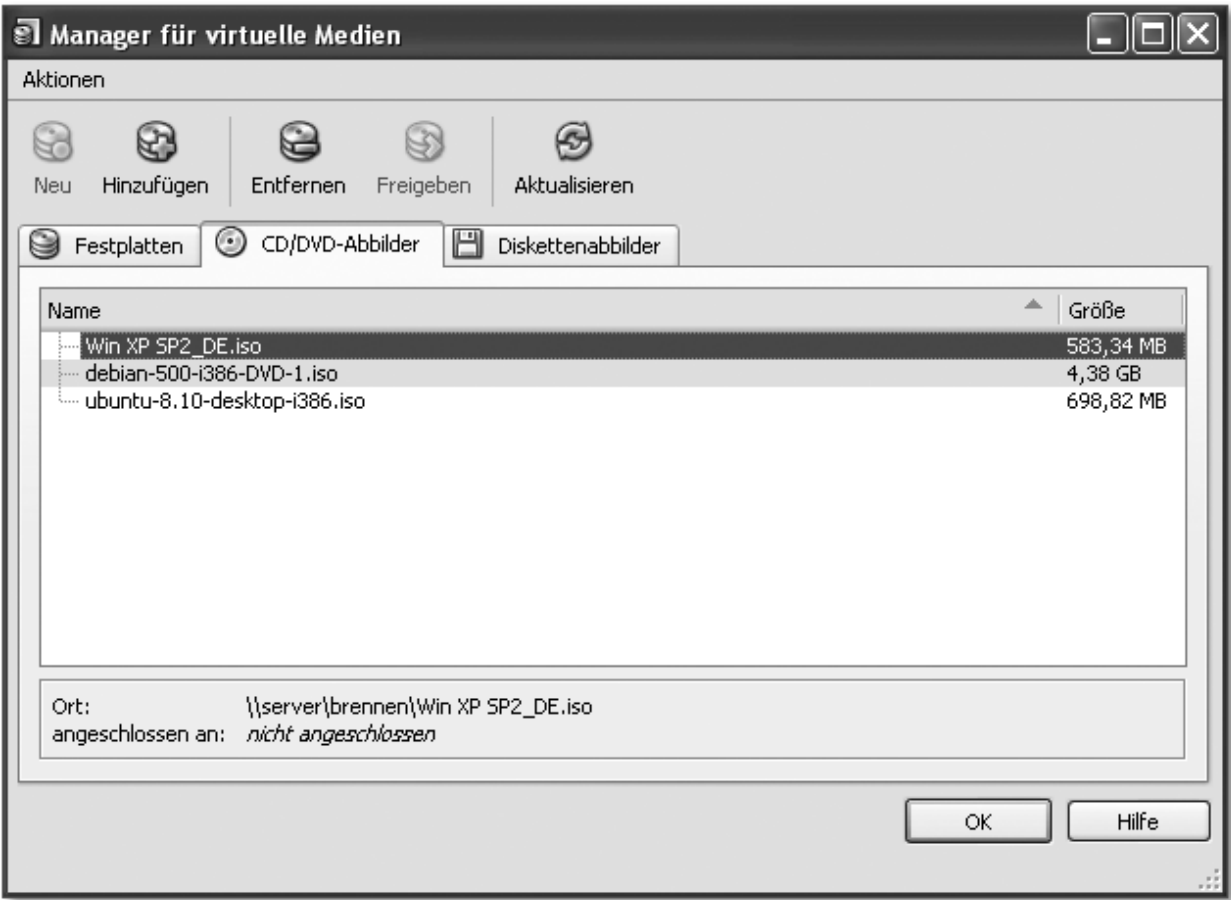


Abbildung 3.31 Verwaltung von CD/DVD-Abbildern

Diskettenabbilder

Diskettenabbilder verwalten Sie genau wie CD/DVD-Abbilder. Ein neues Abbild erstellen Sie über eine neue, leere Textdatei mit dem gewünschten Namen. Diese können Sie dann über HINZUFÜGEN registrieren und als Diskettenabbild verwenden.

Fehler und Probleme

Wenn ein Image, egal von welchem Typ, nicht mehr erreichbar ist, weist Virtual-Box Sie mit einer Fehlermeldung darauf hin. Im Manager für virtuelle Medien hebt VirtualBox das entsprechende Image dann mit einem Warndreieck hervor. In diesem Fall kopieren Sie entweder das Abbild wieder an den Ursprungsort zurück oder, falls es nicht mehr benötigt wird, klicken Sie auf den Eintrag ENTFERNEN.

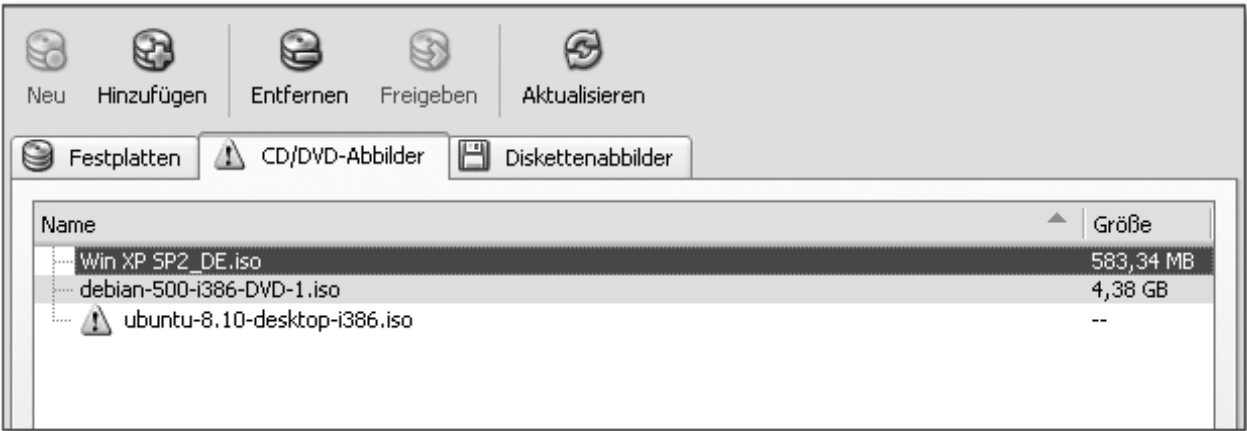


Abbildung 3.32 Ein nicht mehr erreichbares ISO-Image

3.5.4 Konfiguration einer virtuellen Maschine

Das Anlegen und Konfigurieren von virtuellen Maschinen werden Sie in den Praxisbeispielen genauer kennen lernen. In diesem Abschnitt erkläre ich Ihnen vorab nur den Aufbau und die Verwendung des Konfigurationsdialogs. Diesen öffnen Sie, indem Sie die entsprechende Maschine markieren und in der Symbolleiste den Button ÄNDERN anklicken.

In Abbildung 3.33 sehen Sie das Fenster zum Ändern der Konfiguration einer virtuellen Maschine. Im linken Teil befindet sich eine Auflistung der einzelnen Bereiche. Je nachdem, welcher Bereich aktiviert ist, erscheinen rechts die möglichen Optionen – teilweise durch Register getrennt. Wie auch bei den globalen Einstellungen befindet sich über den Buttons OK, ABBRECHEN und HILFE ein Infobereich. In diesem sehen Sie Kurzbeschreibungen der Optionen, über denen sich der Mauszeiger gerade befindet.

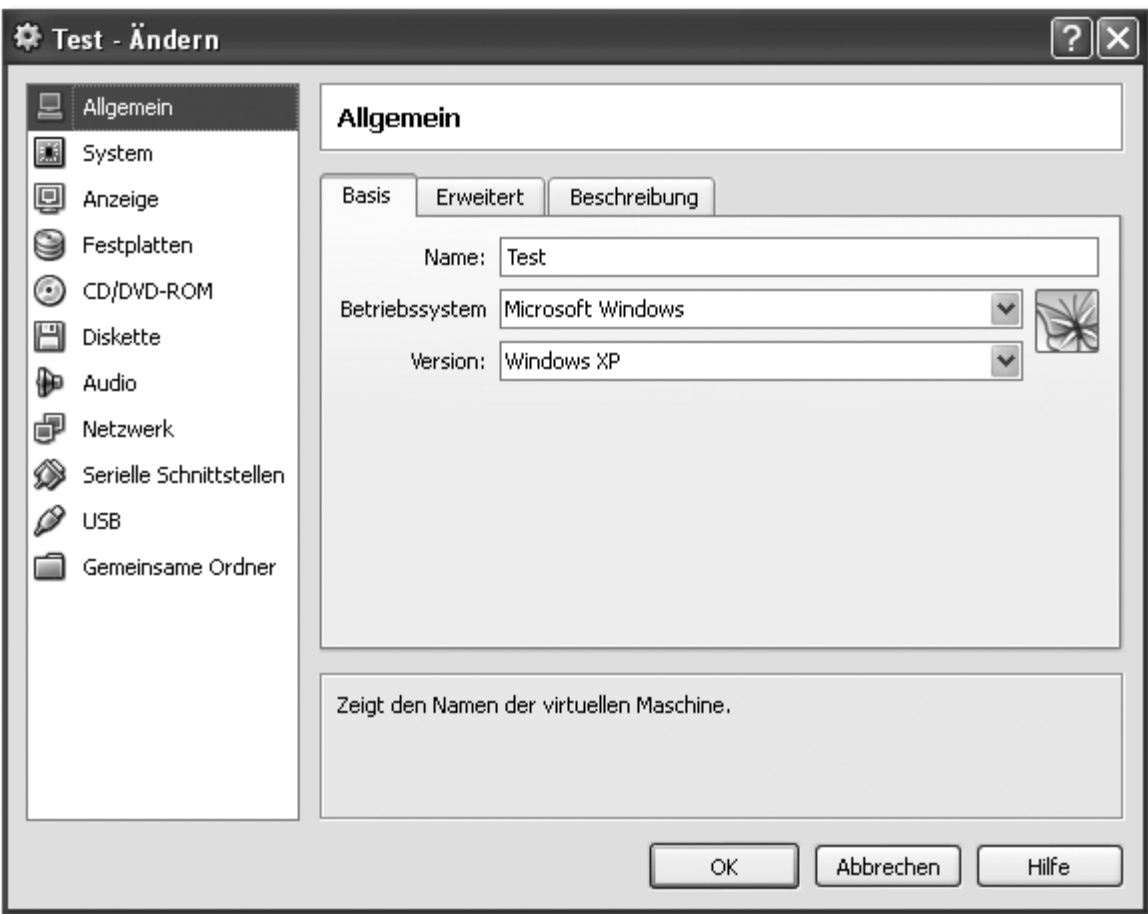


Abbildung 3.33    Einstellungsdialog einer virtuellen Maschine

Änderungen übernehmen Sie erst durch einen Klick auf OK. Achten Sie dabei auf den Infobereich, der auch eventuell fehlerhafte Einstellungen anzeigt (siehe Abbildung 3.34).

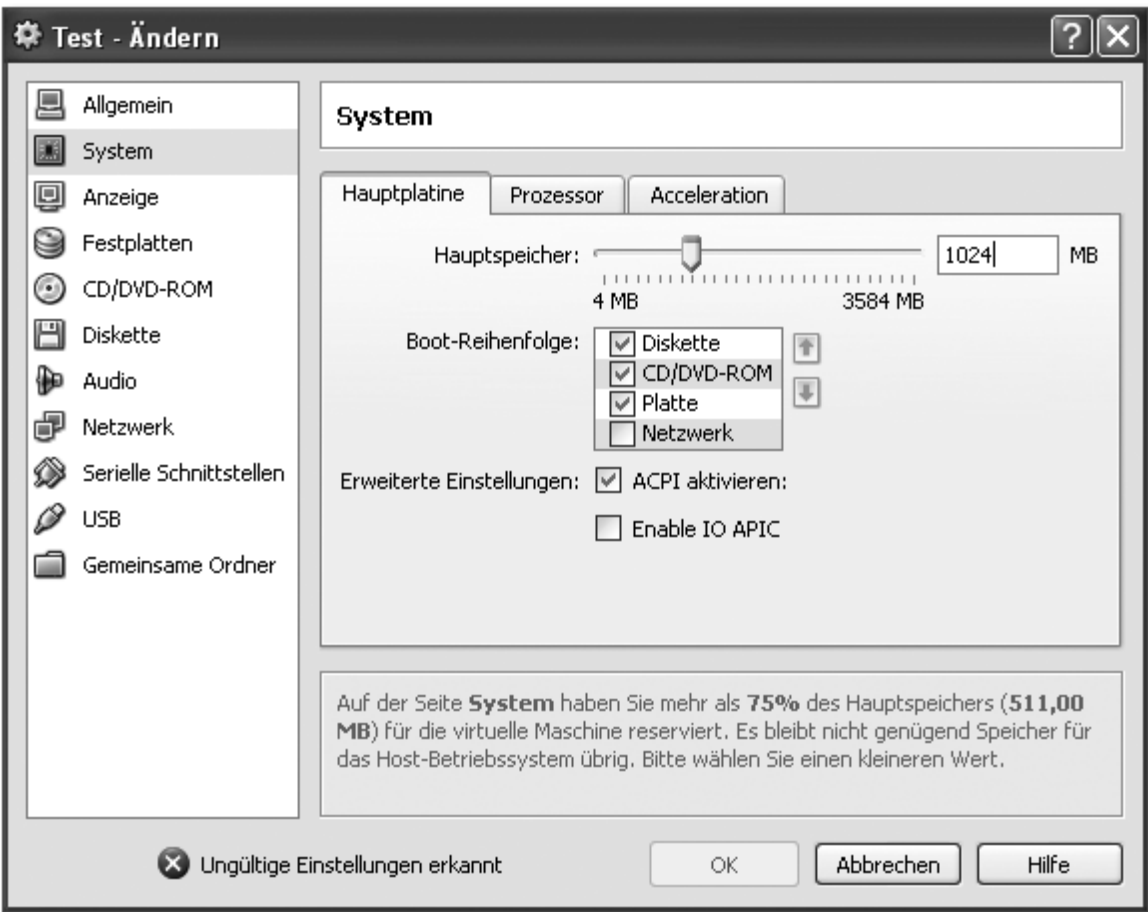
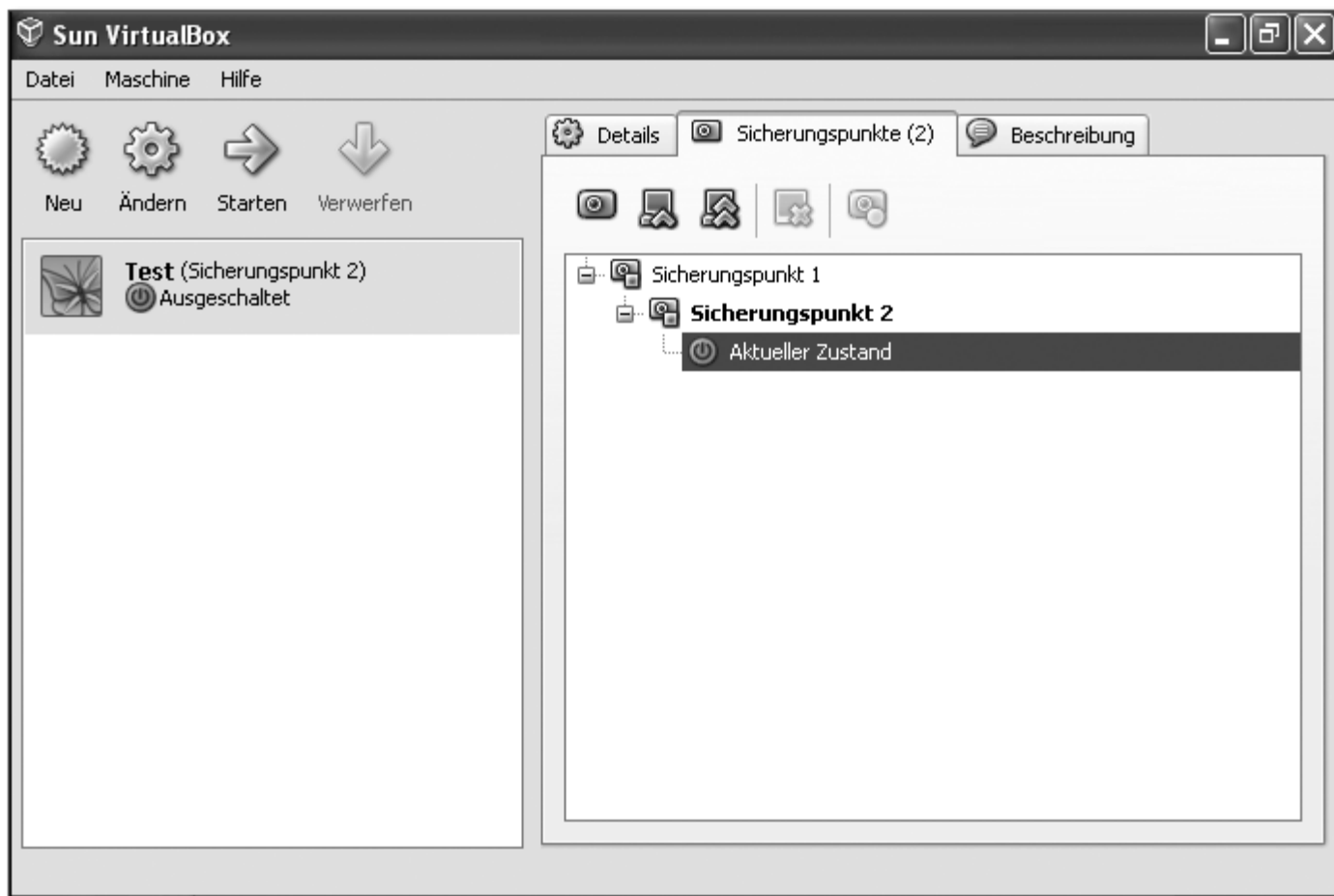


Abbildung 3.34    Der Infobereich zeigt ungültige Einstellungen an.

### 3.5.5 Sicherungspunkte – Snapshots

In diesem Buch werde ich häufiger Sicherungspunkte verwenden. Vielen werden diese bereits unter dem englischen Begriff *Snapshots* bekannt sein. Zu einem solchen Sicherungspunkt kann man jederzeit zurückkehren. Änderungen, die seitdem vorgenommen wurden, verwerfen Sie damit. Alternativ verwerfen Sie den Sicherungspunkt und behalten die Änderungen. Die Verwaltung der Sicherungspunkte ist in der GUI von VirtualBox ziemlich einfach.



**Abbildung 3.35** Verwaltung der Sicherungspunkte in VirtualBox

In Abbildung 3.35 sehen Sie die virtuelle Maschine mit dem Namen *Test*, für die zwei Sicherungspunkte erstellt wurden. Die Sicherungspunkte bauen dabei linear aufeinander auf.

Weitere Details zu den Sicherungspunkten lesen Sie in Abschnitt 4.5.6, »Sicherungspunkt«, und ausführlich in 5.3, »Arbeiten mit Sicherungspunkten (Snapshots)«.

### 3.5.6 Beschreibung

Zu jeder virtuellen Maschine können Sie eine ausführliche Beschreibung angeben. Diese Möglichkeit sollten Sie nicht außer Acht lassen. Gerade bei vielen ähnlichen Maschinen wissen Sie später eventuell nicht mehr, welche Funktion diese hatten. Zum Bearbeiten dieser Beschreibung verwenden Sie den Button EDITIEREN.

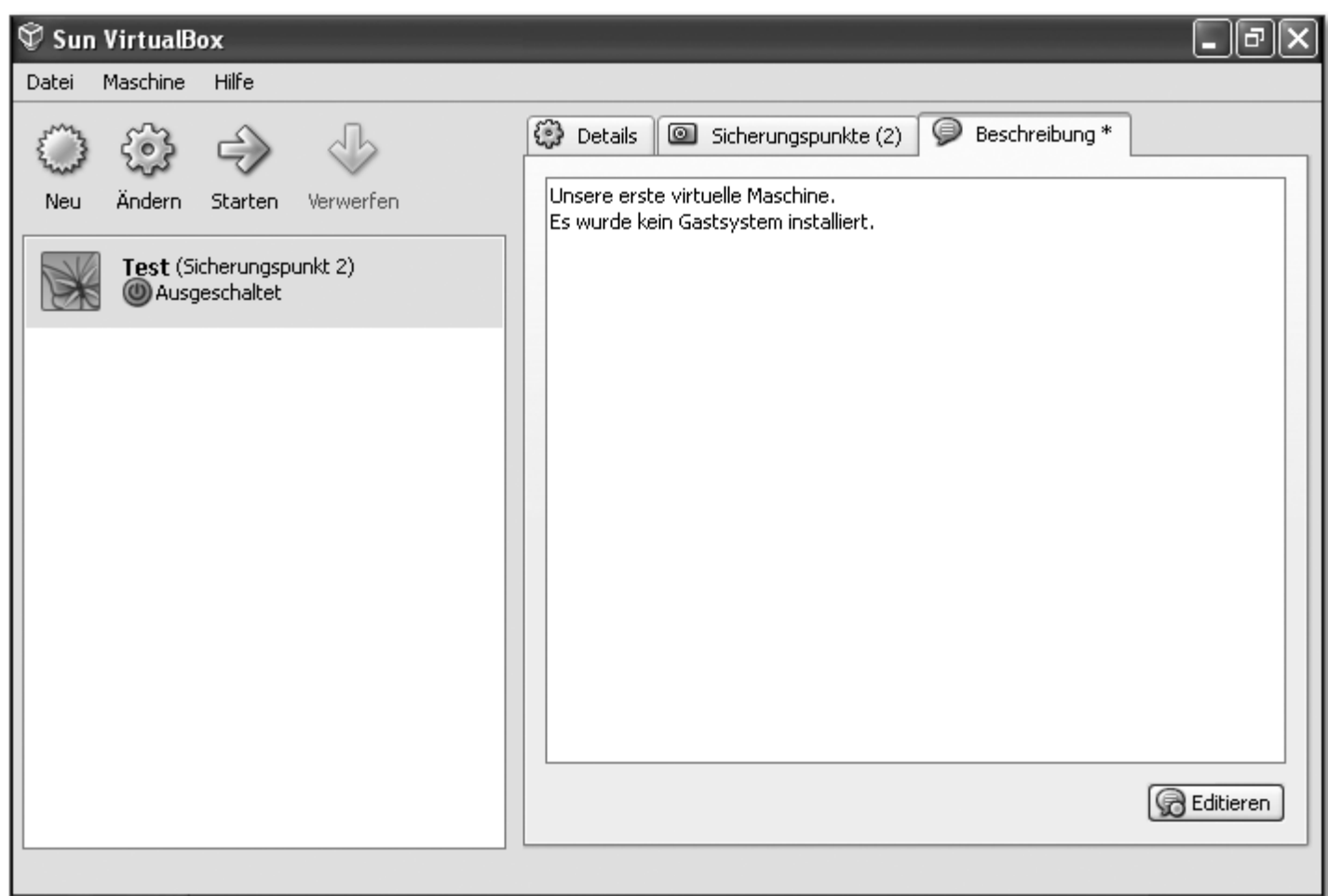


Abbildung 3.36 Beschreibung einer Maschine eintragen

[+]

Zeilenwechsel

Ein Zeilenwechsel kann nicht mit der Taste , sondern nur mit  +  vorgenommen werden!

3.5.7 Fenster für virtuelle Maschinen

Wenn Sie eine Maschine über die grafische Oberfläche starten, öffnet sich ein zusätzliches Fenster, in dem die Ausgabe der Maschine erfolgt. Über dieses Fenster wird die virtuelle Maschine auch gesteuert.

Dieses Fenster besitzt eigene Menüs, über die verschiedene Optionen und Einstellungen an der laufenden virtuellen Maschine geändert werden können, zum Beispiel Wechseln in den Vollbildmodus oder Einbinden von USB-Geräten.

Am unteren Rand befindet sich eine Statusleiste, die Sie über verschiedene Aktivitäten informiert. Bewegen Sie einfach die Maus über eines der Symbole, und es erscheint eine Beschreibung des Objekts. Bei einigen der Symbolen können Sie auch ein Kontextmenü durch einen Rechtsklick öffnen, in dem, ähnlich wie in den Menüs, verschiedene Änderungen möglich sind.





Abbildung 3.37 Laufende virtuelle Maschine

### 3.5.8 Hilfe

Zusätzlich zu der Dokumentation auf der Homepage von VirtualBox gibt es eine Hilfe in der grafischen Oberfläche. Diese ist jedoch ebenfalls nur in englischer Sprache verfügbar. Sie finden sie im Menü HILFE • INHALT.

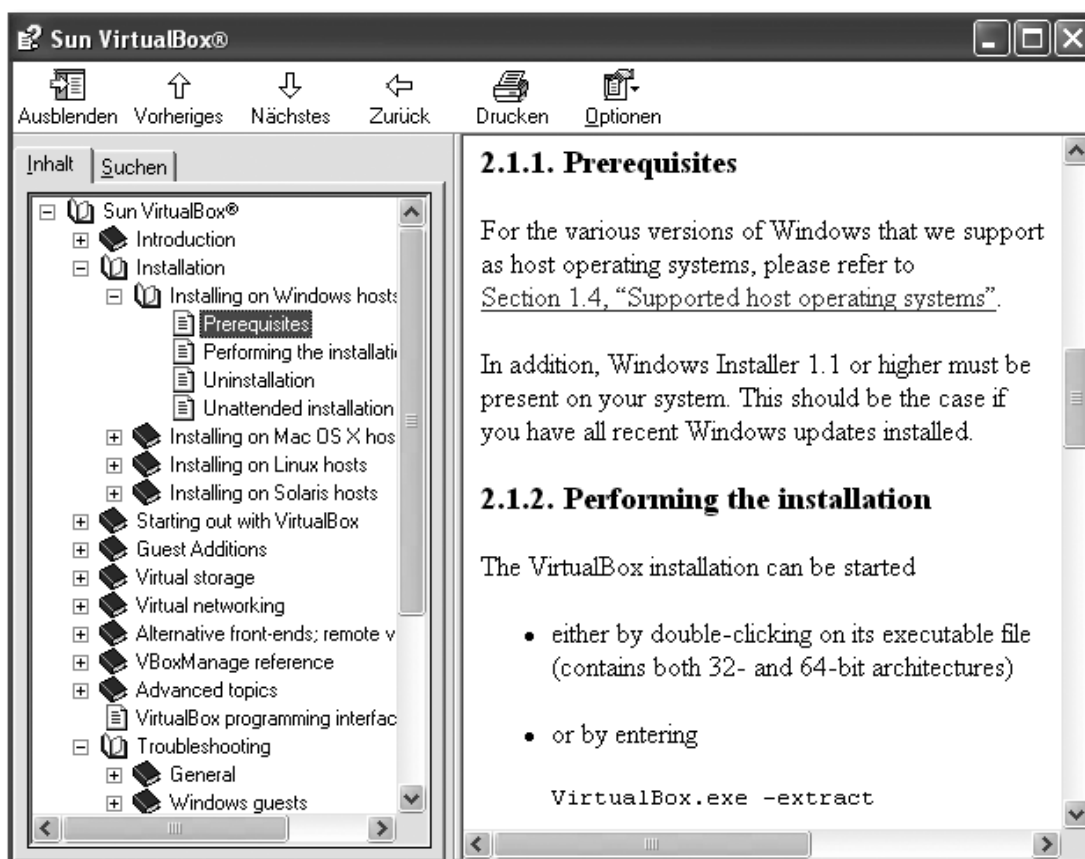


Abbildung 3.38 Die Hilfe ist nur in Englisch verfügbar.

Der Menüpunkt HILFE • ALLE WARNUNGEN ZURÜCKSETZEN sorgt dafür, dass Meldungen, die durch Klicken der Checkbox NICHT MEHR ANZEIGEN deaktiviert wurden, wieder erscheinen.

## 3.6 Kommandozeilentools

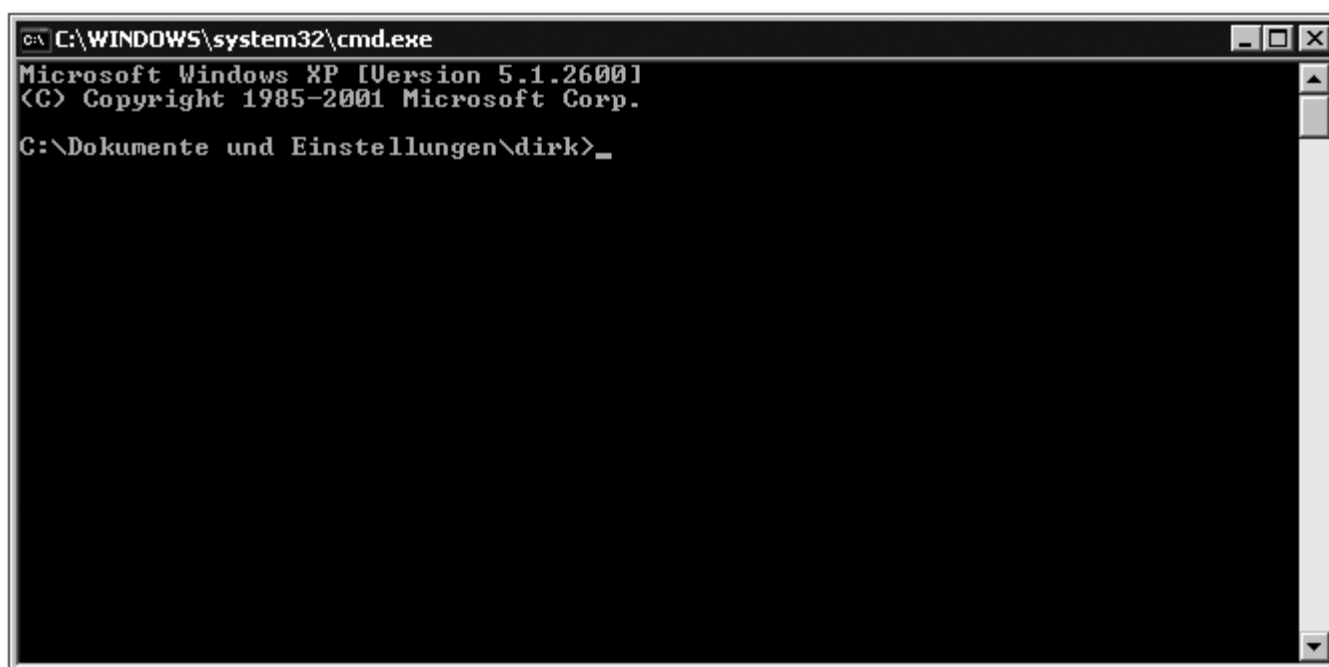
Die Kommandozeilentools dienen dazu, virtuelle Maschinen auch ohne grafische Oberfläche zu konfigurieren, zu verwalten und zu starten. Sie bieten außerdem auch mehr Möglichkeiten und Funktionen als die GUI. Wie der Name vermuten lässt, verwenden Sie diese Tools in der Eingabeaufforderung (Windows) bzw. Konsole (Linux).

[+]

### Mehr über Kommandozeile

Den Einsatz der Kommandozeilentools lernen Sie auch im Kapitel 4, »Praxisbeispiele«, kennen. Die detaillierte Beschreibung zu den Optionen und Parametern folgt am Ende in Kapitel 8, »Kommandozeilentools«.

Unter Windows XP öffnen Sie die Eingabeaufforderung über START • ALLE PROGRAMME • ZUBEHÖR • EINGABEAUFFORDERUNG. Alternativ öffnen Sie sie über START • AUSFÜHREN und den Befehl `cmd`.



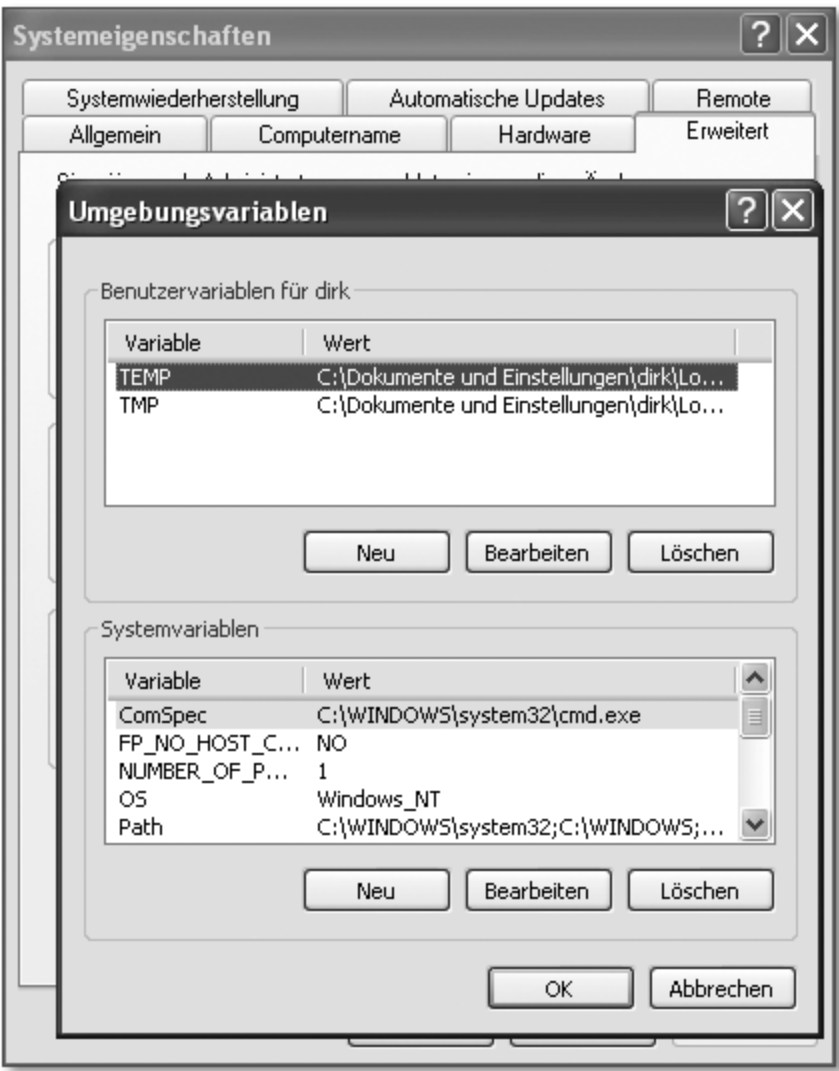
**Abbildung 3.39** Die Eingabeaufforderung unter Windows XP

Unter Windows müssen Sie vor dem Einsatz der Tools vermutlich noch die Variable *Path* anpassen (es sei denn, Sun bzw. Oracle hat inzwischen die Installationsroutine angepasst). Dies testen Sie einfach, indem Sie `VBoxManage` eingeben und  betätigen.

C:\Dokumente und Einstellungen\dirk>vboxmanage  
Der Befehl "vboxmanage" ist entweder falsch geschrieben oder konnte nicht gefunden werden.

**Listing 3.2** Die Ausgabe erscheint, wenn die Path-Variable nicht angepasst wurde.

Falls das Kommando nicht gefunden wurde, muss die Path-Variable angepasst und der Installationsordner von VirtualBox (standardmäßig *C:\Programme\Sun\VirtualBox*) mit aufgenommen werden. Dazu öffnen Sie zunächst die SYSTEMEIGENSCHAFTEN über START • SYSTEMSTEUERUNG • LEISTUNG & WARTUNG • SYSTEM. Dort wechseln Sie in das Register ERWEITERT und klicken auf den Button UMGEBUNGSVARIABLEN (siehe Abbildung 3.40).



**Abbildung 3.40** Dialog zur Einrichtung der Umgebungsvariablen

Nun entscheiden Sie, ob die Variable nur für den aktuellen (Bereich BENUTZERVARIABLEN FÜR BENUTZER) oder für alle Benutzer (Abschnitt SYSTEMVARIABLEN) angepasst werden soll. Für eine systemweite Einrichtung bearbeiten Sie die vorhandene Path-Variable. Achten Sie darauf, die vorhandenen Informationen nicht zu verändern, und trennen Sie die Eingaben mit einem Semikolon (Abbildung 3.41).

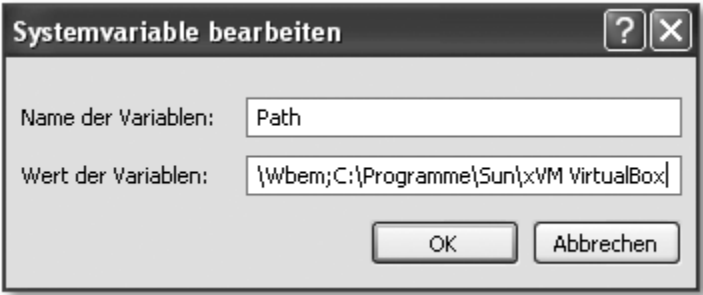


Abbildung 3.41 Path-Variable für alle Benutzer einrichten

Um die Variable nur für den aktuellen Benutzer zu setzen, erstellen Sie eine neue Benutzervariable.

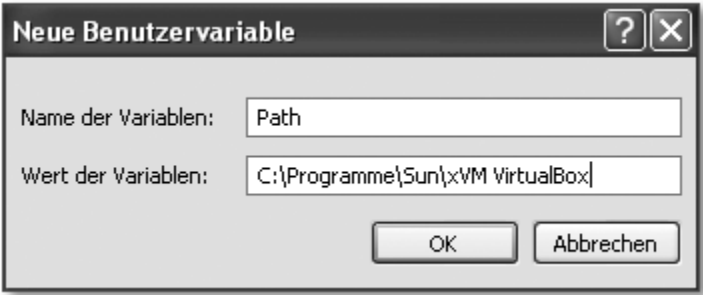


Abbildung 3.42 Variable nur für den aktuellen Benutzer einrichten

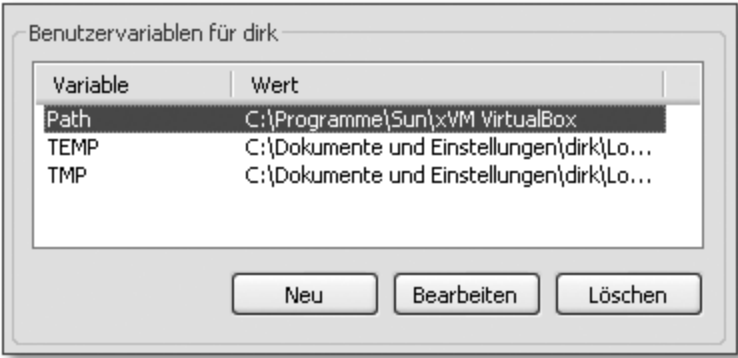


Abbildung 3.43 Die neue Path-Variable

Danach können Sie die Eingabeaufforderung neu starten. Falls noch ein Programmfenster vom vorherigen Test geöffnet ist, schließen Sie dieses, da die Änderungen erst bei einem Neustart übernommen werden. Jetzt zeigt der Befehl `VBoxManage` die Optionen des Kommandozeilentools an.

Bei den meisten Linux-Distributionen finden Sie die Konsole im Startmenü unter dem Namen `KONSOLE` oder `TERMINAL`. Vermutlich werden die Leser, die Virtual-Box unter Linux einsetzen, diese Information jedoch nicht benötigen. Das Problem mit fehlenden Variablen besteht hier nicht, da die Tools bei der Installation in den entsprechenden Verzeichnissen (`/usr/bin`) abgelegt werden. Dafür ist unter Linux die Groß- und Kleinschreibung wichtig. Schreiben Sie `VBoxManage` auch wirklich mit den drei Großbuchstaben (siehe Abbildung 3.45), da der Befehl sonst nicht gefunden wird. Einige Distributionen setzen bei der Installation einen Link namens `vboxmanage`, sodass Sie das Kommando auch kleinschreiben können.

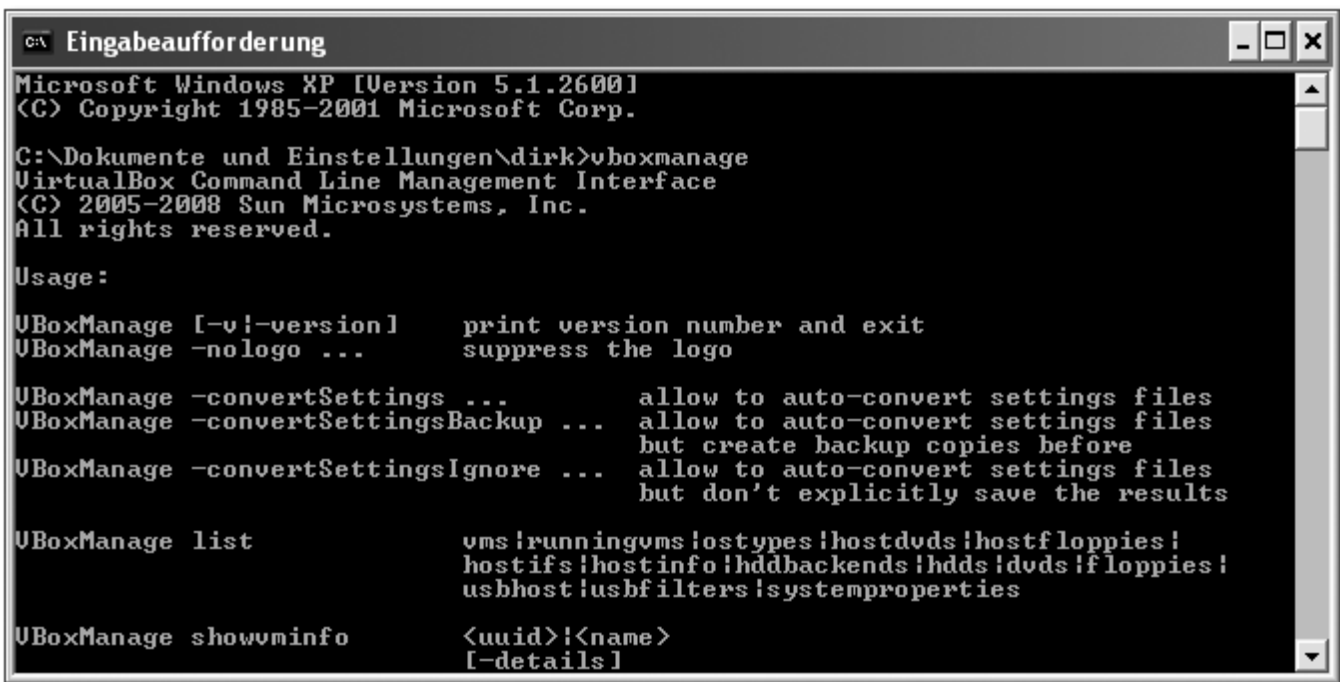


Abbildung 3.44 Das Kommando »VBoxManage« wurde gefunden.

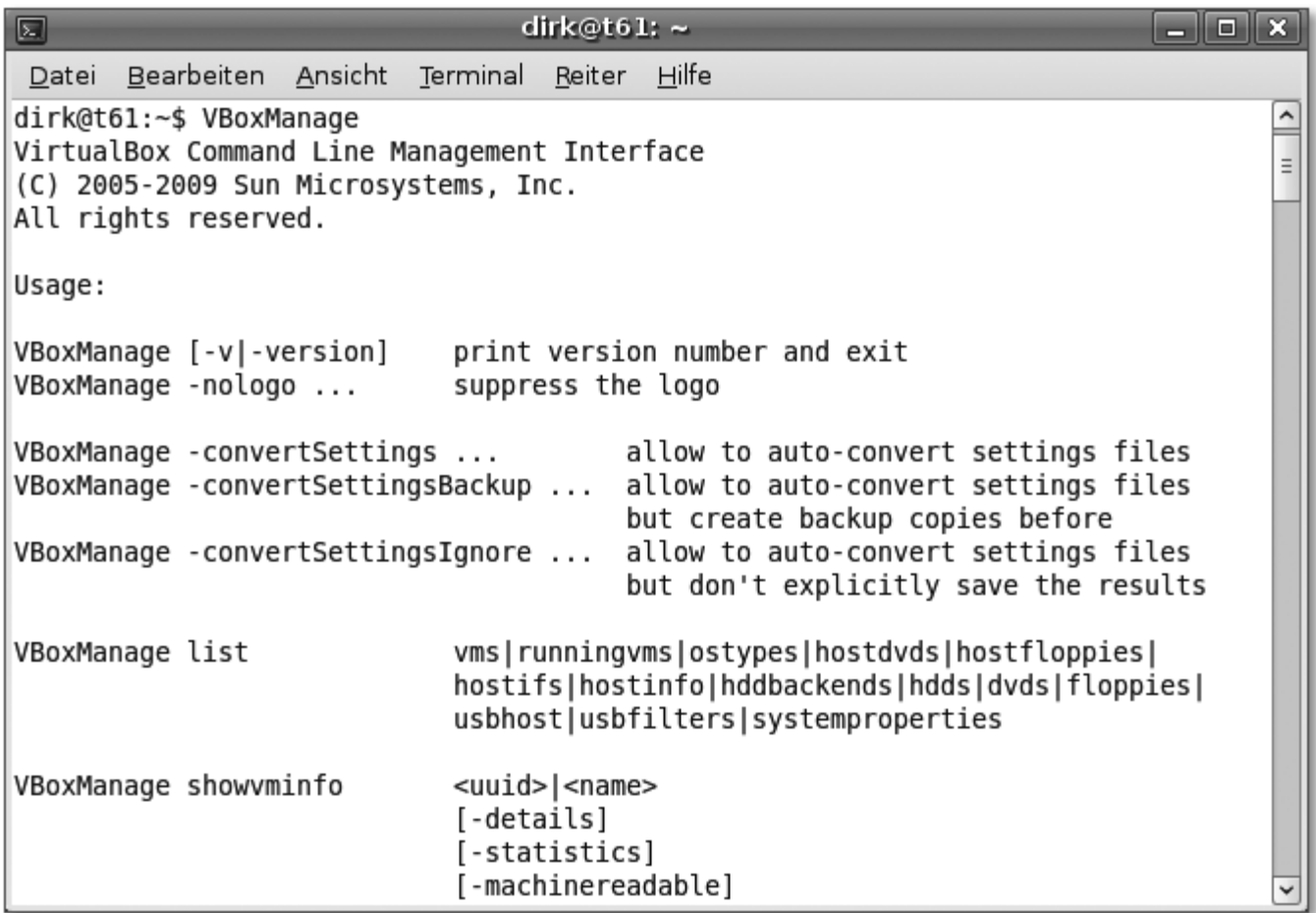


Abbildung 3.45 VBoxManage unter Linux

**Prüfen Sie, ob es Neuerungen gibt**

Die Kommandozeilentools und ihre Funktionen ändern sich fast mit jeder neuen VirtualBox-Version. Prüfen Sie bei Problemen, ob sich eventuell wieder Neuerungen ergeben haben und die von mir beschriebenen Funktionen geändert oder sogar verworfen wurden.

[+]

### 3.6.1 VBoxManage

Die grafische Oberfläche kann leider nicht dazu verwendet werden, sämtliche Optionen einer virtuellen Maschine zu konfigurieren. Ein mächtiges Werkzeug zum Verwalten, Steuern und Einrichten der virtuellen Maschinen ist das *Virtual-Box Command Line Management Interface*, kurz *VBoxManage*. Mit diesem Tool erstellen Sie neue Maschinen und bearbeiten sämtliche ihrer Optionen. Weiterhin verwenden Sie VBoxManage zum Steuern einer Maschine, die mit VBoxHeadless oder VBoxSDL gestartet wurde.

Beim Start von VBoxManage übergeben Sie dem Programm verschiedene Parameter. Der Aufbau ist dabei meist ähnlich:

```
VBoxManage BEFEHL PARAMETER
```

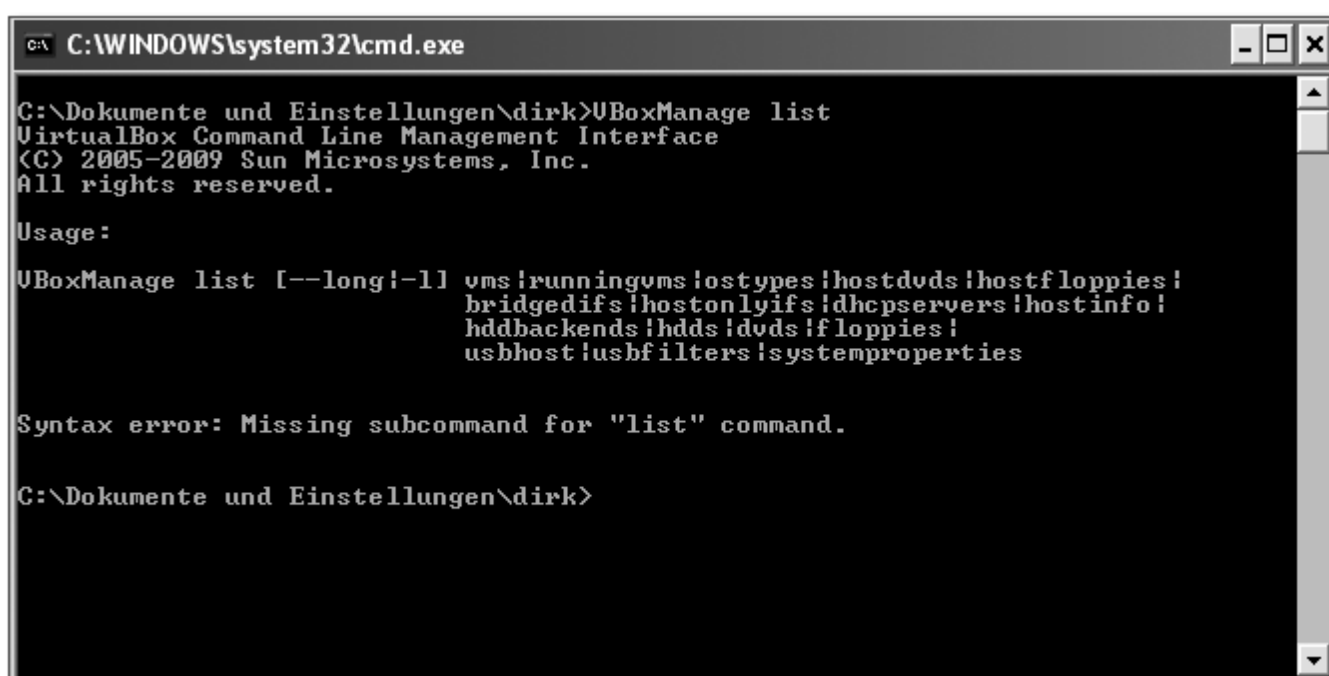
Es stehen viele Funktionen zur Verfügung. Eine komplette Übersicht erhalten Sie durch Eingabe von VBoxManage ohne Parameter. Allerdings laufen die Informationen dann einfach durch das Fenster. Zusammen mit dem Programm `more` erfolgt die Anzeige seitenweise:

```
VBoxManage | more
```

Alternativ lenken Sie die Programmausgabe in eine Datei um und lesen diese dann in einem Texteditor:

```
VBoxManage > hilfe.txt
```

Die Datei *hilfe.txt* können Sie nun in einem Editor Ihrer Wahl öffnen und bei Bedarf auch ausdrucken. Wenn Sie nur die Informationen zu einem Befehl suchen, geben Sie diesen ohne weitere Parameter ein (Abbildung 3.46).



**Abbildung 3.46** »VBoxManage list« zeigt eine Kurzhilfe zum Befehl »list«

## Universally Unique Identifier (UUID)

Jede virtuelle Maschine, jedes Festplatten-Image oder CD/DVD-Abbild erhält in VirtualBox einen *Universally Unique Identifier* (UUID). Diesen können Sie auch beim Einsatz von VBoxManage verwenden, um Zugriff auf ein entsprechendes Objekt zu erhalten. In der grafischen Oberfläche wird die UUID übrigens nicht angezeigt.

Jede virtuelle Maschine hat eine eindeutige UUID. Sie besteht aus einer 16 Byte großen Zahl, die hexadezimal in fünf Blöcken dargestellt wird. Man kann mit ziemlicher Sicherheit davon ausgehen, dass eine korrekt erstellte UUID weltweit eindeutig ist; die Wahrscheinlichkeit von zwei gleichen UUIDs geht zumindest gegen Null.

Um die UUID zu ermitteln, verwenden Sie den Befehl `list`. Dieser zeigt zusammen mit dem Parameter `vms` Informationen über sämtliche registrierte Maschinen des Hostsystems an.

### Ausgabe VBoxManage list vms:

```
C:\Dokumente und Einstellungen\dirk>VBoxManage list vms
"Test" {8cb70952-3ed5-41ef-bd0a-ee4dd3295c89}
C:\Dokumente und Einstellungen\dirk>
```

#### Listing 3.3 Ausgabe »VBoxManage list vms«

Im Listing sehen Sie die Ausgabe zur virtuellen Maschine *Test*. Die UUID (8cb70952-3ed5-41ef-bd0a-ee4dd3295c89) ist fett markiert. Diese UUID können Sie nun in Verbindung mit VBoxManage verwenden, um Befehle an die entsprechende Maschine zu senden. Bei den meisten Kommandos können Sie jedoch auch den Namen der Maschine verwenden – wobei Sie hier (auch unter Windows!) auf Groß- und Kleinschreibung achten müssen!

Verweis auf eine virtuelle Maschine:

- ▶ *per UUID*  
VBoxManage showvminfo **8cb70952-3ed5-41ef-bd0a-ee4dd3295c89**
- ▶ *oder mit dem Namen*  
VBoxManage showvminfo **Test**

### List

Den Befehl `list` haben wir im vorherigen Abschnitt verwendet, um Informationen über alle registrierten Maschinen auszugeben und deren UUID zu ermitteln.

Allerdings zeigt `list` nicht nur die virtuellen Maschinen an, sondern alle registrierten Objekte und weitere wichtige Informationen.

- ▶ Alle registrierten virtuellen Maschinen anzeigen:  
`VBoxManage list vms`
- ▶ Alle registrierten virtuellen Maschinen in einem ausführlicheren Format anzeigen:  
`VBoxManage list --long vms`
- ▶ Alle zurzeit laufenden Maschinen auflisten:  
`VBoxManage list runningvms`
- ▶ Alle registrierten Festplatten-Images:  
`VBoxManage list hdds`
- ▶ Alle registrierten CD/DVD-Abbilder:  
`VBoxManage list dvds`
- ▶ Alle von VirtualBox unterstützten Gastssysteme auflisten:  
`VBoxManage list ostypes`
- ▶ Einstellungen und Systemeigenschaften auflisten:  
`VBoxManage list systemproperties`

In den folgenden zwei Beispielen sehen Sie die Ausgaben von `VBoxManage list dvds` unter Linux und unter Windows XP. Dabei erkennen Sie auch die Unterschiede der beiden Betriebssysteme. Unter Linux existieren keine Laufwerksbuchstaben, und die Pfadangaben erfolgen mit einem Schrägstrich (/). Windows XP verwendet hingegen Laufwerksbuchstaben und den Backslash (\).

#### Ausgabe `VBoxManage list dvds` unter Linux:

```
dirk@t61:~$ VBoxManage list dvds
UUID:           4100527e-5d65-4d5b-e5b3-db8309de0ec4
Path:           /usr/share/virtualbox/VBoxGuestAdditions.iso
Accessible: yes
UUID:           b4663009-c743-4eff-f398-9e6c211883de
Path:           /nfs/brennen/ubcd411.iso
Accessible: yes
UUID:           bcf59f0e-56a9-4974-a2c2-66d38978e1da
Path:           /nfs/brennen/Citrix Xenserver-5.0.0-Install-Cd.iso
Accessible: yes
UUID:           38296d64-f5e6-45a2-9bd5-f8993d4957df
Path:           /nfs/brennen/ubuntu-8.10-desktop-i386.iso
Accessible: yes
UUID:           8f7f7ff8-f7e9-41ea-b3fa-7343a60bf52e
```



```
Path:      /nfs/brennen/debian-500-i386-netinst.iso
Accessible: yes
```

**Listing 3.4** Ausgabe »VBoxManage list dvds« unter Linux

### Ausgabe VBoxManage list dvds unter Windows:

```
C:\Dokumente und Einstellungen\dirk>VBoxManage list dvds
UUID:      c8c1c672-ccd8-40ac-94dd-cafb4aae8bd2
Path:      E:\ISOs\Win XP SP2_DE.iso
Accessible: yes
UUID:      c6994a0e-0e22-4088-a0bd-0009a36db0de
Path:      \\server\brennen\debian-500-i386-DVD-1.iso
Accessible: yes
UUID:      bee1a354-f9bb-4819-ae1b-e42744bc21ce
Path:      \\server\brennen\FreeDOS 1.0_full.iso
Accessible: yes
```

**Listing 3.5** Ausgabe »VBoxManage list dvds« unter Windows

### VBoxManage – Fehlermeldungen

Sollten Sie eine fehlerhafte Eingabe getätigt haben, meldet sich VBoxManage ausführlich zurück. Die folgenden Abschnitte stellen die häufigsten Fehlerquellen vor.

#### Syntaxfehler

Die einfachsten Fehler sind Syntaxfehler. In diesem Fall handelt es sich meist nur um Schreibfehler oder Buchstabendreher.

```
C:\Dokumente und Einstellungen\dirk>VBoxManage list vmd
Usage:
VBoxManage list [--long|-l]
vms|runningvms|ostypes|hostdvds|hostfloppies|
        bridgedifs|hostonlyifs|dhcpserver|hostinfo|
        hddbackends|hdds|dvds|floppies|
        usbhost|usbfilters|systemproperties
Syntax error: Unknown subcommand "vmd".
C:\Dokumente und Einstellungen\dirk>
```

**Listing 3.6** Syntaxfehler

Hier wurde der Parameter `vms` fälschlicherweise als `vmd` geschrieben.

## Anwenderfehler

In diese Kategorie fallen unter anderem Befehle, die nicht oder nur zurzeit nicht angewandt werden können. Dazu zählt zum Beispiel das Entfernen des Festplatten-Images einer virtuellen Maschine, während sie läuft, oder das Abmelden, obwohl noch Medien registriert sind.

```
dirk@t61:/$ VBoxManage -q unregistervm WinXP
ERROR: Cannot unregister the machine 'Test' because it has an open session
Details: code VBOX_E_INVALID_OBJECT_STATE (0x80bb0007),
component Machine, interface IMachine, callee nsISupports
Context: "UnregisterMachine(uuid, machine.asOutParam())" at line 294
of file VBoxManage.cpp
dirk@t61:/$
```

### Listing 3.7 Anwenderfehler

Im obigen Beispiel wurde versucht, die virtuelle Maschine mit dem Namen *WinXP* abzumelden, obwohl sie zurzeit gestartet ist (... *has an open session*). Der Fehlercode lautet *code VBOX\_E\_INVALID\_OBJECT\_STATE (0x80bb0007)*, was so viel bedeutet wie *falscher Objektstatus*.

Sie sehen also, dass VirtualBox beim Auftreten von Fehlern sehr gesprächig ist. Bei korrekter Interpretation sind diese daher zumeist schnell gefunden und beseitigt.

### 3.6.2 VBoxSDL

Wenn Sie häufiger bestimmte virtuelle Maschinen verwenden und diese dabei nicht immer über die grafische Oberfläche starten wollen, kommt VBoxSDL zum Einsatz. VBoxSDL ist ein einfaches Tool zum Steuern von virtuellen Maschinen. Es kann diese starten und dabei auch verschiedene Änderungen vornehmen. Diese Änderungen sind dann jedoch temporär und werden beim Beenden wieder verworfen. Nach dem Start kann die Maschine wie im Fenster der grafischen Oberfläche verwendet werden. Allerdings fehlen bei VBoxSDL die gewohnten Menü- und Statusleisten.

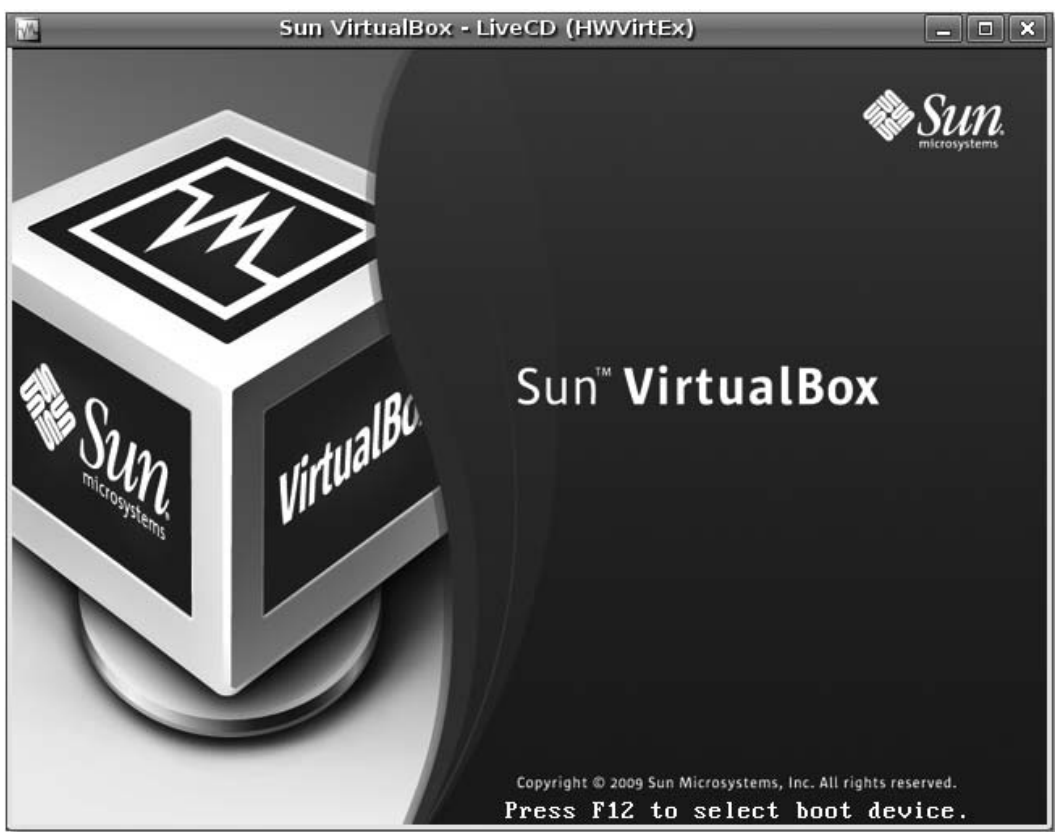


Abbildung 3.47 Eine virtuelle Maschine im VBoxSDL-Fenster

Wie Sie in Abbildung 3.47 erkennen, besitzt das Fenster von VBoxSDL keine Menüleiste wie in der »normalen« grafischen Oberfläche (Abbildung 3.48).



Abbildung 3.48 Die Menüleiste der »normalen« grafischen Oberfläche

Diese Einschränkung ist jedoch von Vorteil, wenn die Maschine einem Benutzer zur Verfügung gestellt wird, der sich nicht mit VirtualBox auskennt und der die Konfiguration auch nicht ändern darf. Außerdem können Sie mit VBoxSDL Maschinen per Verknüpfung, Batchdatei oder Skript starten. Diese Möglichkeit ist praktisch, vor allem wenn sich in der Auflistung der grafischen Oberfläche zahlreiche verschiedene Maschinen befinden und Sie die Übersicht verlieren.

### Virtuelle Maschine starten

Bei VBoxSDL handelt es sich ebenfalls um ein Kommandozeilentool. Sie verwenden es daher auch in der Eingabeaufforderung oder in der Konsole:

```
VBoxSDL --startvm Test
```

**Tastaturprobleme**

Bei Tastaturproblemen auf Linux-Hostsystemen versuchen Sie es am besten mit dem zusätzlichen Parameter `evdevkeymap`.

[+]

Mit der Option `--startvm` starten Sie die Maschine *Test*, und es öffnet sich ein Fenster zum Steuern dieser virtuellen Maschine. Statt diesen Befehl immer wieder eingeben zu müssen, können Sie eine Verknüpfung bzw. einen Starter erstellen (siehe Abbildung 3.49 und Abbildung 3.50).

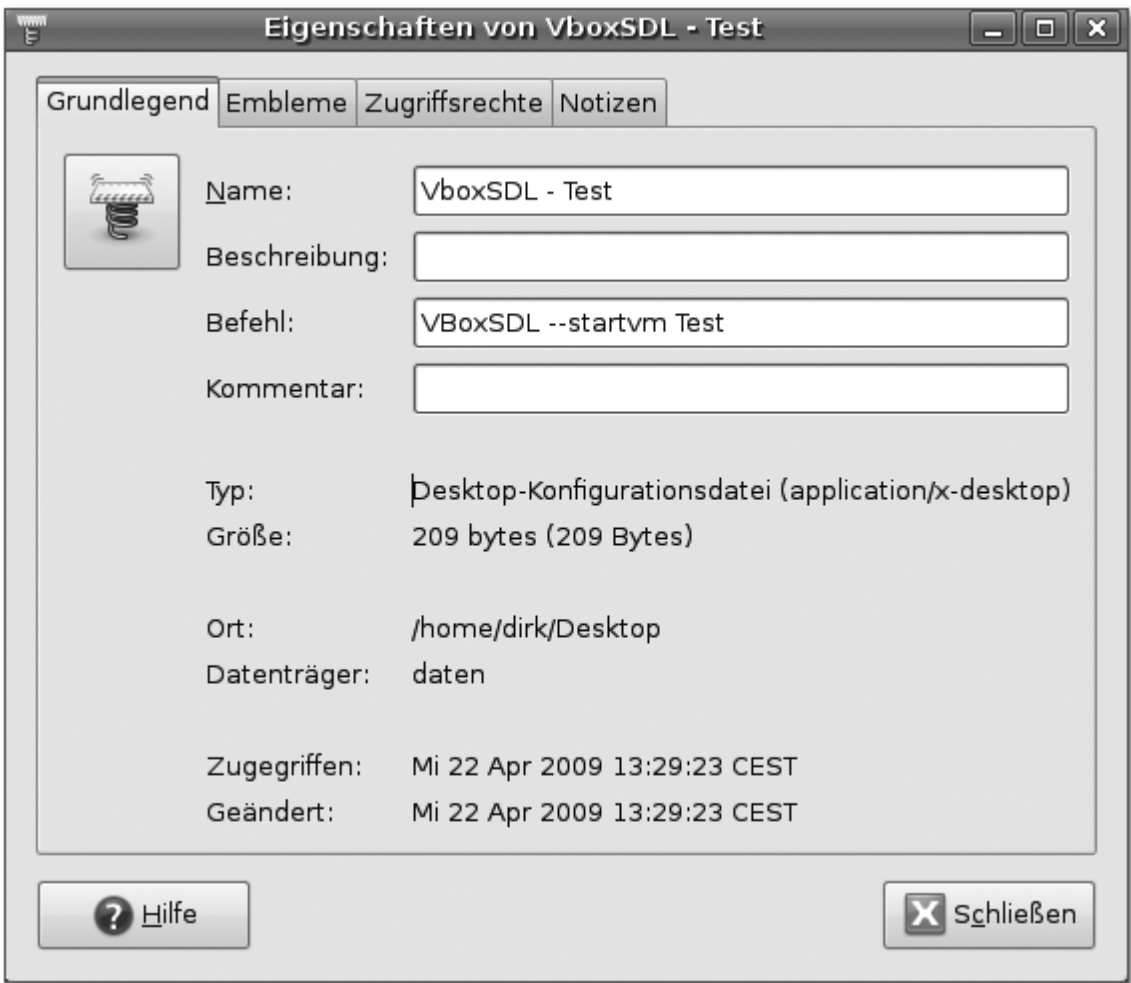


Abbildung 3.49 Einen Starter unter Ubuntu erstellen

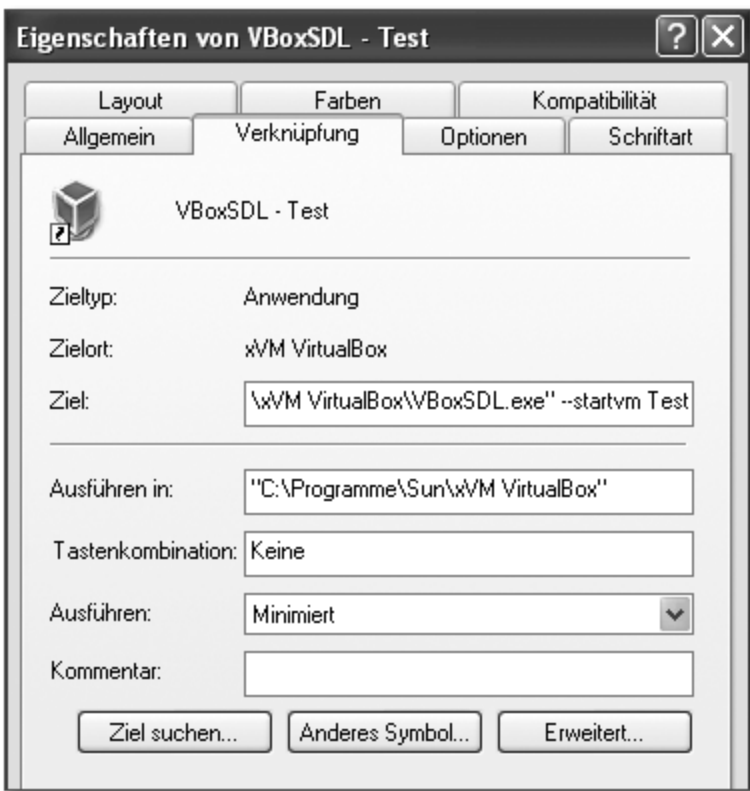


Abbildung 3.50 Verknüpfung unter Windows anlegen

## Host-Taste

Mit der Host-Taste (meist die rechte `[Strg]`-Taste) können Sie, wie auch in der grafischen Oberfläche, verschiedene Befehle an die virtuelle Maschine senden. Mit der Tastenkombination `[Host]-Taste+[F]` wechselt das Fenster zum Beispiel in den Vollbildmodus (englisch »fullscreen«).

## Temporäre Änderungen

Wie bereits erwähnt, können Sie mit VBoxSDL beim Start einer virtuellen Maschine temporäre Änderungen vornehmen, die dann beim Beenden wieder verworfen werden. Die Option `--memory` ändert im folgenden Beispiel den Speicher auf 256 MByte:

```
VBoxSDL --startvm Test --memory 256
```

Weitere nützliche Optionen finden Sie in folgender Liste:

- ▶ Starten einer Maschine:  
`--startvm name`
- ▶ Übergeben eines anderen CD/DVD-Laufwerks oder Abbilds. Um eine Zuordnung aufzuheben, wird *none* übergeben:  
`--cdrom file`
- ▶ Ändern der Größe des Hauptspeichers:  
`--memory Wert`
- ▶ Starten der Maschine im Vollbildmodus:  
`--fullscreen`

## Virtuelle Maschine beenden

In den meisten Fällen werden Sie das Gastsystem der Maschine normal herunterfahren. Durch Schließen des Fensters wird die Maschine jedoch einfach ausgeschaltet. Da dies natürlich nicht sinnvoll ist, sollten Sie diese mit der Option `--termacpi` starten. Die meisten Betriebssysteme fahren dann die Maschine beim Schließen des Fensters korrekt herunter. Gegebenenfalls müssen Sie dies noch in den entsprechenden Energieoptionen anpassen.

```
VBoxSDL --startvm Test --memory 256 --termacpi
```

### 3.6.3 VBoxHeadless

Mit der grafischen Oberfläche und VBoxSDL können Sie virtuelle Maschinen starten und verwenden. Es öffnet sich dabei immer ein Fenster mit der Ausgabe der Maschine. Wenn das Fenster geschlossen wird, beendet sich auch die VM. Nun

kommt es auch vor, dass Maschinen im Hintergrund laufen sollen, ohne dass sich ein Fenster öffnet. Die Benutzer dieser Maschinen können dann per RDP oder SSH auf diese zugreifen. Um virtuelle Maschinen ohne Fenster zu starten, dient das Kommandozeilentool VBoxHeadless.

Die Anwendung von VBoxHeadless ist recht einfach. Im folgenden Beispiel wird die Testumgebung gestartet:

```
VBoxHeadless --startvm Test
```

VBoxHeadless startet hierbei automatisch den VRDP-Server von VirtualBox. Dadurch ist es möglich, mit einem RDP-Client auf die virtuelle Maschine zuzugreifen. Bei mehreren Maschinen, die parallel laufen, kommt es dabei aber zu Port-Problemen. Standardmäßig wird Port 3389 verwendet. Wenn bereits eine VBoxHeadless-Session gestartet wurde, ist dieser Port belegt:

```
VBoxHeadless --startvm Test
Listening on port 3389
Error: failed to start machine. Error message: VRDP server port 3389
is already in use
```

In diesem Fall ändern Sie entweder in der Konfiguration den RDP-Zugriff oder vergeben beim Start temporär einen anderen Port:

```
VBoxHeadless --startvm Test --vrdp on --vrdpport 3400
Listening on port 3400
```

Achten Sie unbedingt darauf, dass Sie Zugriff auf die virtuellen Maschinen erhalten, denn sie können nicht korrekt über VBoxHeadless beendet werden. Auch die Eingabeaufforderung bzw. Konsole, in der VBoxHeadless gestartet wurde, darf nicht geschlossen werden – ansonsten wird die Maschine rigoros beendet, als würden Sie einfach den Stecker ziehen.

### 3.6.4 rdesktop-vrdp

Unter Linux-Hostsystemen wird ein Kommandozeilentool installiert, das als RDP-Client verwendet werden kann. Es basiert auf dem weit verbreiteten Tool `rdesktop` und heißt `rdesktop-vrdp`. Das Tool wurde für den Zugriff auf VirtualBox-Maschinen optimiert. Eine besondere Funktion ist dabei die Möglichkeit, USB-Geräte des Clientrechners an die virtuelle Maschine weiterzureichen.

#### RDP-Verbindung herstellen

```
rdesktop-vrdp ZIELRECHNER:PORT
```

Die Platzhalter ZIELRECHNER und PORT ersetzen Sie dabei durch die entsprechenden Angaben.

*Mit Rechnernamen:*

```
rdesktop-vrdp vboxserver:3400
```

*Mit IP-Adresse:*

```
rdesktop-vrdp 192.168.0.254:3400
```

### Weitere Optionen

rdesktop-vrdp bietet jedoch noch mehr Parameter, zum Beispiel wenn das falsche Tastaturlayout verwendet wird und der Nummernblock nicht korrekt funktioniert:

*Deutsche Tastatur (-k) und Numlock (-N):*

```
rdesktop-vrdp -k de -N 192.168.0.254:3400
```

Falls Sie den RDP-Zugriff auf die virtuelle Maschine mit einem Passwort gesichert haben, verwenden Sie die folgenden Parameter:

*Username (-u) und Passwort (-p):*

```
rdesktop-vrdp -k de -u dirk -p geheim 192.168.0.254:3400
```

Die Audio-Ausgabe kann an den RDP-Client umgeleitet werden:

```
rdesktop-vrdp -k de -r sound:local 192.168.0.254:3400
```

Alle Optionen zeigen Sie mit --help an:

```
rdesktop-vrdp --help
```

## 3.7 Aufbau einer virtuellen Maschine

Der Aufbau von virtuellen Maschinen ist bei den meisten Virtualisierungslösungen ähnlich; sie setzen sich folgendermaßen zusammen:

- ▶ Konfigurationsdatei
- ▶ Festplatten-Image(s)
- ▶ evtl. CD/DVD-Abbilder
- ▶ Logdateien
- ▶ gegebenenfalls Snapshots

### 3.7.1 Konfigurationsdatei

Eine virtuelle Maschine besteht mindestens aus einer Konfigurationsdatei. Diese ist ähnlich aufgebaut wie die der globalen Einstellungen. Die XML-Datei ist in mehrere Abschnitte aufgeteilt. Hier sehen Sie einen Auszug aus einer Konfigurationsdatei einer virtuellen Maschine, die unter einem Linux-Hostsystem erstellt wurde; wichtige Punkte habe ich fett markiert:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sun VirtualBox Machine Configuration -->
<VirtualBox xmlns="http://www.innotek.de/VirtualBox-
settings" version="1.7-windows">
  <Machine uuid="{8cb70952-3ed5-41ef-bd0a-
ee4dd3295c89}" name="Test" OSType="WindowsXP" lastStateChange="2009-
04-22T09:06:07Z">
    <ExtraData>
      <ExtraDataItem name="GUI/FirstRun" value="yes"/>
    </ExtraData>
    <Hardware>
      <CPU count="1">
        <HardwareVirtEx enabled="true"/>
      </CPU>
      <Memory RAMSize="192"/>
      <Boot>
        <Order position="1" device="Floppy"/>
        <Order position="2" device="DVD"/>
        <Order position="3" device="HardDisk"/>
      </Boot>
    [...]
    <Network>
      <Adapter slot="0" enabled="true" MACAddress= "080027C79FF4"
cable="true" speed="0" type="Am79C973">
        <NAT/>
      </Adapter>
    [...]
  </Network>
  [...]
  <AudioAdapter controller="AC97" driver="DirectSound" enabled="
true"/>
  [...]
</Hardware>
<StorageControllers>
  <StorageController name="IDE" type="PIIX4" PortCount="2">
    <AttachedDevice type="HardDisk" port="0" device="0">
      <Image uuid="{81faed9d-7381-48ac-adbc-bea041eec25c}"/>
    </AttachedDevice>
  </StorageController>
</StorageControllers>
</Machine>
```



```

        </AttachedDevice>
    </StorageController>
</StorageControllers>
</Machine>
</VirtualBox>

```

**Listing 3.8** Auszug aus einer Konfigurationsdatei einer virtuellen Maschine

## Hostsystem

Im ersten markierten Abschnitt erkennen Sie schon gleich einen sehr wichtigen Punkt: Die Angabe `version="1.7-windows"` verrät neben der Versionsnummer, dass die Maschine auf einem Windows-Hostsystem erstellt wurde. Diese Konfigurationsdatei kann daher nicht auf einem Linux-, Apple- oder anderen System verwendet werden. Bei der Übertragung einer virtuellen Maschine auf eine andere Plattform können Sie die Konfigurationsdatei nicht übernehmen, sondern müssen sie neu erstellen. (Profis könnten diese natürlich auch in einem Editor »konvertieren«.) Schuld sind die Eigenarten jedes Betriebssystems, auf die ich bereits hingewiesen habe und vermutlich auch noch öfter hinweisen werde (Laufwerksbuchstaben, Pfadangaben usw.).

## Universally Unique Identifier (UUID)

Der nächste Punkt zeigt den bereits in Abschnitt 3.6.1, »VBoxManage«, angesprochenen Universally Unique Identifier (UUID) gefolgt vom Namen und Gastsystem.

```

uuid="{8cb70952-3ed5-41ef-bd0a-ee4dd3295c89}" name="Test"
OSType="WindowsXP" ...

```

Der Name wird in der grafischen Oberfläche angezeigt und beim Aufruf verschiedener Kommandozeilentools verwendet. Er sollte daher eindeutig sein. Ein Verweis auf das Festplatten-Image einer Maschine wird in der Konfigurationsdatei nicht durch die Pfadangabe und den Namen des Images, sondern ebenfalls über die UUID realisiert.

## Hardware

Im Abschnitt `Hardware` sind die zugeordnete Hardware und die Systemeinstellungen aufgeführt. Im Beispiel sehen Sie die Größe des Speichers der virtuellen Maschine:

```
Memory RAMSize="192"
```

In der vollständigen Konfigurationsdatei stehen natürlich bedeutend mehr Informationen, unter anderem zu ACPI, zum Netzwerk, zur Soundkarte usw.

## HardDiskAttachments

Der Abschnitt `HardDiskAttachments` listet die UUIDs der Festplatten-Images auf, die der virtuellen Maschine zugewiesen wurden.

Die meisten dieser Abschnitte sind auch für Laien (mehr oder weniger) leicht zu lesen und zu interpretieren. Da die Konfiguration jedoch in der grafischen Oberfläche oder mit dem Kommandozeilentool `VBoxManage` vorgenommen wird, ist es nicht unbedingt wichtig, alle Abschnitte der Datei zu kennen. Bei Problemen durch Abstürze oder Ähnliches kann es jedoch hilfreich sein, die Datei in einem Texteditor zu öffnen und zu bearbeiten.

### 3.7.2 Festplatten-Images

Die meisten virtuellen Maschinen benötigen natürlich auch Zugriff auf eine Festplatte. Dieser Zugriff erfolgt durch sogenannte Festplatten-Images (auch Festplattenabbilder genannt). Ein solches Image ist, vereinfacht gesagt, eine binäre Datei, die von der Maschine wie eine Festplatte behandelt wird.

Wie bereits erwähnt, werden diese Images nicht im selben Ordner wie die anderen Informationen einer virtuellen Maschine gespeichert. Die Festplatten-Images befinden sich standardmäßig in dem Verzeichnis, das bei der globalen Konfiguration angegeben wurde. Beim Erstellen einer neuen Maschine kann dieser Ablauf jedoch manuell geändert werden.



#### Festplatten-Images anderer Virtualisierungen

VirtualBox kann übrigens auch auf Festplatten-Images der Konkurrenz zugreifen bzw. diese verwenden. Zu den kompatiblen Images gehören VMware- (VMDK) und VirtualPC-Images (VHD).

### 3.7.3 CD/DVD-Abbilder

Einer Maschine können Sie entweder einen direkten Zugriff auf das CD/DVD-Laufwerk des Hostsystems gestatten, oder ihr ein CD/DVD-Abbild zuweisen. Dies sind Abbilder von CDs bzw. DVDs im ISO-Format. Die meisten Brennprogramme ermöglichen es, von einer CD/DVD ein ISO-Abbild zu erstellen. Unter Linux eignet sich beispielsweise K3b dazu, unter Windows bietet sich unter anderem die kostenlose Software CDBurnerXP (<http://www.cdburnerxp.se/>) an.

### 3.7.4 Logdateien

Beim ersten Start von VirtualBox erstellt das Programm im Verzeichnis der virtuellen Maschinen automatisch einen Ordner mit dem Namen *Logs*. In diesem be-

findet sich dann ein Logbuch (*Vbox.log*), in dem verschiedene Informationen und gegebenenfalls auch Fehler protokolliert werden. Ein Blick in diese Protokolle kann bei Problemen sehr hilfreich sein. Außer dem aktuellen bleiben auch immer die letzten drei Logbücher erhalten (*VBox.log.1*, *VBox.log.2* und *VBox.log.3*).

### 3.7.5 Sicherungspunkte (Snapshots)

Neben dem Ordner *Logs* wird standardmäßig auch ein Verzeichnis namens *Snapshots* erstellt. In diesem werden beim Anlegen eines Sicherungspunkts die Änderungen an der virtuellen Maschine gespeichert. Ein Sicherungspunkt ermöglicht das Zurücksetzen einer Maschine auf den Zustand, den sie *vor* dem Erstellen des Sicherungspunkts besaß. Den Snapshot-Ordner können Sie in der Konfiguration der Maschine ändern.

Fakten zum Aufbau einer virtuellen Maschine

- ▶ Eine virtuelle Maschine besteht aus einer Konfigurationsdatei, Festplatten-Images, Logdateien und Snapshot-Dateien.
- ▶ Die Konfigurationsdatei liegt im XML-Format vor und verwendet das Zeilenwechselzeichen von Linux.
- ▶ VirtualBox verwendet UUIDs (Universally Unique Identifier), um jede virtuelle Maschine, jedes Festplatten-Image usw. eindeutig zu identifizieren.
- ▶ Statt CDs oder DVDs können Sie auch ISO-Abbilder als Laufwerk einbinden.
- ▶ In den Logdateien werden verschiedene Informationen und auch Fehler protokolliert.

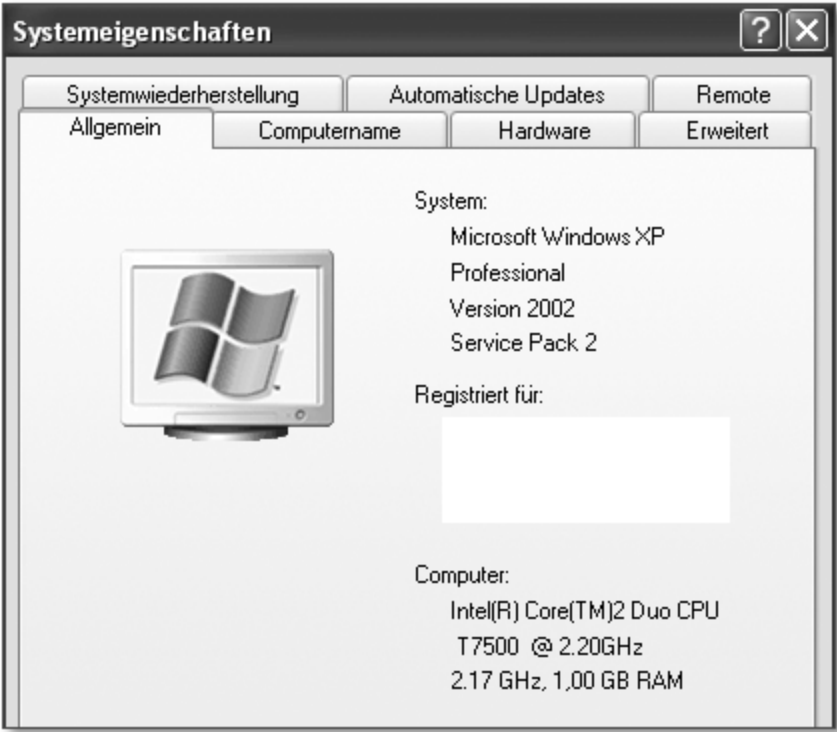
!!

## 3.8 Hardware

VirtualBox emuliert verschiedene Hardware und stellt diese der virtuellen Maschine zur Verfügung. Einige Komponenten der vorhandenen Hardware werden auch einfach an die Maschinen »durchgereicht«. In diesem Abschnitt beschreibe ich den Hardwareaufbau einer virtuellen Maschine. Ich gehe dabei aber nicht allzu sehr ins Detail, sondern gehe nur auf die wichtigsten Geräte ein.

### 3.8.1 Allgemein

VirtualBox stellt den Maschinen ein virtuelles Mainboard zur Verfügung. Der Prozessor wird hingegen nicht komplett emuliert. VirtualBox stellt den Maschinen eine CPU zur Verfügung, die der des Hostsystems entspricht. In Programmen, die Systeminformationen anzeigen, können Sie diese Daten ablesen. Abbildung 3.51 zeigt die Systemeigenschaften eines Windows-XP-Gastsystems unter einem Hostsystem, das einen Intel-Core2-Duo-Prozessor verwendet.



**Abbildung 3.51** Intel-Core2-Duo-CPU auf Host und Gast

In Abbildung 3.52 sehen Sie den Auszug einer Maschine auf einem anderen Hostsystem, das eine Intel-Xeon-CPU verwendet.



**Abbildung 3.52** Intel Xeon CPU

Daraus ergibt sich auch ein kleines Problem für Linux-Gastsysteme. Wenn Sie Profi sind und einen eigenen Kernel kompilieren, müssen Sie in Betracht ziehen, dass Ihre Maschine auf anderen Hostsystemen nicht lauffähig ist. Sie sollten daher möglichst alle CPUs mit einbinden. Die Standard-Kernel hingegen beinhalten meist alle gängigen CPUs und dürften daher, genau wie Windows-Systeme, keine Probleme bereiten.

Als Chipsatz wird der Intel 440FX emuliert. Sie haben dabei auch die Auswahl zwischen verschiedenen IDE-, SATA- und SCSI-Controllern zum Einbinden der Festplatten-Images.

Abbildung 3.53 zeigt einen Ausschnitt aus dem Windows-XP-Gerätemanager mit verschiedenen emulierten Geräte. Einige dieser Komponenten heißen *VirtualBox ...* oder *VBOX ...*, und manche Geräte tragen in den Eigenschaften sogar noch den Herstellernamen *innotek GmbH*.

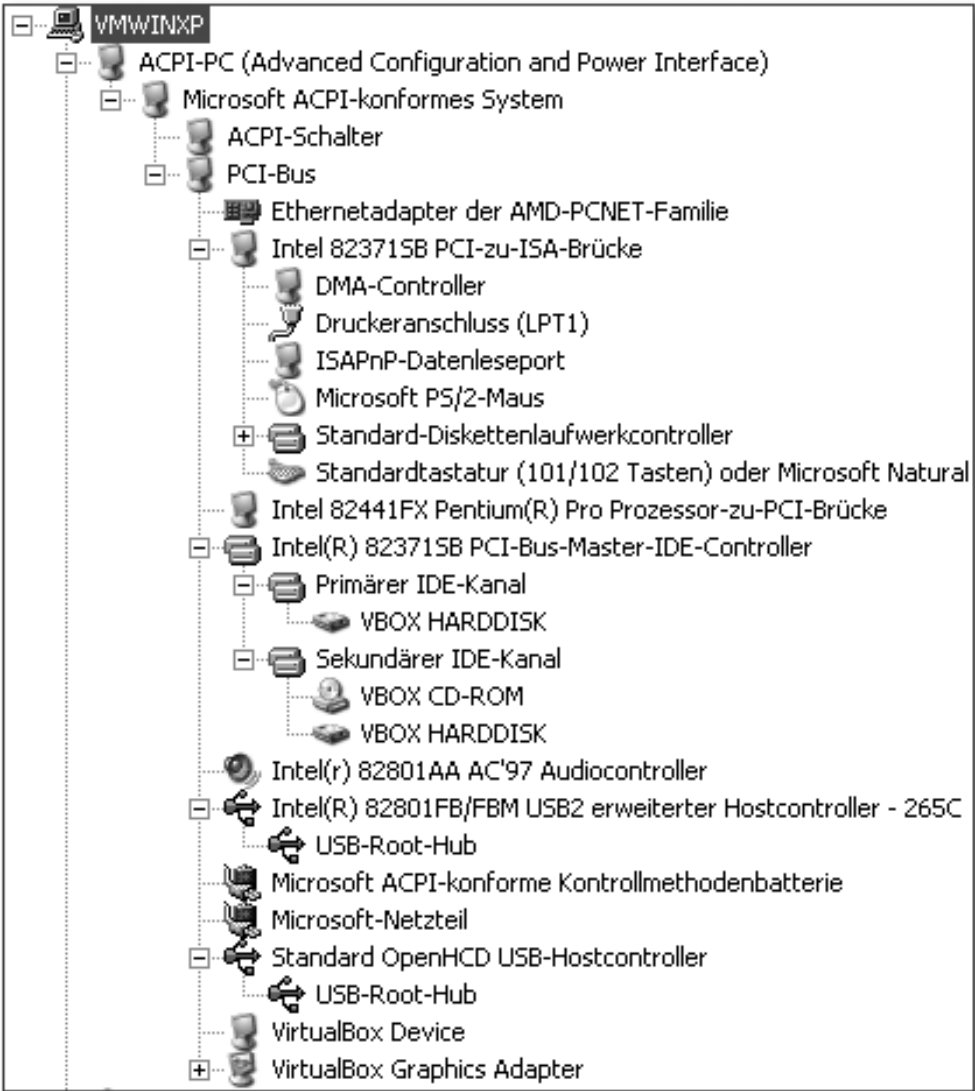


Abbildung 3.53    Gerätemanager unter Windows XP

Den Intel-440FX-Chipsatz erkennen Sie an der Teilenummer 82441FX. Als IDE-Controller (bzw. Southbridge) kommt PIIX3 zum Einsatz – zu erkennen an der Teilenummer Intel 82371SB.

Weitere Informationen bei Intel

Nähere Informationen (und auch Treiber) zu den Chipsätzen, Teilenummern usw. finden Sie auf der Homepage von Intel und dort unter »Chipsätze« (<http://www.intel.com/products/chipsets>).

[+]

### 3.8.2 Festplatten-Controller

Als Festplatten-Controller stehen die drei gängigen Arten zur Verfügung:

- ▶ IDE
- ▶ SATA
- ▶ SCSI

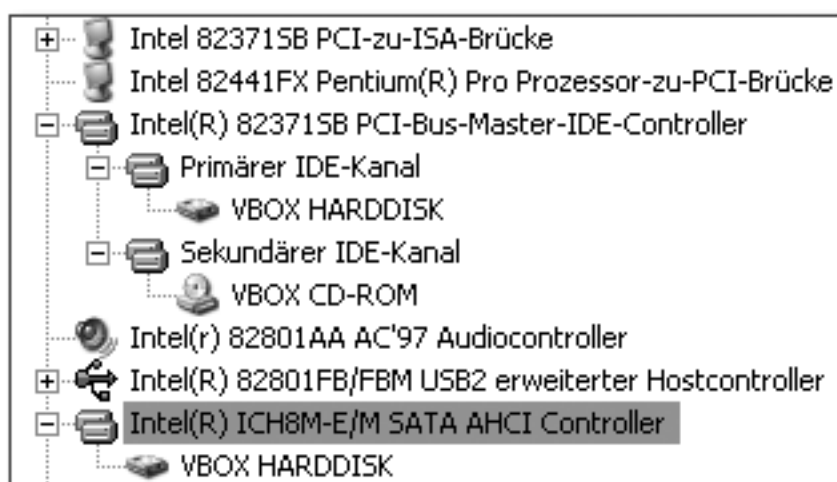
#### IDE

Für IDE können die Chipsätze PII3X, PII4X und ICH6 verwendet werden. Dabei ist zu beachten, dass der IDE-Controller im Nachhinein nicht geändert werden sollte. Windows XP ist dann zum Beispiel nicht mehr bootfähig. Der IDE-Controller ist immer aktiviert, da über ihn das CD/DVD-Laufwerk angeschlossen ist. Außer dem CD/DVD-Laufwerk können noch drei zusätzliche Festplatten-Images eingebunden werden.

#### SATA (Serial ATA)

SATA ist der Nachfolger des IDE-Controllers. Er hat allerdings IDE noch immer nicht vollständig verdrängt und wird auch nicht von allen Betriebssystemen ohne zusätzliche Treiber unterstützt. Aktuelle Treiber für verschiedene Betriebssysteme finden Sie auf der im vorigen Abschnitt angegebenen Homepage von Intel. Mit SATA können Sie bis zu 30 Festplatten-Images verwenden.

SATA wird über die *AHCI*-Schnittstelle (*Advanced Host Controller Interface*) angesprochen. AHCI ist ein offener Schnittstellen-Standard für SATA-Controller.



**Abbildung 3.54** Windows XP mit aktiviertem SATA

Der Einsatz von SATA soll gegenüber IDE einen Geschwindigkeitsvorteil bringen. Unter anderem wird die CPU-Auslastung bei Festplattenaktivitäten verringert. Ich habe selbst jedoch noch keine Tests hierzu vorgenommen.

## SCSI

Seit der Version 2.1 können Festplatten auch per SCSI eingebunden werden. Dabei stehen als Controller LSI-Logic und BusLogic zur Verfügung. Mit SCSI können Sie bis zu 16 Laufwerke einbinden.

VMware unterstützt diese beiden Controller. Daher ist es leichter, virtuelle Maschinen, die unter VMware mit SCSI eingerichtet wurden, unter VirtualBox zu migrieren. (Zur Erinnerung: Es können auch VMDK-Dateien eingebunden werden.)

### 3.8.3 Audio-Adapter (Soundkarte)

Als Audio-Adapter stehen Ihnen zwei verschiedene Geräte zur Auswahl:

- ▶ Intel AC97 (ICH AC97)
- ▶ Creative Soundblaster 16

Der Intel AC97 wird von allen aktuellen Betriebssystemen unterstützt, und auch bei der Soundblaster dürfte dies der Fall sein. Außerdem wird dieser auch von älteren Systemen unterstützt. Die Soundblaster war einer der ersten Audio-Adapter, die im Privatbereich zum Einsatz kamen. Viele Hersteller boten danach »Soundblaster-kompatible« Adapter an.

### 3.8.4 3-D-Beschleunigung

Die 3-D-Beschleunigung ist ein großes Plus von VirtualBox; allerdings arbeiten auch die Konkurrenten an einer solchen Funktion. Dieses Feature ist für Windows- und auch für Linux-Gastsysteme verwendbar. Die Entwicklung befindet sich jedoch noch im »Experimental«-Stadium. Daher kommt es bei verschiedenen Anwendungen noch zu Problemen. Die Gastsysteme können dabei seit der Version 3.0 OpenGL 2.0 verwenden. Direct3D 8 und 9 werden seit dieser Version ebenfalls unterstützt – natürlich nur unter Windows-Gastsystemen und auch nur experimentell.

Um die 3-D-Beschleunigung effektiv nutzen zu können, muss im Hostsystem auch eine OpenGL-Karte vorhanden sein. Eine weitere Voraussetzung für die 3-D-Beschleunigung ist die Installation der Gasterweiterungen.

## 3.9 Netzwerk

Eigentlich gehören die Netzwerkeigenschaften auch in den Bereich »Hardware«, aber aufgrund deren Komplexität habe ich ihnen einen eigenen Abschnitt gewidmet.

Damit die virtuelle Maschine Zugriff auf das Netzwerk erhalten kann, muss diese entsprechend konfiguriert sein. In vielen Fällen reicht hier die Standardeinstellung, die von VirtualBox bei der Neuerstellung vom Assistenten vergeben wird. Um die Einstellungen zu überprüfen oder zu ändern, wechseln Sie in die Eigenschaften der virtuellen Maschine und öffnen den Punkt NETZWERK.

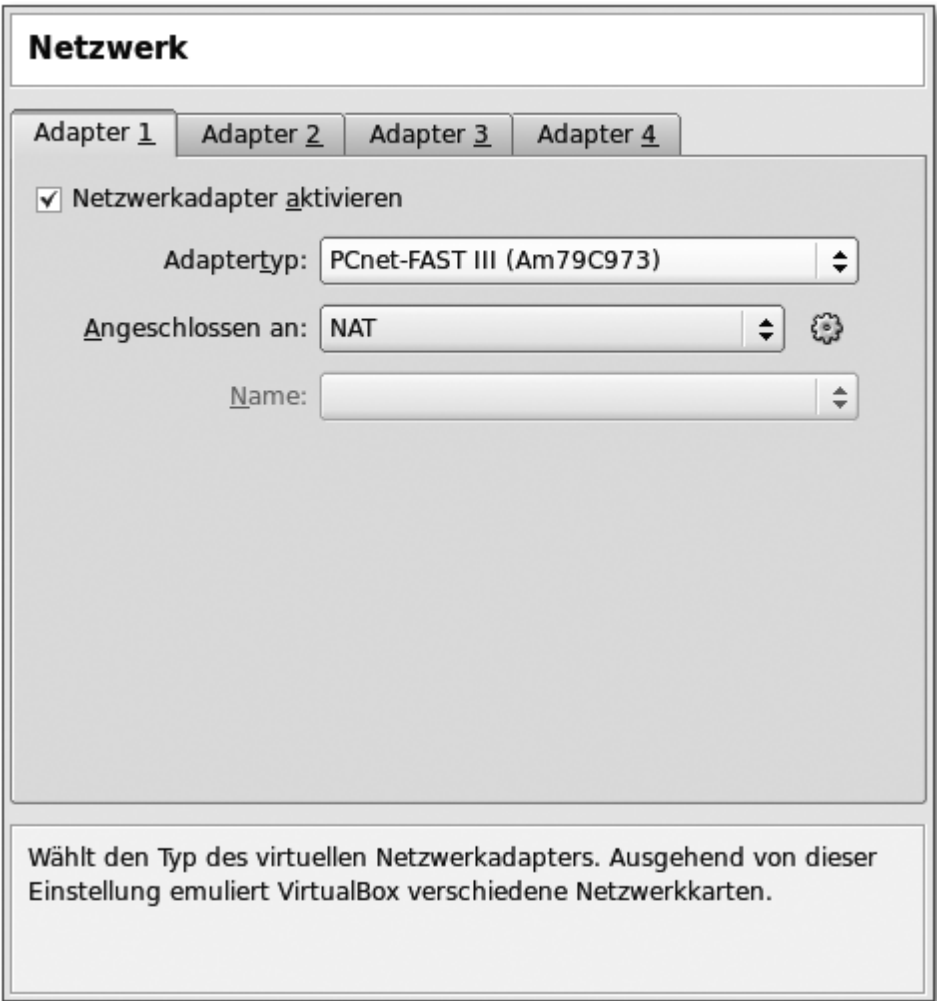


Abbildung 3.55 Die Netzwerkeigenschaften unserer Testmaschine

3.9.1 Adaptertyp (Netzwerkkarte)

In der grafischen Oberfläche einer virtuellen Maschine können Sie bis zu vier Netzwerkadapter bzw. Netzwerkkarten einrichten (Register ADAPTER 1, ADAPTER 2 usw.). Insgesamt sind in Verbindung mit VBoxManage bis zu acht Geräte möglich. VirtualBox bietet verschiedene Netzwerkadapter-Typen an (siehe auch Abbildung 3.56).

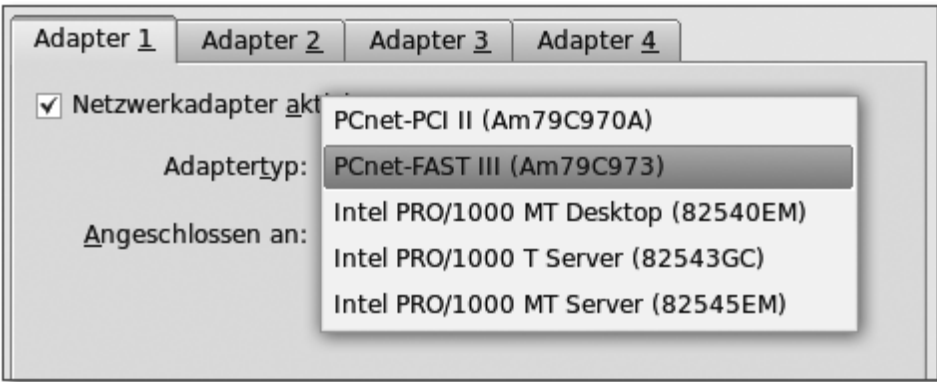


Abbildung 3.56 Auswahl der Netzwerkadapter



Welchen Adapter Sie einsetzen, hängt von Ihrem Gastsystem und dem Einsatz der virtuellen Maschine ab. Es sind unterschiedliche Modelle vorhanden, um zu gewährleisten, dass nicht nur für aktuelle, sondern auch für ältere Gastsysteme ein »bekannter« Netzwerkadapter vorhanden ist. Der Adapter kann dann meist vom entsprechenden Betriebssystem verwendet werden, ohne nachträglich Treiber installieren zu müssen.

**Die verschiedenen Adaptertypen**

- ▶ PCnet-PCI II (Am79C970A)
- ▶ PCnet-FAST III (Am79C973)
- ▶ Intel PRO/1000 MT Desktop (82540EM)
- ▶ Intel PRO/1000 T Server (82543GC)
- ▶ Intel PRO/1000 MT Server (82545EM)

Die ersten beiden Adapter sind Modelle der Firma AMD. Das Modell PCnet-PCI II ist eine 10-MBit-Karte und kann auf älteren Betriebssystemen wie zum Beispiel Windows 95 verwendet werden. Die Karte hat unter anderem einen BNC-Anschluss. Beim Modell PCnet-FAST III handelt es sich um einen Nachfolger von PCnet-PCI II, der bereits eine Datenrate von 100 MBit anbietet. Dieser Adapter kann unter Windows 2000/XP und fast allen Linux-Derivaten zum Einsatz kommen.

Die Intel-Adapter Intel PRO/1000 Desktop und Server eignen sich für aktuelle Gastsysteme; sie sind Vertreter der 1-GBit-Netzwerke. Sie kommen zum Beispiel unter Microsoft Vista oder Windows 7 zum Einsatz.

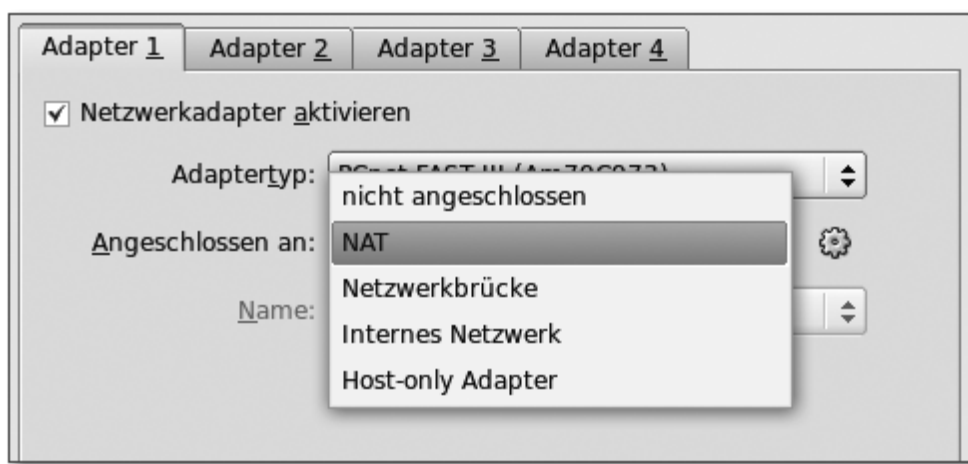
**Geschwindigkeit von Netzwerken**

Die Geschwindigkeit des Netzwerks ist natürlich von dem Netzwerkadapter des Hostsystems abhängig. Es ist logischerweise nicht möglich, eine 1-GBit-Verbindung mit dem Gastsystem aufzubauen, wenn der Hostadapter nur 100 MBit übertragen kann.

[+]

**3.9.2 Anschlussart**

Bei der Erstellung einer neuen Maschine mit dem Assistenten verwendet dieser normalerweise NAT (*Network Address Translation*) als Anschlussart, was in den meisten Fällen ausreichen dürfte.



**Abbildung 3.57** Auswahl der Anschlussarten

In Abbildung 3.57 sehen Sie alle verfügbaren und hier noch einmal aufgeführten Anschlussarten:

- ▶ nicht angeschlossen (keine Netzwerkverbindung)
- ▶ NAT (Network Address Translation)
- ▶ Netzwerkbrücke (Bridged Networking)
- ▶ Internes Netzwerk
- ▶ Host-only Adapter

NICHT ANGESCHLOSSEN bedeutet im Klartext, dass VirtualBox der Maschine den Adapter zunächst nur virtuell zur Verfügung stellt. Er verhält sich jedoch so, als wäre kein Netzkabel angeschlossen.

### Network Address Translation (NAT)

Bei NAT (Network Address Translation) verwendet die virtuelle Maschine den Adapter des Hostsystems. Alle Anfragen, die das Gastsystem ins Netzwerk stellt, laufen über die IP-Adresse des Hostsystems. Intern baut VirtualBox dabei ein virtuelles Netzwerk zwischen dem virtuellen und physikalischen Adapter auf. VirtualBox stellt dabei auch einen DHCP-Server zur Verfügung, damit die Gastsysteme nicht manuell konfiguriert werden müssen. Es ist dabei auch möglich, dass mehrere Maschinen NAT verwenden.

In Abbildung 3.58 sehen Sie (grob) den Aufbau beim Einsatz von NAT. Nach außen in das physikalische Netzwerk hat das Hostsystem die Adresse 192.168.0.10. Die drei virtuellen Maschinen verwenden alle NAT als Anschlussart und haben daher von VirtualBox über den internen DHCP-Server die Adressen 10.2.0.15–17 erhalten. VirtualBox besitzt im virtuellen Netz die Adresse 10.2.0.2 und ist gleichzeitig das Gateway für die virtuellen Maschinen. In Abbildung 3.59 sehen Sie die Ausgabe von `ipconfig` in einer Maschine mit Windows XP als Gastsystem.

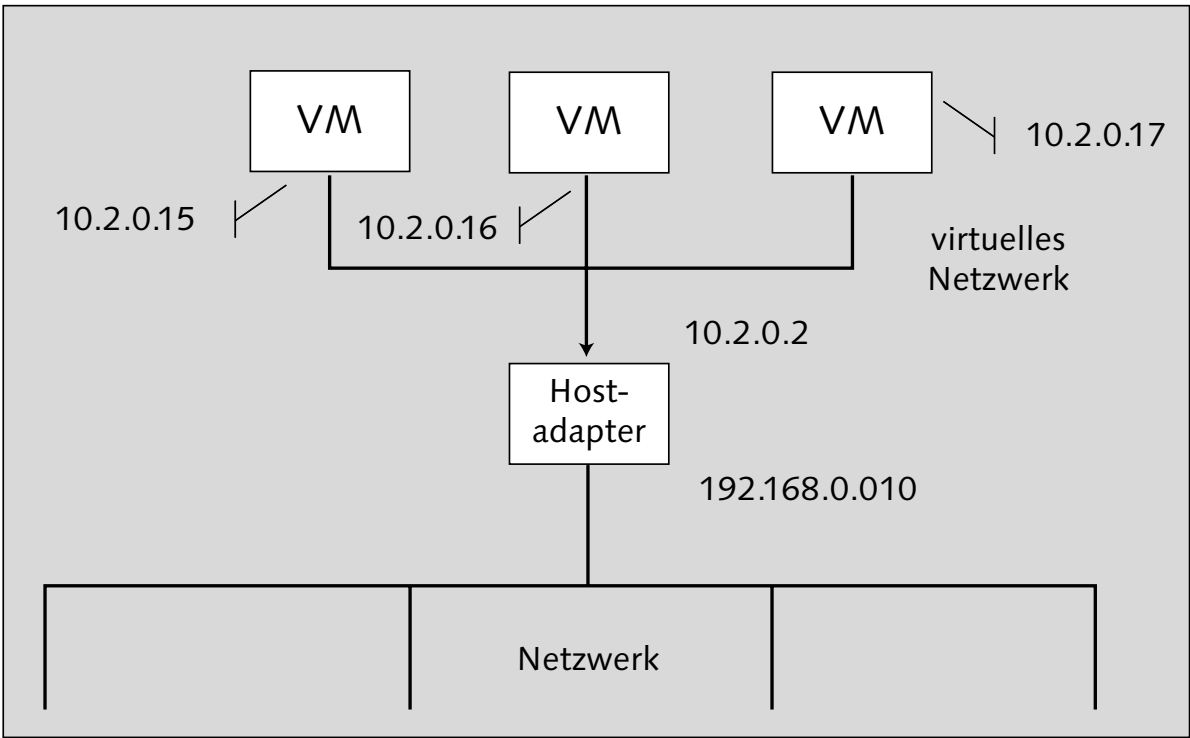


Abbildung 3.58 Schematischer Aufbau von NAT



Abbildung 3.59 ipconfig unter Windows XP

Ein Zugriff aus dem physikalischen Netzwerk auf die virtuelle Maschine ist dabei nicht bzw. nur über Umwege möglich; Informationen hierzu folgen in Abschnitt 5.12, »Port-Forwarding mit setextradata«. Ein Ping zum Beispiel wird von dem Hostsystem nicht beantwortet und erreicht nie das Gastsystem.

Netzwerkadressen bei NAT

Die Netzwerkadressen werden bei NAT wie folgt vergeben: Adapter 1 erhält die Netzwerkadresse 10.0.2.0, Adapter 2 die Adresse 10.0.3.0 usw.

[+]

Für die Vergabe dieser Adressen ist ein in VirtualBox integrierter DHCP-Server zuständig. Sie konfigurieren diesen mit VBoxManage dhcpserver (siehe Abschnitt 8.2, »VBox Manage«).

## Netzwerkbrücke

Wenn Sie eine Netzwerkbrücke für eine Maschine einrichten, erhält sie vollen Zugriff auf den entsprechenden Netzwerkadapter. Dabei muss der Maschine der physikalische Netzwerkadapter, mit dem sie arbeiten soll, zugewiesen werden (siehe Abbildung 3.60). Bei dieser Zuweisung wird eine sogenannte Netzwerkbrücke erstellt.



**Abbildung 3.60** Netzwerkbrücke mit der Schnittstelle eth0

Unter Linux-Hostsystemen werden für den Adapter der Netzwerkbrücke die Bezeichnungen *eth0*, *eth1* usw. für LAN-Karten und *wlan0*, *wlan1* usw. für WLAN-Karten verwendet. Unter Windows sind es die Gerätenamen, die Sie auch in der Detailansicht der Netzwerkverbindungen finden (siehe Abbildung 3.61).

Mit der Netzwerkbrücke verhält sich die virtuelle Maschine nun so, als hätte sie einen eigenen Netzwerkadapter. Der Einsatz für diese Anschlussart ist unter anderem bei einem Gastsystem wichtig, das Netzwerkdienste anbieten soll. Dieses kann dann zum Beispiel Ping- oder DHCP-Anfragen von Netzwerkteilnehmern beantworten. Das Hostsystem kann den zugewiesenen Adapter im Übrigen ebenfalls weiter verwenden.

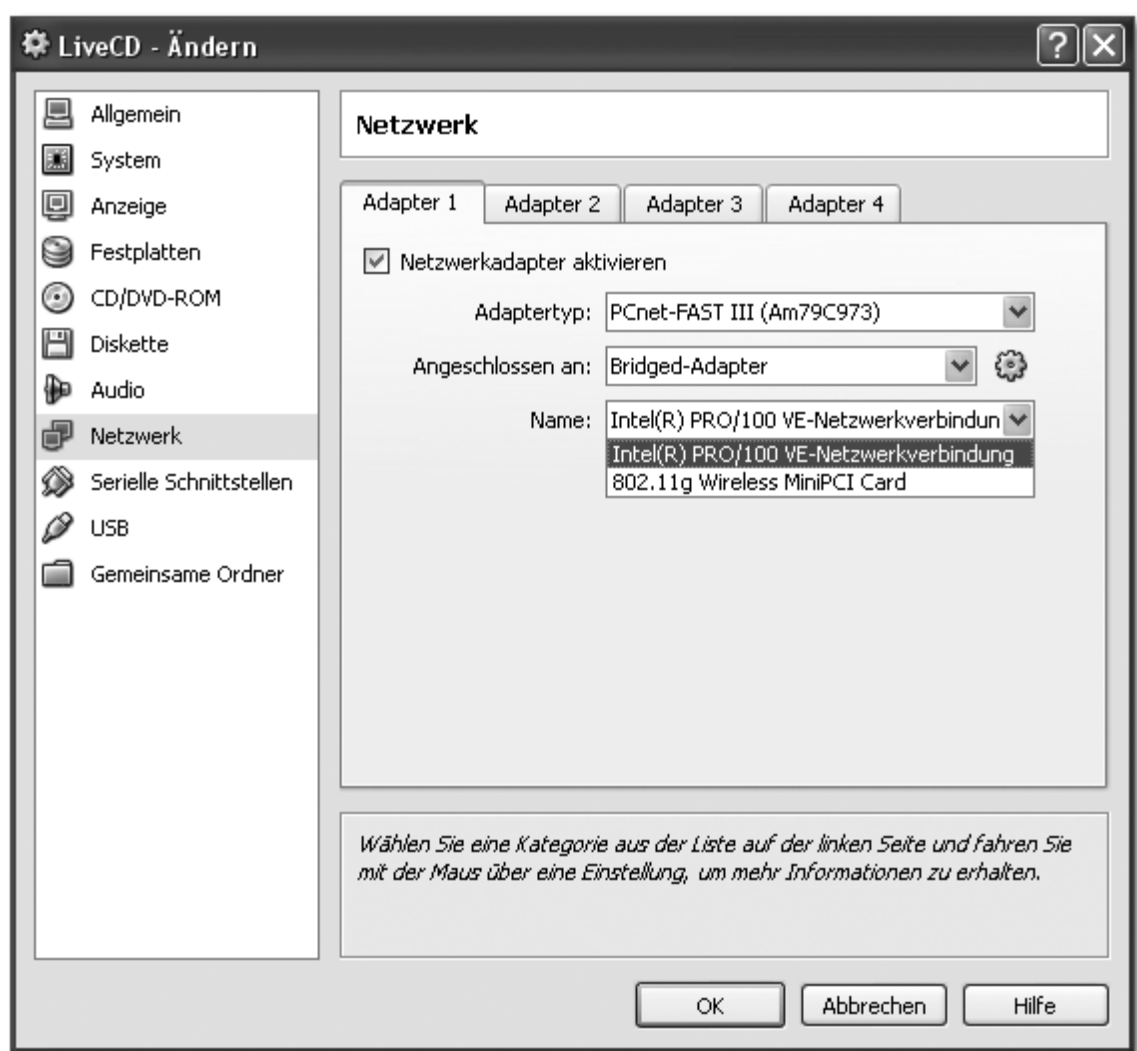


Abbildung 3.61 Netzwerkbrücke unter Windows

Bis zur Version 2.1 war die Einrichtung eines Host-Interfaces unter Linux ganz schön kompliziert. Dabei mussten noch Tap-Devices und Bridging konfiguriert werden. Inzwischen hat sich das jedoch durch ein spezielles Netzwerkmodul geändert.

**Internes Netzwerk**

Die nächste Methode, den Netzwerkzugriff zu regeln, ist die Anschlussart INTERNES NETZWERK. Mit dieser können Sie mehrere virtuelle Maschinen in einem eigenen Netzwerk betreiben. Diese haben dann keinen Zugriff auf Rechner, die sich außerhalb Ihres internen Netzwerks befinden, und von außen erhält niemand Zugriff auf eben diese. Um das interne Netzwerk zu nutzen, wählen Sie dieses bei der Anschlussart (Dropdown-Menü ANGESCHLOSSEN AN) aus und geben im Feld darunter einen Netzwerknamen dafür an.

Sämtliche Maschinen, die in diesem Netzwerk arbeiten sollen, müssen den gleichen Netzwerknamen erhalten. Es können auch mehrere interne Netzwerke mit unterschiedlichen Netzwerknamen parallel verwendet werden.

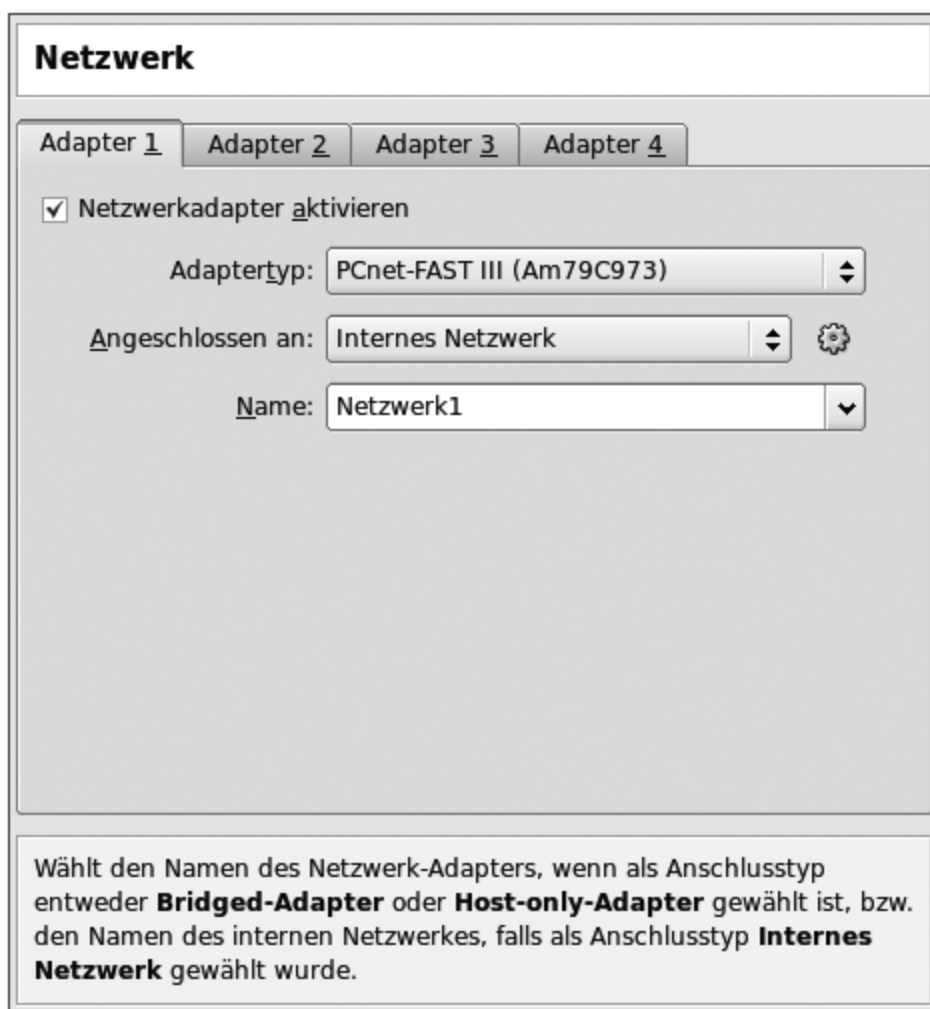


Abbildung 3.62 Internes Netzwerk einrichten

### Host-only Netzwerk

Das Host-only Netzwerk ist eine Erweiterung des internen Netzwerks und erlaubt den Datenaustausch zwischen allen virtuellen Maschinen, die dieses verwenden. Darüber hinaus ist ein Zugriff auf das Hostsystem möglich. Dabei wird auf dem Hostsystem ein virtueller Netzwerkadapter erstellt. Gleichzeitig steht auch ein virtueller DHCP-Server zur Verfügung, der die IP-Adressen der virtuellen Maschinen vergibt.

Die Konfiguration der Host-only Netzwerke erreichen Sie in der grafischen Oberfläche unter DATEI • GLOBALE EINSTELLUNGEN • NETZWERK. Über die Plus- und Minussymbole können weitere Host-only Netzwerke hinzugefügt bzw. vorhandene entfernt werden. Jedes dieser Netzwerke muss unbedingt eine eigene Netzwerkadresse erhalten. Diese darf auch nicht mit der des physikalischen Netzwerks übereinstimmen. Bei der Konfiguration sind folgende Punkte zu beachten:

- ▶ IP-Adresse der virtuellen Netzwerkkarte des Hostsystems
- ▶ IP-Adresse des virtuellen DHCP-Servers
- ▶ IP-Bereich für die virtuellen Maschinen

Die Änderung der Konfiguration erfolgt durch einen Klick auf das Schraubenschlüssel-Symbol. Wechseln Sie zunächst auf den Reiter DHCP-SERVER. Dort

geben Sie die Adresse des Servers und den IP-Bereich an (Abbildung 3.63). Bei dieser Vergabe muss zumindest eine Adresse für den Adapter des Hostsystems übrig bleiben!

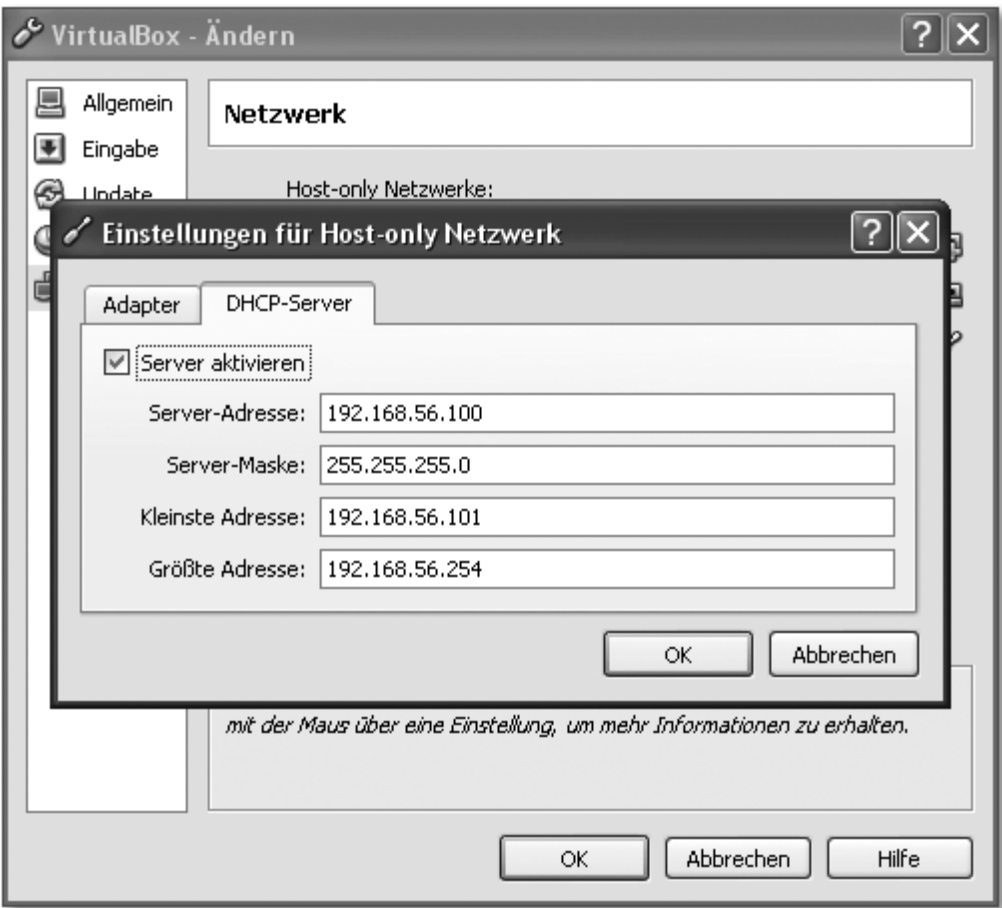


Abbildung 3.63 Den DHCP-Server einrichten

Auf dem ersten Register (ADAPTER) vergeben Sie danach die IP-Adresse des virtuellen Adapters. Diese Adresse muss sich logischerweise im selben Netzwerk befinden wie die Angaben des DHCP-Servers (siehe Abbildung 3.64).

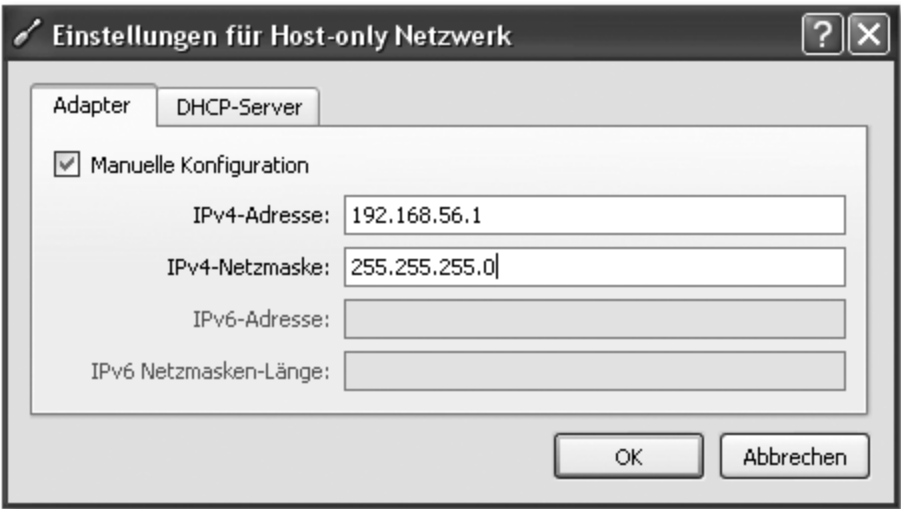
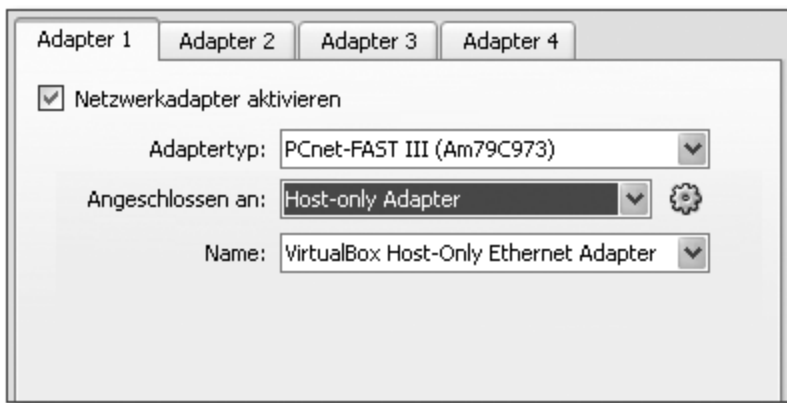


Abbildung 3.64 IP des virtuellen Adapters

Nach der Grundkonfiguration eines Host-only Netzwerks kann dieses bei den virtuellen Maschinen verwendet werden. Dazu müssen Sie nur die entsprechende Anschlussart auswählen.



**Abbildung 3.65** Host-only Netzwerk auswählen

Wenn mehrere solche Netzwerke vorhanden sind, können Sie das gewünschte im Dropdown-Menü unter NAME auswählen.

### 3.10 Gasterweiterungen

VirtualBox bietet Ihnen viele Sonderfunktionen beim Umgang mit einer virtuellen Maschine. Um diese jedoch nutzen zu können, müssen auf den Gastsystemen die Gasterweiterungen installiert werden. Diese stellen Treiber und Tools zur Verfügung, die das Arbeiten zwischen dem Gastsystem und VirtualBox optimieren. Zu den Vorteilen, welche die Gasterweiterungen bieten, gehören unter anderem:

- ▶ bessere Auflösungen durch einen Grafikkartentreiber
- ▶ gemeinsame Ordner zum Datenaustausch
- ▶ nahtloser Modus (seamless windows)
- ▶ gemeinsame Zwischenablage
- ▶ Synchronisation der Uhrzeit zwischen Host und Gast

Die Gasterweiterungen werden leider nicht für alle Gastssysteme angeboten. VirtualBox unterstützt zurzeit nur die folgenden Betriebssysteme:

- ▶ Windows NT 4.0
- ▶ Windows 2000
- ▶ Windows XP
- ▶ Windows Server 2003
- ▶ Windows Vista
- ▶ Windows 7
- ▶ Ubuntu
- ▶ Red Hat



- ▶ Suse
- ▶ openSUSE
- ▶ Fedora

#### Empfehlung

Da VirtualBox nicht alle Linux-Distributionen unterstützt, empfehle ich Ihnen, eines der aufgeführten Systeme zu verwenden, um die Vorteile der Gasterweiterungen auch unter Linux nutzen zu können. Auf den weiteren Einsatz der Gasterweiterungen gehe ich in Kapitel 4, »Praxisbeispiele«, ein.

[+]

## 3.11 Open Virtualization Format (OVF)

Seit Version 2.2 beherrscht VirtualBox auch den Im- und Export von virtuellen Maschinen. Dazu wird das Open Virtualization Format verwendet. Dieses ist ein offenes Format, das von mehreren Firmen (unter anderem Dell, Microsoft und VMware) entwickelt wurde, um Software oder Betriebssysteme, die in virtuellen Maschinen (auch Virtual Appliances genannt) laufen, auszutauschen.

Ein Beispiel zum Einsatz von OVF folgt in Abschnitt 6.2, »Open Virtualization Format«.

## 3.12 Fertige virtuelle Maschinen (Virtual Appliances)

Für VirtualBox gibt es, im Vergleich zu VMware, nur wenige Quellen mit fertigen virtuellen Maschinen (Virtual Appliances). Auf den folgenden Seiten finden Sie Maschinen, die Sie herunterladen und verwenden können, ohne das Gastsystem installieren zu müssen:

- ▶ <http://virtualboximages.com/>
- ▶ <http://virtualbox.wordpress.com/>

Auf den meisten dieser Maschinen ist Linux als Gastsystem installiert. Laufzeitbeschränkte Windows-Systeme habe ich selbst noch keine im Netz gefunden.



*Praxisbeispiele dürfen in diesem Buch natürlich nicht fehlen. In diesem Kapitel bekommen Sie Gelegenheit, VirtualBox und dessen Funktionen genauer kennen zu lernen.*

## 4 Praxisbeispiele

Die Beispiele stammen aus vielen verschiedenen Einsatzszenarien. In jedem dieser Beispiele lernen Sie immer wieder neue Funktionen, Einstellungen und Möglichkeiten von VirtualBox kennen, daher ist die Reihenfolge der Beispiele wichtig. Selbst Sie wenn ein bestimmtes Szenario nicht interessiert, sollten Sie dennoch den Abschnitt zumindest »überfliegen«.

### Hinweise zu den Beispielen

[+]

Arbeiten Sie die Praxisbeispiele unbedingt der Reihe nach durch, da in jedem dieser Abschnitte neue Funktionen und Einstellungen erklärt werden. Löschen Sie die virtuelle Maschine nach Beenden eines Abschnitts nicht, da ich in den darauf folgenden Abschnitten wieder auf diese Maschinen zurückgreife.

Die Beispielszenarien wurden natürlich alle von mir selbst installiert, eingerichtet und getestet. Allerdings werde ich bei den Praxisbeispielen nur oberflächlich auf die Installationen der Gastsysteme eingehen. Diese Themen würden den Umfang des Buchs sprengen.

Ich werde Ihnen in den Praxisbeispielen auch den Einsatz von VBoxManage näherbringen. Dazu führe ich die entsprechenden Befehle an. Die Befehle müssen dabei immer in einer einzigen Zeile ohne Umbruch eingegeben werden – selbst wenn im Listing ein Zeilenwechsel vorkommt (siehe Abschnitt 1.1.2, »Hinweise zum Umgang mit dem Buch«, unter dem Stichwort »Backslash«).

### 4.1 Live-CDs testen

In diesem Abschnitt werden Sie unter anderem Folgendes kennen lernen:

- ▶ eine neue virtuelle Maschine erstellen
- ▶ Einbinden von CDs bzw. DVDs
- ▶ Starten und Beenden von virtuellen Maschinen
- ▶ Funktion der Host-Taste

- Fernsteuerung von Maschinen (Remotezugriff)

Viele von Ihnen lesen sicherlich Computerzeitungen und haben auch die eine oder andere abonniert. Mit den Zeitschriften werden immer wieder sogenannte Live-CDs, vorwiegend mit Linux-Systemen, angeboten. Diese sind für Windows-Benutzer eine gute Möglichkeit, das Betriebssystem Linux kennen zu lernen, ohne es installieren zu müssen. Das »Problem« an Live-Systemen ist, dass der Rechner neu gestartet werden muss. Mit VirtualBox kann dagegen eine Live-CD ohne Neustart in der Windows-Umgebung gestartet werden. Auch Linux-Anwender können diese Möglichkeiten nutzen, um neue Versionen ihrer Distribution zu testen oder verschiedene Systeme miteinander zu vergleichen.

[+]

#### ISO-Images verwenden

Es ist auch möglich, anstelle von gebrannten CDs/DVDs ein ISO-Image zu verwenden. Zum Einbinden von ISO-Images kommen erfahren Sie mehr in Abschnitt 4.2.2, »Manager für virtuelle Medien«.

### 4.1.1 Quellen

Für dieses Beispiel benötigen Sie eine Live-CD. Wie bereits erwähnt, finden Sie in vielen Computerzeitungen immer wieder aktuelle Systeme, vorwiegend mit der Linux-Distribution Ubuntu. Wenn Ihnen jedoch keine CD zur Verfügung steht, können Sie von den folgenden Webseiten ein Image herunterladen. Ich empfehle Ihnen die Distribution Ubuntu, da mit dieser Distribution in einem folgenden Abschnitt eine Testumgebung eingerichtet wird.

- Ubuntu (Desktopversion): kann als Testumgebung eingesetzt oder als System installiert werden, circa 700 MByte, <http://www.ubuntu.com/getubuntu/download>
- Knoppix: Distribution, die zum Testen von Linux und als Rettungssystem eingesetzt werden kann, circa 700 MByte, <http://www.knopper.net/knoppix>
- LernTux: eine Lernumgebung für Kinder und Jugendliche, circa 700 MByte, <http://www.lerntux.de/>
- Parted Magic: ein System mit grafischer Oberfläche zum Partitionieren und Sichern von Festplatten, circa 80 MByte, <http://partedmagic.com/>
- Damn Small Linux: eine »verdammt kleine Linux-Distribution«, die auch auf älteren Systemen mit veralteter Hardware läuft, circa 50 MByte, <http://www.damnsmalllinux.org>

Wählen Sie zum Download möglichst immer ein ISO-Image aus und brennen Sie dieses nach dem Herunterladen auf eine CD.

### 4.1.2 Virtuelle Maschine einrichten

Um unser Live-System zu testen, benötigen wir nun eine virtuelle Maschine. Hierzu starten Sie die grafische Oberfläche und wählen in der Symbolleiste den Button NEU. Danach startet der Assistent zur Erstellung von virtuellen Maschinen (Abbildung 4.1).



**Abbildung 4.1** Assistent zum Erstellen neuer virtueller Maschinen

Mit einem Klick auf WEITER gelangen Sie zum nächsten Schritt. In diesem können Sie entscheiden, wie die virtuelle Maschine heißen und welches Gastsystem darauf laufen soll (Abbildung 4.2).

Der Typ des Gastbetriebssystems ist wichtig, da VirtualBox anhand dieser Auswahl bestimmte Voreinstellungen trifft, um die virtuelle Maschine für das entsprechende Betriebssystem zu optimieren. Der Typ ist dabei unterteilt in das BETRIEBSSYSTEM und die eigentliche VERSION. Falls die von Ihnen gewählte Distribution nicht in der Liste auftaucht, wählen Sie eine entsprechende Kernel-Version.



Abbildung 4.2 Name und Typ des Gastsystems eingeben

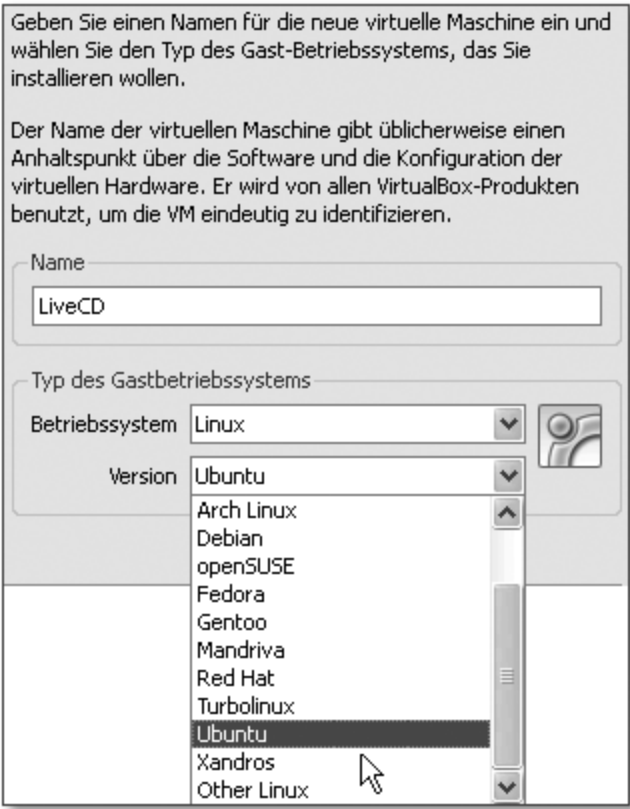


Abbildung 4.3 VirtualBox bietet viele Linux-Systeme in der Liste an.

Ich habe mich für die Distribution Ubuntu entschieden und wähle hier den entsprechenden Eintrag. Wenn Sie sich unsicher sind, funktioniert meist auch Linux 2.6 (für den aktuellen Kernel 2.6).

Danach legen Sie fest, wie viel Speicher der virtuellen Maschine zugeteilt werden soll (Abbildung 4.4). Dabei sollten Sie, abhängig von dem Speicher des Hostsystems, darauf achten, was der Hersteller des Betriebssystems empfiehlt. Falls VirtualBox die Version kennt, schlägt das Programm selbst eine passende Größe vor.

Beachten Sie bei Änderungen, dass für Ihr Hostsystem genügend Hauptspeicher übrig bleibt! Zum Festlegen der Größe können Sie den Schieberegler verwenden oder eine manuelle Eingabe vornehmen.



Abbildung 4.4 Den Speicher für die virtuelle Maschine festlegen

Falls Sie der virtuellen Maschine zu viel Speicher zuweisen, deaktiviert Virtual-Box den Button WEITER.



Abbildung 4.5 VirtualBox achtet darauf, dass Sie nicht zu viel Speicher verteilen.

In Abbildung 4.5 wurde zu viel Speicher zugewiesen, und der Button WEITER ist ausgegraut.

Als Nächstes bietet der Assistent an, eine virtuelle Festplatte für das Gastsystem zu erstellen. In unserem Fall ist dies jedoch unnötig, da die Live-CD keine Festplatte benötigt. Entfernen Sie dazu das Häkchen vor BOOT FESTPLATTE. VirtualBox warnt nun und fragt nach, ob Sie sich sicher sind (Abbildung 4.6). Bestätigen Sie diese Warnung mit einem Klick auf FORTFAHREN.



Abbildung 4.6 VirtualBox warnt vorsichtshalber, wenn Sie keine Festplatte auswählen.

Nun ist die neue virtuelle Maschine fertig gestellt, und der Assistent zeigt Ihnen eine Zusammenfassung Ihrer Einstellungen an (Abbildung 4.7).



Abbildung 4.7 Zusammenfassung der neuen Maschine



Ganz fertig sind wir jedoch noch nicht, da das CD-Laufwerk noch nicht eingebunden ist. Dies sehen Sie in der Detailansicht der virtuellen Maschine (Abbildung 4.8)



Abbildung 4.8 Es wurde keine CD zugewiesen.

Vergewissern Sie sich, dass die CD im Laufwerk liegt, und wählen Sie in der grafischen Oberfläche den Link CD/DVD-ROM. Weisen Sie der Maschine das CD/DVD-Laufwerk des Hostsystems zu.

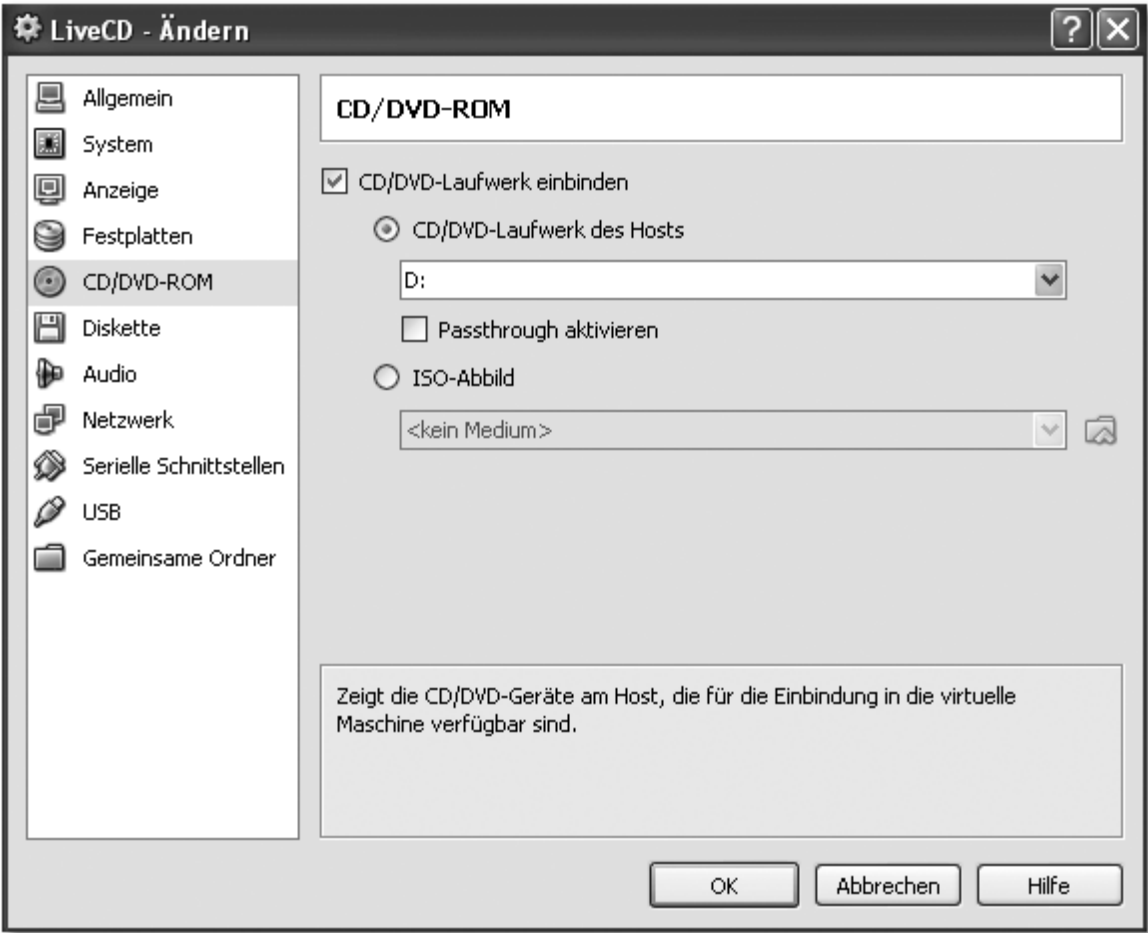


Abbildung 4.9 Ein physikalisches Laufwerk zuordnen

Die Option PASSTHROUGH AKTIVIEREN ist hierbei nicht unbedingt notwendig. Sie ermöglicht dem Gastsystem einen direkten Zugriff auf das Laufwerk. Wenn das CD-Laufwerk zum Beispiel ein Brenner ist und Sie diesen in der virtuellen Umgebung auch als Brenner nutzen wollen, aktivieren Sie die Option.

Bei einem Linux-Hostsystem ist die Zuordnung ein wenig anders als unter Windows. Da Linux keine Laufwerksbuchstaben verwendet, wird dort bei der Einbindung die Laufwerksbeschreibung mit dem Device-Namen verwendet.



Abbildung 4.10 Das Laufwerk ist unter Windows eingebunden.

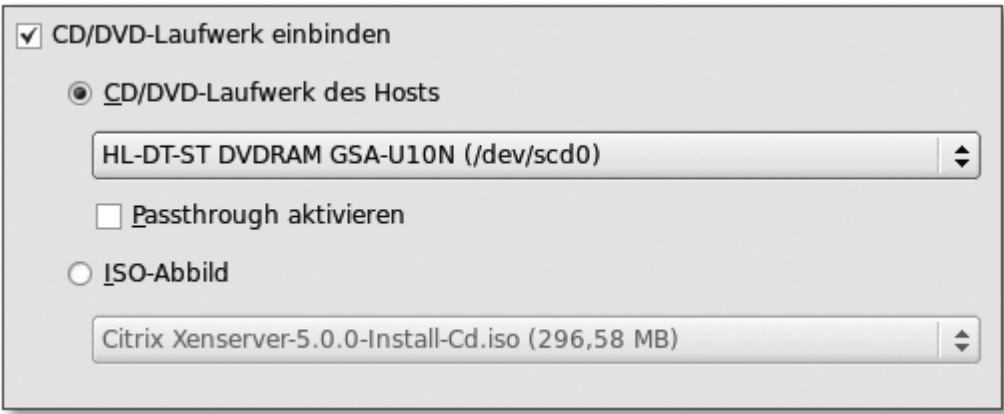


Abbildung 4.11 Laufwerkszuordnung unter Linux

**VBoxManage – CD/DVD-Laufwerk zuordnen**

- ▶ Windows-Hostsystem:  
`VBoxManage modifyvm LiveCD --dvd host:D:`
- ▶ Linux-Hostsystem:  
`VBoxManage modifyvm LiveCD --dvd host:/dev/scd0`

Nebenbei hat VirtualBox der Maschine automatisch noch eine Netzwerkkarte zugeordnet. Sofern Ihr Hostsystem über einen Internetzugang verfügt, können Sie diesen auch im Gastsystem nutzen. Auch ein Datenaustausch ist mit ein wenig Linux-Kenntnissen möglich.



Abbildung 4.12 Ein Netzwerkadapter wurde gleich mit eingerichtet.

**4.1.3 Live-System starten**

Durch einen Klick auf den Button STARTEN können wir unser erstes Praxisbeispiel starten. Zunächst erscheint noch eine Meldung über die Host-Taste. Diese Taste ist notwendig, um die Maus und Tastatur wieder freizugeben und spezielle Funktionen aufzurufen (siehe auch Abschnitt 3.5.2, »Globale Einstellungen«). Wenn

Sie Ubuntu verwenden, werden Sie bemerken, dass der Mauszeiger in dem Fenster der Maschine »gefangen« ist. Durch Betätigen der Host-Taste wird dieser wieder »befreit«.



**Abbildung 4.13** VirtualBox zeigt Information über die Host-Taste an.

Wenn Sie ein Häkchen vor **DIESE MELDUNG SPÄTER NICHT MEHR ZEIGEN** setzen, wird die Information nicht mehr angezeigt. In der Statusleiste (unten rechts) erscheint weiterhin die Information, welche Taste Ihre Host-Taste ist.

Nach dem Schließen des Informationsfensters setzt VirtualBox den Start fort, und das Startmenü von Ubuntu erscheint nach kurzer Zeit auf dem Bildschirm (Abbildung 4.14).

### **VBoxManage – virtuelle Maschine starten**

Alle Systeme:

```
VBoxManage startvm LiveCD
```

Nun können Sie das Live-System nach Belieben testen (Abbildung 4.15). Denken Sie auch daran, dass Sie die Maus und Tastatur nur durch Betätigen der Host-Taste befreien können.



Abbildung 4.14 Das Startmenü von Ubuntu

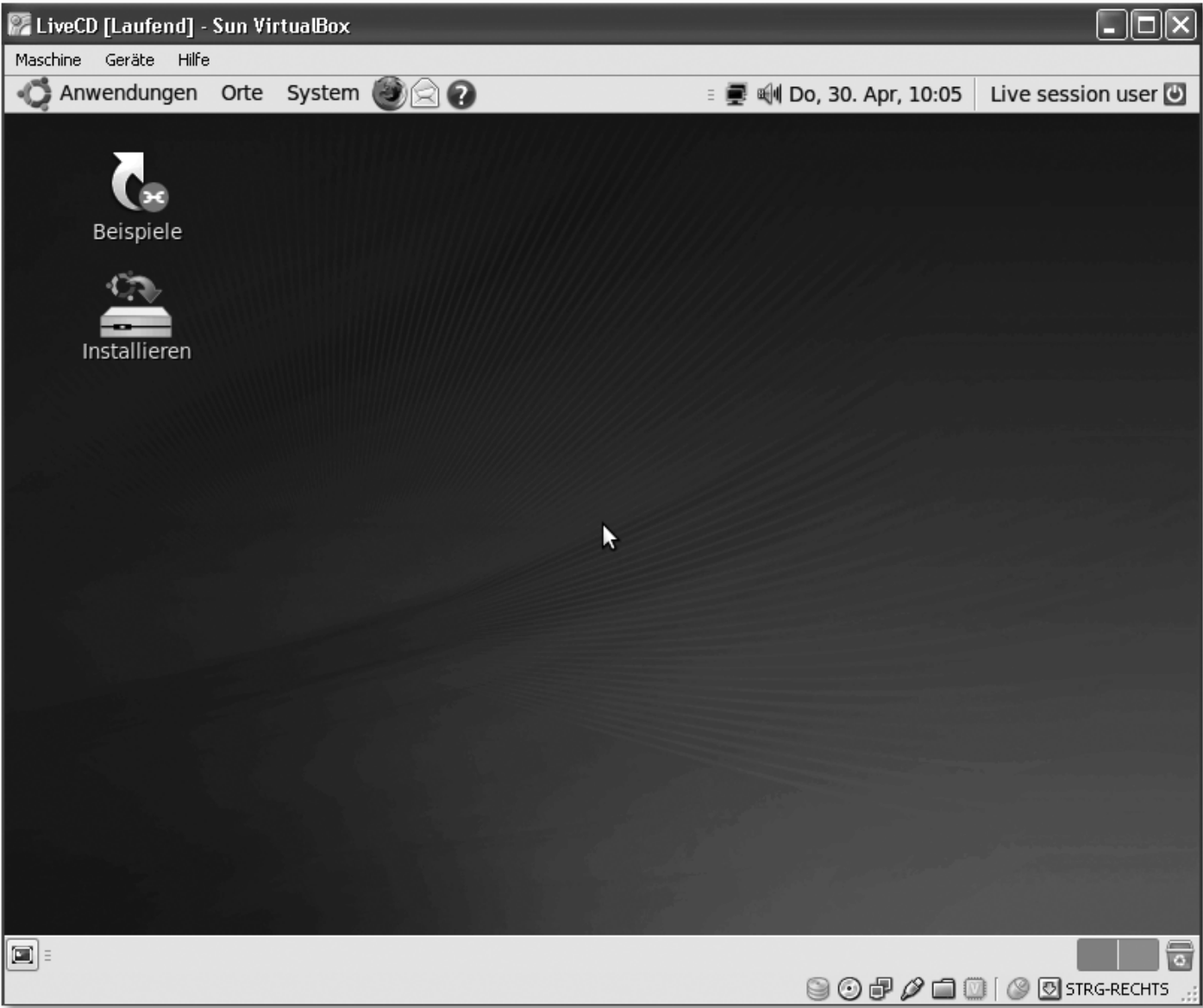


Abbildung 4.15 Der Ubuntu-Desktop nach dem Start der Live-CD

Das Ganze funktioniert natürlich auch unter einem Linux-Hostsystem. In Abbildung 4.16 sehen Sie die Distribution Ubuntu als Live-System in einer virtuellen Maschine, die auf einem Ubuntu-Hostsystem gestartet wurde.

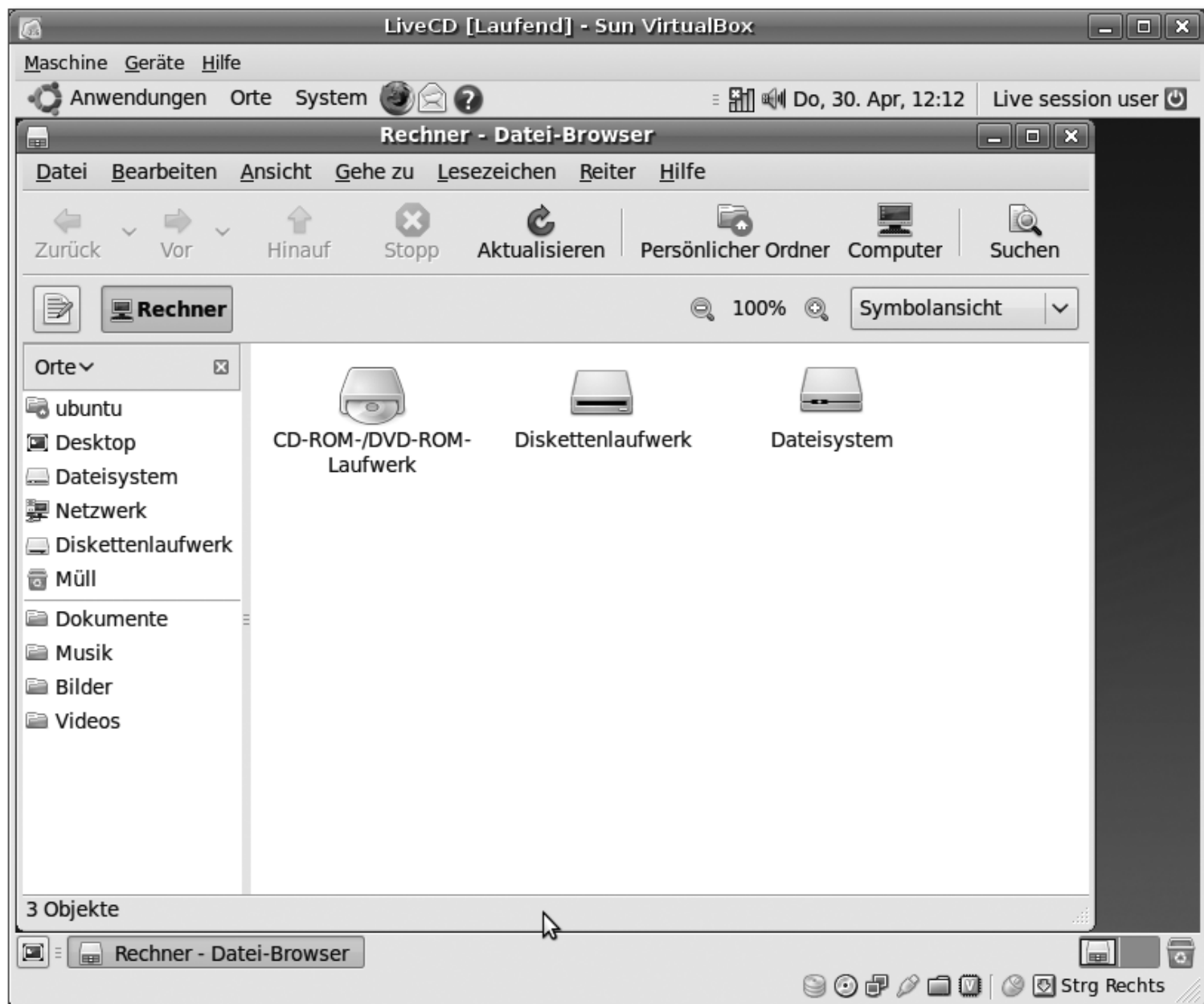


Abbildung 4.16 Ubuntu unter Ubuntu

#### 4.1.4 Beenden der virtuellen Maschine

Wenn Sie nun ihr Live-System ordnungsgemäß herunterfahren, wird in den meisten Fällen auch die virtuelle Maschine beendet – genau wie bei einem physikalischen PC. Wenn dies nicht der Fall sein sollte oder die Maschine aus anderen Gründen beendet werden soll, können Sie dies auf mehrere Arten durchführen:

- ▶ Verwenden Sie das Menü MASCHINE • SCHLIESSEN.
- ▶ Drücken Sie die Host-Taste und **Q** (zum Beispiel **Strg** + **Q**).
- ▶ Schließen Sie das Fenster.



**Abbildung 4.17** Beenden der virtuellen Maschine

In allen Fällen erscheint ein Dialog, in dem Sie festlegen, auf welche Art die Maschine beendet werden soll (Abbildung 4.17):

**DEN ZUSTAND DER VIRTUELLEN MASCHINE SPEICHERN:** Der Speicherinhalt wird auf der Festplatte abgelegt, und die Maschine wird beendet. Dadurch wird die Maschine beim nächsten Start an derselben Stelle fortgeführt. Dies ist ähnlich wie der Ruhezustand bei Notebooks. Infos zu dessen Einsatz folgen in Abschnitt 5.4, »Ruhezustand«.

**DIE VIRTUELLE MASCHINE MITTELS ACPI-EVENT HERUNTERFAHREN** sendet dem Gastsystem ein Signal, dass es sich beenden soll. Dies funktioniert jedoch nicht mit allen Systemen und natürlich nicht, wenn sich das Betriebssystem in einem nicht stabilen Zustand befindet.

**DIE VIRTUELLE MASCHINE AUSSCHALTEN** schaltet die Maschine aus. Diese Funktion ist vergleichbar mit dem Ausschalten am Netzteil oder mit dem Ziehen des Netzsteckers.

### **VBox Manage – virtuelle Maschine beenden**

Alle Systeme:

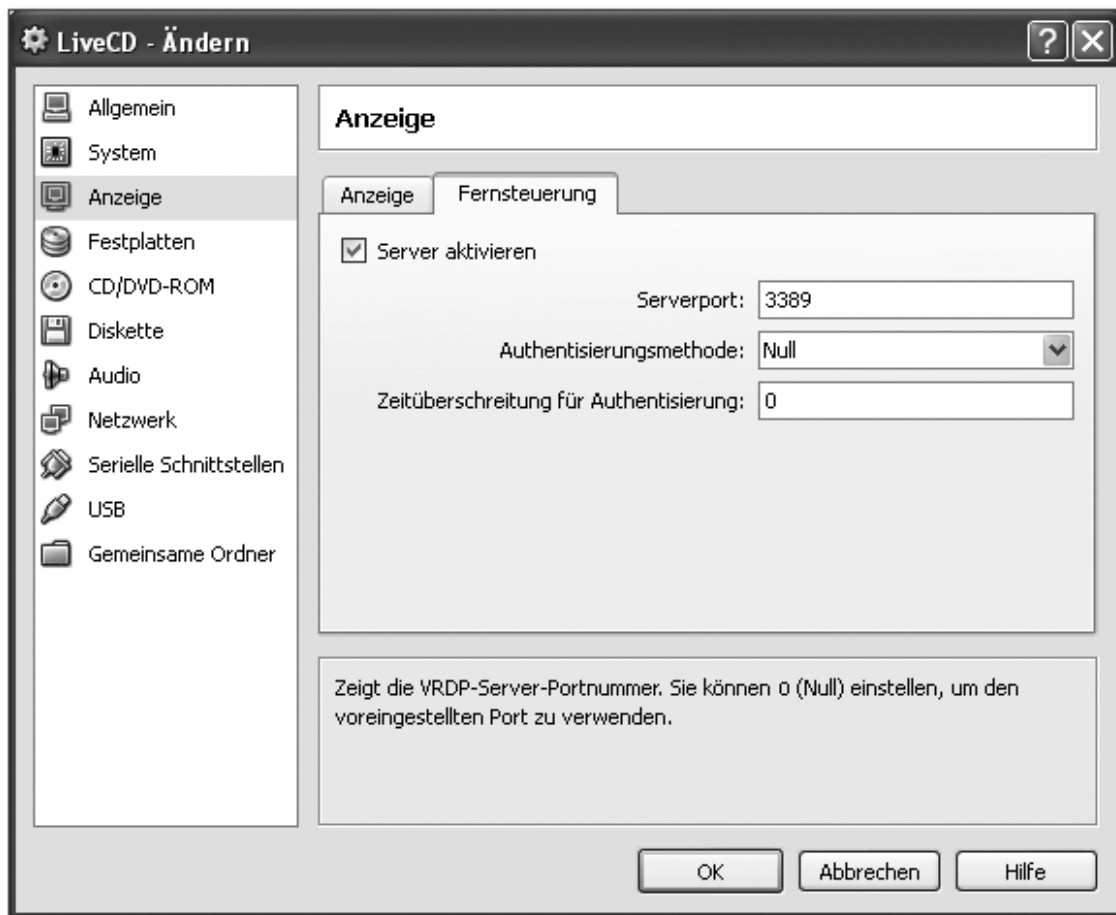
```
VBoxManage controlvm LiveCD acpipowerbutton
VBoxManage controlvm LiveCD poweroff
```

#### **4.1.5 Fernsteuerung (Remotezugriff)**

Beim Start der Live-Systeme hat VirtualBox ein eigenes Fenster für die Steuerung der virtuellen Maschine geöffnet. Dadurch kann die Maschine vom Desktop aus verwendet werden. Dies dürfte in den meisten Fällen auch ausreichend sein. VirtualBox bietet jedoch noch eine weitere Möglichkeit, auf eine laufende virtuelle Maschine zuzugreifen: die Fernsteuerung über einen RDP-Zugriff. Das *RDP* (*Remote Desktop Protocol*) ist ein von Microsoft entwickeltes Netzwerkprotokoll und wird zum Fernsteuern (»Remotearbeiten«) von Windows-Systemen verwendet.

Damit ist es möglich, einen Fernzugriff mit einem Desktoprechner oder Terminalserver aufzubauen und an diesem über das Netzwerk oder Internet zu arbeiten. Für die meisten Betriebssysteme werden dafür entsprechende Clientprogramme angeboten.

In VirtualBox kann nun ein Remotezugriff auf die virtuelle Maschine eingerichtet werden – und zwar unabhängig vom Gastsystem. Wechseln Sie dazu wieder in die Konfiguration und dort zum Punkt ANZEIGE in das Register FERNSTEUERUNG.



**Abbildung 4.18** Die Fernsteuerung aktivieren

Standardmäßig bietet VirtualBox unter SERVERPORT DEN Port 3389 an. Dies ist auch der Standard für RDP-Verbindungen. Wenn Sie mehrere Maschinen gleichzeitig starten, vergeben Sie hier für jede einen anderen Port. Die AUTHENTISIERUNGSMETHODE und ZEITÜBERSCHREITUNG belassen Sie bei den Defaultwerten.

Die Maschine starten Sie wie gewohnt. Öffnen Sie dann Ihren Remotedesktop-Client. Unter Windows XP befindet sich dieser unter START • ALLE PROGRAMME • ZUBEHÖR • KOMMUNIKATION und nennt sich REMOTEDESKTOPVERBINDUNG. Für Linux steht neben *tsclient* oder *rdesktop* auch ein von VirtualBox installierter Client mit dem Namen *rdesktop-vrdp* zur Verfügung. Der Zugriff erfolgt über die IP-Adresse bzw. den Rechnernamen des Hostsystems und die vergebene Port-Nummer, getrennt durch einen Doppelpunkt. Wenn Sie vom Hostsystem aus zugreifen, können Sie auch *localhost* als Namen verwendet werden (Abbildung 4.19). Ein Benutzername, Kennwort oder sonstige Eingaben sind nicht notwendig.



Abbildung 4.19 Remotedesktopverbindung unter Windows XP

In Abbildung 4.20 sehen Sie einen Fernzugriff mit *tsclient* unter Ubuntu. Im Vordergrund ist das Fenster von *tsclient* und im Hintergrund das von VirtualBox – beide natürlich mit der gleichen Anzeige.

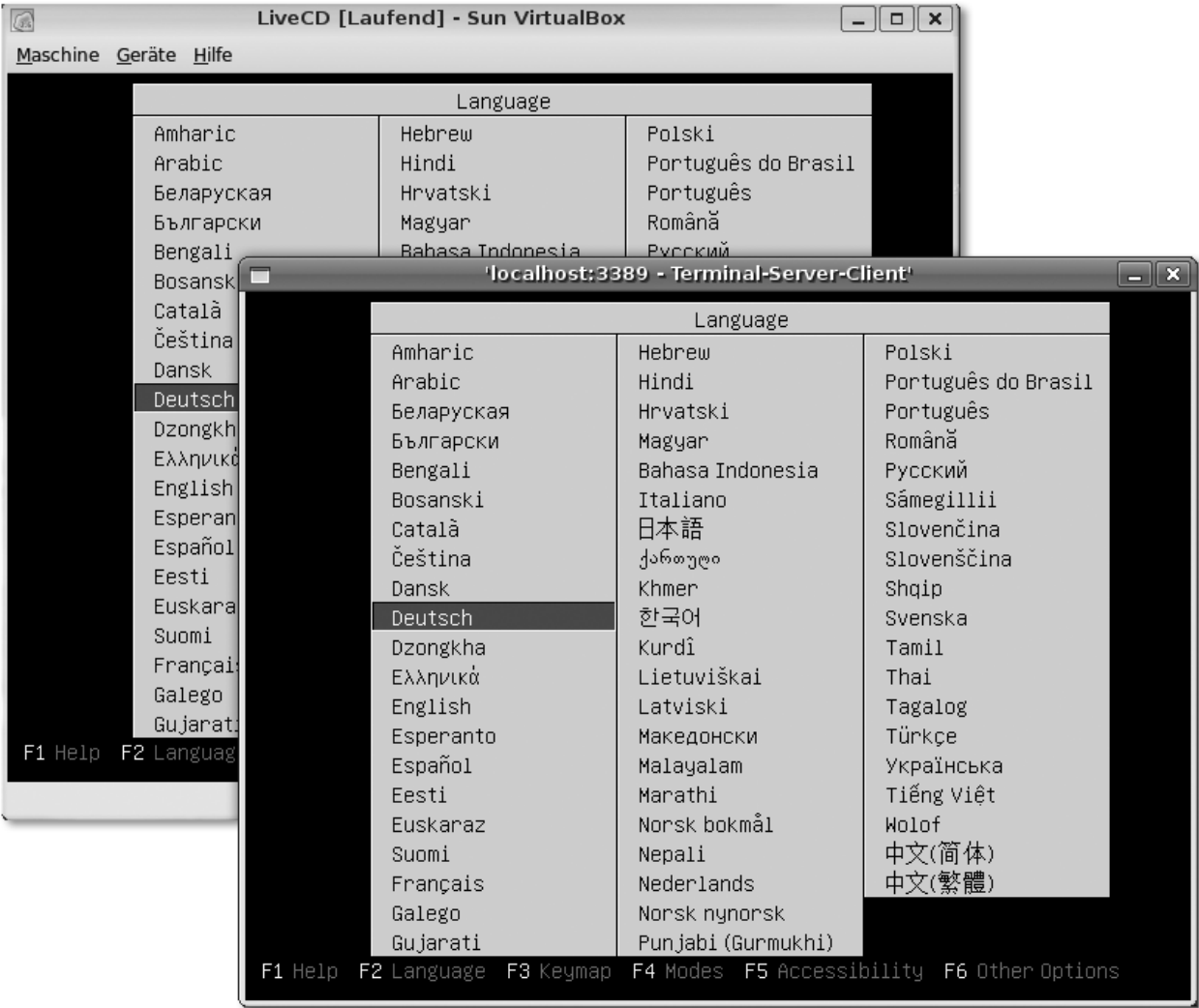


Abbildung 4.20 RDP unter Linux



Bei der Verwendung von RDP haben Sie jedoch ein paar Einschränkungen und gegebenenfalls mit Problemen zu rechnen:

- ▶ Die Host-Taste kann nicht verwendet werden (die Eingabegeräte werden jedoch auch nicht »gefangen«).
- ▶ Es gibt keine Möglichkeit, Geräte oder Ähnliches einzubinden.
- ▶ Verschiedene Sondertasten (`[Alt]`, `[Strg]` usw.) bereiten manchmal Probleme (nur unter einigen RDP-Clients).

### **VBoxManage – Fernzugriff**

- ▶ Alle Systeme, RDP aktivieren:  
`VBoxManage modifyvm LiveCD --vrdp on --vrdpport 3390`
- ▶ Alle Systeme, RDP deaktivieren:  
`VBoxManage modifyvm LiveCD --vrdp off`

## **4.2 Windows XP**

Nachdem Sie nun grundlegende Elemente der grafischen Oberfläche und verschiedene Funktionen kennen gelernt haben, kommen wir nun zum nächsten Praxisbeispiel. Ich gehe davon aus, dass jeder Leser dieses Buchs eine Windows-XP-Version zur Hand hat; falls nicht, kann auch eine Windows-2000-Version verwendet werden.

In diesem Abschnitt werden Sie Folgendes kennen lernen:

- ▶ Festplatten-Images erstellen
- ▶ Manager für virtuelle Medien (ISO-Images) verwenden
- ▶ Gasterweiterungen unter Windows installieren
- ▶ Netzwerkzugriff mit NAT (Network Address Translation)

### **4.2.1 Virtuelle Maschine erstellen**

Der Ablauf zum Erstellen der Maschine ist der gleiche wie im vorherigen Abschnitt. Im Unterschied zum Live-System benötigen wir nun aber ein Festplatten-Image, um Windows zu installieren. Dieses werden wir mit Hilfe des Assistenten erstellen.

Wählen Sie nun in der Symbolleiste wieder den Button NEU, um den Assistenten zu starten. Geben Sie im ersten Schritt das korrekte BETRIEBSSYSTEM und die VERSION an.



Abbildung 4.21 Betriebssystem und Version auswählen

Im nächsten Schritt legen Sie die Größe des Speichers fest. Für Windows XP sollten circa 384–512 MByte ausreichend sein. Denken Sie wieder daran, Ihrem Hostsystem genügend eigene Ressourcen übrig zu lassen.

Nun folgt die Einrichtung der ersten virtuellen Festplatte. Wenn bereits ein Festplatten-Image vorhanden ist, können Sie dieses auswählen (Abbildung 4.22).



Abbildung 4.22 Virtuelle Festplatte einbinden

Wir werden jedoch durch einen Klick auf den Button NEU ein neues, leeres Image erstellen. Danach entscheiden Sie, wo die virtuelle Festplatte abgespeichert wird, wie groß sie sein soll und ob sie diese Größe sofort zugeteilt bekommen oder dynamisch wachsen soll.



Abbildung 4.23 Typ der Festplatte auswählen

In den meisten Fällen dürfte ein dynamisch wachsendes Medium ausreichend sein. Dieses spart Speicherplatz auf Ihrem Hostsystem.



Abbildung 4.24 Lage und Größe der virtuellen Festplatte festlegen

Standardmäßig speichert VirtualBox die Festplatten-Images im VDI-Ordner. Hier können Sie diesen Ort anpassen und den Namen des Images festlegen. Ich wähle als Namen immer den Namen der Maschine und durch einen Unterstrich getrennt den Typ. Als Größe können Sie bis zu 2 TByte angeben. Für Windows XP sind jedoch 10 GByte absolut ausreichend.

Bestätigen Sie nun die weiteren Punkte, und in der grafischen Oberfläche befindet sich eine neue Maschine mit dem Namen *WindowsXP*.

#### 4.2.2 Manager für virtuelle Medien

Mit der Einrichtung sind wir natürlich noch nicht fertig. Sie weisen nun noch die CD mit dem Betriebssystem zu. Dies kann entweder durch Einlegen und Zuweisen der Original-CD oder durch Einbinden eines ISO-Images (CD/DVD-Abbildes) geschehen. Ich verwende im Folgenden ein ISO-Image, das ich als Sicherheitskopie erstellt und abgelegt habe. Dieses machen Sie zunächst im Manager für virtuelle Medien mit VirtualBox bekannt.

Der Vorteil an ISO-Images ist, dass Sie diese, ohne CD-Boxen zu durchsuchen, zur Verfügung haben. Auch Lesefehler wegen Kratzern oder Ähnlichem kommen nicht vor. Daher ist es sinnvoll, von den Betriebssystemen, die man virtualisieren will, Abbilder zu erstellen. Die meisten Linux-Systeme können standardmäßig als ISO-Dateien heruntergeladen werden.

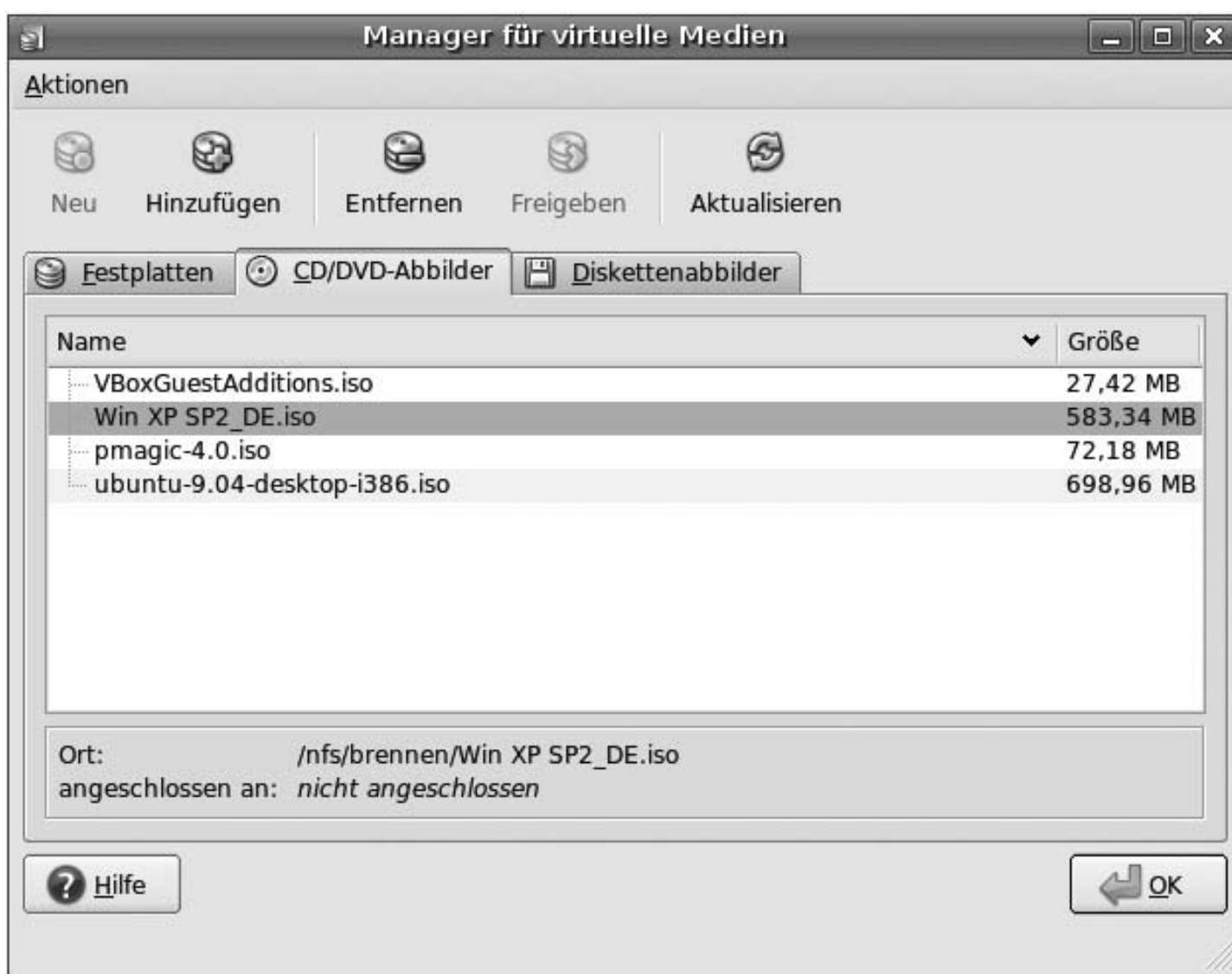
[+]

##### ISO-Images erstellen

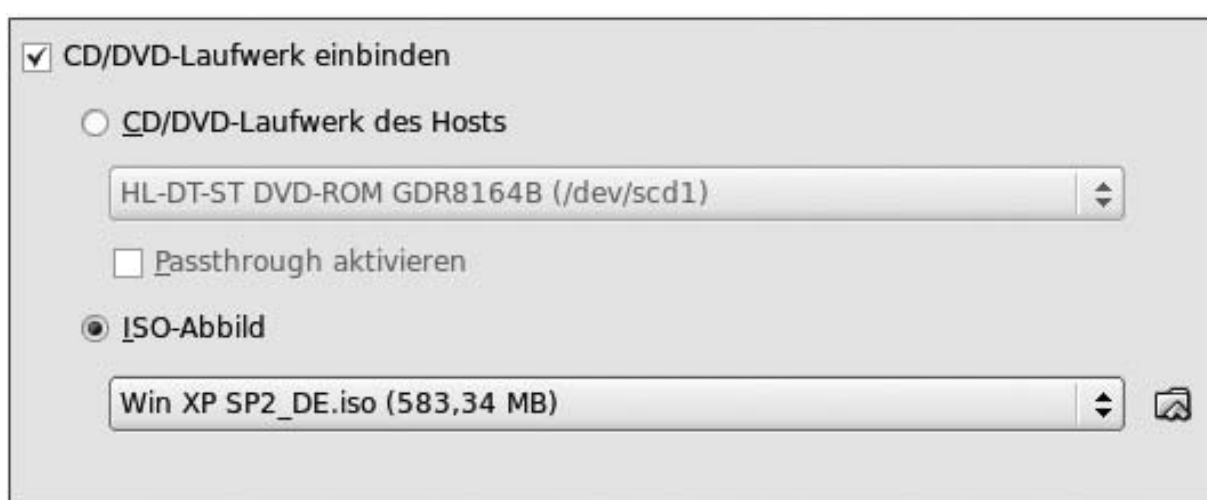
Ein ISO-Image einer CD/DVD können Sie mit den meisten Brennprogrammen erstellen. Wählen Sie dazu die Funktion zum Kopieren des Datenträgers, und suchen Sie als Ziel nicht den CD-Brenner, sondern eine ISO-Datei aus. Wenn Ihre Brennsoftware dies nicht unterstützt, können Sie auf die Freeware CDBurnerXP (<http://www.cdburnerxp.se/>) und unter Linux auf K3B zurückgreifen.

Öffnen Sie nun den Manager für virtuelle Medien über das Menü DATEI • MANAGER FÜR VIRTUELLE MEDIEN und wechseln in das Register CD/DVD-ABBILDER (Abbildung 4.25). Dort registrieren Sie über HINZUFÜGEN ein neues Image.

Wenn dies erledigt ist, können Sie das Image der virtuellen Maschine wie im vorherigen Beispiel zuweisen. Wählen Sie diesmal jedoch die Checkbox ISO-ABBILD aus und nicht das CD/DVD-LAUFWERK DES HOSTS.



**Abbildung 4.25** Der Manager für virtuelle Medien



**Abbildung 4.26** So binden Sie eine ISO-Datei ein.

### VBoxManage – CD/DVD-Abbild zuweisen

► Linux:

```
VBoxManage modifyvm WindowsXP\
    --dvd /media/daten/brennen/Win_XP_SP2.iso
```

► Windows:

```
VBoxManage modifyvm WindowsXP\
    --dvd D:\Images\Win_XP_SP2.iso
```

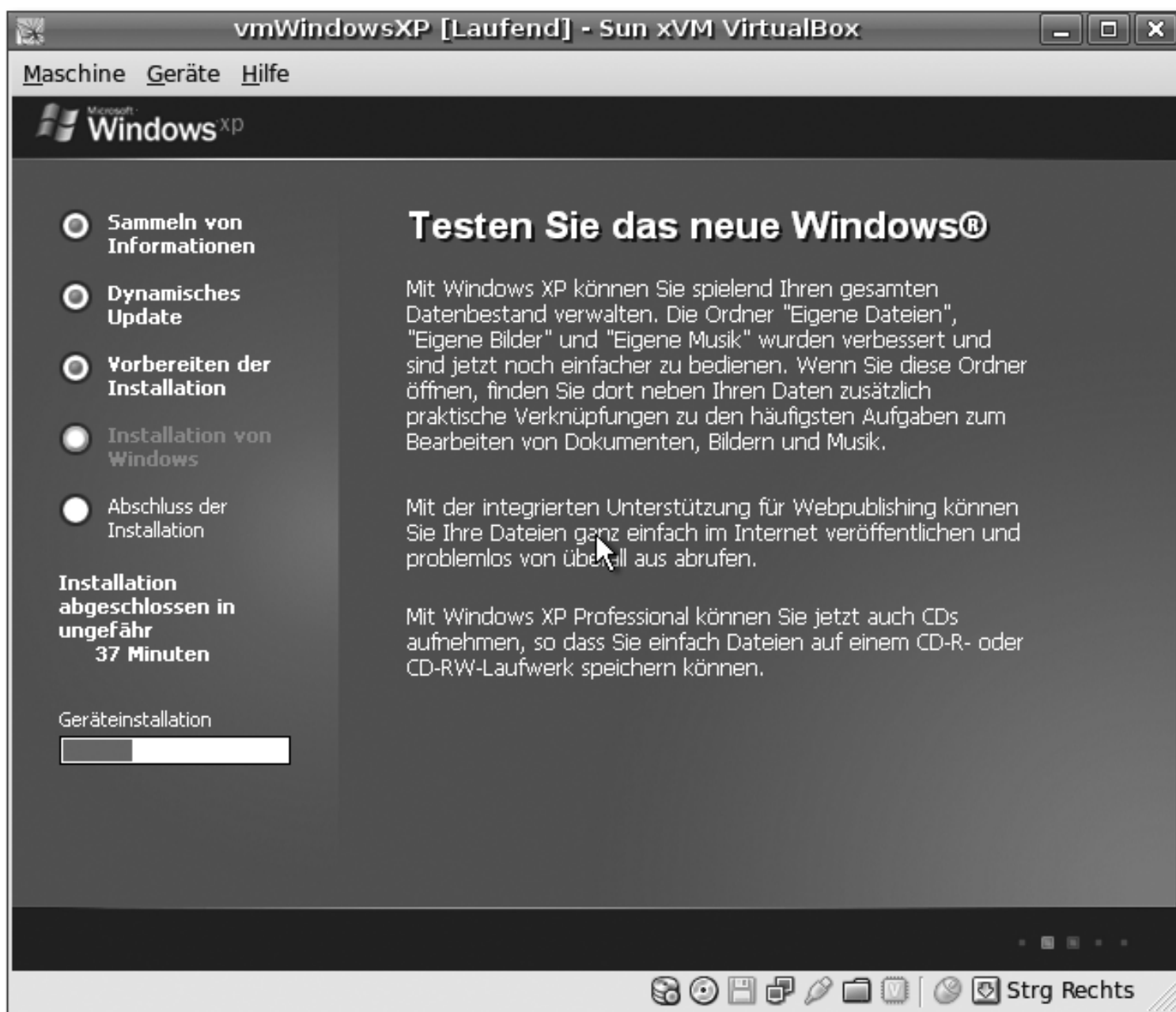
- Alle Systeme, über die UUID:

```
VBoxManage modifyvm WindowsXP\  
--dvd e1faf46b-1081-41bb-8c54-212a8fe9f002
```

Wenn Sie VBoxManage mit der UUID der DVD verwenden wollen, können Sie diese mit dem Kommando `VBoxManage list dvds` ermitteln.

### 4.2.3 Installation von Windows XP

Nach dem Start der virtuellen Maschine können Sie das Windows-System wie gewohnt installieren. Denken Sie hierbei auch an die Funktion der Host-Taste.



**Abbildung 4.27** Windows XP in VirtualBox installieren

Wenn die Installation abgeschlossen ist, können Sie Ihr Windows-System verwenden.



Abbildung 4.28 Windows XP in VirtualBox

#### 4.2.4 Gasterweiterungen

Wie die meisten Konkurrenzprodukte ermöglicht VirtualBox die Verbesserung des Gastsystems durch die sogenannten Gasterweiterungen. Mit diesen werden spezielle Treiber installiert und unter anderem das Problem mit der gefangenen Maus behoben.

Die Installation der Gasterweiterungen ist, zumindest unter Windows, recht einfach. Wählen Sie zunächst im Menü GERÄTE den Punkt GASTERWEITERUNGEN INSTALLIEREN. Dadurch wird ein ISO-Abbild eingebunden, auf dem sich die Erweiterungen befinden. Das neu installierte Windows-System unterstützt von Haus aus den Autostartmodus für CD/DVD-Laufwerke und öffnet daher automatisch die Setup-Routine. Falls die Routine dennoch nicht gestartet wird, rufen Sie sie über START • AUSFÜHREN auf. Starten Sie dann die Datei *VBoxWindowsAdditions.exe* auf dem CD-Abbild.

## VBoxManage – Gasterweiterungen einbinden

### ► Linux:

```
VBoxManage controlvm WindowsXP dvdattach\
    /usr/share/virtualbox/VBoxGuestAdditions.iso
```

### ► Windows:

```
VBoxManage controlvm WindowsXP dvdattach\
    "C:\Programme\Sun\xVM VirtualBox\VBoxGuestAdditions.iso"
```



### Pfad auf Windows-Systemen

Windows-Benutzer müssen darauf achten, dass der Pfad korrekt ist (je nach Installationsverzeichnis von VirtualBox) und diesen, wenn Leerzeichen vorkommen, in Anführungszeichen setzen.



**Abbildung 4.29** Assistent zur Installation der Gasterweiterungen

Über Klick auf WEITER bewegen Sie sich durch die einzelnen Dialoge, in denen Sie unter anderem die Lizenzbedingungen akzeptieren und das Zielverzeichnis angeben. Im Schritt KOMPONENTEN AUSWÄHLEN verzichten Sie für diese Maschine auf die Installation der Direct3D-Unterstützung, denn diese testen wir erst in einem der nächsten Praxisbeispiele.

Bei der Installation der Treiber bestätigen Sie, dass die Installation trotz des nicht bestandenen Windows-Logo-Tests fortgeführt werden soll. Diese Rückfrage erscheint unter anderem für den Grafikkarten- und die Netzwerktreiber (siehe Abbildung 4.31).



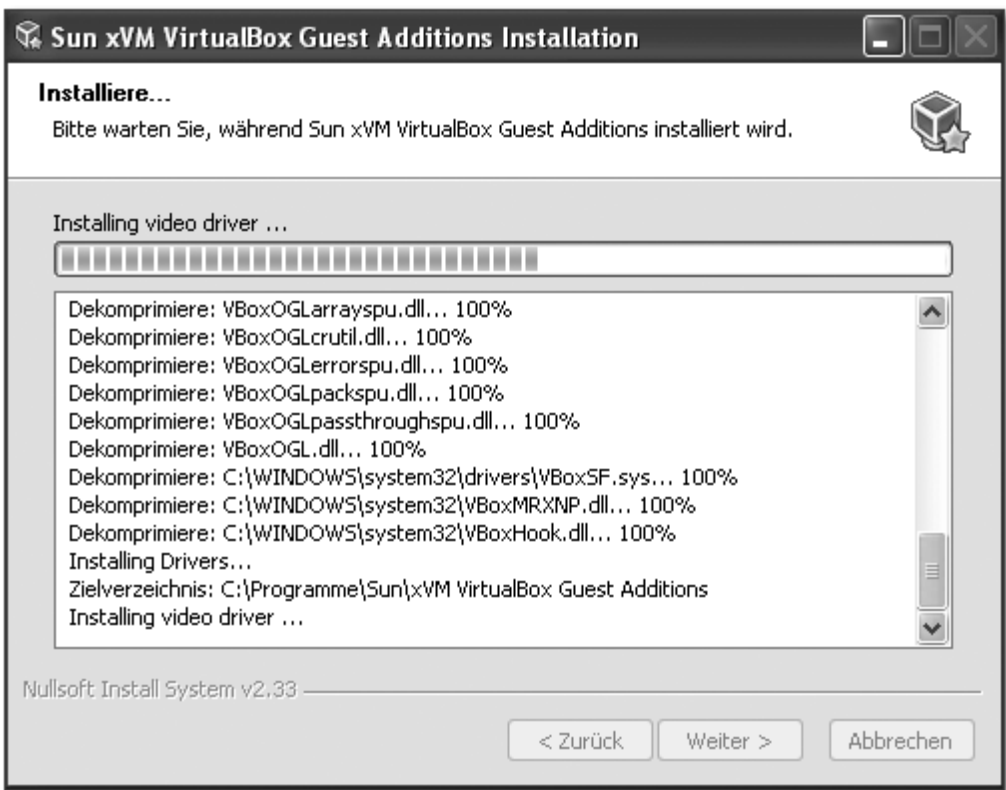


Abbildung 4.30 Treiberinstallation der Gasterweiterungen



Abbildung 4.31 Der Windows-Logo-Test für den Grafikkartentreiber schlägt fehl.

Nach erfolgreicher Installation und dem anschließenden Neustart des Gast-systems befindet sich in der Taskleiste das Symbol der Gasterweiterungen von VirtualBox.



Abbildung 4.32 Die Gasterweiterungen sind aktiviert (vgl. Symbol in der Windows-Startleiste rechts).

Sie können nun gleich einige der Vorzüge dieser Erweiterung und neuen Treiber testen. Als Erstes fällt auf, dass die Maus nicht mehr »gefangen« ist. Durch den neuen Grafikkartentreiber wird die Auflösung von Windows nun automatisch an die Größe des Fensters der virtuellen Maschine angepasst (Abbildung 4.33).



Abbildung 4.33 »Mini-Windows« dank Gasterweiterungen

Wer diese automatische Größenänderung nicht mag, kann sie im Menü MASCHINE • GRÖSSE DES GASTES AUTOMATISCH ANPASSEN deaktivieren.

4.2.5 Netzwerk mit NAT

Wie schon im Abschnitt 4.1, »Live-CDs testen«, erwähnt, hat VirtualBox automatisch eine Netzwerkkarte hinzugefügt. Windows XP hat diese bei der Installation erkannt und eingerichtet (Abbildung 4.34). Wenn Ihr Hostsystem einen Internetzugang besitzt, kann dieser nun auch vom Gastsystem verwendet werden.

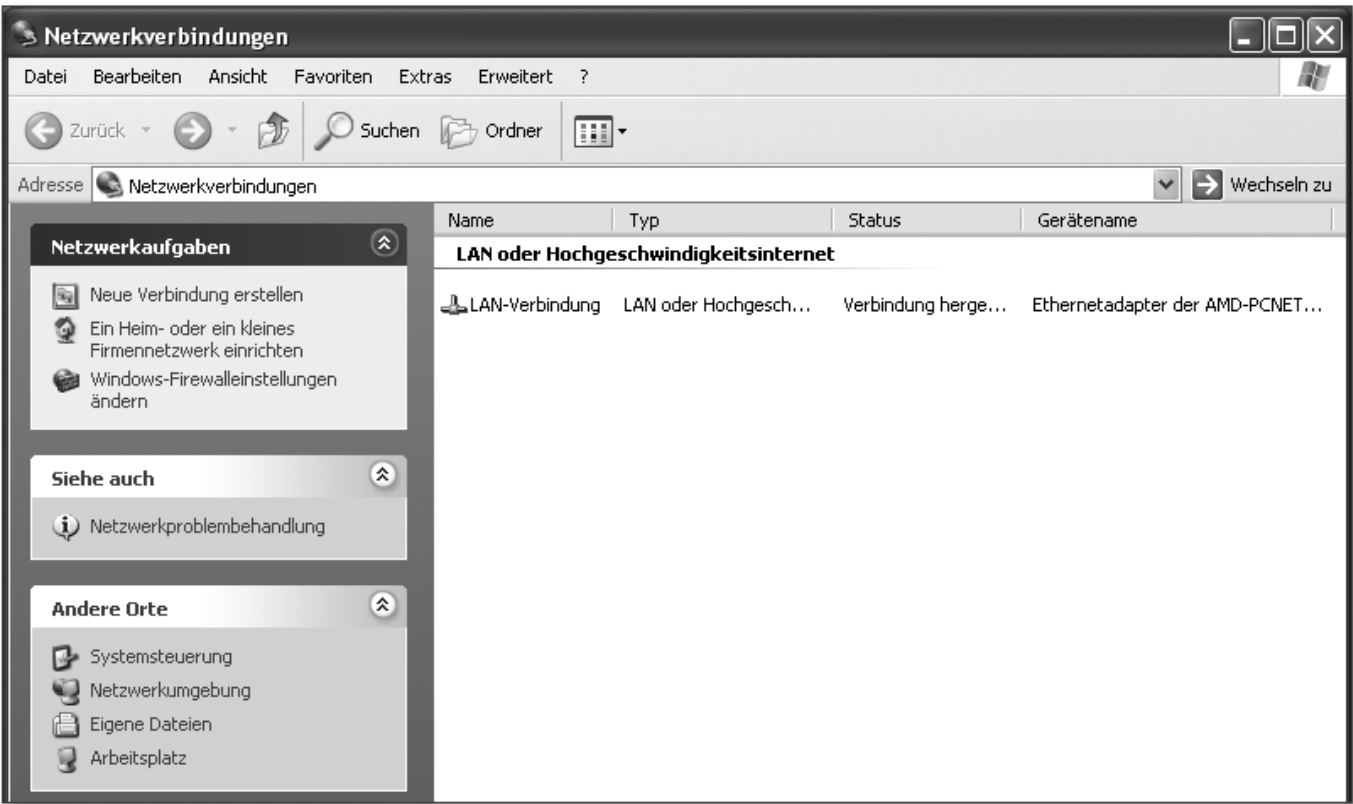
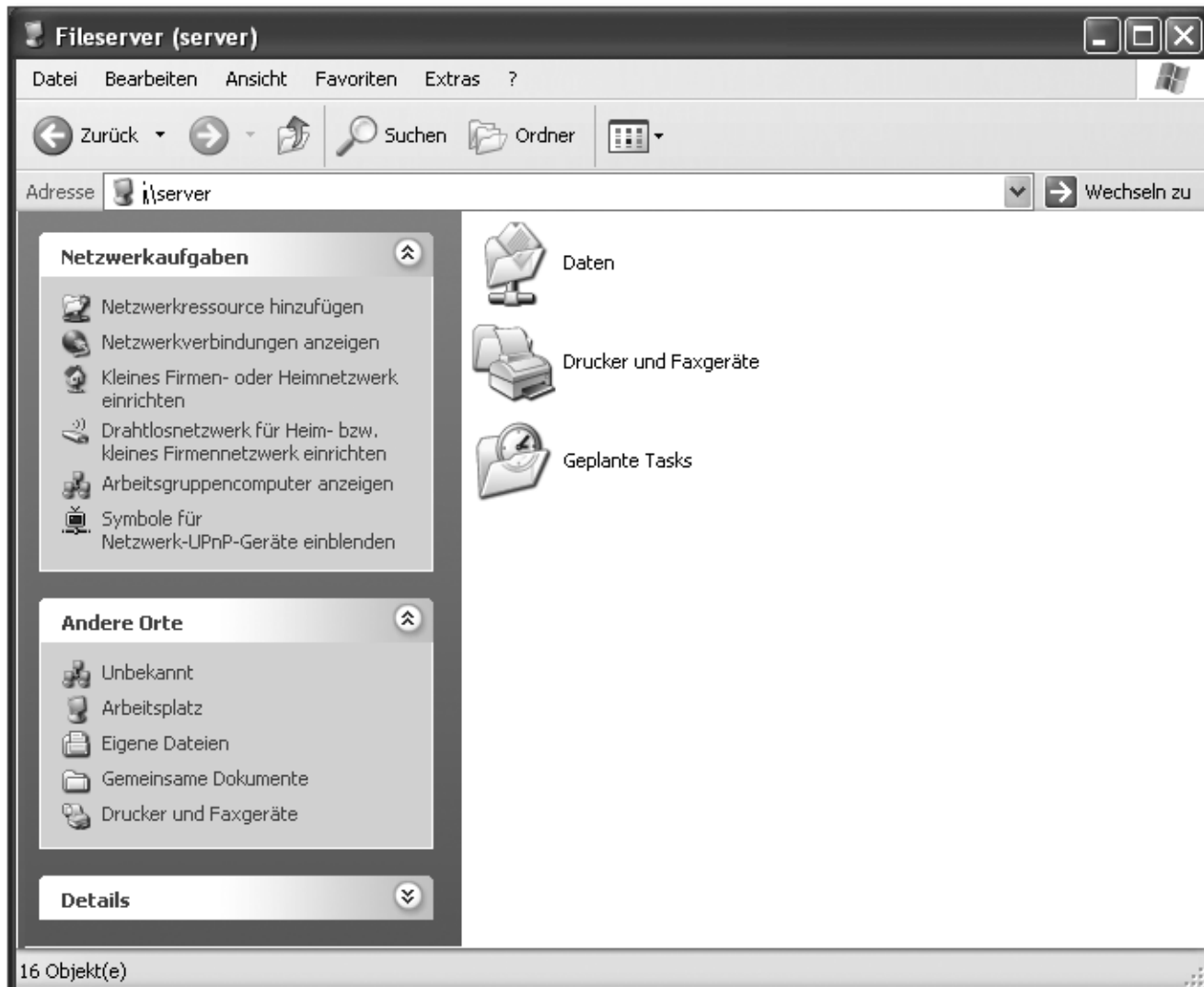


Abbildung 4.34 Netzwerkadapter unter Windows XP

Es ist darüber hinaus möglich, aus dem Gastsystem auf Freigaben anderer Netzwerkteilnehmer zuzugreifen. Geben Sie dazu einfach über **START • AUSFÜHREN** den Netzwerknamen des Zielsystems ein (zum Beispiel `\\server`).



**Abbildung 4.35** Zugriff auf Netzwerkfreigaben

Ein umgekehrter Zugriff aus dem Netzwerk auf Freigaben des Gastsystems gelingt hingegen nicht so einfach. Details hierzu lesen Sie in Abschnitt 3.9, »Netzwerk«.

## 4.3 Linux-System

Im ersten Praxisbeispiel in Abschnitt 4.1, »Live-CD testen«, haben wir eine virtuelle Maschine erstellt und dabei eine Linux-Live-CD getestet. In diesem Abschnitt installieren wir nun die Distribution Ubuntu auf einem Windows-Hostsystem. Mit dieser Maschine können Sie dann Linux testen, ohne dass die Änderungen wie bei einer Live-CD nach dem Beenden verworfen werden.

In diesem Abschnitt werden Sie Folgendes kennen lernen:

- ▶ Gasterweiterungen unter Linux installieren
- ▶ eine Maschine in den Ruhezustand versetzen

Laden Sie zunächst die ISO-Datei für das Linux-System herunter, und binden Sie diese im Manager für virtuelle Medien ein. Die aktuelle Ubuntu-Version finden Sie unter <http://www.ubuntu.com/getubuntu/download>.

[+]

Empfehlung für Linux-Einsteiger: Ubuntu

Wenn Sie Linux-Anfänger sind, sollten Sie auf jedem Fall Ubuntu verwenden, da ich in verschiedenen weiteren Abschnitten auf diese virtuelle Maschine zurückgreife. Dabei werden unter anderem Netzwerkeinstellungen vorgenommen, die ich nur für Ubuntu erkläre. Andere Distributionen mit einzubeziehen, würde den Umfang des Buchs sprengen. Weiterhin habe ich für Linux-Anfänger absichtlich die Installation von Ubuntu ausführlicher erläutert.

4.3.1 Virtuelle Maschine erstellen

Nachdem Sie die ISO-Datei heruntergeladen haben, können Sie die neue virtuelle Maschine erstellen. VirtualBox bietet, wie bereits erwähnt, Unterstützung für diese Distribution. Dazu starten Sie die grafische Oberfläche und wählen den Button NEU in der Symbolleiste. Nun vergeben Sie einen Namen für die virtuelle Maschine und setzen den korrekten Typ des Gastsystems.



Abbildung 4.36 Eine neue virtuelle Maschine mit Ubuntu als Gastsystem anlegen

Im nächsten Schritt erfolgt wieder die Einstellung des Hauptspeichers. Um die grafische Oberfläche von Ubuntu nutzen zu können, sollten Sie mindestens 384 MByte vergeben.

Ubuntu benötigt ein dynamisches Festplatten-Image mit einer Größe von circa 20 GByte (Abbildung 4.37).



**Abbildung 4.37** Festplatten-Image für Ubuntu erstellen

Nach dem Fertigstellen der virtuellen Maschine binden Sie nun die Ubuntu-CD oder das ISO-Abbild ein. Danach können Sie mit der Installation des Gastsystems beginnen.

### 4.3.2 Installation von Ubuntu

Beim Starten der virtuellen Maschine öffnet sich zunächst ein Menü, in dem Sie die gewünschte Sprache des Assistenten einstellen. Im folgenden Menü wählen Sie nun UBUNTU INSTALLIEREN statt UBUNTU AUSPROBIEREN (siehe Abbildung 4.38).

Ein Assistent begrüßt Sie und führt Sie in mehreren Schritten durch die Installation (Abbildung 4.39). Die einzelnen Schritte sind dabei ziemlich selbsterklärend und dürften Ihnen keine Schwierigkeiten bereiten.



Abbildung 4.38 Das Ubuntu-Bootmenü

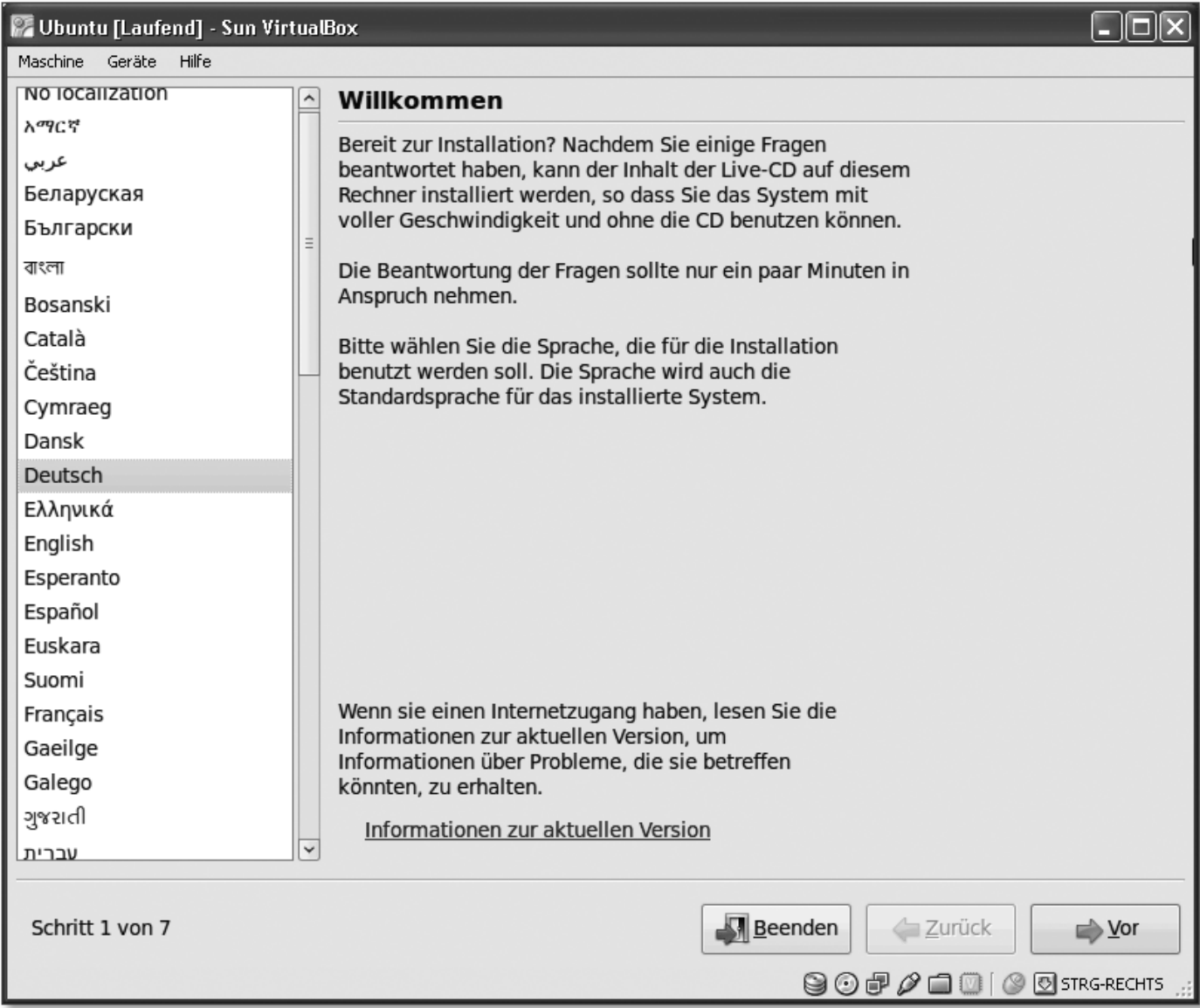
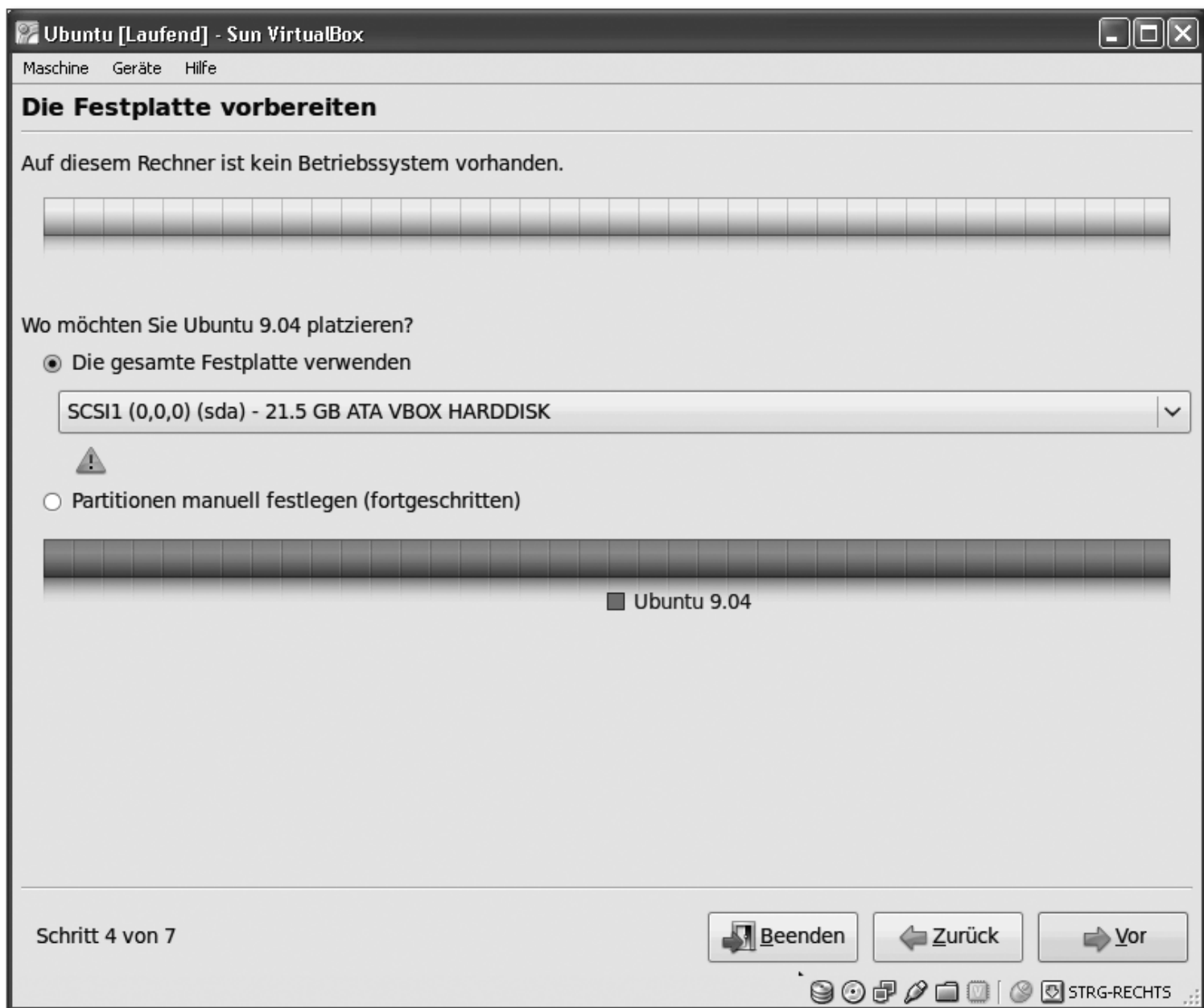


Abbildung 4.39 Der Ubuntu-Installer führt Sie in wenigen Schritten durch die Installation der Linux-Distribution.

Im vierten Schritt geht es um die Partitionierung der Festplatte. Hier können Kenner gegebenenfalls manuelle Einstellungen vornehmen. (Linux-)Anfänger sollten jedoch, wie in Abbildung 4.40 zu sehen, den Punkt DIE GESAMTE FESTPLATTE VERWENDEN wählen.



**Abbildung 4.40** Partitionierung der Festplatte

Als Nächstes richten Sie einen Benutzer ein und geben den Namen des Computers an. Ich verwende bei Netzwerknamen immer das Kürzel »vm« vor dem eigentlichen Namen. Dadurch erkennt man in einer Netzwerkumgebung sofort, ob es sich um einen physikalischen Rechner oder eine virtuelle Maschine handelt.

Nachdem Sie alle Eingaben getätigt haben, beginnt die eigentliche Installation. Den aktuellen Stand zeigt ein Statusfenster an.

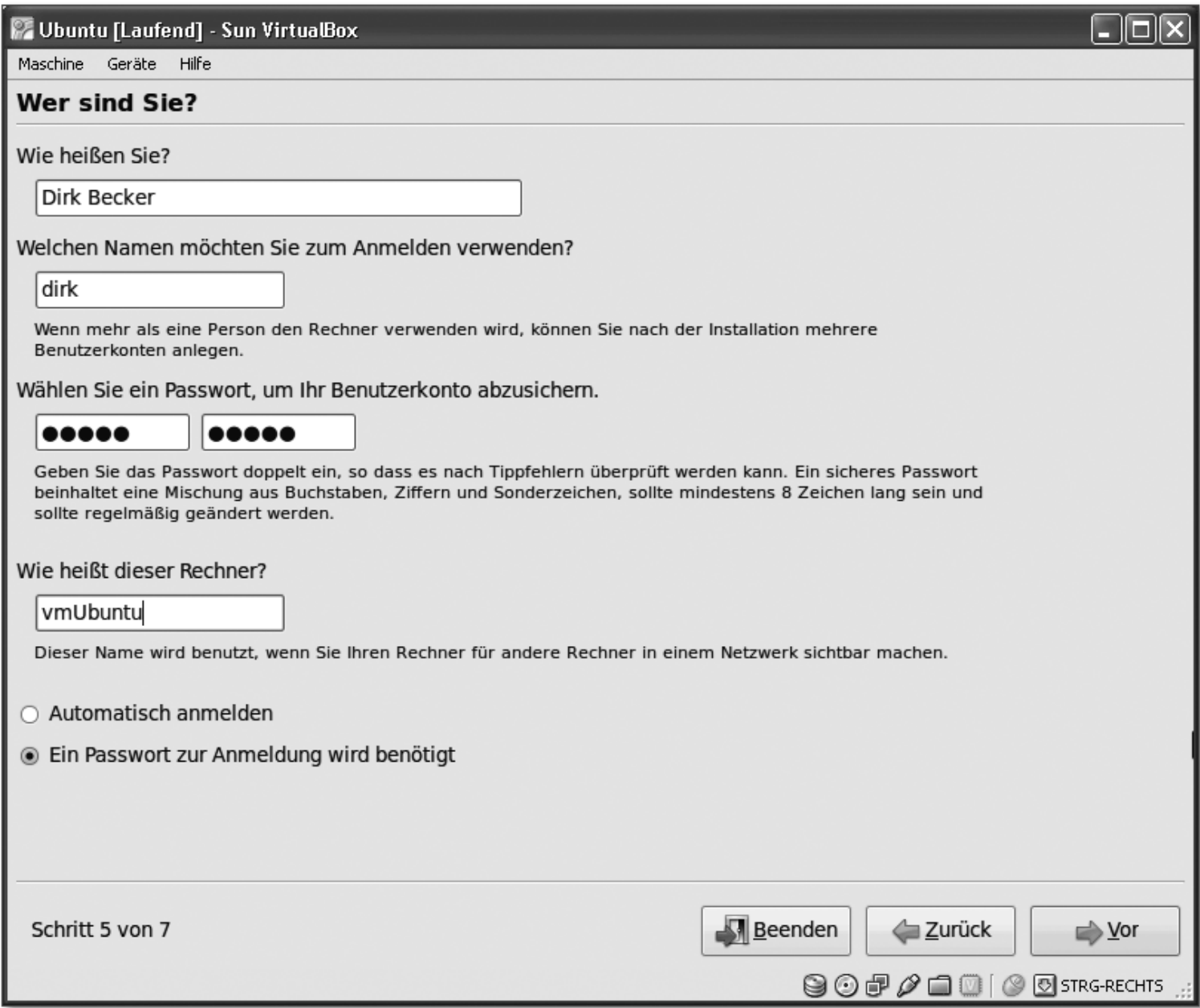


Abbildung 4.41 In diesem Dialog richten Sie einen Benutzeraccount und den Hostnamen ein.

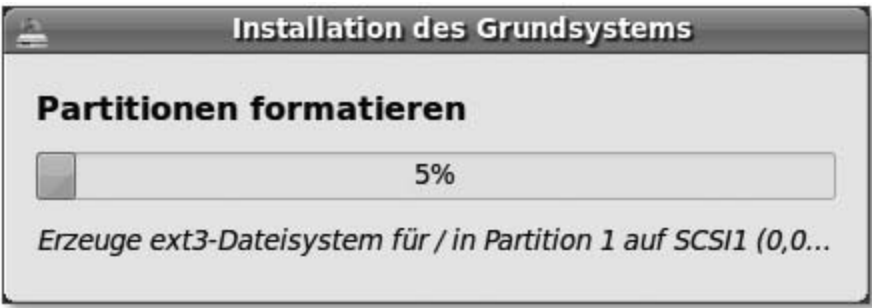


Abbildung 4.42 Während der Installation sehen Sie diese Fortschrittsanzeige.

Nachdem die Installation beendet ist, wird das Gastsystem neu gestartet. Bootet die virtuelle Maschine wieder von der CD, sollten Sie im Menü GERÄTE das Laufwerk trennen (Abbildung 4.43).

Nach dem Neustart erscheint zunächst der Anmeldebildschirm. Hier authentisieren Sie sich mit den eingerichteten Benutzernamen und dem dazugehörigen Passwort. Nach erfolgreicher Anmeldung startet der Ubuntu-Desktop (in der Voreinstellung die GNOME-Oberfläche).





Abbildung 4.43 CD/DVD-ROM trennen

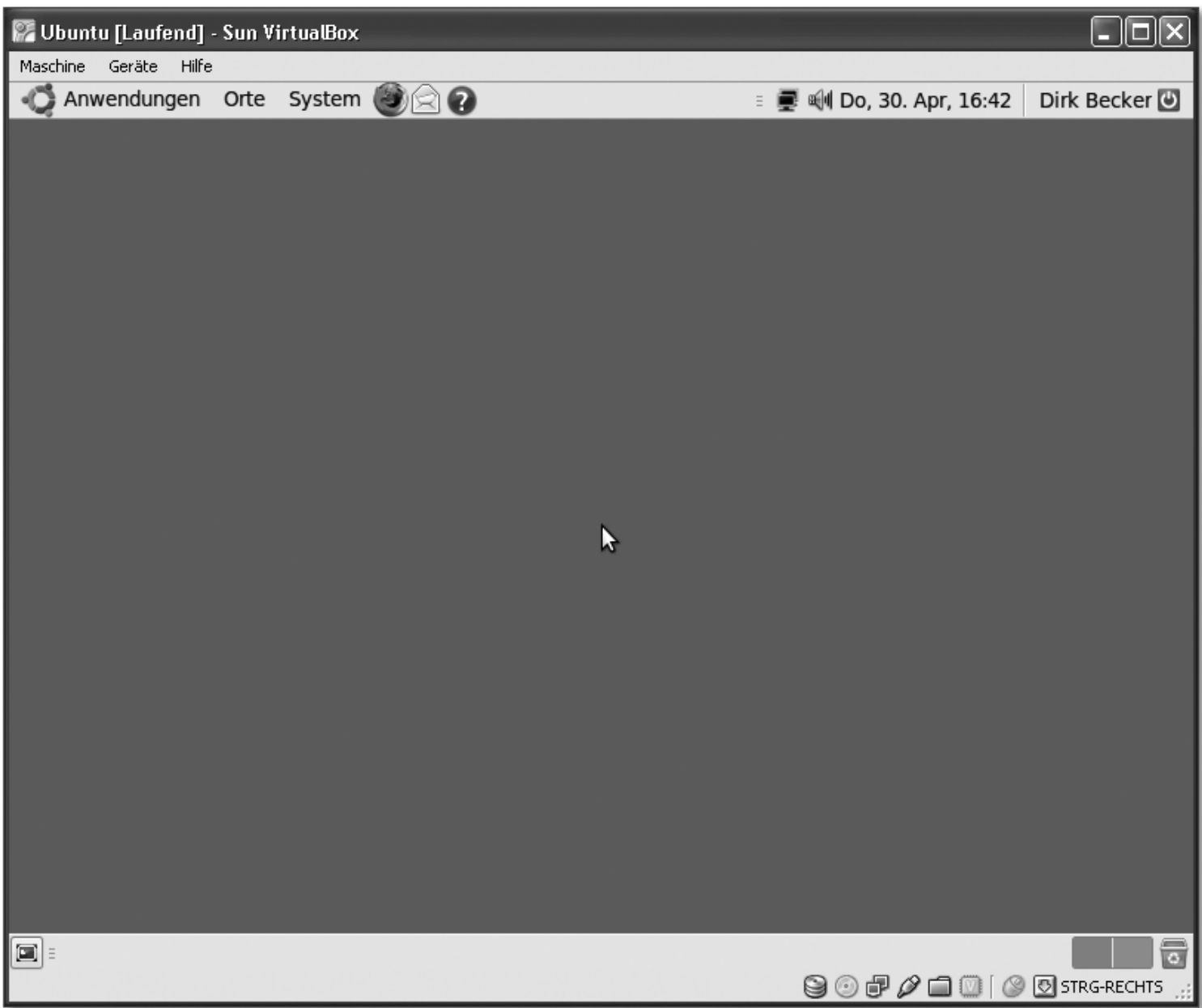


Abbildung 4.44 Der Ubuntu-Desktop nach der Anmeldung

### 4.3.3 Gasterweiterungen unter Linux

Wie in Abschnitt 4.2, »Windows XP«, erwähnt, können Sie unter Linux ebenfalls die Gasterweiterungen installieren. Wählen Sie dazu wieder im Menü GERÄTE den Punkt GASTERWEITERUNGEN INSTALLIEREN. Da es auch unter Linux eine Art

»Autostart« für CDs gibt, öffnet sich nun ein Fenster mit einem Hinweis auf die eingelegte CD. Dieses können Sie mit ABBRECHEN schließen – die Installation erfolgt in der Konsole. Die Konsole befindet sich im Startmenü unter ANWENDUNGEN • ZUBEHÖR • TERMINAL. In diesem Terminal geben Sie nun folgende Befehle ein:

Zunächst wechseln Sie in das Verzeichnis der CD:

```
cd /media/cdrom
```

Dann starten Sie die Installation:

```
sudo ./VBoxLinuxAdditions-x86.run
```

Wenn Sie ein Hostsystem mit einer AMD-CPU im 64-Bit-Modus verwenden, wählen Sie statt *VBoxLinuxAdditions-x86.run* die Routine *VBoxLinuxAdditions-amd64.run*.

## [+]

### Umgang mit Gasterweiterungen

Eventuell ändern sich die Dateinamen der Gasterweiterungen. Prüfen Sie daher bei Problemen mit dem Befehl `ls`, wie der korrekte Dateiname lautet. Je nachdem, welche Linux-Distribution Sie verwenden, müssen Sie eventuell weitere Komponenten installieren, damit die Gasterweiterungen funktionieren. Meistens sind das

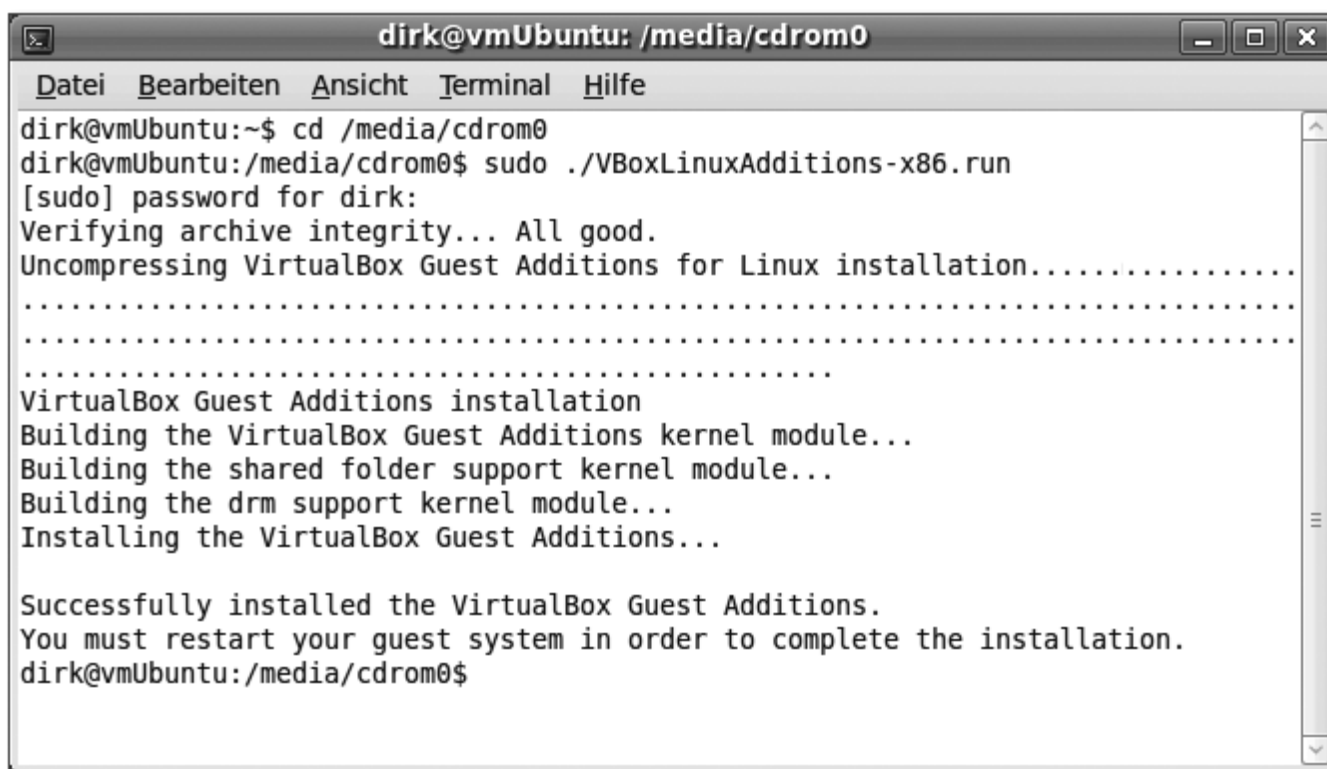
- ▶ die GNU Compiler Collection (GCC),
- ▶ die Kernel-Header oder die Kernel-Sourcen und
- ▶ das Programm *make*.

Die Installationsroutine zeigt fehlende Pakete an.

Zur Installation verwenden Sie unter Ubuntu das Programm *sudo*; dieses verlangt die Eingabe Ihres eigenen Passworts. Danach wird die Installation der Gasterweiterung mit Root-Rechten ausgeführt.

Wie auch unter Windows muss nach der Installation das Gastsystem neu gestartet werden. Nach dem Neustart können Sie unter MASCHINE • SESSION-INFORMATION prüfen, ob die Gasterweiterungen korrekt gestartet wurden.

In Abbildung 4.46 sehen Sie im Abschnitt LAUFZEIT-ATTRIBUTE, dass die Erweiterungen aktiviert wurden; hier steht VERSION 1.4 anstelle von NICHT ERKANNT.



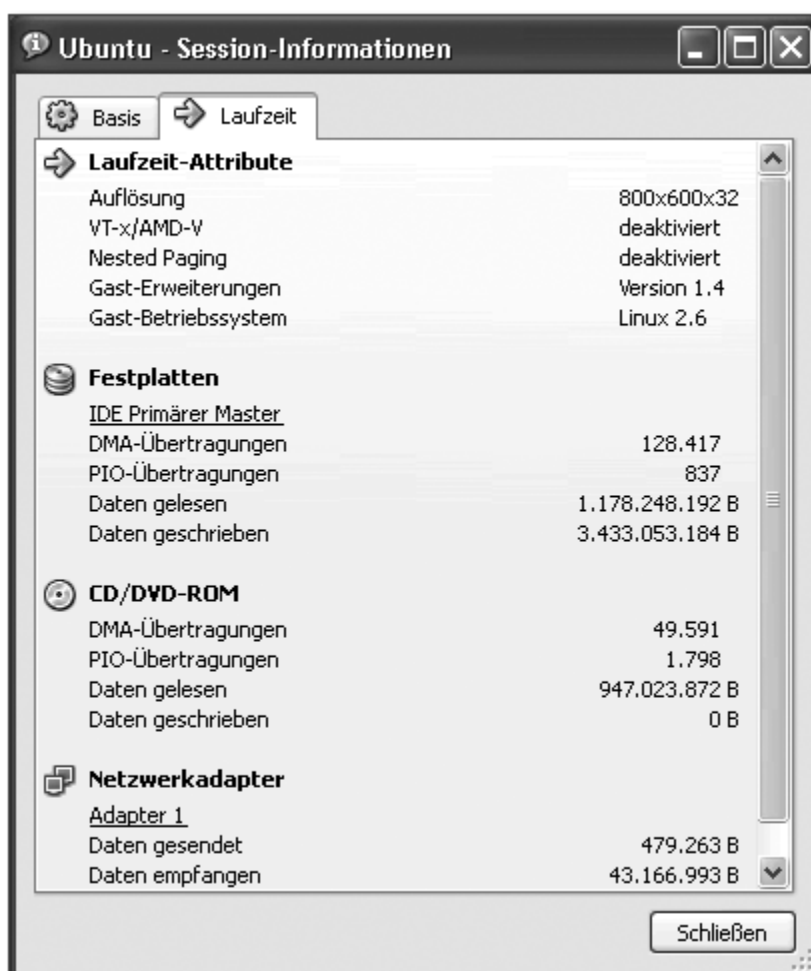
```

dirk@vmUbuntu: /media/cdrom0
Datei Bearbeiten Ansicht Terminal Hilfe
dirk@vmUbuntu:~$ cd /media/cdrom0
dirk@vmUbuntu:/media/cdrom0$ sudo ./VBoxLinuxAdditions-x86.run
[sudo] password for dirk:
Verifying archive integrity... All good.
Uncompressing VirtualBox Guest Additions for Linux installation.....
.....
VirtualBox Guest Additions installation
Building the VirtualBox Guest Additions kernel module...
Building the shared folder support kernel module...
Building the drm support kernel module...
Installing the VirtualBox Guest Additions...

Successfully installed the VirtualBox Guest Additions.
You must restart your guest system in order to complete the installation.
dirk@vmUbuntu:/media/cdrom0$

```

**Abbildung 4.45** Die Gasterweiterungen unter Linux installieren



**Abbildung 4.46** Die Gasterweiterungen in Version 1.4

Nun können Sie testen, ob der Mauszeiger das Fenster ohne Betätigen der Host-Taste verlassen kann und ob der Ubuntu-Desktop auf Größenänderungen des Fensters reagiert (Abbildung 4.47).

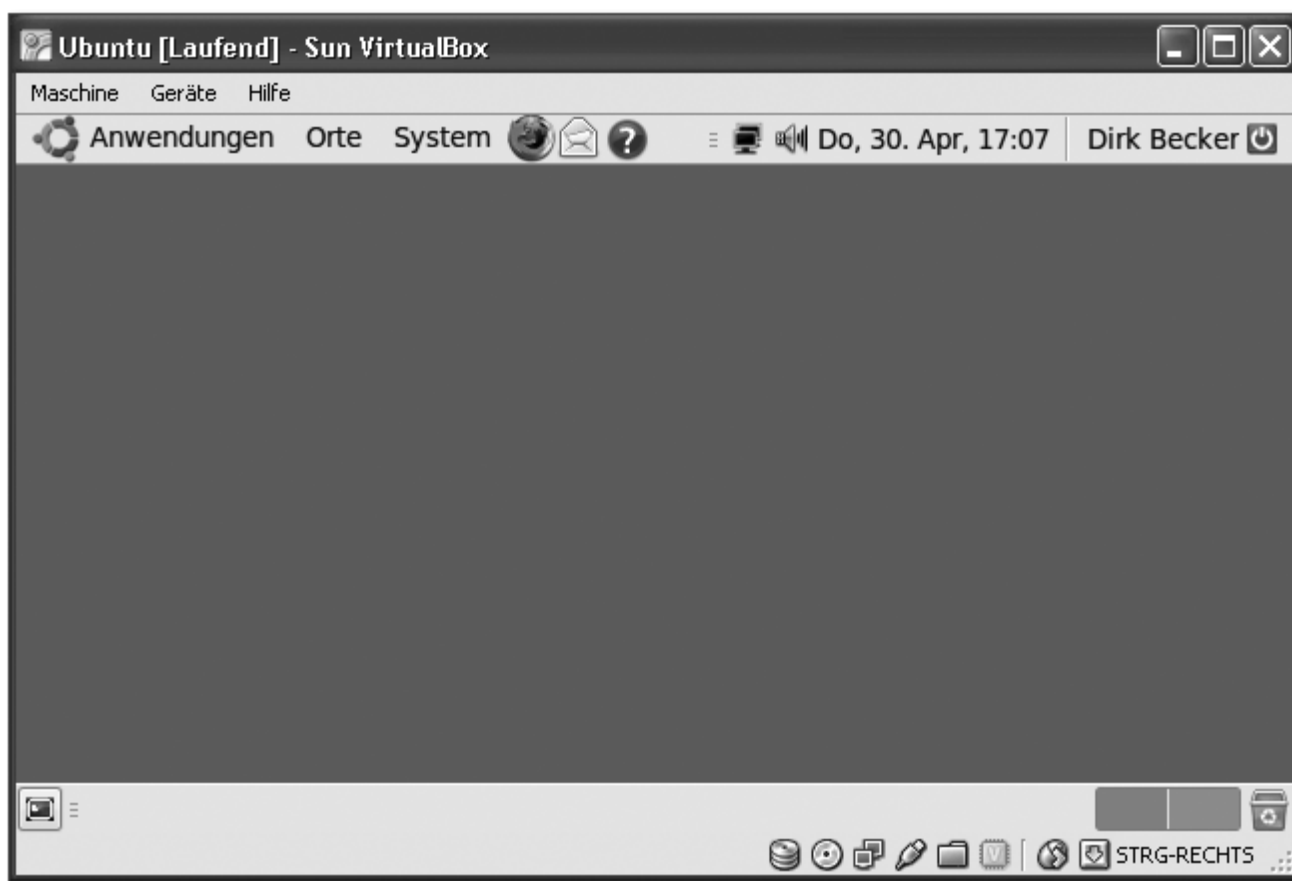


Abbildung 4.47 »Mini-Ubuntu«

## [+]

**Neuinstallation bei Änderung des Kernels**

Bei einer Änderung des Kernels – etwa durch ein Update – müssen Sie die Gasterweiterungen in einem Linux-Gastsystem zumeist neu installieren. Der Grund ist, dass bei der Installation spezielle Kernel-Module kompiliert werden und diese immer zur aktuellen Kernel-Version passen müssen.

**4.3.4 Ruhezustand**

VirtualBox bietet die praktische Möglichkeit, eine Maschine nicht immer beenden zu müssen. Stattdessen kann sie wie ein Notebook in einen Ruhezustand versetzt werden. Dadurch wird die aktuelle Arbeitsumgebung gespeichert und beim nächsten Start der virtuellen Maschine wiederhergestellt. Dieses Feature funktioniert erst nach der Installation der Gasterweiterungen korrekt, da diese die Synchronisation der Uhrzeit zwischen Host und Gast ermöglichen.

Um den Ruhezustand zu aktivieren, verwenden Sie die Funktion **MASCHINE • SCHLIESSEN** und wählen dann den Punkt **DEN ZUSTAND DER VIRTUELLEN MASCHINE SPEICHERN**.

Nach dem Klick auf OK speichert VirtualBox den Status der Maschine und beendet sie.

**VBoxManage – Ruhezustand**

Alle Systeme:

```
VBoxManage controlvm Ubuntu savestate
```



**Abbildung 4.48** Den Ruhezustand einer virtuellen Maschine aktivieren

**4.4 Windows 7**

Alternativen

Wenn Sie keine Windows-7-Version zur Verfügung haben, können Sie auch Windows XP oder Vista verwenden. Eine andere Windows-Version ist nur dann möglich, wenn Sie auf die 3-D-Beschleunigung verzichten, da unter Windows-Systemen nur XP, Vista und Windows 7 diese unterstützen.

[+]

Nachdem Sie nun die meisten grundlegenden Funktionen kennen gelernt haben, richten wir Windows 7 als Gastsystem mit erweiterten Funktionen ein. Da zur Zeit der Drucklegung noch keine Vollversion erhältlich war, verwende ich hier den aktuellen Release Candidate.

Vorab zu Ihrer leichteren Orientierung wieder ein kurzer Ausblick auf die Inhalte dieses Abschnitts. In diesem Abschnitt werden Sie Folgendes kennen lernen:

- ▶ 3-D-Beschleunigung konfigurieren
- ▶ Audio-Controller einrichten
- ▶ gemeinsame Ordner verwenden
- ▶ Netzwerkbrücke einrichten

Ob Sie Windows 7 später auch für den professionellen Einsatz verwenden, sollten Sie sich gut überlegen. Erstens benötigt Windows 7 im Gegensatz zu Windows XP bedeutend mehr Ressourcen, und zweitens handelt es sich hier noch um eine Testversion. Ob die Vorteile und Neuerungen von Windows 7 dies aufwiegen, müssen Sie selbst entscheiden.

4.4.1 Virtuelle Maschine einrichten

Wie bereits erwähnt, benötigt Windows 7 mehr Ressourcen als die Systeme aus den vorherigen Praxisbeispielen. Daher sollte Ihr Hostsystem mindestens 1 GByte, besser 2 GByte, Hauptspeicher besitzen, damit es flüssig läuft.

Ich werde die Installation unter einem Linux-Hostsystem vornehmen. Auf die Unterschiede zu einem Windows-Host werde ich Sie natürlich hinweisen. Da Sie nun bereits wissen, wie man neue Maschinen erstellt, brauchen wir auf diese Schritte nicht mehr einzugehen. In Abbildung 4.49 sehen Sie die Zusammenfassung meines Praxisbeispiels. Anhand dieser kann die Konfiguration in der grafischen Oberfläche vorgenommen werden. Wichtige Punkte hierbei sind:

- ▶ *System*: Windows 7
- ▶ *Speicher*: mindestens 512 MByte, besser 1024 MByte
- ▶ *3-D-Beschleunigung*: aktivieren
- ▶ *CD/DVD-ROM*: das physikalische Laufwerk oder ein ISO-Abbild
- ▶ *Soundkarte*: Intel AC97
- ▶ *Netzwerk*: Intel Pro/1000 MT Desktop mit NAT

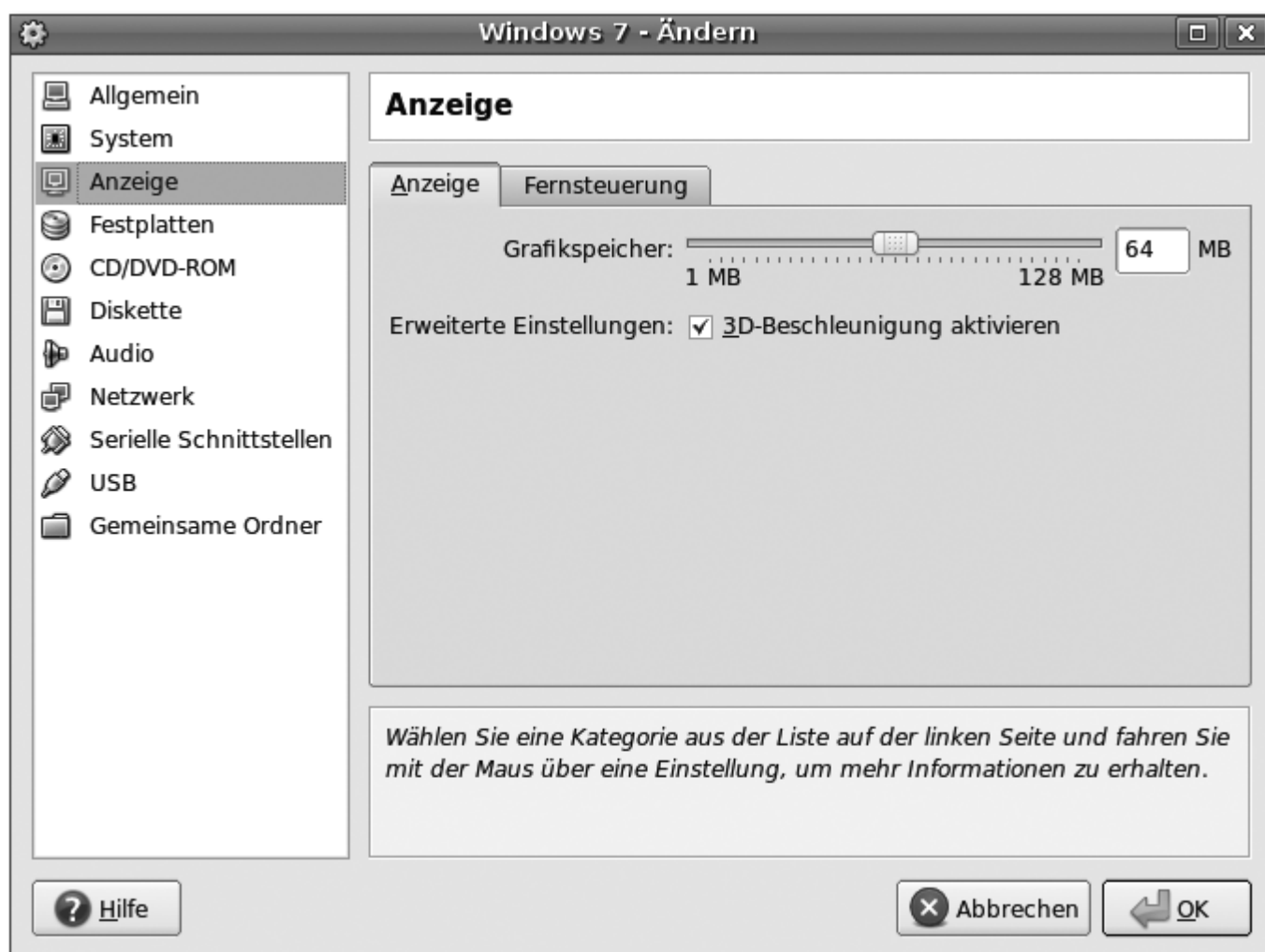


Abbildung 4.49 Die Konfiguration der virtuellen Maschine

Im Test hatte ich Probleme mit der Installation auf einer SATA-Festplatte in der virtuellen Maschine. Die Installation dauerte hierbei mehrere Stunden. Nach einer Umstellung auf IDE lief die Einrichtung jedoch in gewohnter Geschwindigkeit ab. Einen Grund für dieses Verhalten habe ich nicht gefunden. Auch ein späterer Einsatz von SATA-Platten verursachte keine Probleme mehr.

#### 4.4.2 3-D-Beschleunigung

In Abbildung 4.50 sehen Sie, dass die 3-D-Beschleunigung aktiviert wurde. Die Aktivierung erfolgt in der grafischen Oberfläche im Punkt ANZEIGE. Voraussetzung für die Aktivierung ist, dass später die Gasterweiterungen installiert werden.



**Abbildung 4.50** Die 3-D-Beschleunigung aktivieren

Leider können Sie trotz 3-D-Beschleunigung nicht die Aero-Oberfläche von Windows 7 oder Vista unter VirtualBox verwenden. Hintergrund ist, dass die 3-D-Beschleunigung OpenGL und DirectX-Unterstützung zwar anbietet, die virtuelle Maschine jedoch den Windows-Leistungstest nicht besteht. Einer der Gründe ist die fehlende DirectX-10-Unterstützung – VirtualBox bietet zurzeit nur Unterstützung für die Versionen 8 und 9. Ein weiterer Grund ist die meist schlechte Geschwindigkeit einer virtuellen Maschine bei diesem Test. Das Hostsystem muss extrem leistungsfähig sein, um ihn zu bestehen.

Dennoch werden wir später ein OpenGL-Programm installieren, mit dem Sie testen können, ob die 3-D-Beschleunigung funktioniert oder nicht.

### VBoxManage – 3-D-Beschleunigung aktivieren

Alle Systeme:

```
VBoxManage modifyvm "Windows 7" --accelerate3d on
```

#### 4.4.3 Audio-Controller

Die virtuelle Soundkarte bietet verschiedene Konfigurationsmöglichkeiten. Einige der Einstellungen sind dabei vom Hostsystem abhängig. Zum Verwenden der Soundkarte müssen drei Parameter eingestellt werden:

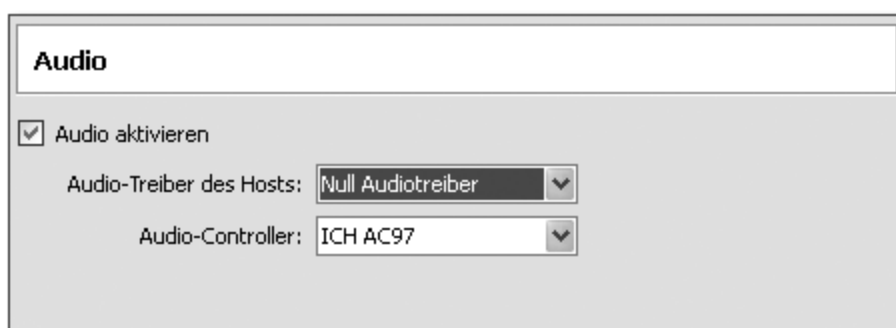
1. die Funktion AUDIO AKTIVIEREN
2. der AUDIO-TREIBER DES HOSTS
3. der AUDIO-CONTROLLER, der emuliert werden soll

#### Zu Punkt 1: Audio aktivieren

Zuerst aktivieren Sie die Soundkarte über einen Klick in die Checkbox AUDIO AKTIVIEREN.

#### Zu Punkt 2: Audio-Treiber des Hosts

Danach legen Sie fest, über welchen Treiber des Hostsystems die Ausgabe erfolgen soll. Diese Auswahl ist je nach Hostsystem bis auf eine Einstellung unterschiedlich. Auf allen Systemen können Sie die Option NULL AUDIOTREIBER aktivieren (Abbildung 4.51). Diese ermöglicht es, dass Sie der virtuellen Maschine eine Soundkarte zuweisen, ohne dass eine Ein- oder Ausgabe erfolgt. Verschiedene Programme, die einen Audio-Adapter erfordern, können dadurch ebenfalls verwendet werden.

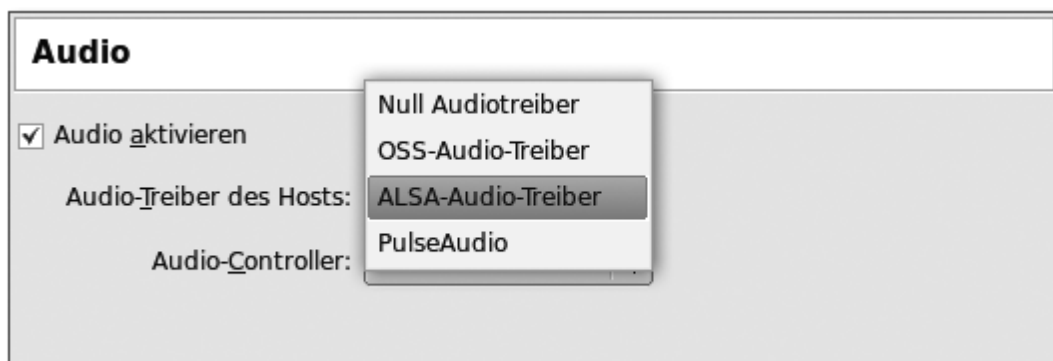


**Abbildung 4.51** Die Option Null Audiotreiber auswählen

Wenn jedoch eine Soundausgabe möglich sein soll, muss der Treiber ausgewählt werden, über den diese erfolgt. Manche Hostsysteme bieten mehr als einen Au-

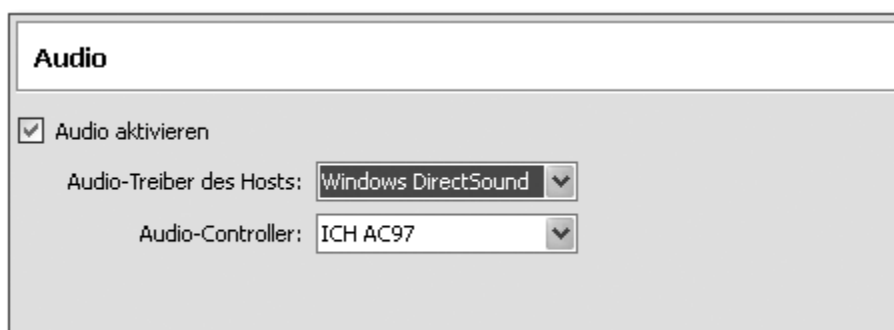


diotreiber an, zum Beispiel ein USB-5.1-System und parallel eine Onboard-Karte. Auf meinem Linux-Hostsystem verwende ich den ALSA-Treiber (Abbildung 4.52).



**Abbildung 4.52** Den Treiber des Linux-Hostsystems auswählen

Windows-Anwender können in den meisten Fällen den DirectSound-Treiber verwenden.



**Abbildung 4.53** DirectSound-Treiber für Windows einstellen

### VBoxManage – Audio aktivieren

- ▶ Alle Systeme:  
VBoxManage modifyvm "Windows 7" --audio **null**
- ▶ Windows:  
VBoxManage modifyvm "Windows 7" --audio **dsound**
- ▶ Linux:  
VBoxManage modifyvm "Windows 7" --audio **alsa**
- ▶ oder  
VBoxManage modifyvm "Windows 7" --audio **oss**

### Zu Punkt 3: Audio-Controller

Unter Punkt 3 legen Sie fest, welche Soundkarte (AUDIO-CONTROLLER) der virtuellen Maschine zur Verfügung gestellt werden soll. VirtualBox bietet den Intel AC97 (ICH AC97) oder SOUNDBLASTER 16 an. Für Windows 7 verwende ich den AC97-Controller. Für ältere Gastsysteme, wie zum Beispiel MS-DOS oder Windows 3.11, sollte der Soundblaster 16 zum Einsatz kommen, da dieser der »Vorreiter« aller Soundkarten war und daher von fast allen Systemen unterstützt wurde.

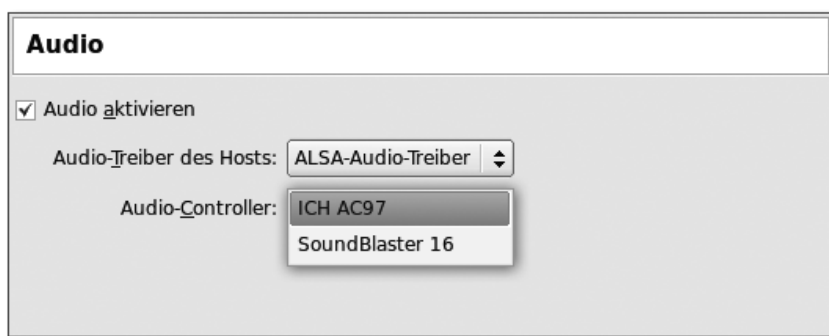


Abbildung 4.54 Den Audio-Controller auswählen

### VBoxManage – Audio-Controller auswählen

- ▶ Alle Systeme:  
VBoxManage modifyvm "Windows 7" --audiocontroller **ac97**
- ▶ oder:  
VBoxManage modifyvm "Windows 7" --audiocontroller **sb16**

#### 4.4.4 Installation von Windows 7

Legen Sie nun Ihre Windows-7-CD ein, oder weisen Sie der virtuellen Maschine das entsprechende CD/DVD-Abbild zu. Nach dem Start der Maschine erscheint zunächst der Startbildschirm der Installation, in dem Sie die Sprache festlegen können (Abbildung 4.55).

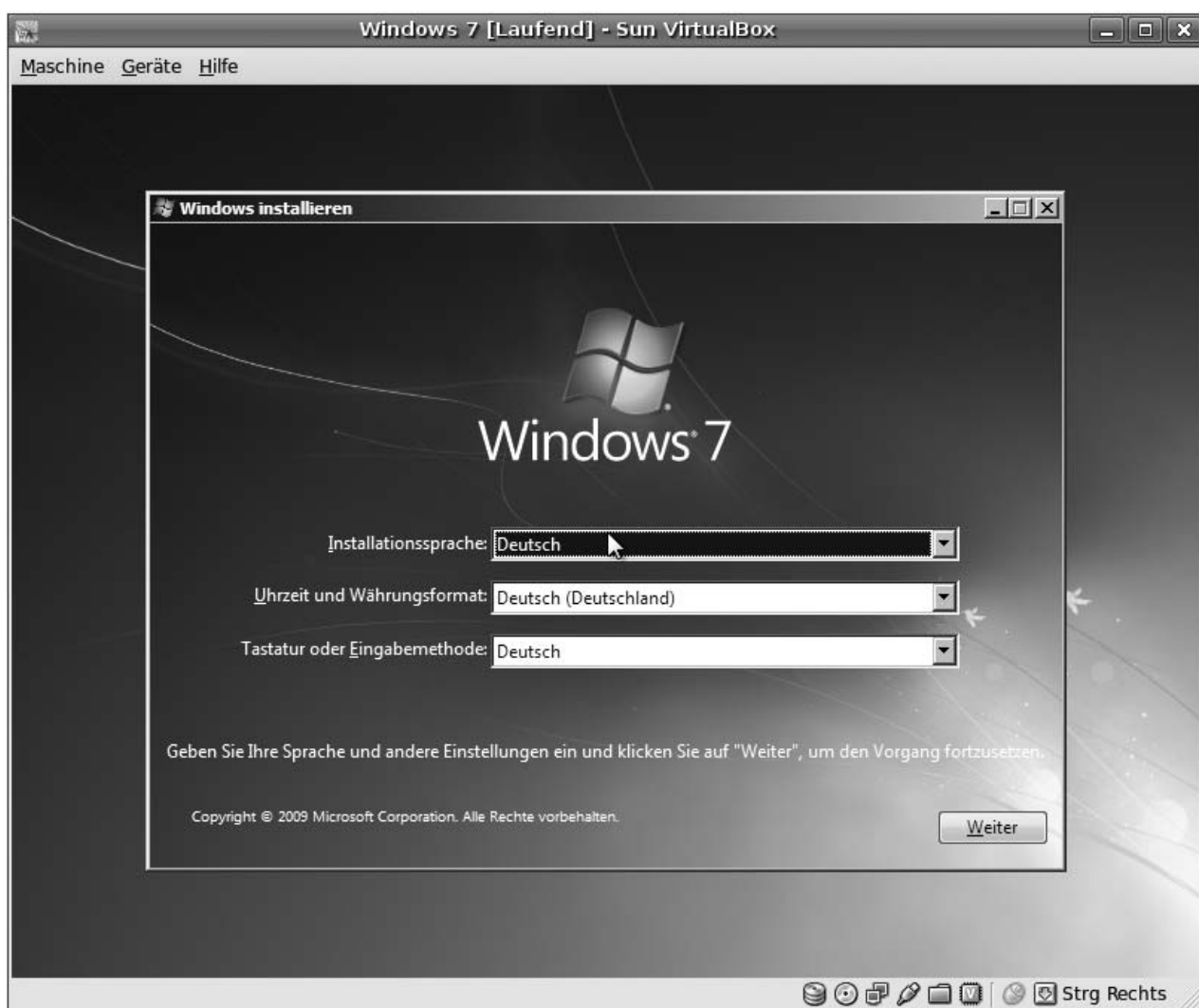


Abbildung 4.55 Installation von Windows 7

Geben Sie in den folgenden Schritten alle notwendigen Informationen ein (wählen Sie am besten BENUTZERDEFINIERTE INSTALLATION und danach die komplette Festplatte). Der Fortschritt der Installation kann dann in einem Infofenster verfolgt werden.



**Abbildung 4.56** Die Installation läuft.

Am Ende der Installation wird noch nach den Netzwerkeinstellungen gefragt. Verwenden Sie dort die Option HEIMNETZWERK (Abbildung 4.57).

Nach der Installation und dem Neustart des Gastsystems können Sie sich unter Windows 7 anmelden.

#### Probleme mit dem Audio-Treiber

[+]

Nach der Installation funktionierte zunächst der Audio-Adapter nicht. Dies konnte ich durch ein Windows-Update beheben. Bei diesem Update wurde der AC97-Treiber angeboten. Dieser musste jedoch explizit aktiviert werden.



Abbildung 4.57 Das Heimnetzwerk einrichten



Abbildung 4.58 Windows 7 nach dem Start

4.4.5 Gasterweiterungen unter Windows 7

Um nun die 3-D-Beschleunigung und die gemeinsamen Ordner testen zu können, installieren Sie die Gasterweiterungen. Nachdem Sie über MASCHINE • GASTERWEITERUNGEN INSTALLIEREN der Maschine das CD-Abbild zugewiesen haben, erscheint die Autostartmeldung. Wählen Sie dort den Punkt VBoxWindowsADDITIONS.EXE AUSFÜHREN, um die Installation zu starten.



Abbildung 4.59 Der Dialog Automatische Wiedergabe

Genau wie bei Windows Vista geben Sie auch bei Windows 7 Ihre Zustimmung, dass das Programm mit administrativen Rechten laufen und Änderungen am System vornehmen darf.

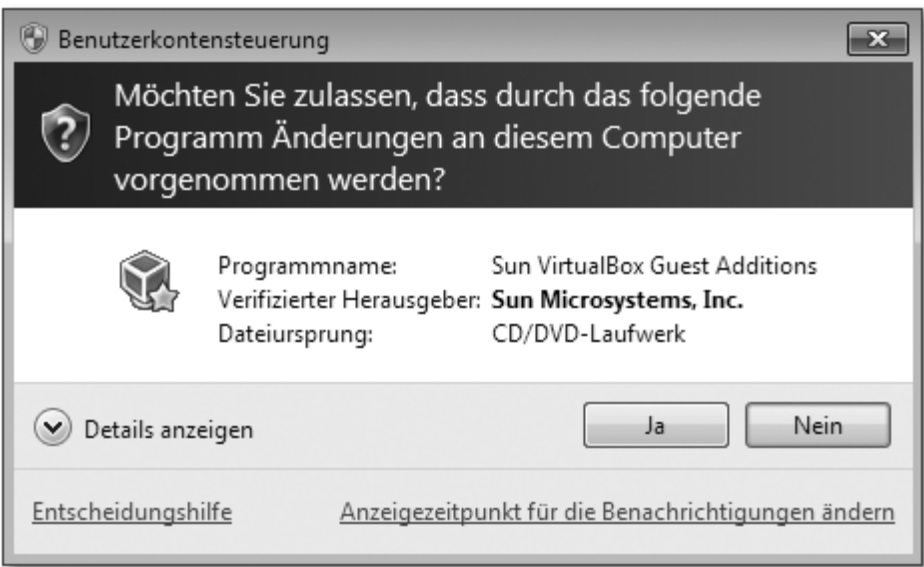


Abbildung 4.60 Der Dialog Benutzerkontensteuerung

Danach erfolgt die bereits in Abschnitt 4.2.4, »Gasterweiterungen«, durchgeführte Installation der Gasterweiterungen für ein Windows-System. In diesem Beispiel aktivieren Sie jedoch die DIRECT3D-UNTERSTÜTZUNG (siehe Abbildung 4.61).



**Abbildung 4.61** Installation der Gasterweiterungen mit Direct3D-Unterstützung

#### 4.4.6 Gemeinsame Ordner

Nach der Installation der Gasterweiterung richten Sie nun einen gemeinsamen Ordner ein, um mit der virtuellen Maschine Daten auch ohne Netzwerk auszutauschen. Dazu öffnen Sie die Konfiguration der Maschine und wechseln zum Punkt GEMEINSAME ORDNER. Dort gibt es mehrere Möglichkeiten, ein neues Verzeichnis hinzuzufügen:

- ▶ durch Betätigen der Taste Einfg
- ▶ über das Kontextmenü (Rechtsklick)
- ▶ durch Klick auf das Ordnersymbol mit dem Pluszeichen

Danach öffnet sich ein Fenster, in dem Sie die notwendigen Eingaben vornehmen (Abbildung 4.62).

Ins Feld ORDNER-PFAD tragen Sie den Ordner oder das Laufwerk ein, auf die das Gastsystem Zugriff erhalten soll. Bei ORDNER-NAME geben Sie den Namen ein, über den die Maschine die Verbindung aufbauen soll (ähnlich wie ein Freigabe-name im Netzwerk). Falls das Gastsystem nur Lesezugriff erhalten soll, setzen Sie ein Häkchen bei NUR LESBAR.

Nun können Sie nach dem erneuten Start des Gastsystems auf diesen Ordner zugreifen. Dazu kann der Befehl `net use` verwendet werden. Wechseln Sie dazu in das Fenster AUSFÜHREN (`[Windows]-Taste+[R]`), und geben Sie dort ein:

```
net use z: \\vboxsvr\daten
```



Abbildung 4.62    Einen neuen gemeinsamen Ordner hinzufügen

Im Beispiel ist `z:` der zu verwendende Laufwerksbuchstabe und `daten` der Name des Ordners.

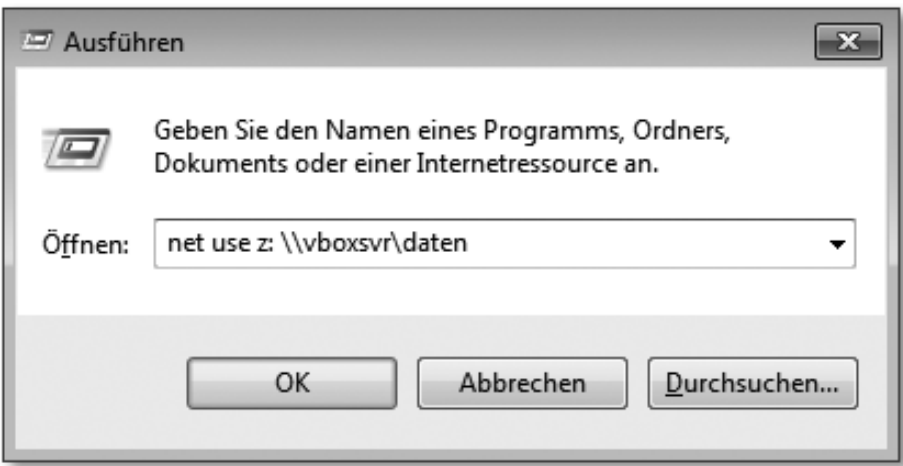
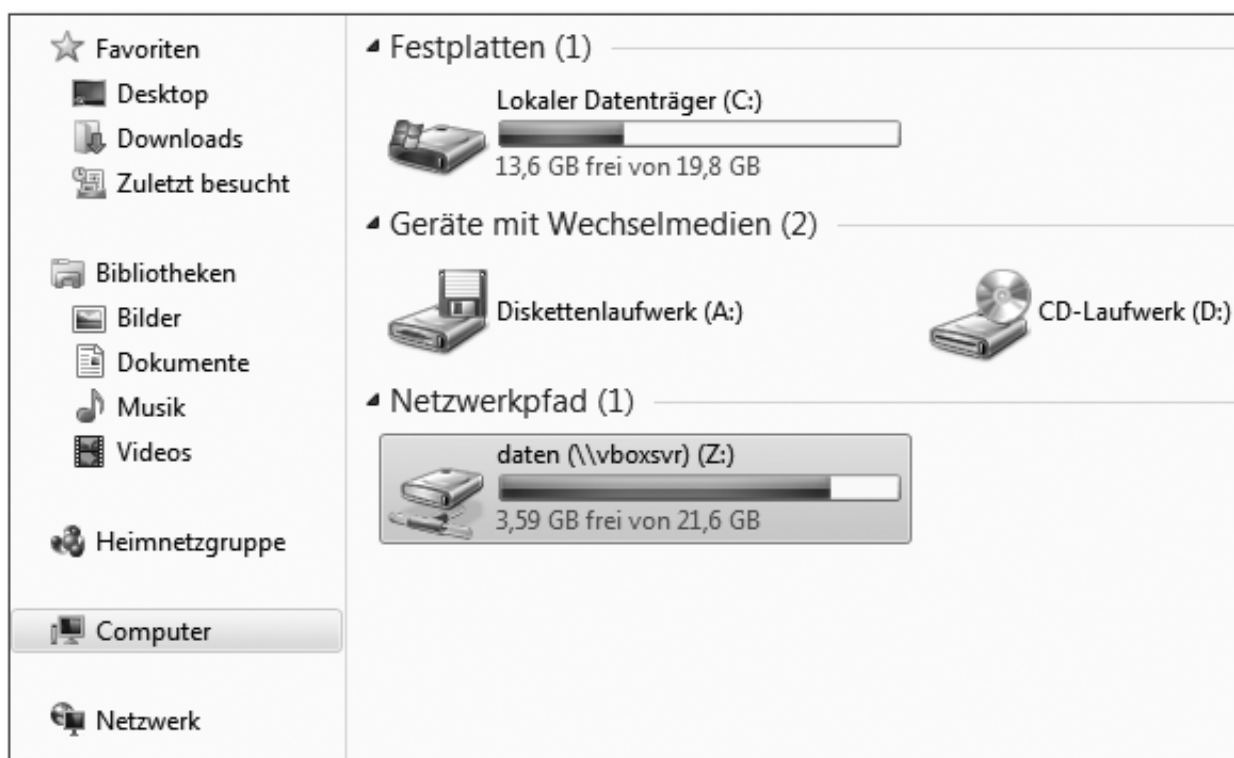


Abbildung 4.63    Das Kommando »net use«

Danach sehen Sie im Explorer das neue Laufwerk und können auf dessen Daten zugreifen. Sollte es zu Problemen kommen, können Sie den Befehl auch in der Eingabeaufforderung (unter `START • ALLE PROGRAMME • ZUBEHÖR`) eingeben. Das hat den Vorteil, dass Sie dann eventuelle Fehlermeldungen in der Konsole sehen.



**Abbildung 4.64** Laufwerk Z: ist der neue gemeinsame Ordner.

### VBoxManage – gemeinsamen Ordner erstellen

► Linux:

```
VBoxManage sharedfolder add "Windows 7" --name daten\  
--hostpath /media/daten
```

► oder schreibgeschützt:

```
VBoxManage sharedfolder add "Windows 7" --name daten\  
--hostpath /media/daten --readonly
```

► Windows:

```
VBoxManage sharedfolder add "Windows 7" --name daten\  
--hostpath "D:\Gemeinsamer Ordner"
```

### 4.4.7 3-D-Beschleunigung testen

Nun kommen wir zu einem Test, in dem wir prüfen, ob die 3-D-Beschleunigung wirklich funktioniert. Dazu stehen Ihnen mehrere Möglichkeiten zur Verfügung. Sie können dazu aus dem Internet freie Testprogramme oder auch OpenGL-Software herunterladen. Dazu zählen zum Beispiel die freie Version der Software Sandra von SiSoftware, die Testversion Everest der Firma Lavalys oder das kostenlose Spiel BillardGL.

#### Quellen:

- Sandra: <http://www.sisoftware.co.uk/>
- Everest: <http://www.lavalys.com/>
- BillardGL: <http://www.billardgl.de/>



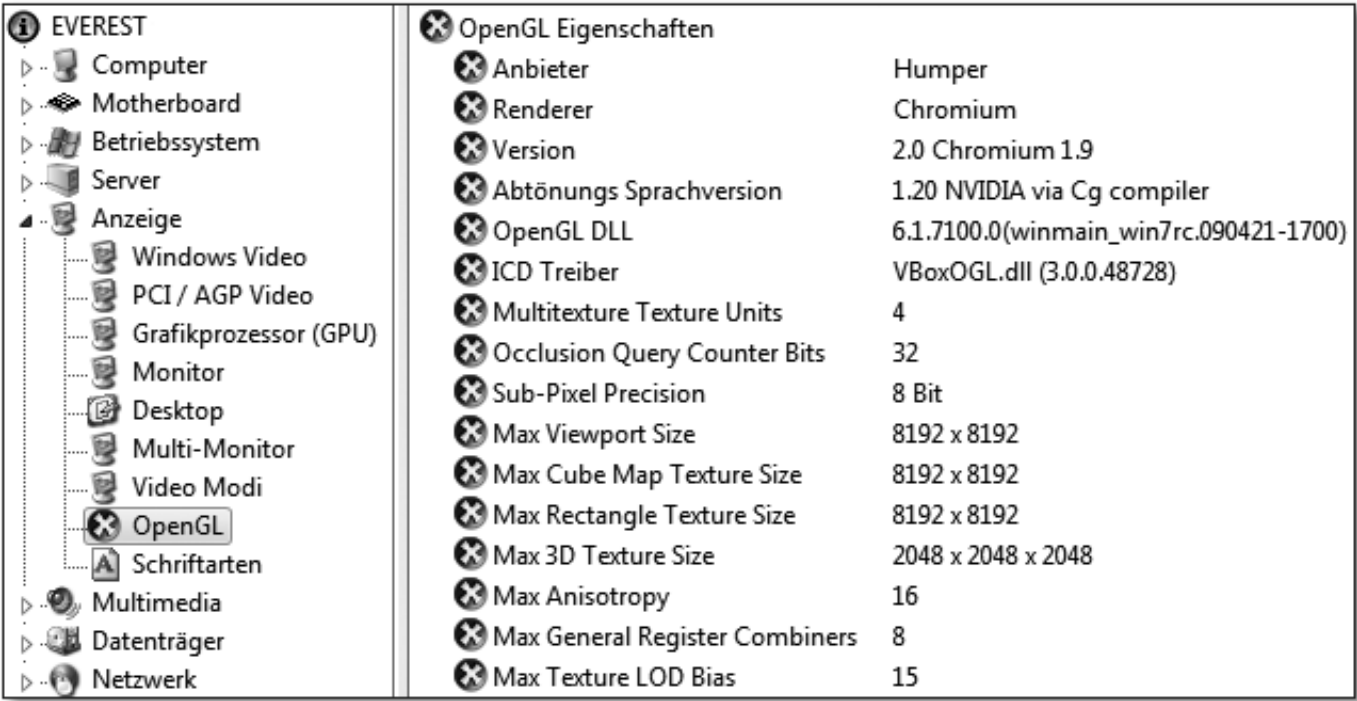


Abbildung 4.65 Infos von Everest mit 3-D-Beschleunigung

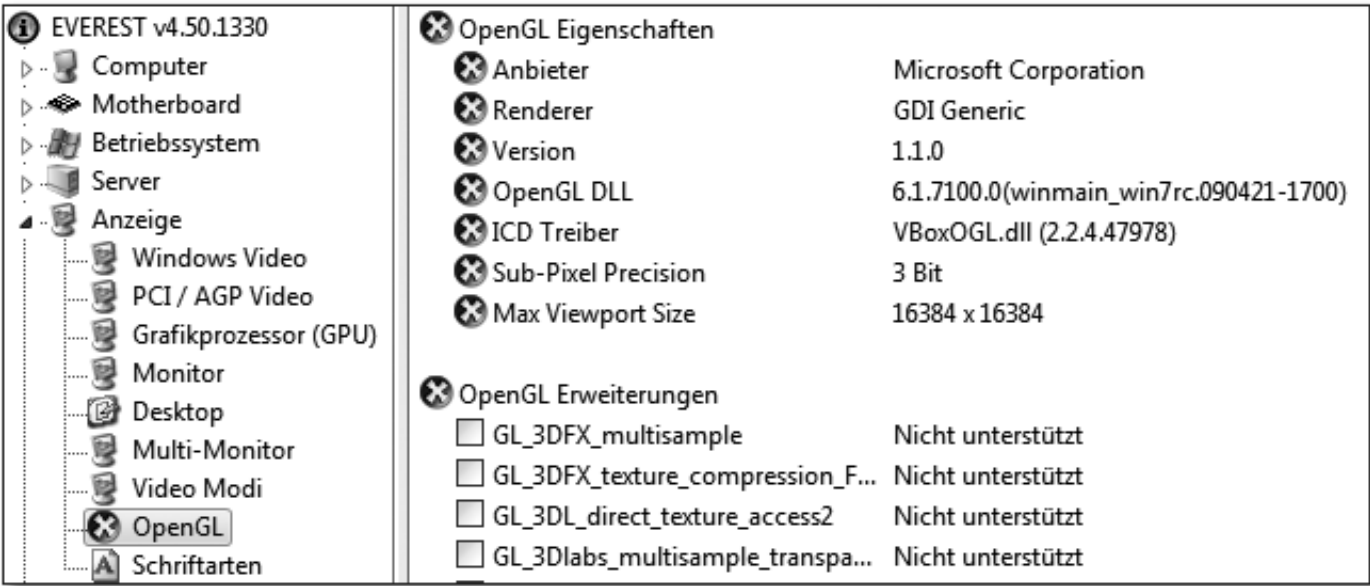


Abbildung 4.66 Infos von Everest ohne 3D-Beschleunigung

In den beiden Abbildungen 4.65 und 4.66 sehen Sie die Ausgabe von Everest – einmal mit und einmal ohne 3-D-Beschleunigung. Dass die Beschleunigung aktiviert ist, erkennen Sie unter anderem an der OpenGL-Versionsnummer 2.0 Chromium 1.9.

Ein Einsatzgebiet der 3-D-Beschleunigung für Linux-Anwender wäre zum Beispiel die Verwendung von 3-D-Spielen unter einem Windows-XP oder Windows-Vista-Gastsystem. Testen wir daher das OpenGL-Spiel BillardGL – vor allem, um zu erkennen, ob die Beschleunigung auch wirklich einen »Schub« in der virtuellen Maschine bringt.



**Abbildung 4.67** Das Spiel BillardGL

Testen Sie das Spiel, wie im vorherigen Test, einmal mit und einmal ohne die Beschleunigung. Sie werden schnell den Unterschied feststellen.

Beachten Sie, dass noch nicht alle Anwendungen und Spiele unter VirtualBox laufen, da die 3-D-Funktion sich noch in der Testphase befindet.

#### 4.4.8 Netzwerkbrücke

Die vorherigen Beispiele verwendeten standardmäßig eine Netzwerkkarte vom Typ PCnet Fast III, und der Netzwerkzugang erfolgte per NAT. Für Windows 7 hat der Assistent nun den Adapter Intel Pro/1000 MT Desktop aktiviert, da Microsoft seit Windows Vista standardmäßig keine Treiber mehr für die AMD PCnet-Adapter anbietet. Als Anschlussart wurde immer noch NAT verwendet. Ihr Gastsystem hat daher Zugriff auf das Netzwerk und auch auf das Internet. Jedoch ist ein Zugriff anderer Netzteilnehmer auf Dienste (Freigaben oder Ähnliches) der Maschine nicht möglich. Um dies zu ändern werden wir nun die Anschlussart ändern. Dazu öffnen Sie die Konfiguration und ändern, wie in Abbildung 4.68 zu erkennen, die Anschlussart auf NETZWERKBRÜCKE. Dann wählen Sie unter Name den Netzwerkadapter, den die Maschine verwenden soll.

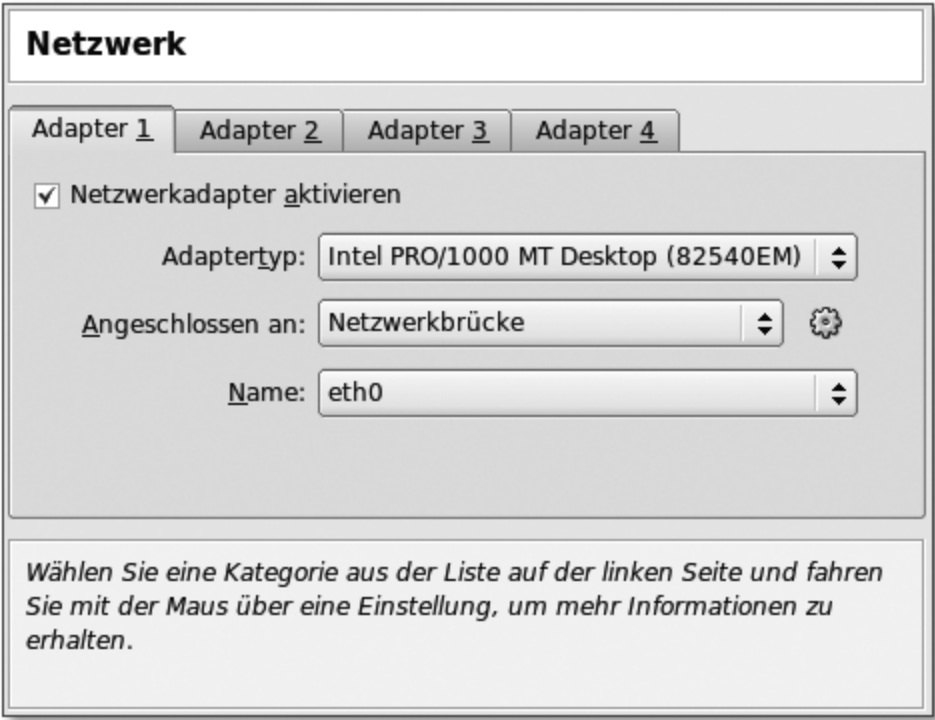


Abbildung 4.68    Netzwerkbrücke für die Schnittstelle eth0

Auf einem Windows-Hostsystem sehen Sie unter NAME den Gerätenamen (Abbildung 4.69) anstelle des für Linux typischen Device-Bezeichners.

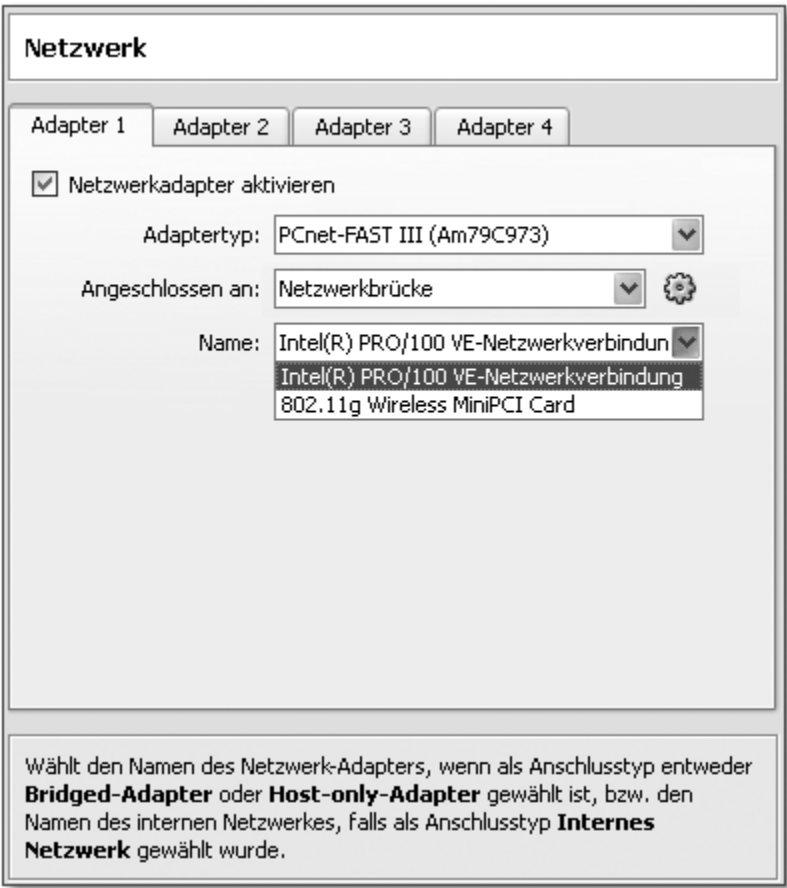
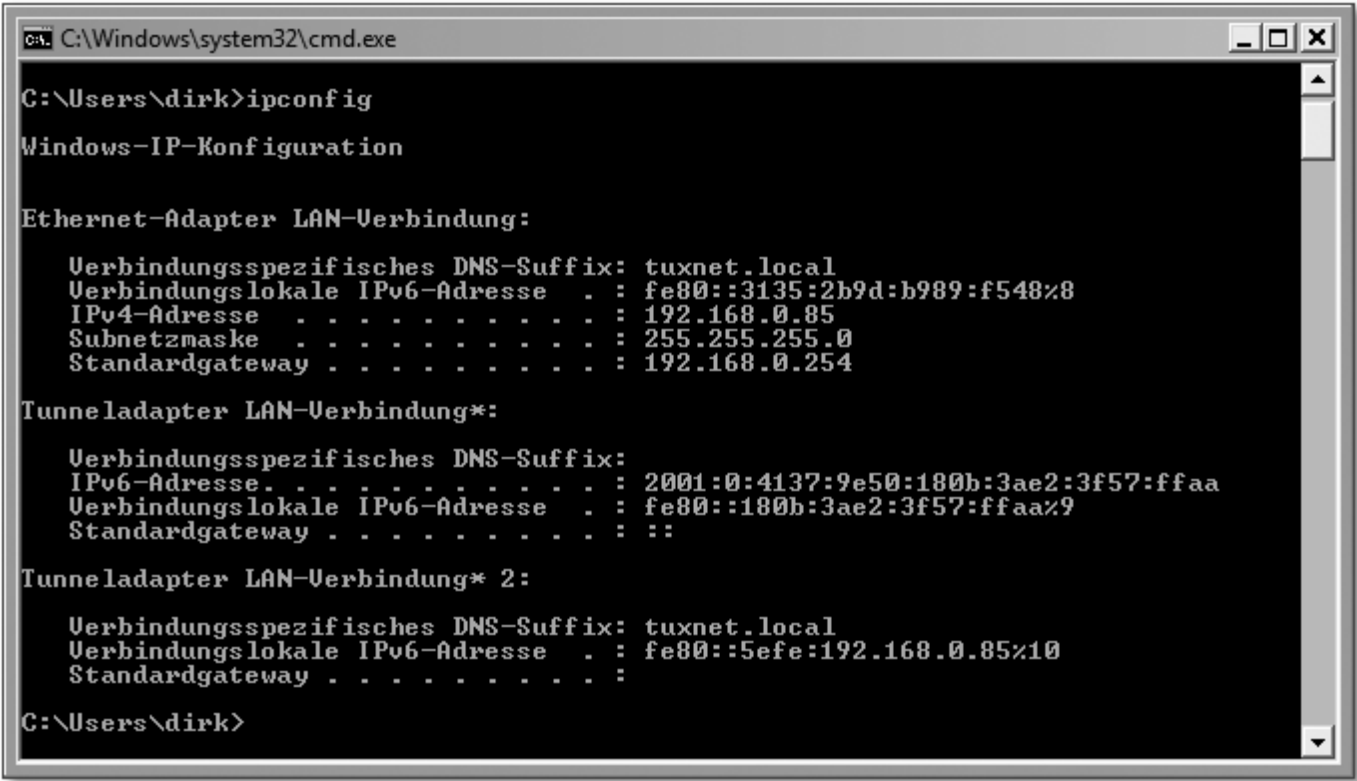


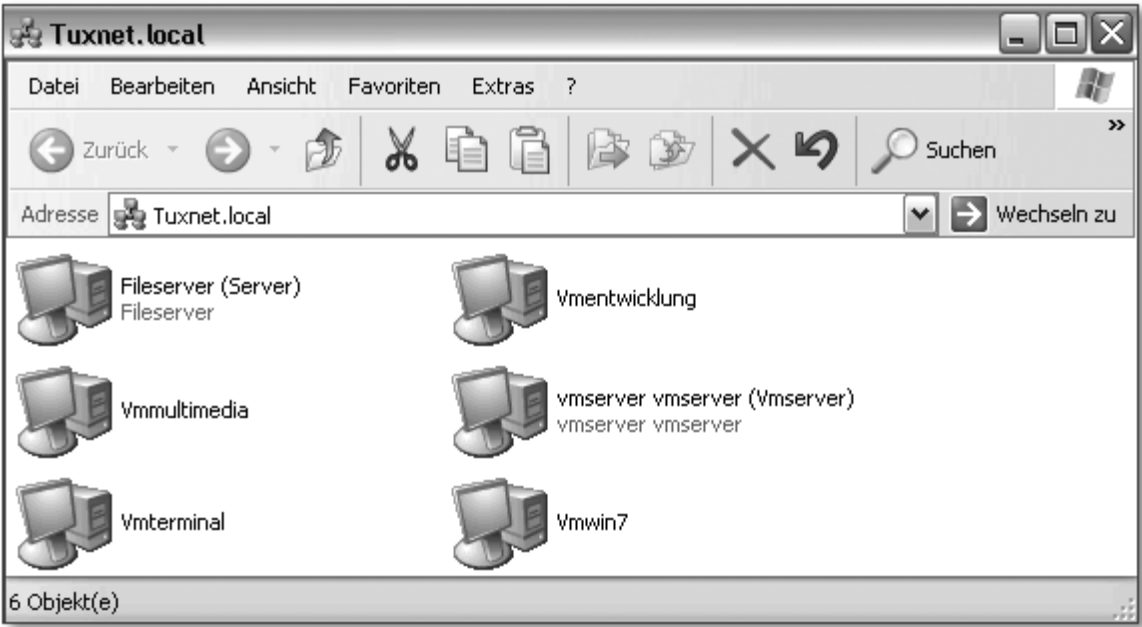
Abbildung 4.69    Gerätenamen unter Windows

Wenn Sie die Maschine nun starten, müssen Sie gegebenenfalls die Netzwerkeigenschaften unter Windows 7 neu konfigurieren. Dies hängt davon ab, ob sich in Ihrem Netzwerk ein DHCP-Server befindet oder nicht.



**Abbildung 4.70** ipconfig zeigt die aktuelle Netzwerkkonfiguration an.

In Abbildung 4.70 sehen Sie die Ausgabe des Kommandozeilentools `ipconfig`. Hier wurde dem Gastsystem von einem DHCP-Server die IP-Adresse 192.168.0.85 zugeteilt. Ab nun verhält sich das Gastsystem wie ein normaler Netzwerkteilnehmer. Sie können es von anderen Rechnern aus unter der vergebenen Adresse »anpingen« oder auf eventuelle Freigaben oder sonstige Dienste zugreifen. Ab nun erscheint die virtuelle Maschine auch in der Netzwerkumgebung anderer Computer (Abbildung 4.71). Gegebenenfalls müssen Sie jedoch noch das Gastsystem Ihrer eigenen Arbeitsgruppe oder Domäne zuordnen. Wenn Sie das Gastsystem nicht erreichen können, prüfen Sie zunächst, ob eine aktivierte Firewall den Zugriff blockiert.



**Abbildung 4.71** Vmwin7 in der Arbeitsgruppe tuxnet.local

### VBoxManage – Netzwerkbrücken (Bridged Adapter)

- Linux:  
VBoxManage modifyvm "Windows 7" --nic1 bridged\  
--bridgeadapter1 eth0
- Windows:  
VBoxManage modifyvm "Windows 7" --nic1 bridged\  
--bridgeadapter1 "Ethernetadapter der AMD-PCNET-Familie"

Adapter korrekt angeben

Achten Sie bei der Verwendung von VBoxManage auf den korrekten Namen des Adapters hinter der Option --bridgeadapter. Das Programm akzeptiert hier leider auch ungültige Bezeichner ohne zu murren.

[+]

## 4.5 Internes Netzwerk

Bisher habe ich Ihnen in den Praxisbeispielen die Netzwerkeinstellungen mit NAT und einer Netzwerkbrücke vorgestellt. Die Gastsysteme der virtuellen Maschinen erhielten damit Zugang zum Internet oder zum lokalen Netzwerk. Eine weitere Möglichkeit ist das interne Netzwerk. Die Einstellung INTERNES NETZWERK dient dazu, mehreren Maschinen ein privates Netzwerk zur Verfügung zu stellen. In diesem können Sie dann Netzwerkanwendungen oder Ähnliches testen, ohne andere Netzwerkteilnehmer zu stören oder von diesen beeinflusst zu werden. Aus dem internen Netzwerk ist, wie bereits in Abschnitt 3.9.2, »Anschlussart«, beschrieben, kein Zugriff auf Ressourcen Ihres physikalischen Netzwerks möglich.

Da in diesem Beispiel mehrere Gastsysteme gestartet werden, sollte Ihr Hostsystem über genügend Ressourcen verfügen. Ansonsten müssen Sie eventuell daran denken, den Maschinen weniger Hauptspeicher zuzuweisen.

Vorkenntnisse sind von Vorteil

Für diesen Abschnitt, in dem Sie lernen, wie Sie interne Netzwerke einrichten und Sicherungspunkte (Snapshots) erstellen, sind grundlegende Netzwerkkenntnisse erforderlich. Da wir bei Verwendung des internen Netzwerks keinen Zugriff auf das vorhandene Netzwerk mehr haben, steht uns standardmäßig auch kein DHCP-Server zur Verfügung. Die Netzwerkadapter müssen daher in den Gastsystemen manuell konfiguriert werden. Sie sollten also unter anderem wissen, wie IP-Adressen vergeben werden und was ein DNS-Server ist.

[+]

4.5.1 Virtuelle Maschinen einrichten

Neue Maschinen werden wir in diesem Abschnitt nicht einrichten. Stattdessen verwenden wir zwei Systeme aus den vorherigen Praxisbeispielen und konfigurieren diese neu. Ich greife dabei auf die Gastsysteme *Windows XP* und *Ubuntu* aus den Abschnitten 4.2, »Windows XP«, und 4.3, »Linux-System«, zurück.

4.5.2 Sicherungspunkte (Snapshots)

Bevor Sie die Konfiguration ändern, erstellen Sie Snapshots der Maschinen, um die Änderungen später wieder rückgängig machen zu können. Dazu markieren Sie die entsprechende Maschine und wechseln in das Register SICHERUNGSPUNKTE (Abbildung 4.72).

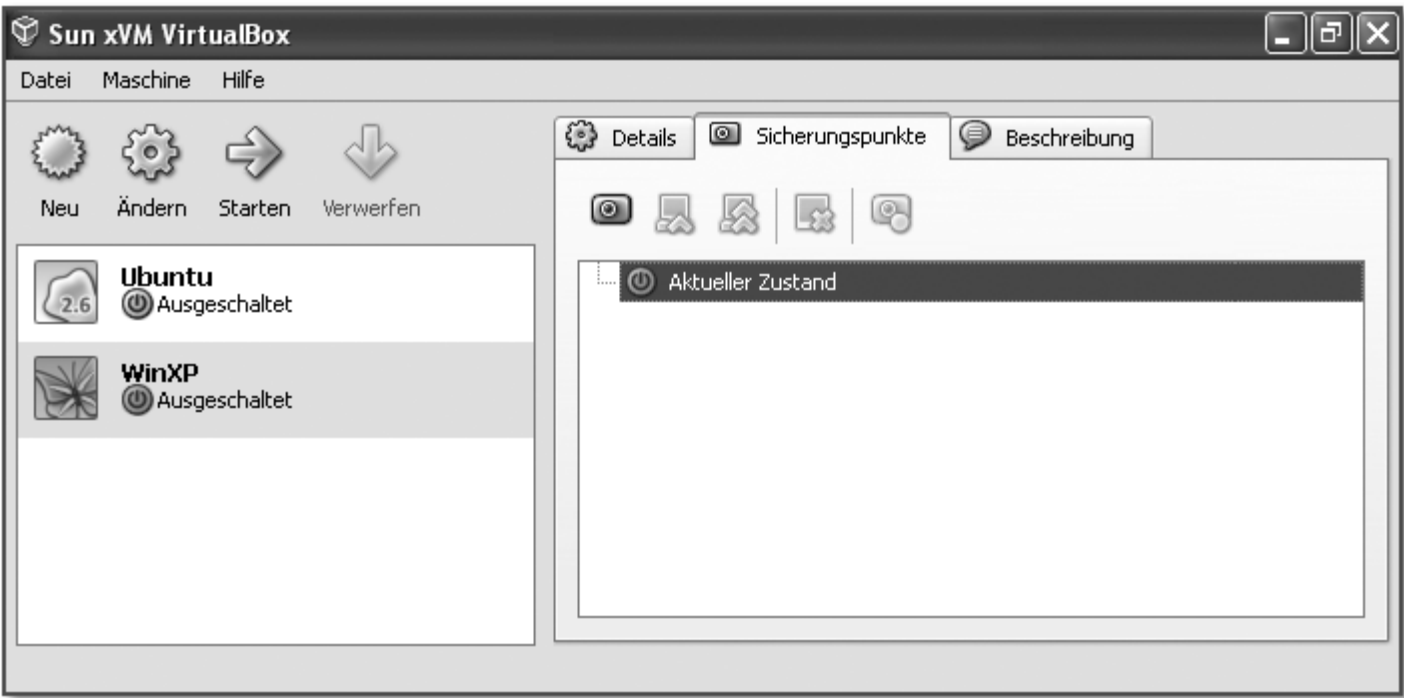


Abbildung 4.72 Im Register »Sicherungspunkte« erstellen Sie Snapshots.

In diesem Register können Sie nun einen Sicherungspunkt erstellen. Dies erfolgt entweder durch einen Klick auf das Fotoapparat-Symbol, durch einen Rechtsklick auf AKTUELLER ZUSTAND oder über die Tastenkombination `Strg` + `Umschalt` + `S`.

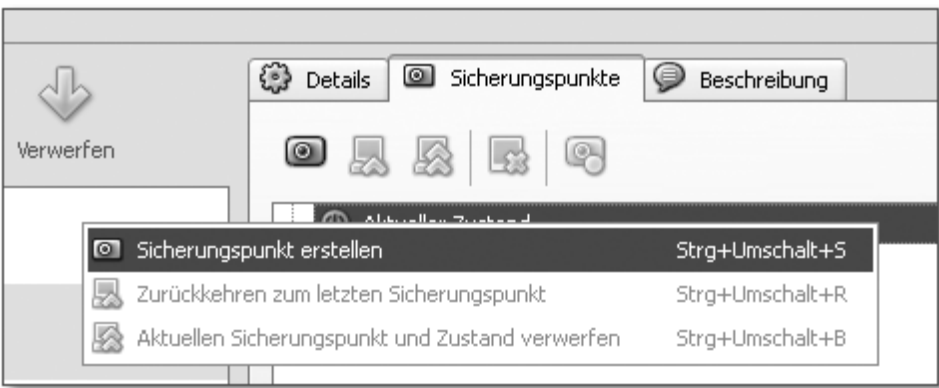


Abbildung 4.73 Einen Sicherungspunkt erstellen

Danach öffnet sich ein Fenster, in dem Sie einen Namen und eine Beschreibung für den Sicherungspunkt vergeben können. Zumindest eine Beschreibung sollten Sie eintragen, da dies bei mehreren Sicherungspunkten sehr hilfreich sein kann, um den korrekten Stand wiederzufinden.

Sie können von jedem Sicherheitspunkt dessen Detailinformationen abrufen. Markieren Sie ihn dazu, und klicken Sie auf den Button EINZELHEITEN ZEIGEN. (siehe Abbildung 4.74).

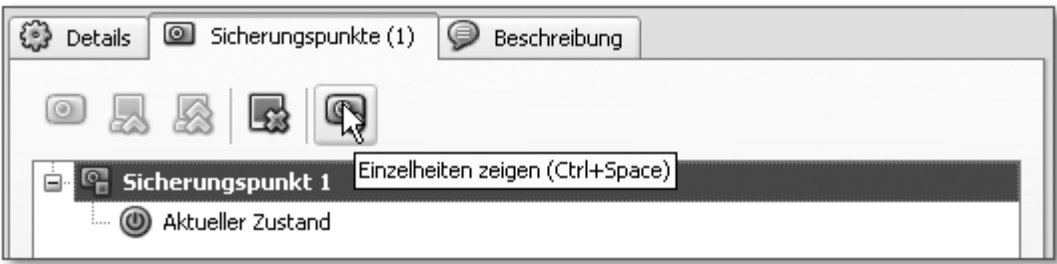


Abbildung 4.74 Die Einzelheiten eines Snapshots anzeigen

In der folgenden Abbildung sehen Sie die Details zu meinem Sicherungspunkt der Windows-XP-Maschine. Der Netzwerkadapter ist in diesem Fall noch über NAT angeschlossen.

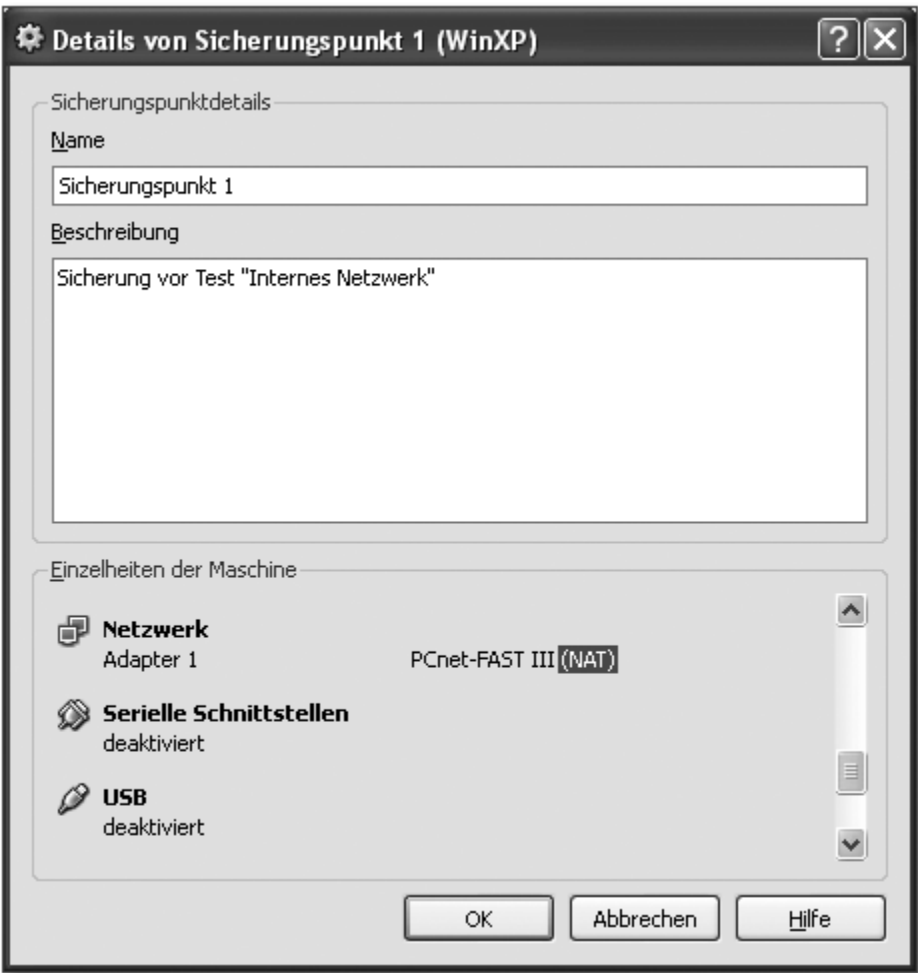


Abbildung 4.75 Details des ersten Sicherungspunkts

Erstellen Sie nun von beiden virtuellen Maschinen die jeweiligen Sicherungspunkte.

### VBoxManage – Sicherungspunkte

Verwenden Sie beim Einsatz von VBoxManage immer Anführungszeichen, um Namen oder Beschreibungen mit Leerzeichen zu übergeben.

- Alle Systeme, Sicherungspunkt erstellen:

```
VBoxManage snapshot WinXP take "Sicherungspunkt 1"\  
--desc "Sicherung vor Test 'Internes Netzwerk'"
```

- Alle Systeme, Einzelheiten zeigen:

```
VBoxManage snapshot WinXP showvminfo "Sicherungspunkt 1"
```

### 4.5.3 Internes Netzwerk einrichten

Ändern Sie nun bei beiden Maschinen die Netzwerkkonfiguration. Dazu wird die Anschlussart auf INTERNES NETZWERK gesetzt und ein NAME vergeben. Damit sich beide Gastssysteme später auch finden, muss bei allen derselbe Netzwerkname eingetragen werden. Ich vergebe in diesem Beispiel einfach den Namen »Netzwerk1«.



**Abbildung 4.76** Internes Netzwerk konfigurieren

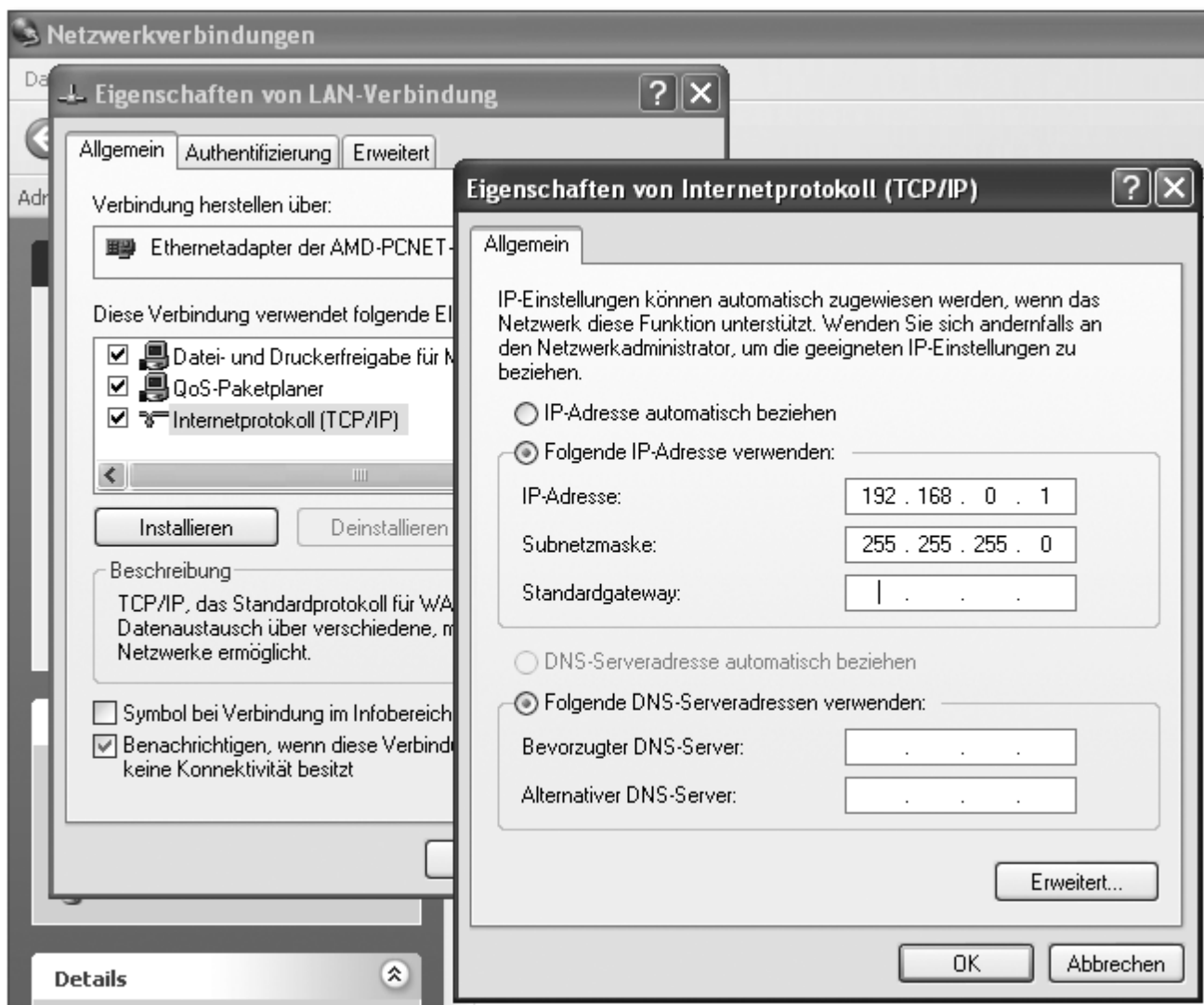
Sie könnten auch mehrere interne Netzwerke parallel einsetzen (vorausgesetzt, die Ressourcen Ihres Hostsystems lassen dies zu). Dazu verwenden Sie einfach verschiedene Netzwerknamen, zum Beispiel »Netzwerk1«, »Netzwerk2«, »Test-netzwerk« usw.

Dadurch ist jedes dieser Netzwerke von den anderen abgekoppelt. Maschinen, die »Netzwerk1« zugeordnet sind, können nicht auf Gastssysteme von »Netzwerk2« zugreifen. Die internen Netzwerke mit verschiedenen Namen können Sie sich dabei wie zwei physikalisch voneinander getrennte Netzwerke vorstellen.



#### 4.5.4 Gastsysteme konfigurieren

Wie bereits in Abschnitt 3.9, »Netzwerk«, beschrieben, stellt VirtualBox bei NAT einen internen DHCP-Dienst zur Verfügung. Bei unserem Beispiel mit der Netzwerkbrücke ging ich davon aus, dass sich ein DHCP-Server in Ihrem Netzwerk befindet (zum Beispiel ein Router), der den Gastsystemen die notwendigen Netzwerkeinstellungen übermittelt und IP-Adressen zuteilt. In diesem Beispiel nehmen Sie die Netzwerkeinstellungen manuell vor, da in unserem internen Netzwerk »Netzwerk1« kein DHCP-Server vorhanden ist. Wir vergeben dabei die IP-Adressen 192.168.0.1 (Windows XP) und 192.168.0.2 (Ubuntu).

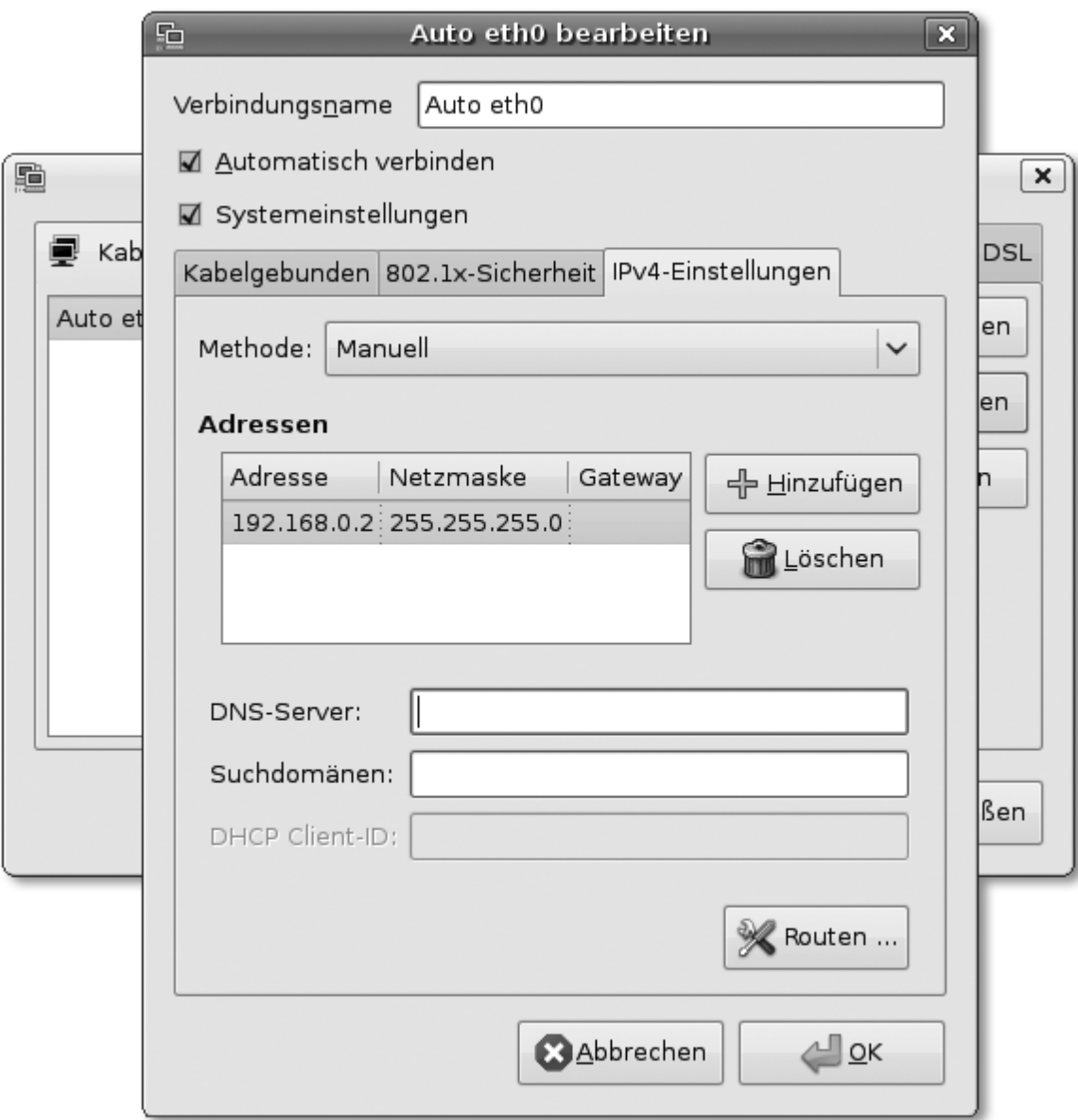


**Abbildung 4.77** Eine IP-Adresse unter Windows XP einrichten

Sie können gegebenenfalls auch andere IP-Adressen verwenden, zum Beispiel welche mit der gleichen Netzwerkmaske Ihres Netzwerks, jedoch mit dort nicht vergebenen IP-Adressen. Dadurch können Sie später mit einem Ping sicher feststellen, ob die Maschinen Zugang zum normalen Netzwerk haben und umgekehrt. Wenn Sie jedoch Adressen verwenden, die in beiden Netzwerken vorhanden, können Sie nicht sicher sein, welche Maschine geantwortet hat (obwohl in diesem Fall normalerweise IP-Adresskonflikte angezeigt werden).

Um unter Windows XP eine IP-Adresse einzurichten, wechseln Sie zunächst über **START • SYSTEMSTEUERUNG • NETZWERK- UND INTERNETVERBINDUNGEN • NETZWERKVERBINDUNGEN** in den Ordner, in dem die Netzwerkadapter aufgeführt werden. Dort öffnen Sie dann die **EIGENSCHAFTEN** des Netzwerkadapters.

Die Netzwerkeinstellungen unter Ubuntu befinden sich im Menü **SYSTEM • EINSTELLUNGEN • NETZWERKVERBINDUNGEN**. Dort tragen Sie im Register **IPv4-Einstellungen** die entsprechende IP-Adresse ein (siehe Abbildung 4.78).



**Abbildung 4.78** Eine IP-Adresse von Hand unter Ubuntu vergeben

Wenn Sie eine andere Distribution verwenden oder Ihre Ubuntu-Version nicht den gleichen Netzwerkmanager verwendet wie in meinem Beispiel, können Sie die Adresse auch in der Konsole vergeben. Diese Einstellungen sind jedoch temporär, und das System »vergisst« die Adresse beim nächsten Neustart. Öffnen Sie dazu die Konsole, und geben Sie den folgenden Befehl ein:

```
sudo ifconfig eth0 192.168.0.2 netmask 255.255.255.0
```

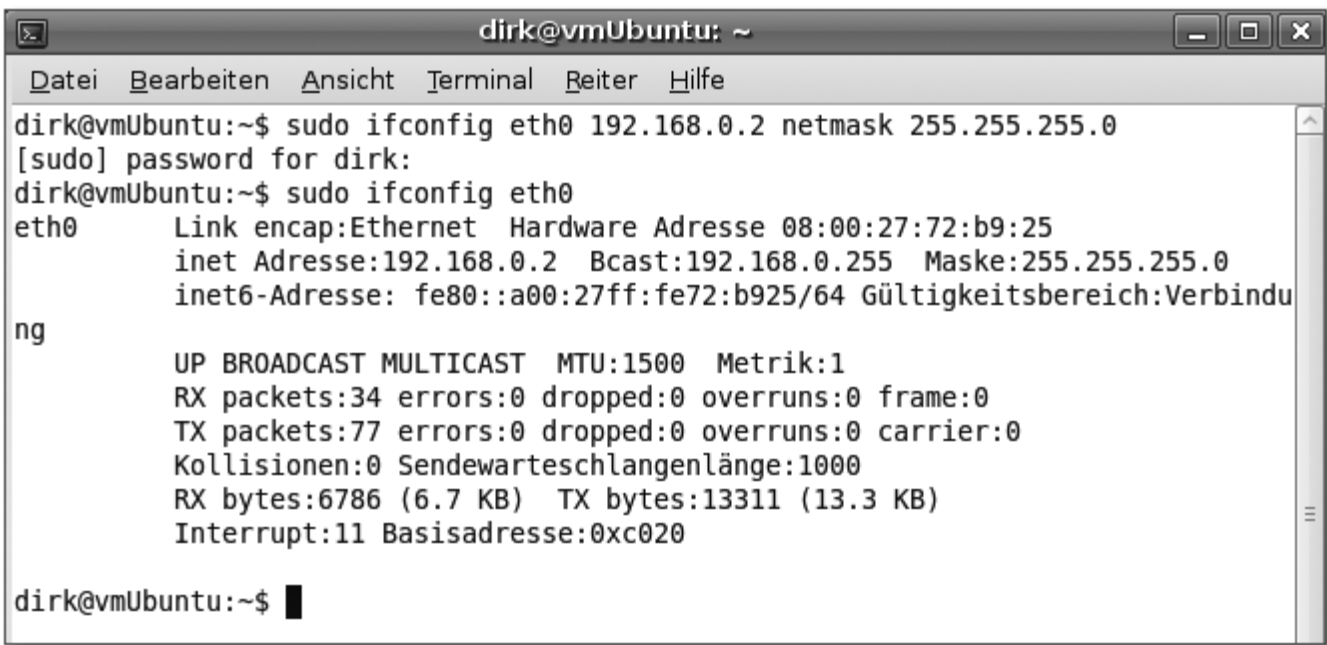


Abbildung 4.79 Adressvergabe mit »ifconfig« auf der Shell

### 4.5.5 Test

Kommen wir nun zu einem einfachen, aber dennoch aussagekräftigen Test des Netzwerks. Dazu starten Sie zunächst beide virtuelle Maschinen und öffnen die Eingabeaufforderung in dem Windows-XP-Gastsystem. Dort überprüfen Sie mit dem Tool `ping`, ob die Ubuntu-Maschine erreichbar ist (Abbildung 4.80).

`ping 192.168.0.2`

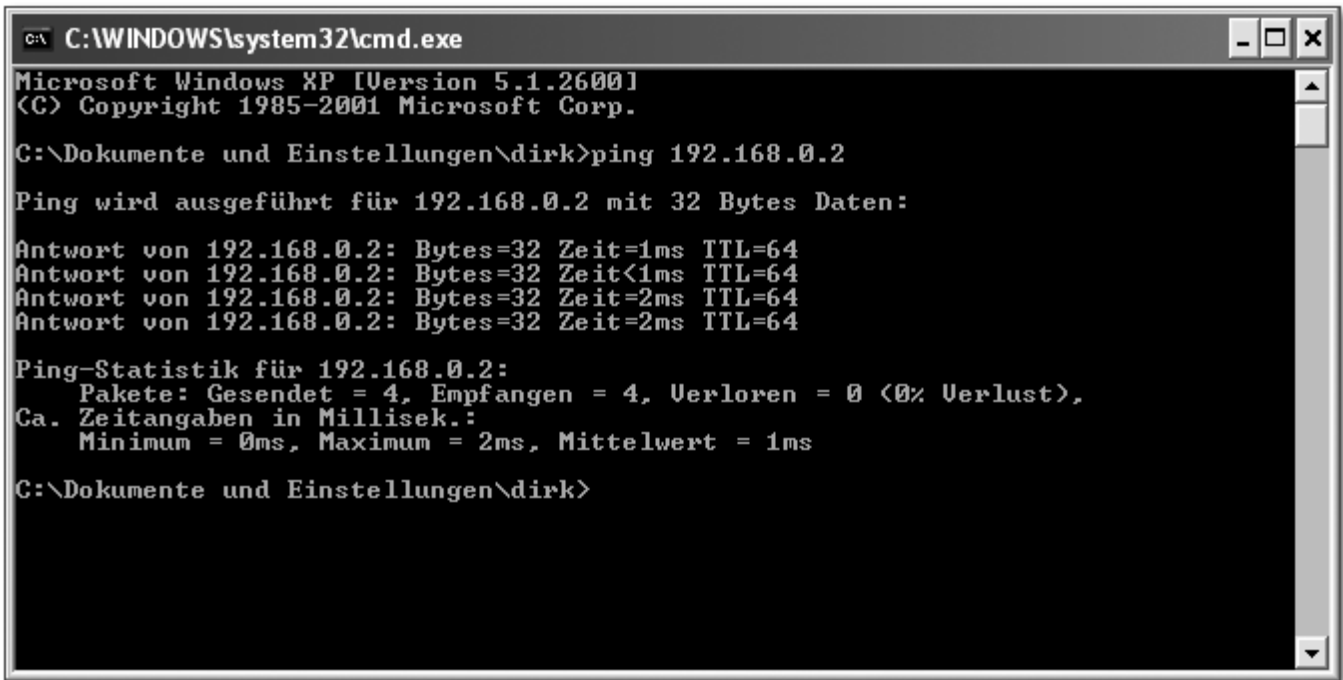


Abbildung 4.80 Pingen Sie den Ubuntu-Rechner (192.168.0.2) von Windows XP (192.168.0.1) aus an.

Falls der Ping bei Ihnen nicht ankommt, prüfen Sie nochmals die Konfiguration der virtuellen Maschinen und Netzwerkeinstellungen (inklusive der Firewall) der Gastsysteme.

Nun können Sie noch testen, ob die PCs Ihres lokalen Netzwerks oder das Internet erreichbar sind. Senden Sie zunächst einen Ping an eine aktuelle IP-Adresse (zum Beispiel an das Hostsystem) und dann an die Adresse einer Webseite. In Abbildung 4.81 habe ich einen Ping an mein Hostsystem (192.168.0.100) und an meinen Webserver abgesetzt. Beide Tests waren nicht erfolgreich.

Wie Sie erkennen, gibt es jedoch zwei verschiedene Gründe dafür. Der Ping an die Adresse 192.168.0.100 ist nicht erfolgreich, da der »Zielhost nicht erreichbar« ist. Der zweite Test hingegen ist erfolglos, weil kein DNS-Server im internen Netz vorhanden ist, der die Namensauflösung übernimmt.

Nach diesem Kurztest können Sie nun, entsprechende Kenntnisse vorausgesetzt, weitere Netzwerkdienste oder Tools einsetzen, um das interne Netzwerk zu verwenden. Entwickler und Administratoren werden sicher viele Einsatzmöglichkeiten erkennen.

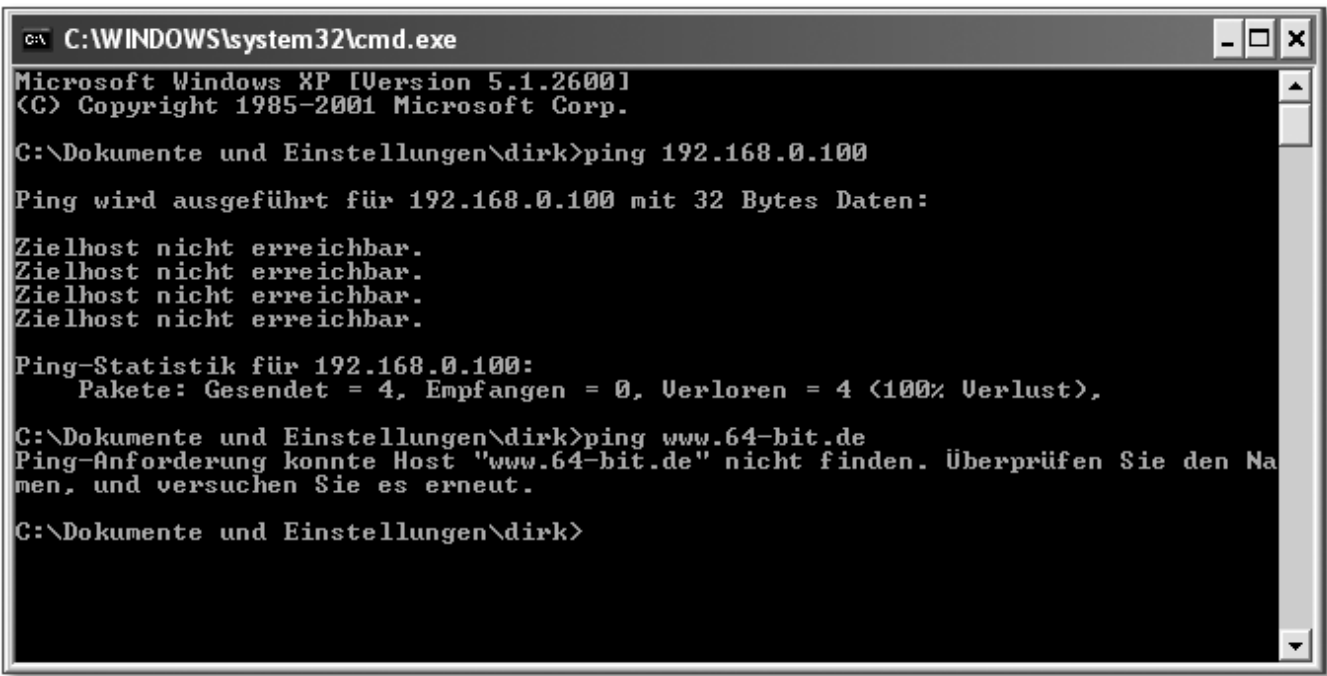


Abbildung 4.81 Kein anderer Teilnehmer ist erreichbar.

4.5.6 Sicherungspunkt

Zu Beginn dieses Beispiels haben wir einen Sicherungspunkt erstellt. Nun können Sie selbst entscheiden, ob Sie die Änderungen für das interne Netzwerk wieder verwerfen oder erhalten. Nehmen wir dazu die Maschine mit Windows XP als Gastsystem. Im Register SICHERUNGSPUNKTE erkennen Sie, dass sich der Zustand der Maschine geändert hat (Abbildung 4.82).

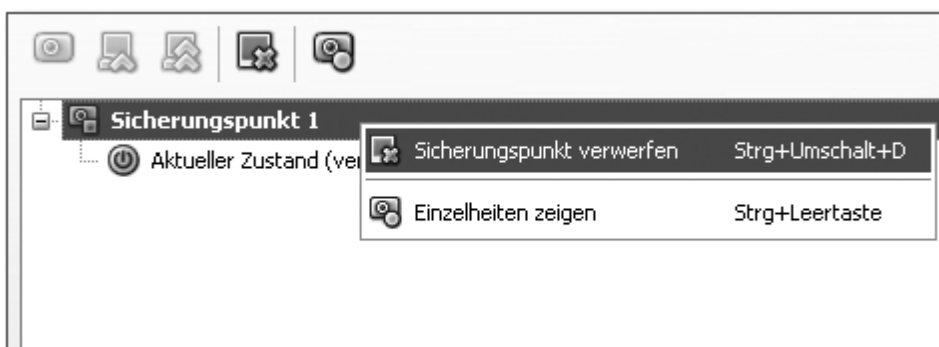


Abbildung 4.82 Der aktuelle Zustand wurde verändert.

## Änderungen erhalten

Wenn die Änderungen erhalten bleiben sollen, muss der Sicherungspunkt verworfen werden. In diesem Fall übernimmt VirtualBox die Änderungen in das gesicherte Festplatten-Image. Der erstellte Sicherungspunkt wird danach entfernt. Markieren Sie dazu den entsprechenden Sicherungspunkt, und wählen Sie im Kontextmenü SICHERUNGSPUNKT VERWERFEN.

Je nachdem, wie viele Änderungen vorgenommen wurden, kann dieser Vorgang mehrere Minuten dauern. Bedenken Sie dabei, dass eventuell mehrere GByte Daten kopiert werden müssen (mehrere Software-Installationen, neues Servicepack usw.). In unserem Beispiel dürften die Änderungen jedoch nach kurzer Zeit erledigt sein, da wir nur die Netzwerkkonfiguration angepasst hatten.



**Abbildung 4.83** Den Sicherungspunkt verwerfen

Nach dem Verwerfen des Sicherungspunkts können Sie nun kontrollieren, ob die Netzwerkkonfiguration noch auf INTERNES NETZWERK eingestellt ist.

## VBoxManage – Änderungen erhalten (Sicherungspunkt verwerfen)

Alle Systeme:

```
VBoxManage snapshot WinXP discard "Sicherungspunkt 1"
```

## Änderungen verwerfen

Soll der vorherige Status wieder hergestellt werden (unter anderem die Netzwerkeinstellung NAT), gibt es dazu zwei Möglichkeiten:

- ▶ ZURÜCKKEHREN ZUM LETZTEN SICHERUNGSPUNKT `[Strg] + [↶] + [R]`
- ▶ AKTUELLEN SICHERUNGSPUNKT UND ZUSTAND VERWERFEN `[Strg] + [↶] + [B]`

Die erstgenannte Funktion verwirft alle Änderungen und kehrt zu dem Status zurück, den die virtuelle Maschine beim Erstellen des Sicherungspunkts hatte. Der Sicherungspunkt bleibt hierbei erhalten. Sie können nun weitere Änderungen vornehmen und immer wieder zu diesem Punkt zurückkehren.

Die zweite Funktion verwirft ebenfalls alle Änderungen. Im Gegensatz zur ersten Option entfernt diese jedoch auch den vorhandenen Sicherungspunkt. Dadurch erhalten Sie wieder exakt den Zustand, den die Maschine vorher hatte.

Um den aktuellen Zustand zu verwerfen, klicken Sie mit der rechten Maustaste auf AKTUELLER ZUSTAND, und wählen Sie die gewünschte Funktion aus dem Kontextmenü (Abbildung 4.84).

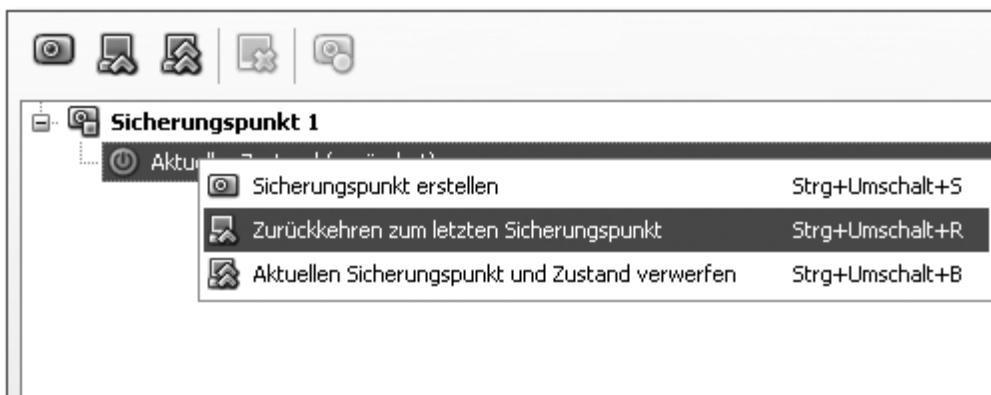


Abbildung 4.84 Die Änderungen verwerfen

### VBoxManage – Änderungen verwerfen (alle Systeme)

- ▶ Alle Systeme, Sicherungspunkt erhalten:  
`VBoxManage snapshot WinXP discardcurrent --state`
- ▶ Alle Systeme, Sicherungspunkt entfernen:  
`VBoxManage snapshot WinXP discardcurrent --all`

## 4.6 VirtualBox als Server

In den bisherigen Beispielen habe ich VirtualBox immer nur als das eingesetzt, was es eigentlich ist: als Desktopvirtualisierer. Es gibt jedoch auch die Möglichkeit, VirtualBox als Virtualisierungsserver zu verwenden. Ich möchte jedoch gleich noch einmal daran erinnern, dass dies kein Ersatz für VMware ESX oder Citrix XenServer ist (siehe auch Abschnitt 2.1.6, »Virtualisierungssoftware«).

Privat verwendete ich bis vor kurzem immer die freie Version VMware Server zur Virtualisierung. Danach stellte ich meine virtuellen Maschinen auf VirtualBox um. Auf meinem kleinen Privatserver laufen meistens fünf bis sechs Maschinen gleichzeitig:

- ▶ Mailserver
- ▶ Desktop
- ▶ Multimedia

- ▶ Entwicklung
- ▶ Windows-XP- und Ubuntu-Testumgebung

Diese Maschinen sind, bis auf den Mailserver, ausschließlich Desktopsysteme. Die Desktopmaschine dient dazu, dass ich von überall mit derselben Oberfläche im Internet surfen, Mails abrufen oder mit OpenOffice arbeiten kann. Der Zugriff erfolgt zu Hause über WLAN und von außen per OpenVPN. Auf der Multimedia-maschine schneide und komprimiere ich aufgenommene Filme, und in der Entwicklungsumgebung schreibe ich Programme und gelegentlich Bücher. Die Testumgebungen sind »frisch« installierte Systeme, auf denen ich neue Software teste. Danach setze ich diese per Sicherungspunkt einfach wieder zurück. Das Hostsystem besteht dabei unter anderem aus folgender Hardware:

- ▶ 2 Xeon 3 GHz
- ▶ 5 GByte Hauptspeicher
- ▶ vier mal 500 GByte Festplatten als RAID 5
- ▶ System: Debian

Viele aktuelle Desktopsysteme erreichen ebenfalls schon diese Werte (bis auf das RAID-System vielleicht). Kleinere Unternehmen dürften auch einen ähnlichen Server zur Verfügung haben.

#### 4.6.1 Virtuelle Maschinen starten

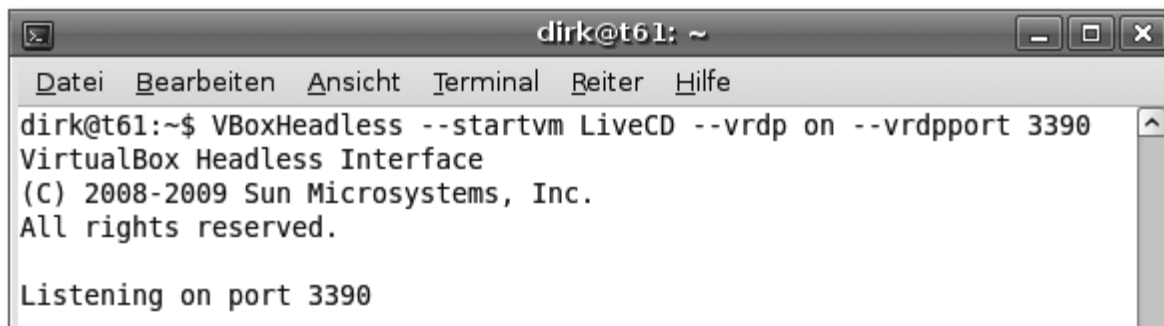
Bisher haben wir die Maschinen immer über die grafische Oberfläche gestartet. Dies kann auch auf einem Server so fortgeführt werden – vorausgesetzt, der Server hat eine eigene grafische Oberfläche. Aus Stabilitäts- und Performancegründen wird jedoch bei Servern, die Linux als Betriebssystem verwenden, häufig auf eben diese verzichtet. Dort kann daher auch die grafische Oberfläche von VirtualBox nicht verwendet werden. VirtualBox bietet für diesen Fall die Möglichkeit, die Maschinen über das Kommandozeilentool VBoxHeadless zu starten. Der Zugriff erfolgt dann über einen Fernzugriff auf den VRDP-Server von VirtualBox.

##### VBoxHeadless

- ▶ Wenn bereits ein Fernzugriff aktiviert ist:  
`VBoxHeadless --startvm LiveCD`
- ▶ Fernzugriff beim Start aktivieren:  
`VBoxHeadless --startvm LiveCD --vrdp on --vrdpport 3390`

Der Fernzugriff kann natürlich auch fest in den Eigenschaften der virtuellen Maschine konfiguriert werden. Dadurch sparen Sie sich die ständige Übergabe der Parameter.

Abbildung 4.85 zeigt VBoxHeadless nach dem Start. Ein Zugriff auf die virtuelle Maschine kann nun über RDP erfolgen.



**Abbildung 4.85** VBoxHeadless starten Sie auf der Kommandozeile.

### 4.6.2 Virtuelle Maschine beenden

Um eine Maschine, die mit VBoxHeadless gestartet wurde, zu beenden, gibt es mehrere Möglichkeiten. Zunächst können Sie VBoxHeadless einfach mit der Tastenkombination `Strg` + `C` beenden. Dies ist jedoch nicht sinnvoll, da es mit der Funktion *poweroff* vergleichbar ist – die Maschine wird also einfach ausgeschaltet.

Eine weitere Möglichkeit ist, je nach Gastsystem, der Einsatz des Kommandozeilentools VBoxManage:

```
VBoxManage controlvm LiveCD acpipowerbutton
```

Die letzte und sicher geläufigste Methode ist, das Gastsystem über den Fernzugriff sauber herunter zu fahren. Falls dieses keine ACPI-Funktion verwendet, um die Sitzung zu beenden, können Sie VBoxHeadless danach einfach per Tastenkombination schließen.

### 4.6.3 Sitzung aufzeichnen

Nun kommen wir zu einer Besonderheit von VBoxHeadless. Eine mit diesem Kommandozeilentool gestartete Maschine kann man »aufzeichnen«. Dabei wird im Hintergrund ein Video erstellt, in dem sämtliche Vorgänge aufgenommen werden. Dieser Film kann später zum Beispiel für Vorführungen oder Schulungen genutzt werden. Allerdings sollten Sie hierzu über genügend Festplattenspeicher (oder besser noch einen separaten Datenträger) für die Aufzeichnung verfügen. Es kommen hierbei sehr schnell mehrere GByte zusammen.



**Aufzeichnungsfunktion****[+]**

Diese Funktion ist nicht im Originalhandbuch von VirtualBox beschrieben. Es könnte daher sein, dass sie demnächst wieder entfernt wird.

Zum Starten einer Aufzeichnung setzen Sie folgende Parameter:

- ▶ `capture`,
- ▶ `width`
- ▶ `height`
- ▶ `bitrate`
- ▶ `filename`

Mit `capture` aktivieren Sie die Aufnahme. Über `width`, `height` und `bitrate` legen Sie die Auflösung und Qualität des Videos fest. Die Bitrate geben Sie dabei in Bytes an; sie muss zwischen 300.000 und 1.000.000 liegen. Mit dem Parameter `filename` definieren Sie den Speicherort und Namen des Videos.

Das Format des Videos ist das bewährte MPEG-Format. Eine Voraussetzung dafür ist, dass die Container-Sammlung FFmpeg installiert ist. Diese enthält verschiedene freie Video- und Audio-Codecs für Windows- und Linux-Systeme. Sie laden FFmpeg von <http://ffmpeg.org/> oder <http://ffdshow-tryout.sourceforge.net/> herunter. Viele Linux-Distributionen enthalten diese Sammlung bereits in ihren Paketarchiven, sodass Sie sie bequem über den Paketmanager einspielen können.

**VBoxHeadless – Aufnahmen**

```
VBoxHeadless --startvm WindowsXP\
  --capture --width 800 --height 600 --bitrate 500000\
  --filename D:\Aufnahmen\XP01.mpeg
```

Mit den Parametern `width`, `height` und `bitrate` müssen Sie ein wenig experimentieren. Eine hohe Auflösung oder niedrige Bitrate fordern auch hohe Systemressourcen. Je nach Hostsystem leidet darunter natürlich die Ausführung der virtuellen Maschine.



Die zentralen Funktionen von VirtualBox kennen Sie bereits. In diesem Kapitel lesen Sie nun, wie Sie VirtualBox noch gezielter Ihren Bedürfnisse anpassen und die Konfiguration optimieren können.

## 5 Weitere Funktionen und Tipps

Nach dem Durcharbeiten der Praxisbeispiele haben Sie nun einen sehr guten Überblick über die Anwendungsmöglichkeiten von VirtualBox. Sie können virtuelle Maschinen erstellen und diese verwalten. Allerdings gibt es noch einige weitere Funktionen und Möglichkeiten, die ich Ihnen auf den folgenden Seiten vorstellen werde.

### 5.1 Arbeiten mit dem Manager für virtuelle Medien

Das Arbeiten mit Images aller Art ist unter VirtualBox mit dem Manager für virtuelle Medien ziemlich einfach. Allerdings kann es gerade für Anfänger auch zu »verworrenen« Situationen kommen, in denen sich registrierte Medien scheinbar nicht entfernen lassen. Daher stelle ich in diesem Abschnitt den Manager für virtuelle Medien noch genauer vor.

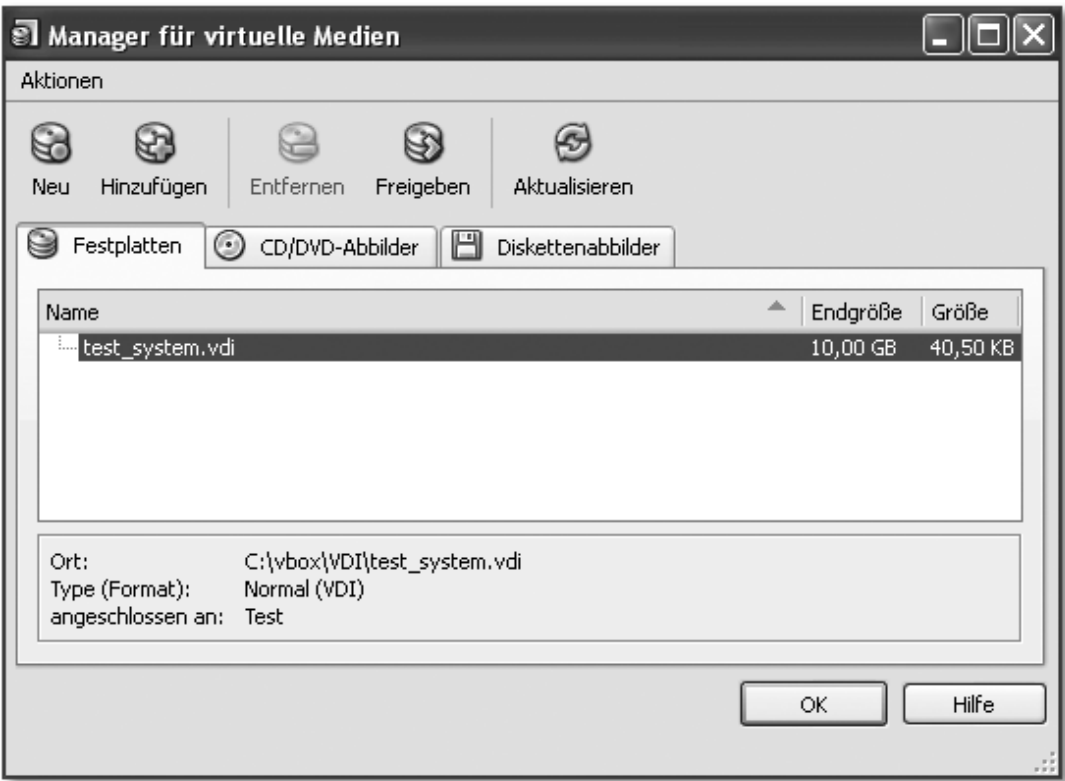


Abbildung 5.1 Der Manager für virtuelle Medien

Das Fenster des Managers enthält drei Register (siehe Abbildung 5.1):

- ▶ Festplatten
- ▶ CD/DVD-Abbilder
- ▶ Diskettenabbilder

Jedes dieser Register verwaltet die entsprechenden Medien. In der Infobox im unteren Fensterbereich erscheinen die Informationen zum aktuell markierten Medium (Ablageort, Typ und ob es einer virtuellen Maschine zugewiesen ist).

### 5.1.1 Festplatten

Im Manager für virtuelle Medien können Sie weitere Festplatten-Images erstellen, entfernen oder vorhandene wieder registrieren.

#### Neu

Mit einem Klick auf NEU in der Symbolleiste erstellen Sie ein neues Festplatten-Image. Sie können hierbei zunächst entscheiden, ob es sich um ein DYNAMISCHES MEDIUM oder ein MEDIUM MIT FESTER GRÖSSE handeln soll (Abbildung 5.2).



**Abbildung 5.2** Ein neues Festplatten-Image erstellen

**DYNAMISCH WACHSENDES MEDIUM:** Das Festplatten-Image wird erst dann vergrößert, wenn die virtuelle Maschine diesen Speicherplatz auch benötigt. Dadurch wird auf dem Hostsystem bedeutend weniger Festplattenspeicher belegt. Gerade bei vielen virtuellen Maschinen macht sich dies sehr schnell bemerkbar. Ein wei-

terer Vorteil ist, dass mehr und größere Images angelegt werden können, als der Festsplattenspeicher des Hostsystems hergibt. Dieser Vorteil ist aber auch gleichzeitig ein Nachteil. Sie müssen aufpassen, dass der freie Festsplattenspeicher des Hostsystems nicht unbemerkt zur Neige geht.

**MEDIUM MIT FESTER GRÖSSE:** In diesem Fall belegt das Image sofort so viel Speicherplatz auf dem Hostsystem, wie Sie der virtuellen Festplatte zugeteilt haben.

Im nächsten Schritt legen Sie den Ablageort, den Dateinamen und die Größe des Images fest.



**Abbildung 5.3** Die Lage und Größe des Images bestimmen

Standardmäßig speichert VirtualBox die Images in dem in der Konfiguration festgelegten VDI-Ordner. Über einen Klick auf das in Abbildung 5.3 zu sehende Ordnersymbol können Sie den Ablageort ändern. Als Name verwende ich immer einen Zusammenschluss aus dem Maschinennamen und einer Beschreibung, zum Beispiel *winxp\_system.vdi*, *ubuntu\_daten.vdi* oder *server\_sql.vdi*.

Die Größe stellen Sie über den Schieberegler oder von Hand ein. Bei manueller Eingabe verwenden Sie die Kürzel »MB« für MByte, »GB« für GByte und »TB« für TByte.

### **VBoxManage – neues Image erstellen**

- Linux; dynamisch wachsendes Festplatten-Image von 10.000 MByte Größe:

```
VBoxManage createhd --filename /media/vbox/VDI/test_system.vdi\  
--size 10000 --variant Standard
```

- ▶ Windows; Festplatten-Image von 1.000 MByte Größe (mit Registrierung):

```
VBoxManage createhd --filename E:\VBox\VDI\win_daten.vdi\  
--size 1000 --variant Fixed --remember
```

Ein Festplatten-Image, das mit dem Manager für virtuelle Medien erstellt wurde, ist auch sofort in diesem registriert. Wenn Sie VBoxManage auf der Kommandozeile verwenden, geben Sie zusätzlich den Parameter `--remember` an, oder registrieren Sie es nach der Erstellung mit dem Befehl `openmedium`.

### Hinzufügen

Über den Button HINZUFÜGEN registrieren Sie ein bereits vorhandenes Image. Auf diese Weise spielen Sie gesicherte Medien wieder ein oder melden eine Maschine auf einem anderen Hostsystem an.

### VBoxManage – vorhandenes Image hinzufügen

- ▶ Linux:  
VBoxManage **openmedium disk** /media/vbox/VDI/test\_system.vdi
- ▶ Windows:  
VBoxManage **openmedium disk** E:\VBox\VDI\win\_daten.vdi

### Entfernen und Freigeben

Über ENTFERNEN kann ein Festplatten-Image aus der Registrierung entfernt und, wenn nötig, komplett gelöscht werden. Allerdings sollten Sie hierbei verschiedene Punkte beachten:

- ▶ Solange ein Image einer Maschine zugewiesen ist, kann es nicht entfernt werden. Diese Zuordnung muss dann zunächst durch Klick auf FREIGEBEN aufgehoben werden.
- ▶ Wenn ein Festplatten-Image in einem Snapshot enthalten ist, können Sie es ebenfalls nicht entfernen. In dem Fall muss zunächst der Snapshot verworfen und dann das Festplatten-Image freigegeben werden.

Wenn Sie diese Punkte nicht beachten, kann es zu den erwähnten »verworrenen« Situationen kommen, und Sie können nicht sofort erkennen, warum sich ein Image nicht entfernen lässt.

In nächsten Schritt (Abbildung 5.4) entscheiden Sie nun, ob Sie das Image komplett (LÖSCHEN) oder nur aus der Registrierung entfernen (BEHALTEN).



Abbildung 5.4 Löschen oder nur Abmelden

### VBoxManage – Image freigeben

- ▶ Linux:  
VBoxManage **closemedium disk** /media/vms/vbox/VDI/test\_system.vdi
- ▶ Windows:  
VBoxManage **closemedium disk** E:\VBox\VDI\win\_datan.vdi

Bei der Verwendung von VBoxManage wird das Image nicht gelöscht. Diesen Schritt nehmen Sie dann selbst in einem Dateimanager oder mit dem entsprechenden Befehl in der Konsole bzw. Eingabeaufforderung vor.

### Probleme

Gelegentlich kann es beim Arbeiten mit Festplatten-Images zu Problemen kommen. Wenn Sie zum Beispiel ein Image nicht über den Manager sondern auf andere Weise löschen, ist dieses noch in VirtualBox registriert und gegebenenfalls sogar einer virtuellen Maschine zugeordnet. In diesem Fall warnt Sie VirtualBox bereits beim Start, dass ein registriertes Image nicht mehr vorhanden ist.

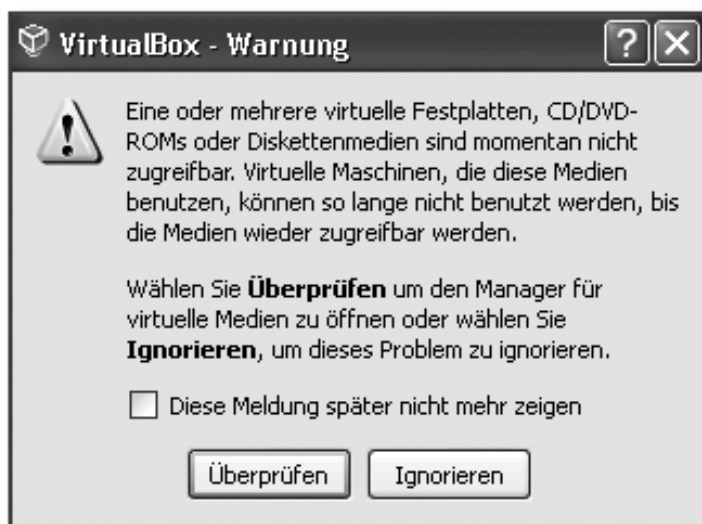
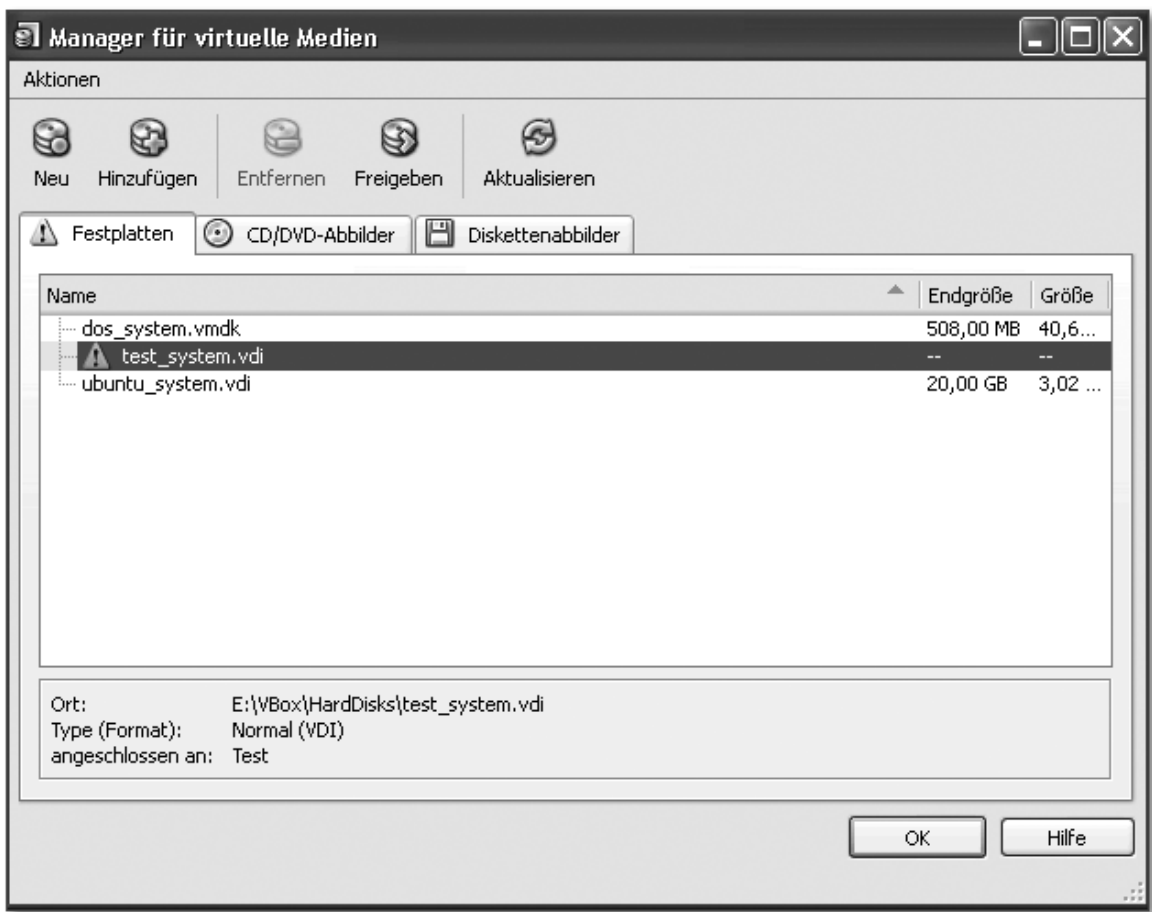


Abbildung 5.5 VirtualBox zeigt einen Warnhinweis, wenn ein Image nicht mehr verfügbar ist.

Wenn Sie nun den Manager für virtuelle Medien aufrufen, ist der Übeltäter schnell gefunden. In Abbildung 5.6 sehen Sie, dass die Datei *test\_system.vdi* nicht gefunden wurde. Weiterhin verrät der Infobereich, dass die Datei der Maschine **Test** zugeordnet ist.



**Abbildung 5.6** Die Datei *test\_system.vdi* wurde offenbar entfernt.

Das Problem schaffen Sie aus der Welt, indem Sie beispielsweise ein Backup einspielen oder das Festplatten-Image aus der Registrierung entfernen.

**5.1.2 CD/DVD-Abbilder**

CD/DVD-Abbilder können Sie genau wie Festplatten-Images einfach über HINZUFÜGEN registrieren und mit ENTFERNEN wieder aus der Registrierung löschen. Auch hier kann es zu Problemen kommen, wenn ein Abbild ohne korrekte Abmeldung entfernt wird.

Abbilder, die registriert werden sollen, müssen im ISO-Format vorliegen. Das ebenfalls weit verbreitete Format NRG von Nero oder die Formate anderer Hersteller sind nicht kompatibel. Wenn Sie aus vorhandenen CDs/DVDs selbst ISO-Abbilder erstellen wollen, können Sie dazu beispielsweise ein Brennprogramm wie CDBurnerXP (<http://www.cdburnerxp.org/>) oder K3b (<http://www.k3b.org/>) zu Hilfe nehmen.



**VBoxManage – CD/DVD-Abbild einbinden**

## ► Linux:

```
VBoxManage openmedium dvd /nfs/brennen/pmagic-4.0.iso
```

## ► Windows:

```
VBoxManage openmedium dvd E:\Images\pmagic-4.0.iso
```

**Mit VBoxManage ein CD/DVD-Abbild abmelden**

## ► Linux:

```
VBoxManage closemedium dvd /nfs/brennen/pmagic-4.0.iso
```

## ► Windows:

```
VBoxManage closemedium dvd E:\Images\pmagic-4.0.iso
```

**5.1.3 Disketten**

Disketten-Images werden meist nur von älteren Gastsystemen benötigt. Allerdings können sie auch verwendet werden, um Daten mit virtuellen Maschinen auszutauschen, da Floppy-Laufwerke von fast allen Gastsystemen unterstützt werden. Um ein eigenes Image zu erstellen, legen Sie einfach eine leere Datei an. Diese kann dann im Manager für virtuelle Medien als Disketten-Image eingebunden und von den virtuellen Maschinen verwendet werden. Eine Neuanlage über den Manager ist leider nicht möglich. Ein neues Image verhält sich dabei wie eine unformatierte 1,44-MByte-Diskette und ist anfangs weder lesbar noch schreibbar. Sie muss vor der Verwendung mit einem entsprechenden Programm des Gastsystems formatiert werden. Um eine leere Datei zu erstellen, gibt es mehrere Möglichkeiten. Die einfachste Methode, die unter den meisten Betriebssystemen möglich ist, ist die Anlage einer Textdatei. Unter Ubuntu und in der Desktopumgebung GNOME wählen Sie hierzu im Kontextmenü eines Ordners DOKUMENT ANLEGEN • LEERE DATEI (siehe Abbildung 5.7). Im Windows-Explorer rufen Sie NEU • TEXTDOKUMENT auf.

Der neuen Datei geben Sie dann die Dateiendung *.img*. Danach können Sie sie mit dem Manager für virtuelle Medien einbinden.

**Dateinamenerweiterung in Windows**

Windows-Systeme zeigen die Erweiterung der Dateinamen standardmäßig nicht an. Um diese sichtbar zu machen, deaktivieren Sie im Explorer unter EXTRAS • ORDNEROPTIONEN • ANSICHT die Option ERWEITERUNGEN BEI BEKANNTEN DATEITYPEN AUSBLENDEN.

**[+]**

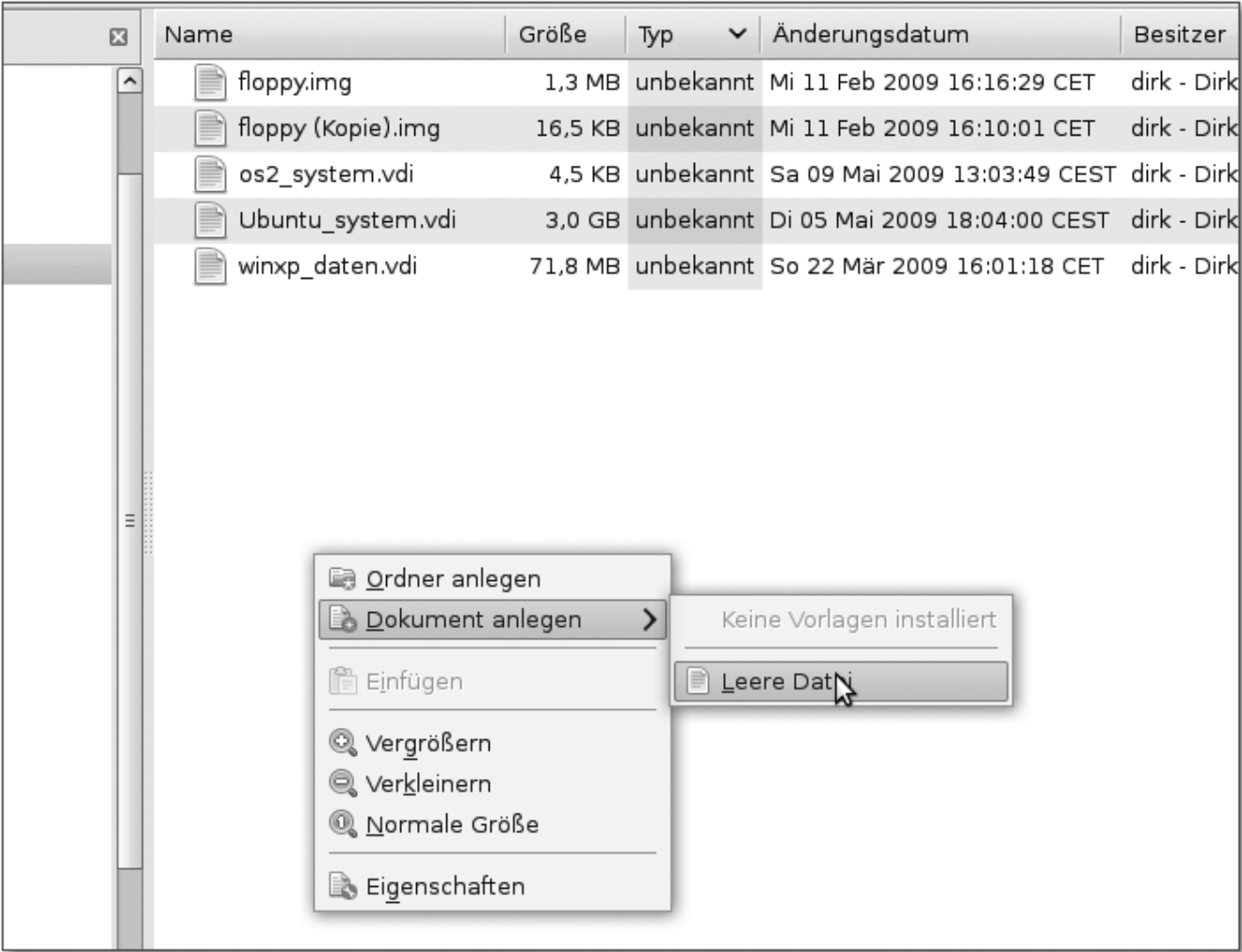


Abbildung 5.7 Eine leere Datei unter Ubuntu (bzw. GNOME) erstellen

### VBoxManage – Disketten Image einbinden

- ▶ Linux:  
`VBoxManage openmedium floppy /media/vms/Diskette1.img`
- ▶ Windows:  
`VBoxManage openmedium floppy D:\images\Diskette1.img`

## 5.2 Arbeiten mit Festplatten-Images

Die folgenden Abschnitte zeigen, wie Sie Festplatten-Images vergrößern, verkleinern, kopieren (klonen) und konvertieren.

### 5.2.1 Festplatten-Image vergrößern

Das Vergrößern eines Festplatten-Images ist mit den Bordmitteln von VirtualBox nicht möglich. Sie müssen dazu leider einen Umweg gehen und ein neues Festplatten-Image mit der gewünschten Größe erstellen, dieses einbinden und das System bzw. die Daten darauf kopieren. Dazu sind folgende Schritte notwendig:

- ▶ ein neues Festplatten-Image erstellen und einbinden
- ▶ Live-CD Parted Magic einbinden und booten

- ▶ mit `dd` den Inhalt des alten 1:1 auf das neue Image kopieren
- ▶ mit GParted die Größe der Partitionen an neues Festplatten-Image anpassen
- ▶ in der virtuellen Maschine das neue Image als erste Festplatte einbinden

Zu den ersten beiden Punkten muss ich sicherlich nichts mehr erklären. Der dritte Punkt hingegen ist Ihnen vielleicht weniger bekannt. Mit diesem Befehl können Sie unter Linux unter anderem Datenträger kopieren. Dabei muss zunächst ermittelt werden, wie die Device-Namen der einzelnen virtuellen Festplatten lauten (siehe auch Abschnitt 2.2.1, »Virtuelle Maschine einrichten«). Dies kann in der Konsole mit dem Befehl `fdisk -l` erfolgen. Das eigentliche Kopieren starten Sie dann mit `dd` (siehe Abbildung 5.8).

```

root@PartedMagic:~# fdisk -l

Disk /dev/hda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0xd9f3d9f3

   Device Boot      Start         End      Blocks    Id  System
/dev/hda1  *           1         1304     10474348+    7  HPFS/NTFS

Disk /dev/hdb: 21.4 GB, 21474836480 bytes
16 heads, 63 sectors/track, 41610 cylinders
Units = cylinders of 1008 * 512 = 516096 bytes
Disk identifier: 0x00000000

Disk /dev/hdb doesn't contain a valid partition table
root@PartedMagic:~# dd if=/dev/hda of=/dev/hdb
20971520+0 records in
20971520+0 records out
10737418240 bytes (11 GB) copied, 1112.04 s, 9.7 MB/s
root@PartedMagic:~#

```

**Abbildung 5.8** `fdisk` und `dd` unter Parted Magic

Achten Sie bei der `fdisk`-Ausgabe unbedingt darauf, welches Device die Quelle mit dem Gastssystem und welches das Ziel ist (bei einem neuen Festplattenimage ist das Ziel ohne Partitionstabelle- »**doesn't contain a valid partition table**«). Je nachdem, wie die Festplatten-Images eingebunden wurden, ändert sich auch der Device-Name.

Beispiele:

- ▶ Beide Images am IDE-Controller (primärer Master und Slave):  
`dd if=/dev/hda of=/dev/hdb`
- ▶ Beide Images an SATA oder SCSI:  
`dd if=/dev/sda of=/dev/sdb`
- ▶ Quelle an SATA oder SCSI und das Ziel am IDE-Controller:  
`dd if=/dev/sda of=/dev/hda`

dd kopiert den kompletten Inhalt (die Partitionsinformationen und Dateien) der Quelle auf das Ziel. Der Parameter `if` (*input file*) gibt die Quelle und `of` (*output file*) das Ziel an. Dieser Vorgang kann, je nach Größe, mehrere Minuten dauern. Leider gibt dd keine Statusinformationen preis. Nach dem erfolgreichen Kopiervorgang vergrößern Sie als Nächstes mit GParted die Partition auf dem neuen Image. Da dd die Partition 1:1 kopiert hat, ist sie zwar auf dem neuen Image vorhanden, allerdings noch in ihrer alten Größe.



**Mögliche Fehlermeldung von GParted**

Gelegentlich kann es vorkommen, dass GParted zunächst einen Fehler meldet und die Partitionen nicht erkennt. In diesem Fall starten Sie die virtuelle Maschine mit Parted Magic neu.

Nach dem Anpassen der Größe können Sie nun die Konfiguration der Maschine ändern und das alte durch das neue Festplatten-Image ersetzen. Falls Sie vorhaben, das alte Image zu löschen, empfehle ich, dies erst nach erfolgreichen Tests vorzunehmen.

Unter Linux-Systemen kann es nun zu Bootproblemen kommen. Die meisten neueren Systeme verwenden die UUID der Festplatte im Bootmenü. Lesen Sie dazu auch den Abschnitt 7.3, »Allgemeine Probleme«.

**5.2.2 Festplatten-Image verkleinern**

Das direkte Verkleinern eines Images ist ebenfalls nicht möglich und eigentlich bei Verwendung eines dynamisch wachsenden Mediums auch nicht notwendig. Wenn Sie dennoch zum Beispiel ein 100 GByte großes Festplatten-Image auf 50 GByte verkleinern wollen, können Sie dazu wie im vorherigen Abschnitt gezeigt vorgehen.

Bei dynamischen Festplatten-Images gibt es jedoch die Möglichkeit, dies zu komprimieren. Wenn es auf eine bestimmte Größe gewachsen ist und Sie Dateien entfernt haben, können Sie es mit VBoxManage wieder komprimieren. Vorab müssen diese Dateien jedoch wirklich entfernt werden. Wie Sie sicher wissen, werden beim Löschen mit einem Dateimanager diese nur aus dem Inhaltsverzeichnis des Dateisystems entfernt und sind physikalisch noch vorhanden. Sie müssen nun mit entsprechenden Tools diese gelöschten Daten mit Nullen überschreiben. Für Windows eignet sich dazu beispielsweise das Programm *sdelete* von Microsoft (früher von Sysinternals) und für Linux empfehle ich *zerofree*. Anschließend können Sie das Festplatten-Image verkleinern. Unter Windows kann ein Defragmentieren der Partition vor dem Einsatz von *sdelete* die Komprimierung noch zusätzlich verbessern.

### Windows-Gastsystem

- ▶ Im Gastsystem:  
`sdelete -c E:`
- ▶ Auf dem Hostsystem:  
`VBoxManage modifyhd /media/vms/vdi/winxp_daten.vdi --compact`

Auf einem Linux-Gastsystem stoßen Sie eventuell auf Probleme, wenn es sich bei dem Festplatten-Image um die Root-Partition handelt. Das Programm `zerofree` erfordert es, dass die entsprechende Partition schreibgeschützt gemountet ist, was mit der Root-Partition im laufenden Betrieb nicht möglich ist. Am besten starten Sie eine Live-CD und führen die folgenden Kommandos aus; ersetzen Sie `/dev/hda5` im Beispiel durch den Namen Ihrer Partition:

### Linux-Gastsystem

- ▶ Im Gastsystem:  
`sudo mount -o ro,remount /dev/hda5`  
`sudo zerofree -v /dev/hda5`
- ▶ Auf dem Hostsystem:  
`VBoxManage modifyhd E:\vms\vdi\ubuntu_daten.vdi --compact`

#### Probleme mit dem Parameter `--compact`

[+]

Mir ist aufgefallen, dass es Probleme mit dem Parameter `--compact` zu geben scheint. Ich musste `VBoxManage` gelegentlich mehrfach aufrufen, bis das Image wirklich komprimiert wurde.

### 5.2.3 Kopieren (Klonen)

Wenn ein vorhandenes Festplatten-Image mehrmals verwendet werden soll, kann dieses wegen der eindeutigen UUID nicht einfach kopiert und neu registriert werden. Stattdessen erstellen Sie die Kopie mit dem speziellen Befehl `clonehd`. Dabei wird das Image kopiert, und ihm wird eine neue UUID zugewiesen.

#### VBoxManage – klonen unter Windows

```
VBoxManage clonehd ubuntu_system\  
E:\vbox\vdi\ubuntu2_system --remember
```

Der zusätzliche Parameter `--remember` sorgt dafür, dass das neue Festplatten-Image nach dem Klonen auch registriert wird.

Wie auch bei den Beispielen aus dem vorherigen Abschnitt, kann es hier unter Linux-Gastsystemen wegen inkorrektter UUIDs zu Bootproblemen kommen. Lesen Sie dazu den Abschnitt 7.3, »Allgemeine Probleme«.

### 5.2.4 Einbinden von fremden Festplatten-Images

Ich hatte bereits angesprochen, dass VirtualBox auch Festplatten-Images von VMware oder Virtual PC verwenden kann. Sie können diese wie VDI-Dateien einbinden und einer virtuellen Maschine zuordnen. Das Gastsystem kann dann auf die darauf befindlichen Daten zugreifen. Weiterhin können fertige virtuelle Maschinen von diesen Systemen eingebunden und verwendet werden. Letzteres ist jedoch nur unter Vorbehalt möglich, da das Gastsystem der Maschine auch VirtualBox-kompatibel sein muss (siehe auch Kapitel 6, »Migration und Import«).

Um Festplatten-Images eines Fremdsystems zu verwenden, binden Sie diese ganz einfach mit dem Manager für virtuelle Medien oder dem Programm VBoxManage ein. Früher gab es bei diesem Vorgehen noch Einschränkungen, inzwischen versteht VirtualBox jedoch die beiden Fremdformate. Dennoch neige ich dazu, diese nur zu Tests oder zum Migrieren von Systemen einzubinden. Wenn ich sie dauerhaft verwenden will, kopiere ich die Daten auf ein Image vom Typ VDI oder konvertiere sie (siehe nächster Abschnitt). So bin ich auf der sicheren Seite, falls es in Zukunft wieder neue Funktionen gibt, die nur den VDIs zur Verfügung stehen.

### 5.2.5 Festplatten-Images konvertieren

VBoxManage bietet Ihnen die Möglichkeit, Festplatten-Images verschiedener Formate umzuwandeln. Der Befehl `internalcommands converthd` konvertiert VDI-, VMDK- und VHD-Images – und zwar in beide Richtungen.

#### VBoxManage – Images konvertieren

##### ► VMDK in VDI:

```
VBoxManage internalcommands converthd\  
-srcformat VMDK -dstformat VDI\  
/media/cdrom/WinXP.vmdk\  
/media/virtuelle Maschinen/VDI/WinXP.vdi
```

##### ► VDI in VMDK:

```
VBoxManage internalcommands converthd\  
-srcformat VDI -dstformat VMDK\  
E:\VDI\linux_system.vdi\  
/media/VDI/linux_system.vmdk
```

Die Parameter `-srcformat` und `-dstformat` bestimmen dabei das Quell- und Zielformat.

## 5.3 Arbeiten mit Sicherungspunkten (Snapshots)

Bisher haben Sie nur das einfache Arbeiten mit den Sicherungspunkten kennen gelernt. In Abschnitt 4.5.2, »Sicherungspunkte«, wurde ein Sicherungspunkt erstellt, und danach konnten die Änderungen wieder rückgängig gemacht werden. VirtualBox bietet Ihnen aber die Möglichkeit, mehr als nur einen Sicherungspunkt anzulegen und zu verwenden. Dadurch können Sie schrittweise mehrere Stände festhalten und bei Bedarf wieder zu diesen zurückkehren.

Dabei gibt es jedoch ein paar Regeln, die Sie beachten sollten:

- ▶ Sicherungspunkte können nur *fortlaufend* erstellt werden.
- ▶ Sie können nur vom Punkt *AktuellerZustand* aus einen Sicherungspunkt erstellen. (Anwender von VMware Workstation werden es auch anders kennen.)
- ▶ Sie können immer nur zum letzten Sicherungspunkt zurückkehren.
- ▶ Sicherungspunkte können Sie beliebig verwerfen.
- ▶ Festplatten-Images vom Typ *write-through* sind von Sicherungspunkten ausgeschlossen, und ihre Änderungen bleiben immer erhalten.

Nach dem Erstellen eines Sicherungspunkts wird nichts mehr in das Festplatten-Image geschrieben. VirtualBox legt ein neues Image im für die Sicherungspunkte vorgesehenen Ordner an. Der Dateiname besteht aus der UUID und der Endung *.vdi*. Alle Änderungen und neue Dateien werden nun in diese Datei geschrieben.

### 5.3.1 Sicherungspunkte erstellen

Legen Sie für das nächste Beispiel mehrere Sicherungspunkte von einer Maschine an, und ändern Sie bei jeder Variante eine Einstellung. Ich habe, wie in Abbildung 5.9 zu sehen, drei Sicherungspunkte erstellt. Der Ausgangspunkt hatte NAT als Netzwerkeinstellung. Nach dem ersten Sicherungspunkt habe ich die Konfiguration auf INTERNES NETZWERK geändert und dabei *Netzwerk 1*, dann *Netzwerk 2* und zum Schluss (AKTUELLER ZUSTAND) *Netzwerk 3* als Namen verwendet.

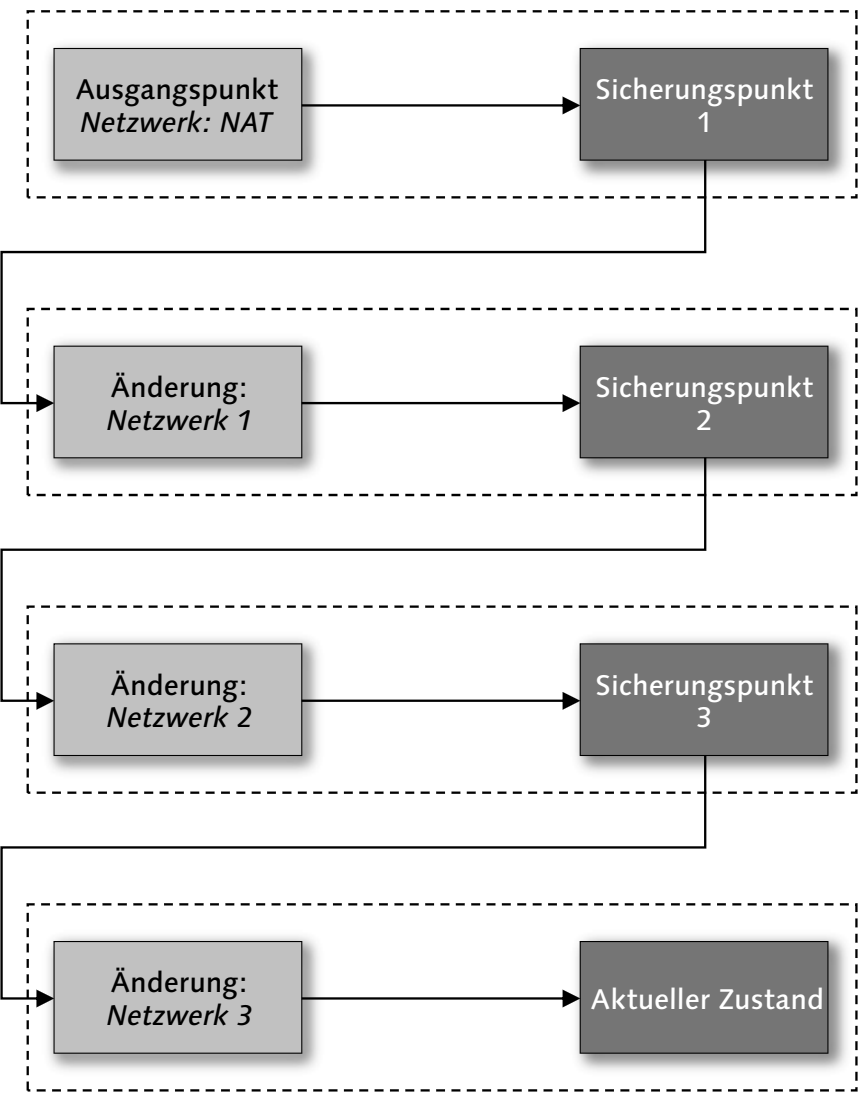


Abbildung 5.9 Änderungshistorie der Sicherungspunkte

In Abbildung 5.10 sind die drei fortlaufenden Sicherungspunkte zu sehen. Je nachdem, ob AKTUELLER ZUSTAND oder einer der Sicherungspunkte markiert ist, ändert sich die darüber befindliche Symbolleiste. Ein Rechtsklick auf einen der Punkte öffnet ein Kontextmenü mit den gleichen Funktionen wie sie die Symbolleiste bietet – nur mit einer zusätzlichen Beschriftung.



Abbildung 5.10 Drei Sicherungspunkte der Testmaschine

[+]

**Hinweis zu eingebundenen CD/DVD/Floppy-Abbildern**

Ich achte immer darauf, dass bei der Erstellung eines Sicherungspunkts keine CD/DVD-Abbilder oder Disketten-Images in der virtuellen Maschine registriert sind. Diese müssen dann nämlich immer vorhanden sein, um zu diesem Sicherungspunkt zurückkehren zu können (siehe Abschnitt 7.3, »Allgemeine Probleme«).



### VBoxManage – Sicherungspunkt erstellen

Alle Systeme:

```
VBoxManage snapshot WinXP take "Sicherungspunkt 1"
```

#### 5.3.2 Informationen abrufen

In Abbildung 5.11 sehen Sie den Dialog DETAILS VON SICHERUNGSPUNKT 1. Die Konfiguration des Netzwerkadapters ist markiert und steht auf NAT. Sie erkennen übrigens auch, dass der Button zum Erstellen eines neuen Snapshots deaktiviert ist. Der Grund dafür ist, dass von einem Sicherungspunkt kein neuer erstellt werden kann, sondern nur vom Punkt AKTUELLER ZUSTAND aus.

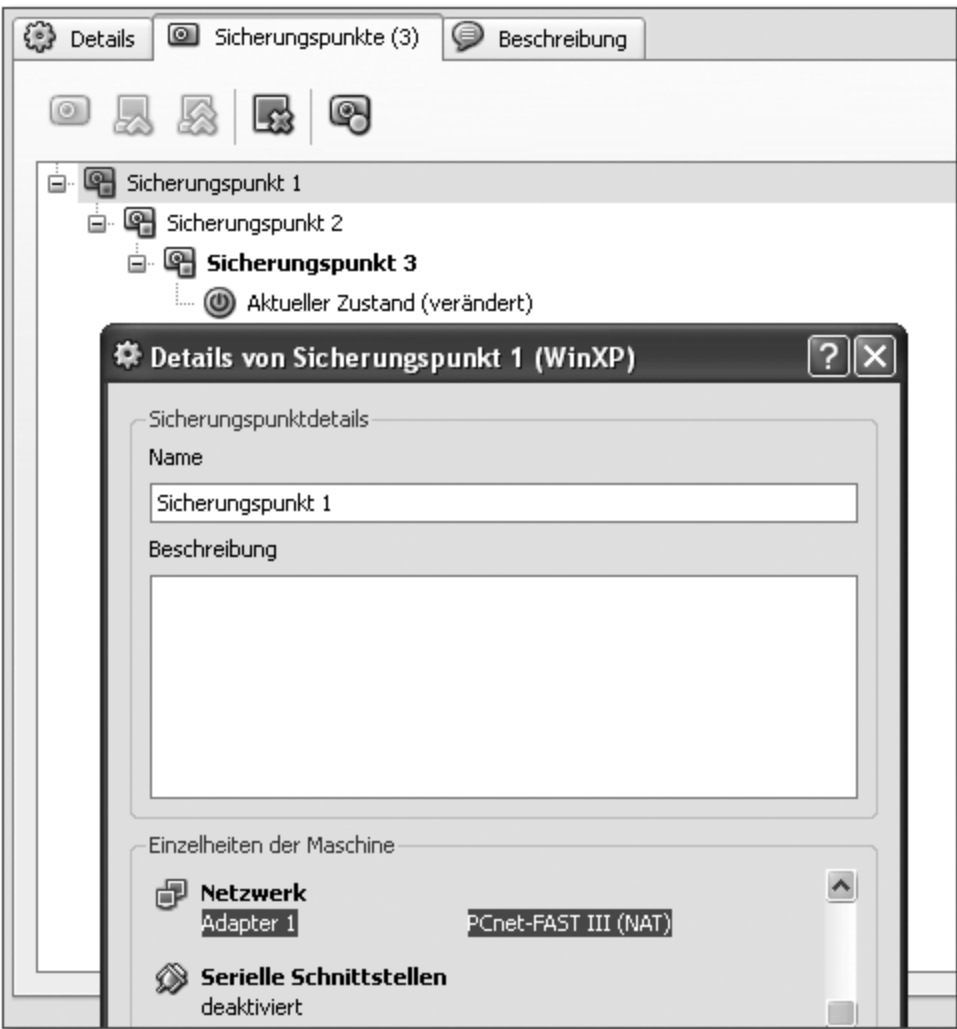


Abbildung 5.11 Der Dialog »Details von Sicherungspunkt 1«

Sie können nun noch die Details der anderen Sicherungspunkte prüfen, indem Sie aus dem Kontextmenü den Eintrag EINZELHEITEN ZEIGEN wählen.

### VBoxManage – Einzelheiten zeigen

Alle Systeme:

```
VBoxManage snapshot WinXP showvminfo "Sicherungspunkt 1"
```

5.3.3 Sicherungspunkt verwerfen

Nun können Sie das Detailfenster schließen und SICHERUNGSPUNKT VERWERFEN wählen; vergewissern Sie sich, dass SICHERUNGSPUNKT 1 markiert ist. Dadurch bleiben die Änderungen, die nach Sicherungspunkt 1 und vor Sicherungspunkt 2 vorgenommen wurden, erhalten. Das bedeutet, dass die Einstellung NAT nicht mehr wiederhergestellt werden kann.

VBoxManage – Sicherungspunkt verwerfen

Alle Systeme:

```
VBoxManage snapshot WinXP discard "Sicherungspunkt 1"
```

[!]

»Sicherungspunkt verwerfen«

»Sicherungspunkt verwerfen« bedeutet, dass VirtualBox alle Änderungen übernimmt, die nach diesem Punkt und vor dem nächsten Snapshot vorgenommen wurden. Danach wird der entsprechende Sicherungspunkt verworfen (also entfernt).

5.3.4 Zurückkehren zum letzten Sicherungspunkt

Als Nächstes wollen wir die letzten Änderungen wieder verwerfen – also den vorherigen Zustand wiederherstellen. Zur Erinnerung:

Vor Sicherungspunkt 2 hatten wir die Konfiguration INTERNES NETZWERK mit dem Namen *Netzwerk 1* gewählt,  
vor Sicherungspunkt 3 den Namen *Netzwerk 2*,  
und unser aktueller Zustand ist *Netzwerk 3*.

Zunächst verwerfen wir den aktuellen Zustand, um zu der Konfiguration *Netzwerk 2* zurückzukehren. Markieren Sie dazu den Punkt AKTUELLER ZUSTAND, und wählen Sie ZURÜCKKEHREN ZUM LETZTEN SICHERUNGSPUNKT im Kontextmenü. Sie erkennen in Abbildung 5.12, dass die Änderungen verworfen wurden und wir wieder den Zustand direkt nach Sicherungspunkt 3 haben (hinter AKTUELLER ZUSTAND befindet sich kein Hinweis VERÄNDERT).



Abbildung 5.12 Der aktuelle Zustand wurde verworfen.

Zur Überprüfung schauen wir nochmal in die Konfiguration des Netzwerks. Dieses muss nun wieder *Netzwerk 2* als Namen haben (Abbildung 5.13).

Abbildung 5.13 Die Konfiguration »Netzwerk 2«

**VBoxManage – zurückkehren zum letzten Sicherungspunkt**

Alle Systeme:

```
VBoxManage snapshot WinXP discardcurrent --state
```

**5.3.5 Aktuellen Sicherungspunkt und Zustand verwerfen**

Nun schlagen wir zwei Fliegen mit einer Klappe: Wir verwerfen einen Sicherungspunkt und den aktuellen Zustand auf einmal. Dadurch werden die Änderungen nach Sicherungspunkt 3 wieder verworfen, und der Sicherungspunkt 3 wird entfernt.

Markieren Sie wieder den Punkt AKTUELLER ZUSTAND, und wählen Sie im Kontextmenü AKTUELLEN SICHERUNGSPUNKT UND ZUSTAND VERWERFEN.



Abbildung 5.14 Der Sicherungspunkt 3 und der Zustand wurden verworfen.

Nun ist nur noch der Sicherungspunkt 2 vorhanden, und die Konfiguration wurde auf *Netzwerk 1* (Abbildung 5.15) zurückgesetzt.



Abbildung 5.15 Die Einstellung »Netzwerk 1«

**VBoxManage – aktuellen Sicherungspunkt und Zustand verwerfen**

Alle Systeme:

```
VBoxManage snapshot WinXP discardcurrent --all
```



**Verwerfen des aktuellen Sicherungspunktes**

Wenn Sie den aktuellen Sicherungspunkt und Zustand verwerfen, löscht das den Sicherungspunkt und verwirft alle Änderungen, die nach diesem Sicherungspunkt vorgenommen wurden.

**5.4 Arbeiten mit dem Ruhezustand**

Viele von Ihnen verwenden sicherlich die Möglichkeit, ein Betriebssystem in den sogenannten Ruhezustand zu versetzen. Dabei wird der Speicherinhalt auf die Festplatte geschrieben und der PC ausgeschaltet. Diese Funktion kommt vorwiegend bei Notebooks zum Einsatz – praktisch, da geöffnete Anwendungen nach dem Wiedereinschalten einfach weiter verwendet werden können. VirtualBox bietet ein ähnliches Feature an. Im Praxisbeispiel (vgl. Abschnitt 4.3.4, »Ruhezustand«) haben wir diese Funktion bereits eingesetzt. Einzige Voraussetzung dafür ist, dass die Gasterweiterungen auf dem Gastsystem installiert sind. Ohne sie würde die Uhrzeit und das Datum beim nächsten Start nicht aktualisiert. Im Gegensatz zu der Ruhezustandfunktion eines Betriebssystems wird hier der Ruhezustand von VirtualBox gesteuert. Dadurch kann es auch bei Gastsystemen eingesetzt werden, die diese Funktion selbst nicht anbieten.

**5.4.1 Ruhezustand aktivieren**

Um den Ruhezustand zu nutzen, beenden Sie die virtuelle Maschine. Je nachdem, wie Sie sie gestartet haben, kann dies über die grafische Oberfläche oder das Programm VBoxManage erfolgen. In der grafischen Oberfläche wählen Sie im Menü MASCHINE • SCHLIESSEN oder betätigen die Host-Taste + . Danach öffnet sich ein Dialogfenster, in dem Sie den Punkt DEN ZUSTAND DER VIRTUELLEN MASCHINE SPEICHERN auswählen (siehe Abbildung 5.16).



**Abbildung 5.16** Den Zustand der Maschine speichern

### VBoxManage – Zustand der Maschine speichern

Alle Systeme:

```
VBoxManage controlvm Ubuntu_9 savestate
```

Welche Methode Sie auch verwenden, VirtualBox speichert nun den aktuellen Zustand der Maschine in einer Statusdatei im Ordner, der auch für die Sicherungspunkte verwendet wird. Der Dateiname besteht aus einer UUID und der Endung `.sav`. Die Größe der Datei ist dabei abhängig vom aktuellen Speicherbedarf des Gastsystems und kann daher stark variieren.

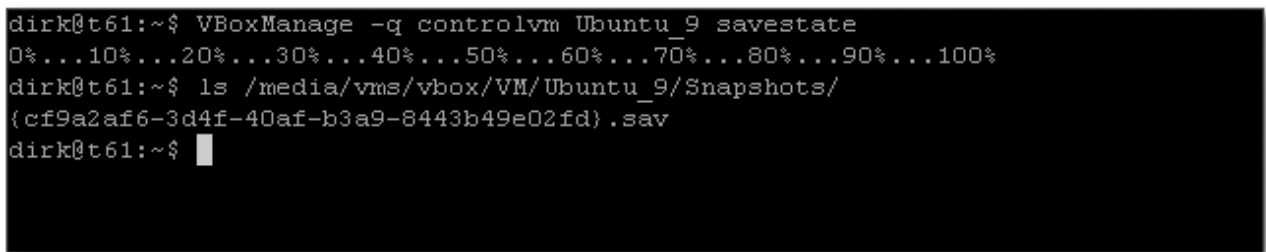


Abbildung 5.17 VBoxManage und die Statusdatei

Speichern des Zustands der Maschine

Bevor Sie die Funktion zum Speichern verwenden, sollten Sie auf dem Gastsystem die Gasterweiterungen installieren.

[!]

### 5.4.2 Virtuelle Maschine wieder starten

Um die virtuelle Maschine aus dem Ruhezustand »aufzuwecken«, sind keine besonderen Maßnahmen notwendig. Sie müssen diese nur mit der gewünschten Methode »einschalten«. Ob eine Maschine sich gerade im Ruhezustand befindet, erkennen Sie in der grafischen Oberfläche an dem Zusatz **GESICHERT**, der unter dem Namen steht.



Abbildung 5.18 Die virtuelle Maschine Ubuntu\_9 ist »gesichert«.

VBoxManage – Status abfragen

Alle Systeme:

```
VBoxManage showvminfo Ubuntu_9
```

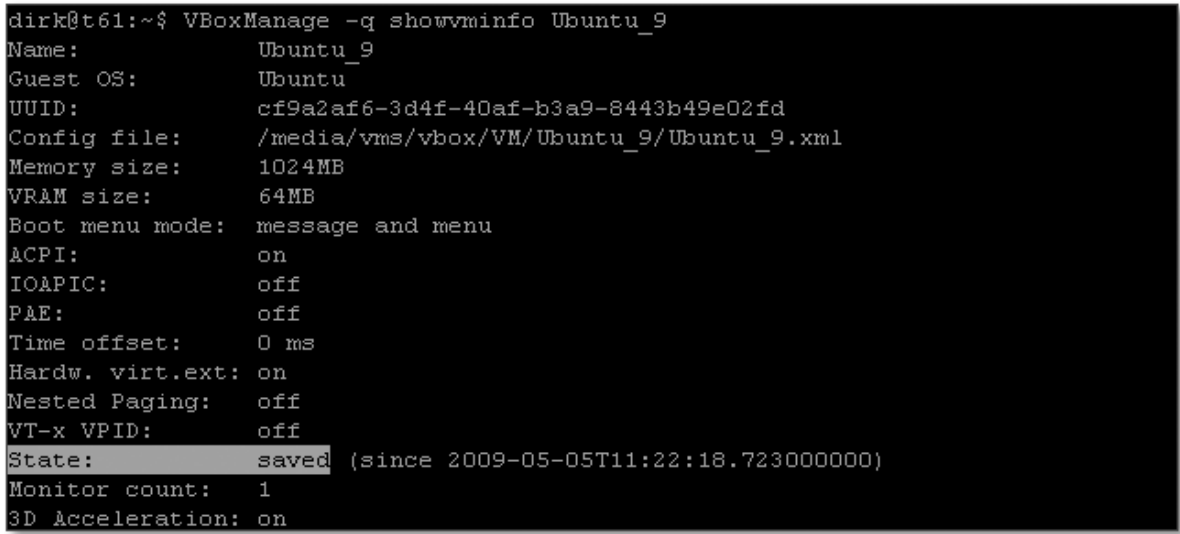


Abbildung 5.19 Der Status der Maschine ist »saved«, das heißt gesichert.

Nach dem Einschalten wird nun der Speicher wieder mit den Daten der Statusdatei gefüllt, die Datei entfernt und die Maschine gestartet. Die meisten Gastssysteme kommen mit diesem Aufwecken klar. Wenn Ihr System keine Gasterweiterungen installiert hat, wird unter anderem jedoch die Systemzeit nicht aktualisiert.



Maschinen ohne Gasterweiterungen: Zeitfunktionen nach Ruhezustand

Ich habe den Ruhezustand auch mit Systemen ohne Gasterweiterung getestet. Grundsätzlich funktioniert sie auch dort, einzig mit der Einschränkung, dass die Uhrzeit und das Datum beim erneuten Einschalten nicht mehr korrekt sind. Ich kann jedoch nicht garantieren, dass es durch Cache oder Ähnliches nicht zu Datenverlusten kommt.

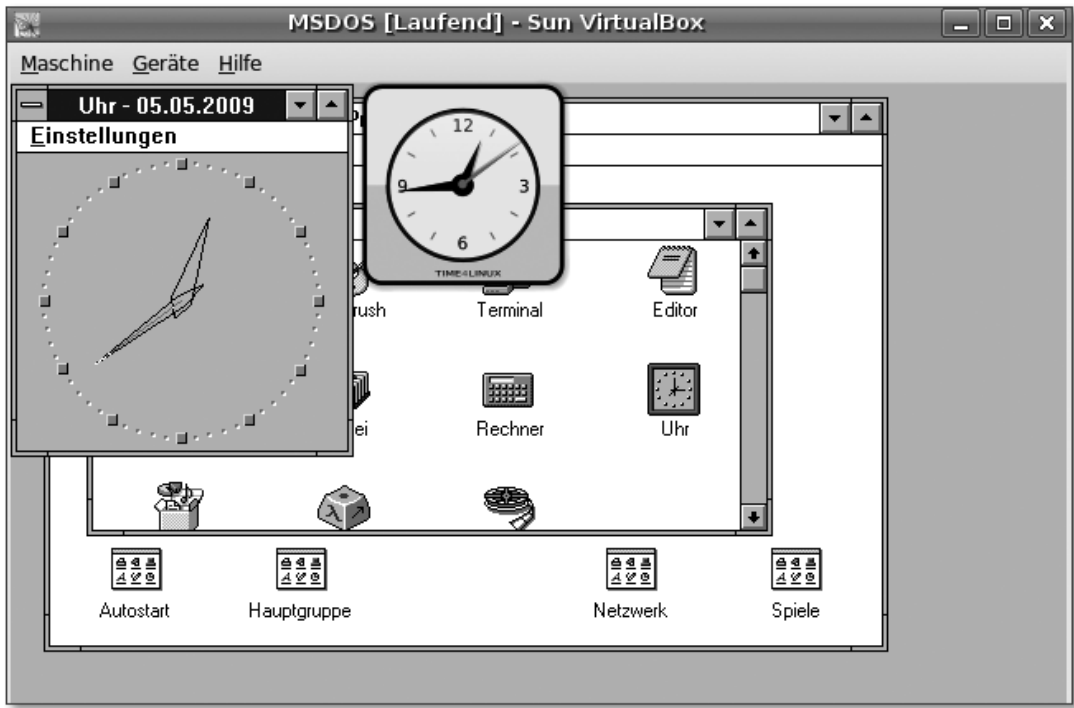


Abbildung 5.20 Fünf Minuten Ruhezustand ohne Gasterweiterungen, und die Uhr ist aus dem Takt.

Auch bei anderen (vorwiegend älteren) Anwendungen kann es zu Problemen kommen. Betroffen sind vor allem Netzwerk- oder Datenbankzugriffe. Nutzer des Ruhezustands bei Notebooks werden diese Probleme kennen.

Auswirkung fehlender Gasterweiterungen

Ohne Gasterweiterungen werden das Datum und die Uhrzeit nicht aktualisiert. Bei Netzwerk- oder Datenbankanwendungen kann es zu Problemen kommen.

[!]

### 5.4.3 Status verwerfen

Gelegentlich kann es vorkommen, dass die Maschine sich nicht mehr aus dem Ruhezustand einschalten lässt. Oder Sie haben eine Live-CD in den Ruhezustand versetzt und benötigen diesen nicht mehr. In solchen Fällen kann der Zustand verworfen werden. Wählen Sie dazu den Punkt MASCHINE • VERWERFEN oder das Symbol mit dem Pfeil nach unten aus der Leiste (Abbildung 5.21).

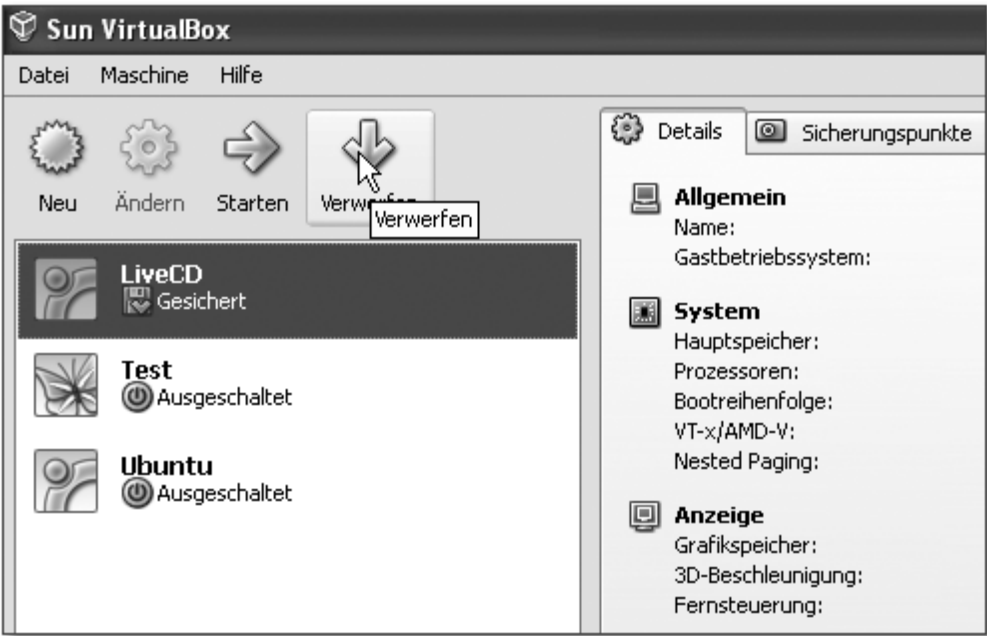


Abbildung 5.21 Den Ruhezustand verwerfen

Hierbei wird der Status der Maschine in AUSGESCHALTET versetzt, und VirtualBox löscht die vorhandene Sicherungsdatei.

#### VBoxManage – Status verwerfen

Alle Systeme:

VBoxManage **discardstate** LiveCD

Vorsicht beim Gebrauch der Funktion »Status verwerfen«

Bei installierten Gastsystemen sollten Sie diese Funktion nur verwenden, wenn nichts anderes mehr möglich ist. Es kann hierdurch zu Datenverlust oder schlimmstenfalls zu einem defekten System kommen.

[+]

#### 5.4.4 Status übernehmen

Eine weitere Möglichkeit bietet der VBoxManage-Befehl `adoptstate`. Mit ihm können Sie einen vorhandenen Status übernehmen und dadurch einen Ruhezustand sichern und mehrmals verwenden. Dieser Befehl steht nicht in der grafischen Oberfläche zur Verfügung, sondern funktioniert nur mit VBoxManage auf der Kommandozeile.

Zunächst sichern Sie eine vorhandene Statusdatei mit einem Dateimanager. Dann weisen Sie diese Kopie einer virtuellen Maschine zu. Denken Sie dabei daran, dass die Datei beim Starten entfernt wird. Wenn sie mehrmals verwendet werden soll, erstellen Sie eine zusätzliche Sicherung.

#### VBoxManage – Status übernehmen

► Linux:

```
VBoxManage adoptstate LiveCD /media/vbox/save/LiveCD_01.sav
```

► Windows:

```
VBoxManage adoptstate LiveCD D:\save\LiveCD_01.sav
```

Beim Anwenden dieser Funktion gibt es jedoch ein paar Punkte zu beachten. So müssen die Statusdateien immer den korrekten virtuellen Maschinen zugeordnet werden. Wenn Sie den Speicherstatus einer Maschine mit Windows XP als Gastsystem einem Linux-System zuordnen, kommt es logischerweise zu Fehlern.



**Abbildung 5.22** VirtualBox warnt vor falschen Zuordnungen.

Weiterhin kann es zu Problemen kommen, wenn sich die Daten der Festplatte ändern und der Zustand einer Statusdatei diese jedoch benötigt. Daher sollten Sie diese Funktion mit Bedacht verwenden. Im Fall eines Fehlers müssen Sie den Status, wie im vorherigen Abschnitt beschrieben, wieder verwerfen.



## 5.5 Gemeinsame Ordner

Wie die meisten Konkurrenzprodukte bietet auch VirtualBox die Funktion GEMEINSAME ORDNER (*shared folders*) an. Der Zugriff auf diese Ordner erfolgt in den Gastsystemen ähnlich wie Zugriffe auf Netzwerkfreigaben. Abschnitt 4.4.6, »Gemeinsame Ordner«, hat die Funktion bereits kurz vorgestellt, allerdings nur mit Windows 7 als Gastsystem. Im Folgenden erläutere ich die Erstellung und Verwendung nun detaillierter und auch für andere Gastsysteme.

Installation der Gasterweiterungen

Vor der Verwendung der gemeinsamen Ordner sollten Sie die Gasterweiterungen installieren.

[+]

### 5.5.1 Gemeinsame Ordner hinzufügen

Die Einrichtung kann, wie schon im Praxisbeispiel beschrieben, in der grafischen Oberfläche oder mit VBoxManage erfolgen. Mit der ersten Methode öffnen Sie die Konfiguration der virtuellen Maschine und aktivieren das entsprechende Register. Dort können Ordner hinzugefügt, bearbeitet und entfernt werden. Dabei können Sie zusätzlich festlegen, ob die Maschine Voll- oder nur Lesezugriff auf den Ordner erhält.

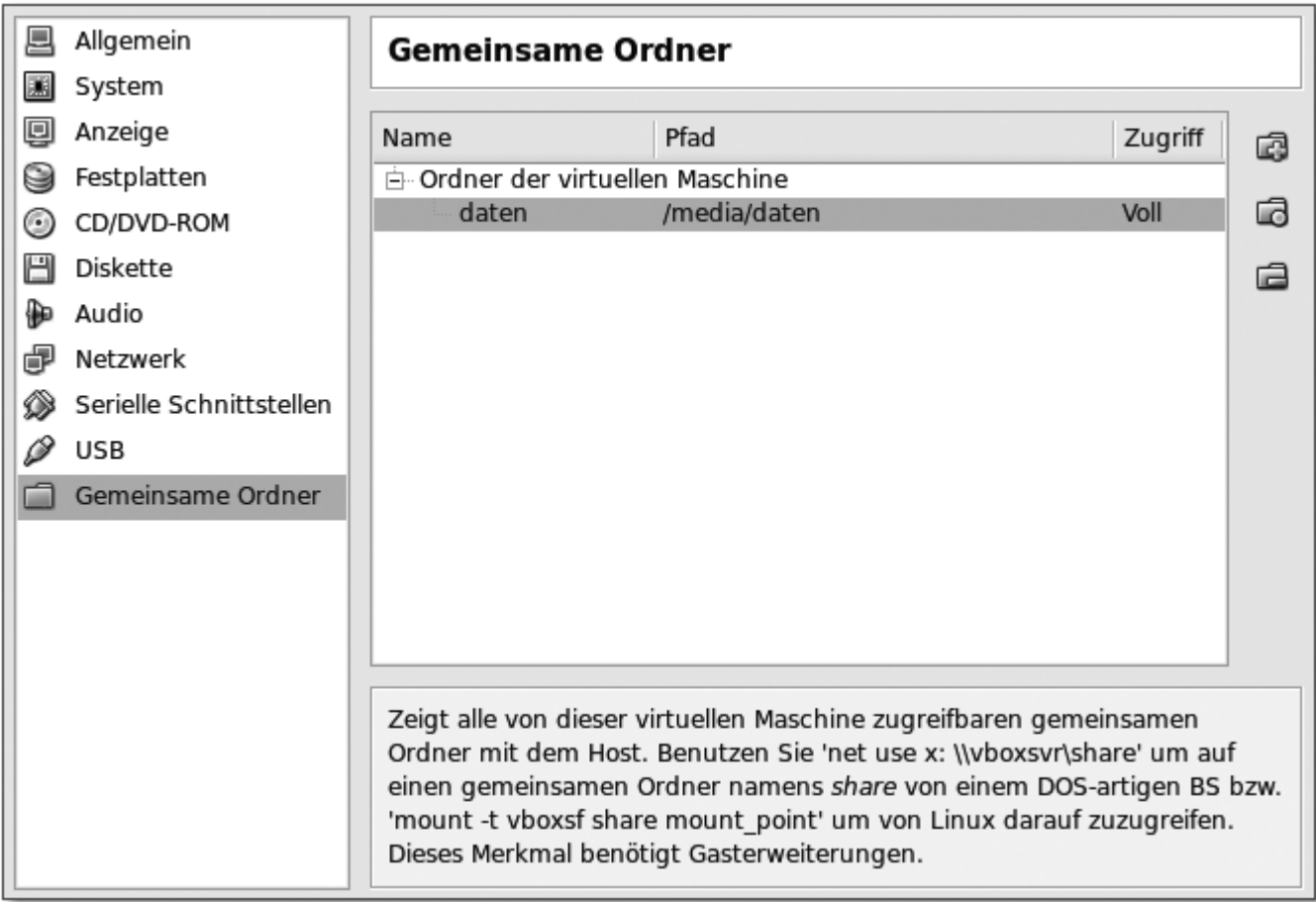


Abbildung 5.23 Gemeinsame Ordner zwischen Host- und Gastsystem

Eine weitere Möglichkeit ist, einen Ordner zur Laufzeit hinzuzufügen. Die Zuordnung kann über das Menü GERÄTE • GEMEINSAME ORDNER oder mit VBoxManage erfolgen. Ein Ordner kann auch »nicht permanent« (*transient*) zugeordnet werden. Das bedeutet, dass er nur zur Laufzeit zur Verfügung steht und nach dem Beenden wieder aus der Konfiguration entfernt wird.



Abbildung 5.24 Einen Ordner während der Laufzeit hinzufügen

Ordner, die nicht permanent hinzugefügt wurden, erscheinen dann im Bereich TRANSIENTE ORDNER.

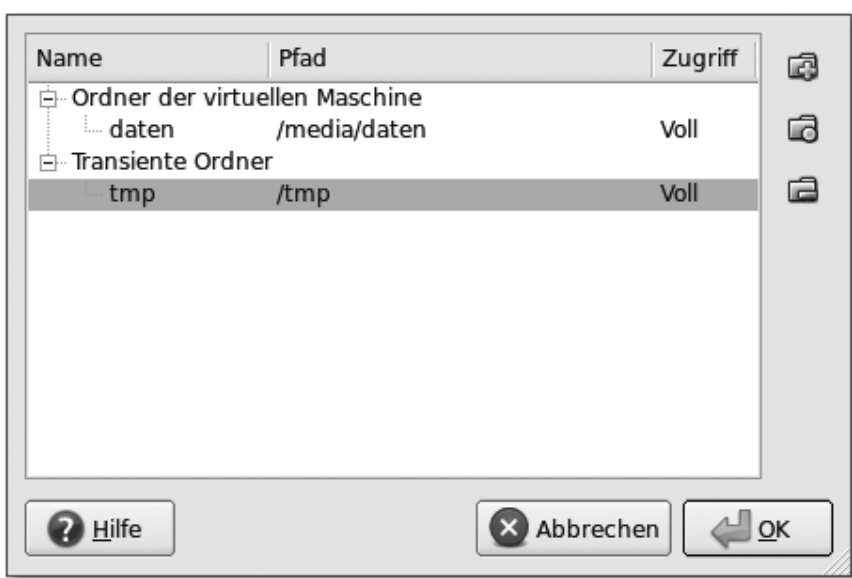


Abbildung 5.25 Transiente Ordner

**VBoxManage – gemeinsame Ordner erstellen oder entfernen**

- Linux:  
Ordner hinzufügen:  
VBoxManage sharedfolder **add** "Windows 7"\  
--name daten --hostpath /media/daten

Ordner schreibgeschützt hinzufügen:

```
VBoxManage sharedfolder add Ubuntu --name daten\
  --hostpath /media/daten --readonly
```

Ordner entfernen:

```
VBoxManage sharedfolder remove Ubuntu --name daten
```

► **Windows:**

Ordner hinzufügen:

```
VBoxManage sharedfolder add "Windows 7" --name daten\
  --hostpath "D:\Gemeinsamer Ordner"
```

oder transient:

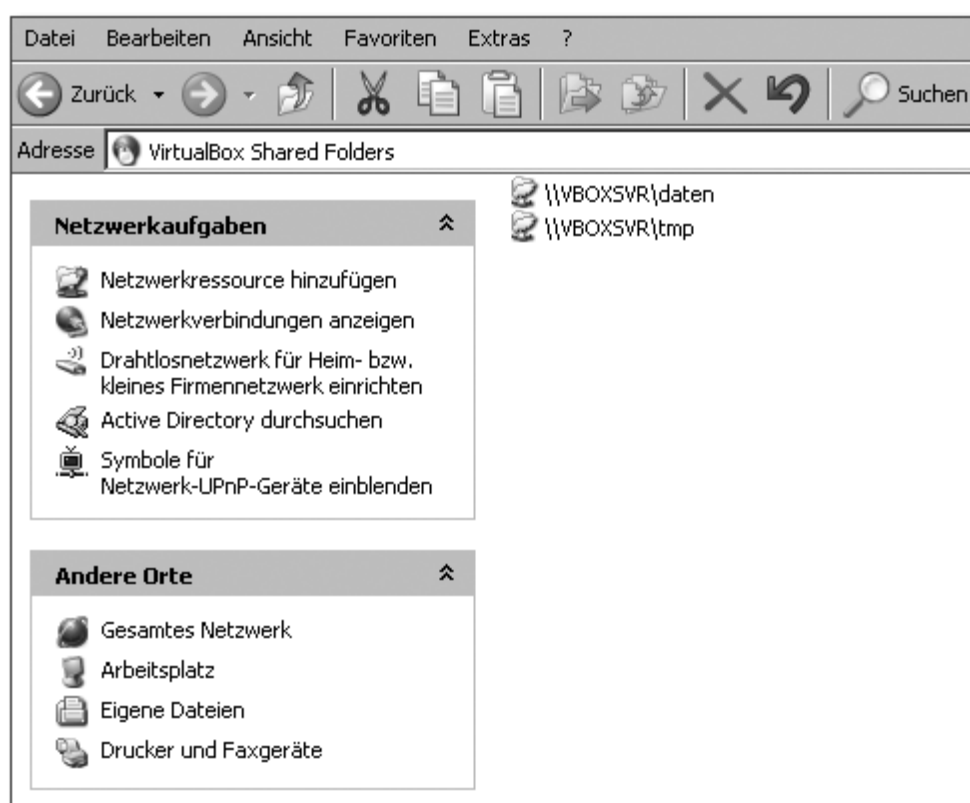
```
VBoxManage sharedfolder add Ubuntu --name tmp\
  --hostpath E:\Temp --transient
```

## 5.5.2 Zugriff auf einen gemeinsamen Ordner

Im Gastsystem der laufenden Maschine können Sie nun auf den Ordner zugreifen. Die Zugriffsart ist dabei von dem jeweiligen System abhängig.

### Windows

Unter Windows-Systemen kann der Zugriff auf zwei Arten erfolgen. Die erste Möglichkeit ist ein einfacher Zugriff über die Netzwerkumgebung.



**Abbildung 5.26** In der Netzwerkumgebung finden Sie gemeinsam genutzte Ordner.

Öffnen Sie diese und schauen Sie, je nach Windows-Version, nach **GESAMTES NETZWERK • VIRTUALBOX SHARED FOLDERS**. Dort erscheinen sämtliche Ordner, die der Maschine zugeordnet wurden.

Die zweite Möglichkeit ist die Verwendung des Befehls `net`. Mit diesem Kommando können Sie in der Eingabeaufforderung oder im Dialog **START • AUSFÜHREN** Netzwerkeinstellungen vornehmen, zum Beispiel:

```
net use x: \\vboxsvr\daten /persistent:yes
```

Durch den optionalen Parameter `persistent` kann der Zugriff permanent eingerichtet werden.

### Linux

Unter Linux hängen Sie den gemeinsamen Ordner mit dem Befehl `mount` ein, um Zugriff zu erhalten:

```
mount -t vboxsf daten /mnt
```

Achten Sie darauf, dass der Mount Point `/mnt` auch vorhanden und nicht bereits in Verwendung ist.

## 5.6 Nahtloser Modus (seamless windows)

Der nahtlose Modus ist eine besondere Erweiterung von VirtualBox, die es ermöglicht, das Gastsystem einer virtuellen Maschine in das Hostsystem zu »integrieren«.

**[+]**

### Gasterweiterungen für nahtlosen Modus installieren

Um den nahtlosen Modus zu verwenden, müssen auf dem Gastsystem die Gasterweiterungen installiert sein. Außerdem gibt es Probleme, wenn die 3-D-Beschleunigung aktiviert ist (siehe Abschnitt 7.3, »Allgemeine Probleme«).

In Abbildung 5.27 sehen Sie eine virtuelle Maschine mit Windows 7 als Gastsystem. Die Maschine wurde unter Ubuntu gestartet und befindet sich im nahtlosen Modus.

Sie erkennen die Windows-7-Startleiste, den Explorer und den Windows-Rechner. Unten befindet sich eine Startleiste von Ubuntu (der Avant Window Navigator) und im Hintergrund der GNOME-Dateimanager Nautilus. Umgekehrt funktioniert das Ganze ebenfalls. In Abbildung 5.28 sehen Sie eine Ubuntu-Umgebung unter einem Windows-Hostsystem.



Abbildung 5.27 Windows 7 im nahtlosen Modus

Sie aktivieren den Modus entweder über das Menü MASCHINE • NAHTLOSER MODUS des Fensters oder über die Tastenkombination Host-Taste +  $\boxed{L}$ . Letztere sollten Sie sich merken, denn auf diese Weise können Sie den Modus auch wieder verlassen. Über das VirtualBox-Menü ist das nicht mehr möglich, da es im nahtlosen Modus nicht angezeigt wird. Achten Sie beim Verwenden der Tastenkombination darauf, dass ein Fenster des Gastsystems den Fokus besitzt. Bei Problemen mit dem nahtlosen Modus finden Sie in Abschnitt 7.3, »Allgemeine Probleme«, gegebenenfalls eine Lösung.

Voraussetzungen für den Einsatz des nahtlosen Modus

- Für den nahtlosen Modus müssen die Gasterweiterungen installiert sein.
- Sie deaktivieren den Modus mit dem Shortcut Host-Taste +  $\boxed{L}$ .
- Sie sollten die 3-D-Beschleunigung nicht zusammen mit dem nahtlosen Modus verwenden.

[!]

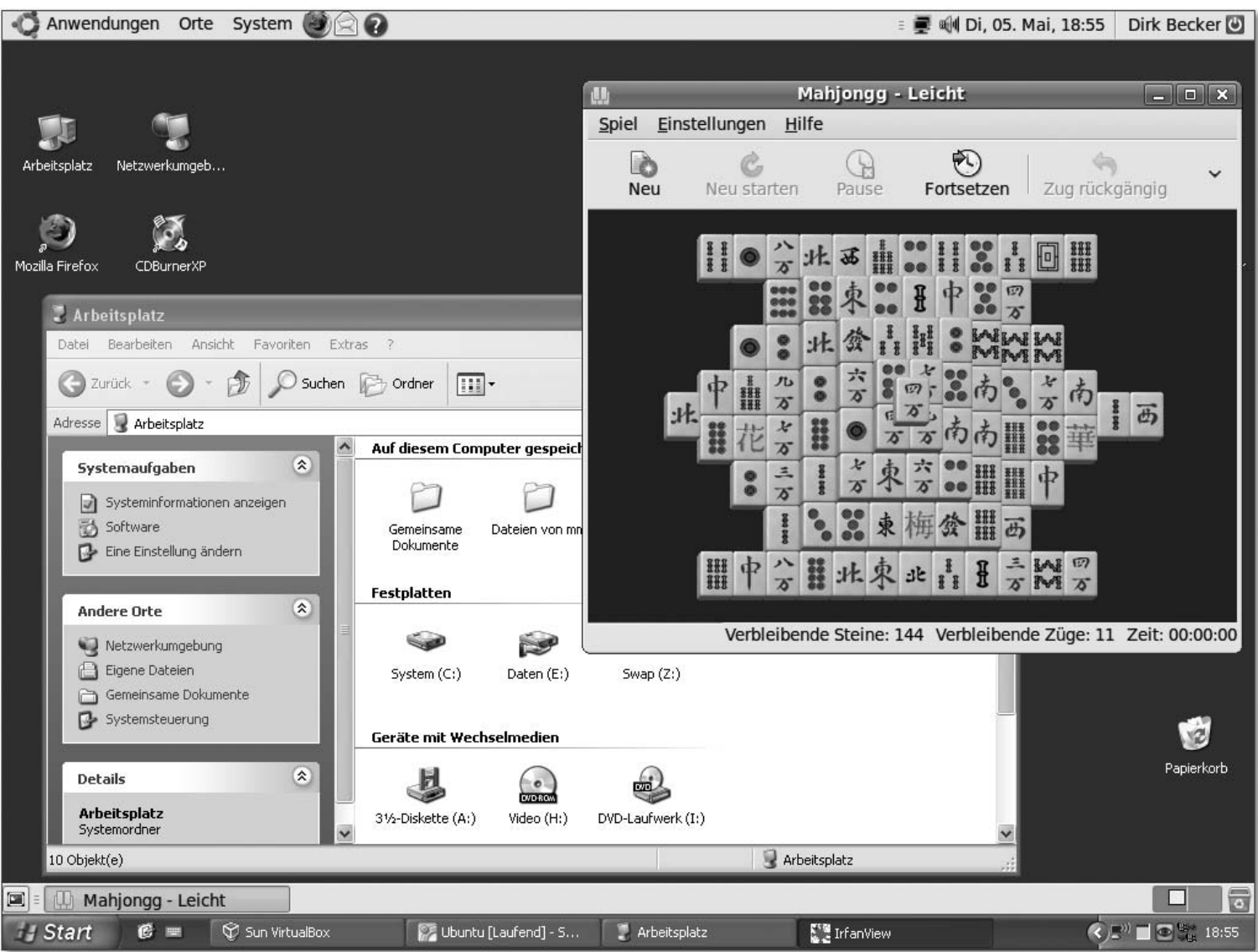


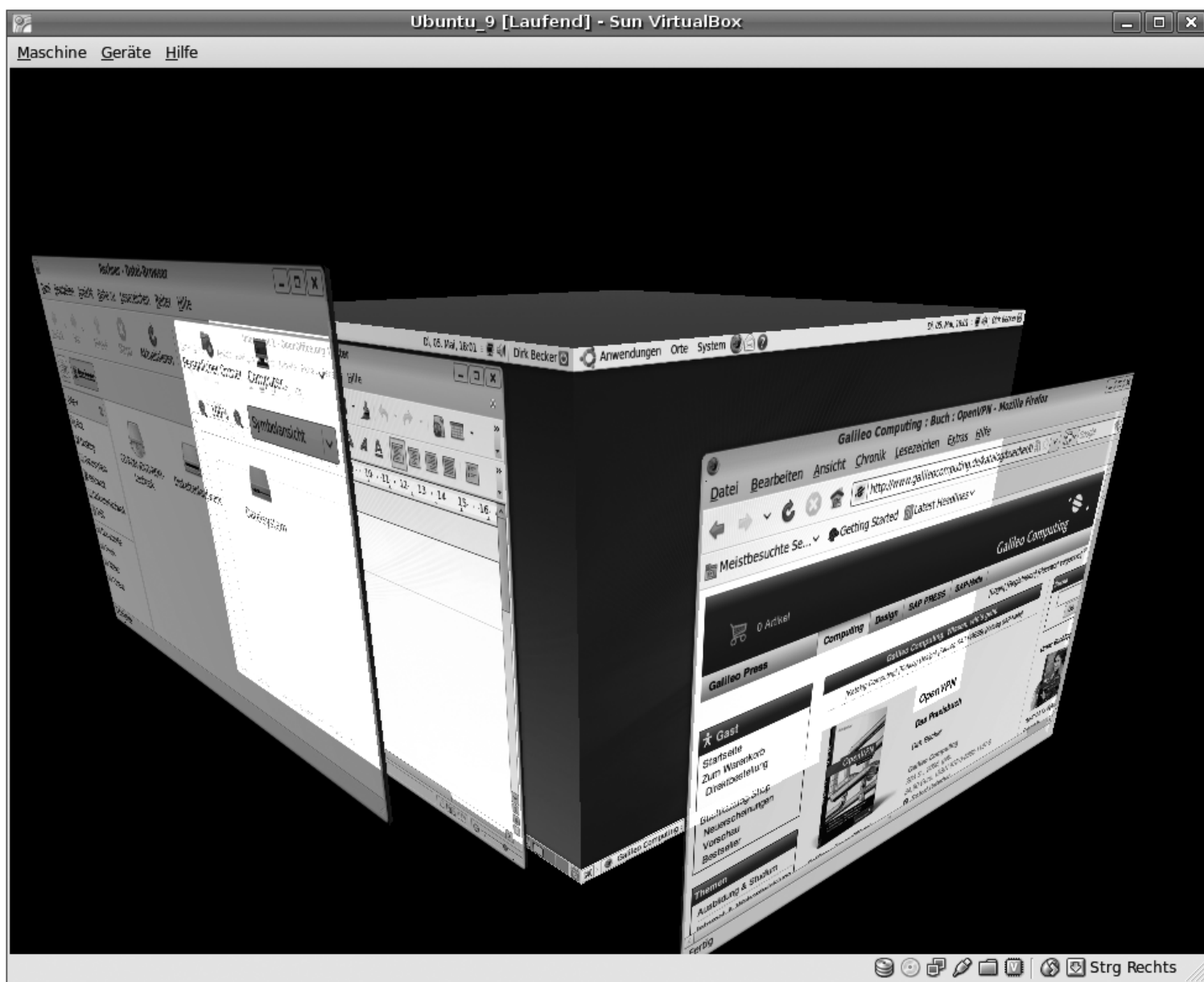
Abbildung 5.28    Ubuntu unter Windows XP

## 5.7    3-D-Beschleunigung

### 5.7.1    OpenGL 2.0

Die 3-D-Beschleunigung und die OpenGL-Funktionen haben wir bereits im Praxisbeispiel (siehe Abschnitt 4.4.7, »3-D-Beschleunigung testen«) unter dem Gast-system Windows 7 getestet. Seit Version 2.2 bietet VirtualBox den Einsatz von 3-D auch unter Linux und Solaris als Gastsystem an. Unter Linux ist dabei der Ein-satz von Composite möglich, um eine 3-D-Oberfläche zu verwenden.

Besondere Einstellungen sind hierzu nicht notwendig – aktivieren Sie einfach die 3-D-Beschleunigung und konfigurieren Sie das Linux-Gastsystem entsprechend.



**Abbildung 5.29** 3-D unter einem Linux-Gastsystem

### 5.7.2 Direct3D

Seit Version 3 können Sie für Windows-Gastsysteme nicht nur OpenGL, sondern auch Direct3D verwenden. Diese Funktion ist zurzeit jedoch noch im »Experimentierstatus«; möglicherweise ist Direct3D aber inzwischen stabil, wenn Sie diese Zeilen lesen.

Um die Aktivierung zu verdeutlichen, teste ich zunächst einmal die Konfiguration eines Windows-XP-Gastsystems ohne 3-D-Beschleunigung. Dazu verwende ich das zu DirectX gehörende Tool Dxdiag (DirectX-Diagnoseprogramm), das Sie über **START • AUSFÜHREN • DXDIAG** starten. Im Register **ANZEIGE** sehen Sie nun im unteren Fensterbereich den Hinweis, dass die Direct3D-Beschleunigung nicht Verfügbar ist (Abbildung 5.30).

Um Direct3D zu verwenden, müssen Sie die 3-D-Beschleunigung aktivieren und die Gasterweiterungen mit der Option **DIRECT3D-UNTERSTÜTZUNG** installieren (Abbildung 5.31).

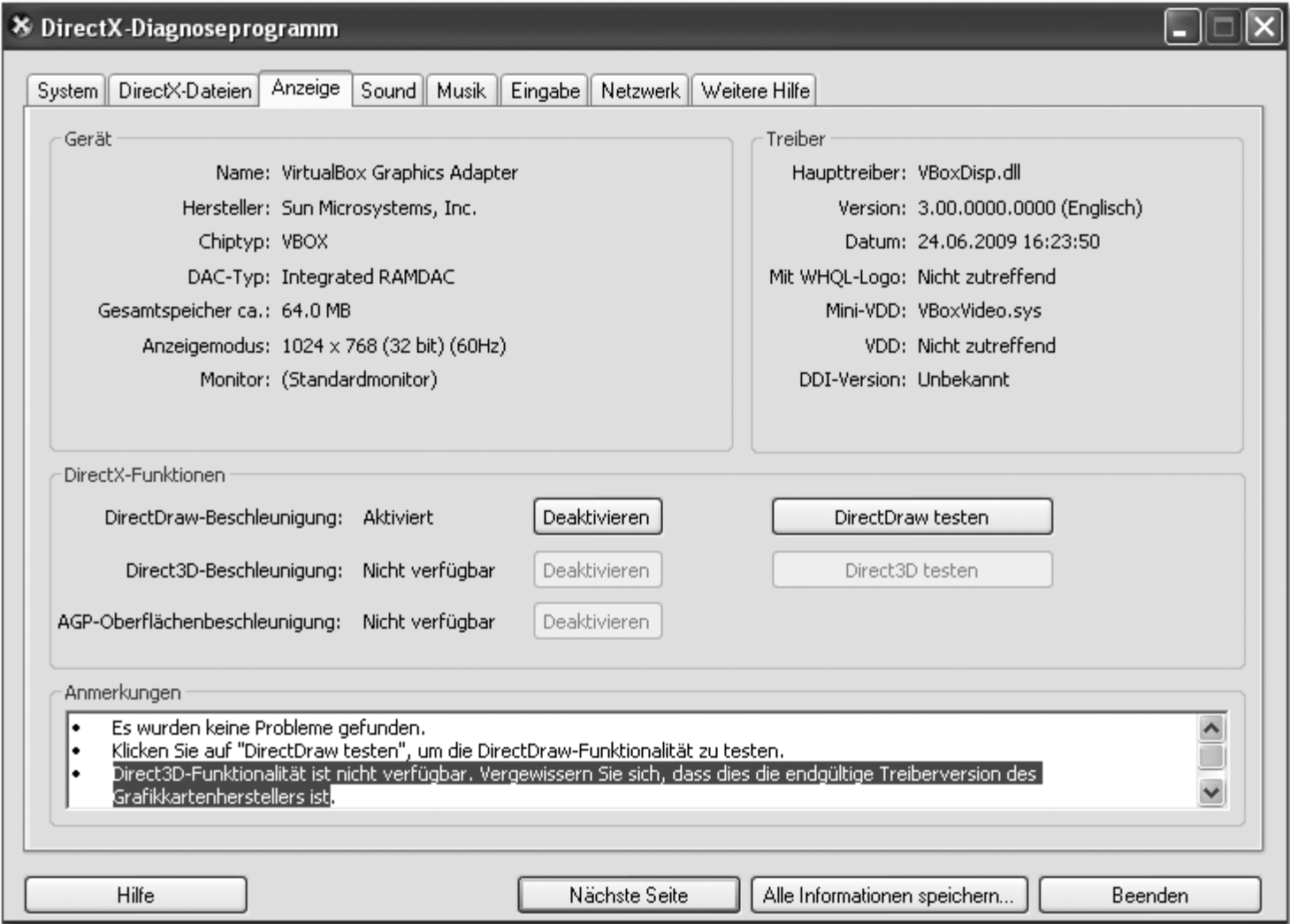


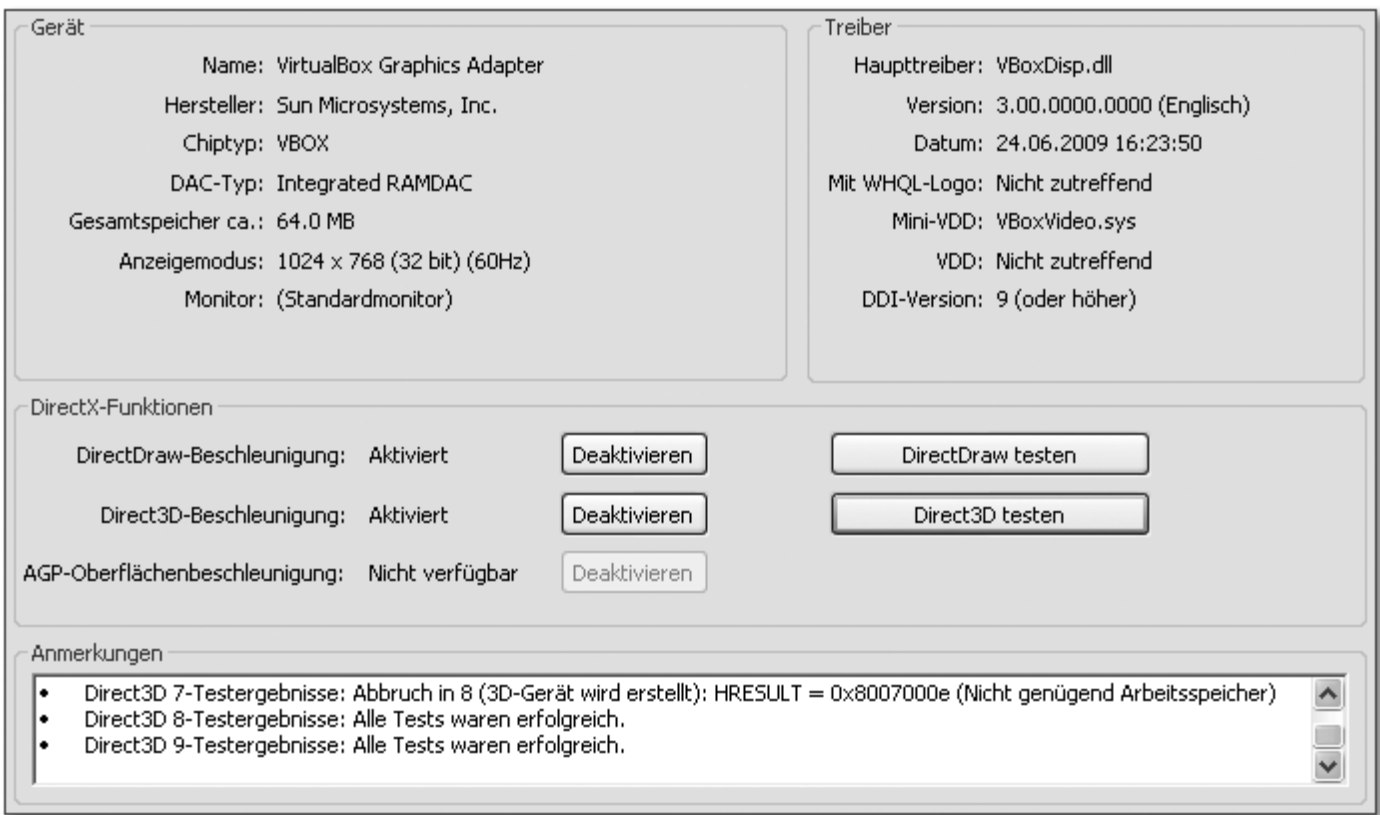
Abbildung 5.30 Die Direct3D-Beschleunigung ist nicht verfügbar



Abbildung 5.31 Die Gasterweiterungen mit Direct3D-Unterstützung installieren

Wenn Sie danach Dxdiag erneut starten und in das Register ANZEIGE wechseln, erkennen Sie, dass die Direct3D-Beschleunigung aktiviert ist.





**Abbildung 5.32** Alle Tests waren erfolgreich, und die Direct3D-Beschleunigung ist aktiviert.

Der Direct3D 7-Test meldete auf meinem Testsystem NICHT GENÜGENDE ARBEITSSPEICHER. Dies ist jedoch nicht wirklich verwunderlich, da VirtualBox nur die Versionen 8 und 9 unterstützt.

Nun können Sie selbst Anwendungen und Spiele testen, die Direct3D 8 oder 9 benötigen. Wenn diese Programme auch mit OpenGL arbeiten, würde ich dies jedoch zurzeit noch vorziehen.

## 5.8 Mehrere Monitore – Multi-Head

Neben der 3-D-Beschleunigung bietet Ihnen VirtualBox noch eine Erweiterung für die virtuelle Grafikkarte an: Multi-Head. Sie können dabei festlegen, wie viele Monitore an die virtuelle Maschine angeschlossen sind. Allerdings ist diese Funktion noch nicht ausgereift:

- ▶ Man kann diese Einstellungen nicht in der grafischen Oberfläche aktivieren.
- ▶ Die Erweiterung funktioniert zurzeit nur mit Windows-Gastsystemen (wie die 3-D-Beschleunigung anfangs auch).
- ▶ Die virtuelle Maschine muss mit VBoxHeadless gestartet werden.

Da Sie diese Einstellung nicht in der grafischen Oberfläche vornehmen können, öffnen Sie zunächst die Konsole bzw. Eingabeaufforderung. Mit dem Parameter `--monitorcount` legen Sie die Anzahl der Monitore fest. Dabei können Sie maximal acht Monitore aktivieren.

**VBoxManage – Multi-Head**

```
VBoxManage modifyvm WinXP --monitorcount 4
```

In diesem Beispiel habe ich vier Monitore »angeschlossen«. Der Zugriff erfolgt dann über vier parallele RDP-Zugriffe.

[+]

**Grafikspeicher der Anzahl der Monitore anpassen**

Je nachdem, wie viele Monitore Sie einrichten und welche Auflösung Sie verwenden, müssen Sie der virtuellen Maschine genügend Grafikspeicher zuweisen.

Da der Zugriff über mehrere RDP-Sitzungen erfolgt, muss für die Maschine auch die Option `--vrdpmulticon` eingeschaltet werden. Dadurch sind mehrere, parallele RDP-Verbindungen möglich. Des Weiteren müssen Sie, wie bereits in Abschnitt 4.1.5, »Fernsteuerung (Remotezugriff)«, die Fernsteuerung (RDP) in den Einstellungen der virtuellen Maschine konfigurieren.

**VBoxManage – RDP konfigurieren**

```
VBoxManage modifyvm WinXP --vrdpmulticon on
```

Wie erwähnt erfolgt der Start nicht über die grafische Oberfläche, sondern mit dem Kommandozeilentool `VboxHeadless`.

**VBoxHeadless – virtuelle Maschine starten**

```
VBoxHeadless --startvm WinXP
```

Auf dem Reiter ANZEIGE nehmen Sie nun noch die Konfiguration für mehrere Monitore vor. Aktivieren Sie die Option **WINDOWS-DESKTOP AUF DIESEM MONITOR ERWEITERN**, um einen Monitor freizuschalten.

Der Zugriff auf die Maschine erfolgt nun mit dem Kommandozeilentool `rdesktop-vrdp` oder über einen anderen (grafischen) RDP-Client. Über den Parameter `-d` (*Domäne*) geben Sie an, auf welchen Monitor zugegriffen werden soll:

So führen Sie einen Fernzugriff mit `rdesktop-vrdp` durch:

```
rdesktop-vrdp 192.168.0.254:3400 -d @1 &
rdesktop-vrdp 192.168.0.254:3400 -d @2 &
```

Das kaufmännische Und (&) in diesem Beispiel bewirkt, dass die Anwendung im Hintergrund gestartet wird. In Abbildung 5.33 erfolgt der Zugriff über einen Windows-RDP-Client.

Nach dem Start der RDP-Clients können Sie diese nun so verschieben, dass sie ein Gesamtbild von vier »Monitoren« ergeben (vgl. Abbildung 5.34: Vier Monitore).



Abbildung 5.33 Zugriff auf den zweiten Monitor



Abbildung 5.34 Vier Monitore

5.9 SMP – mehrere Prozessoren

Version 3 hat neben der verbesserten 3-D-Beschleunigung eine weitere Neuerung gebracht: SMP (*Symmetrisches Multiprozessorsystem*). Sie können einer virtuellen Maschine nun mehr als einen virtuellen Prozessor zuweisen (Abbildung 5.35). Voraussetzung ist, dass Ihr Hostsystem Intel VT oder AMD-V unterstützt und Sie dieses auf dem Register BESCHLEUNIGUNG aktivieren. Weiterhin benötigen Sie die Funktion I/O-APIC im Register HAUPTPLATINE.

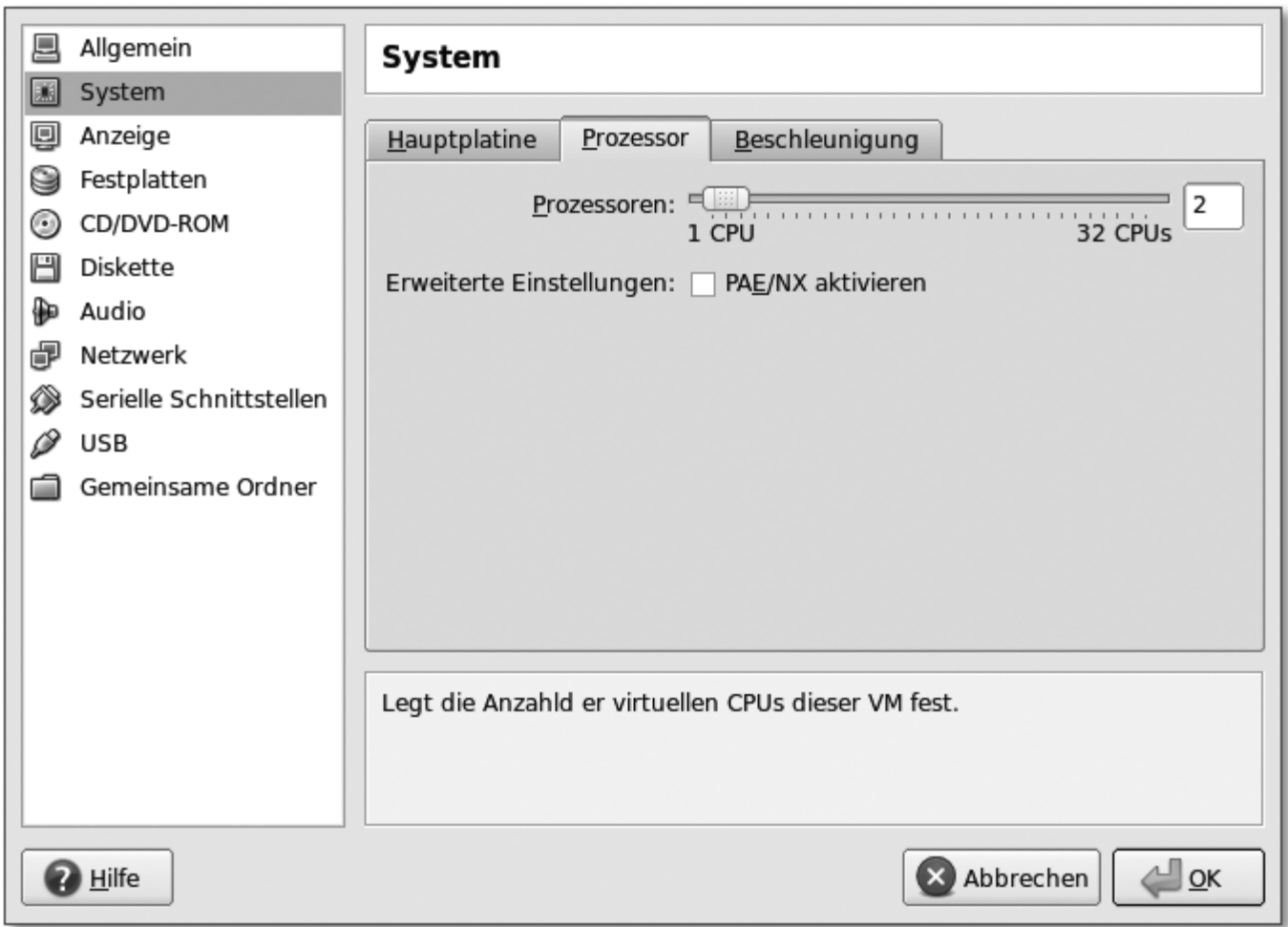


Abbildung 5.35 Die Anzahl der virtuellen CPUs festlegen

VirtualBox empfiehlt hier jedoch, einer Maschine nicht mehr Prozessoren zuzuweisen, als das Hostsystem zur Verfügung stellt. Wenn Sie versuchen, einer Maschine mehr als doppelt so viele Prozessoren zu geben, als vorhanden sind, akzeptiert VirtualBox diese Eingabe nicht.

Abbildung 5.36 zeigt den Windows-Gerätemanager und den Task-Manager. Das Windows-7-Gastsystem besitzt nun zwei Prozessoren.

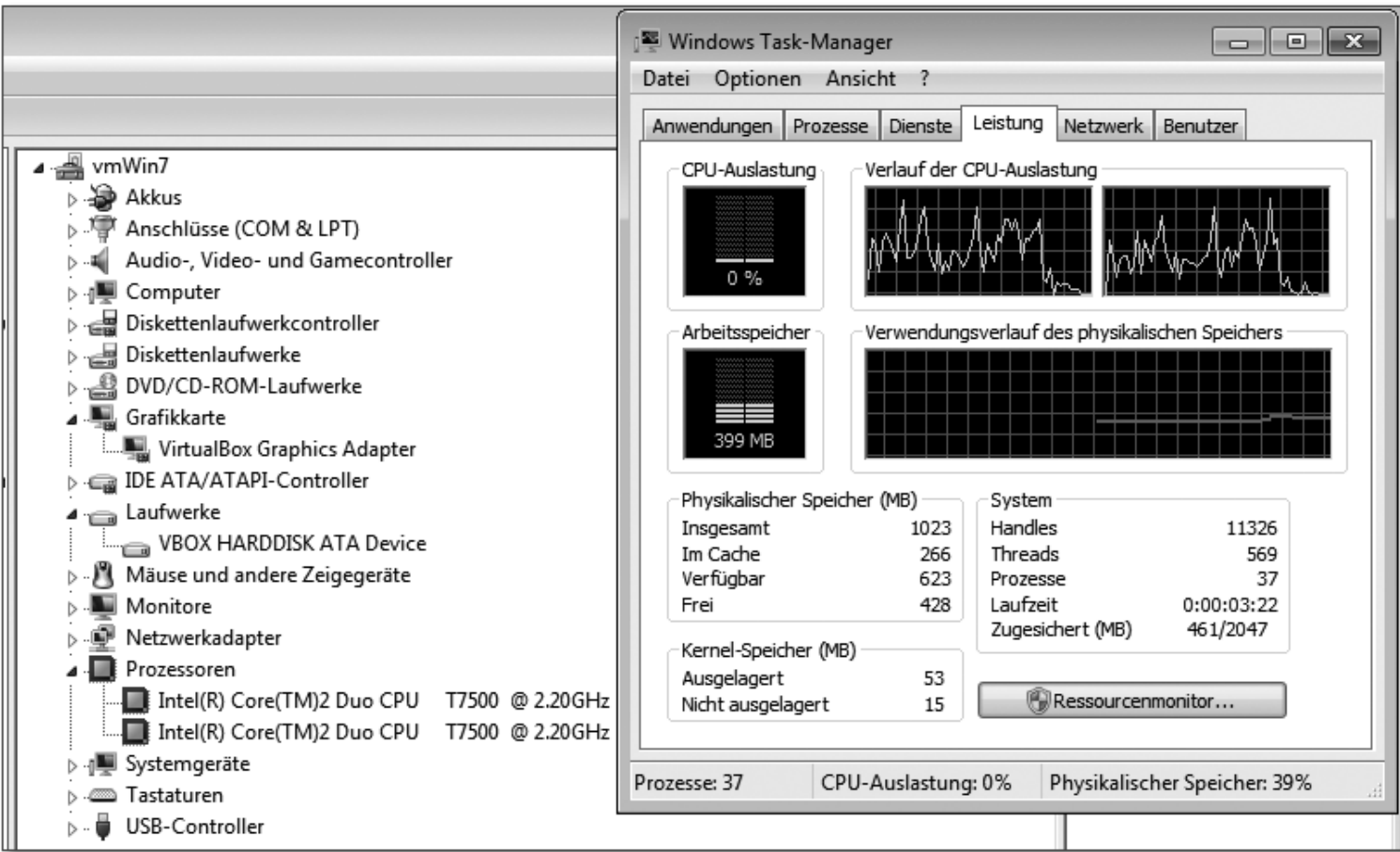


Abbildung 5.36 Windows 7 mit zwei Prozessoren

Auch unter Linux funktioniert dieses Feature. In Abbildung 5.37 sehen Sie die Ubuntu-Testmaschine ebenfalls mit zwei Prozessoren.

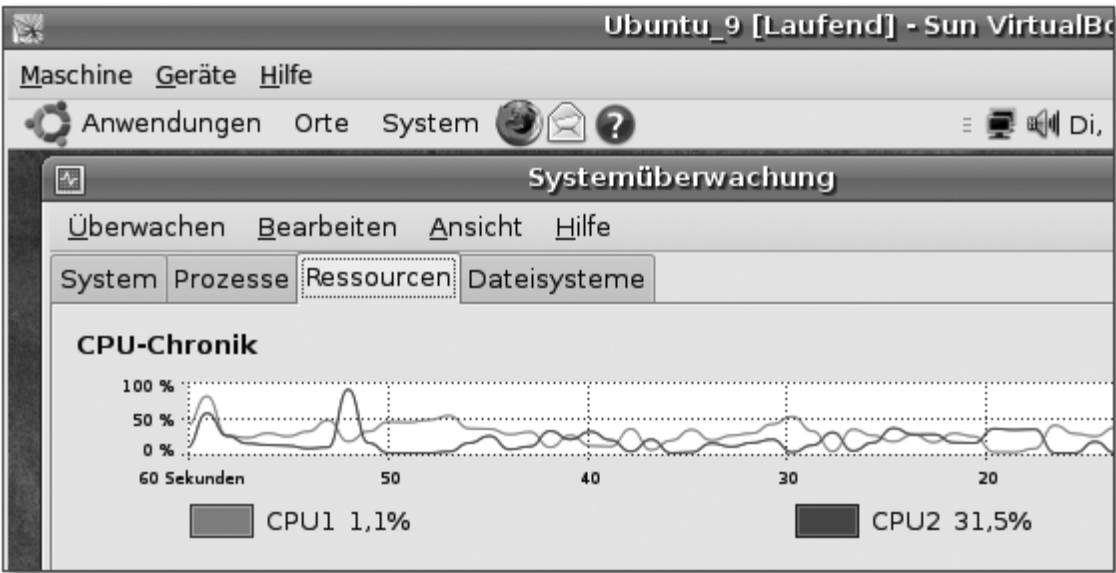


Abbildung 5.37 Ubuntu mit zwei CPUs

Windows-Gastsysteme – I/O-APIC aktivieren

Wenn Sie einem Windows-Gastsystem, das noch kein I/O-APIC verwendet, mehrere Prozessoren zuweisen wollen, sollte zunächst I/O-APIC aktiviert und unter dem Gastsystem eingerichtet werden (automatische Erkennung). Dann erst sollten Sie die Anzahl der Prozessoren ändern. Andernfalls startet Windows in den meisten Fällen nicht korrekt.

[+]

Kommen wir nun noch zu einem Test, durch den wir feststellen, ob die zweite CPU einen Geschwindigkeitsschub mit sich bringt oder nicht. In Abbildung 5.38 ist das Ergebnis eines simplen Benchmarktests mit einer einzigen CPU zu sehen. Das Ergebnis liegt bei 6364.

CPU		CPU Takt	Motherboard
25587	8x Xeon E5462	2800 MHz	Intel S5400SF
13578	4x Core 2 Extreme QX...	2666 MHz	Gigabyte GA-EP35C-DS3R
11950	2x Core 2 Duo E6700	2666 MHz	Abit AB9
10990	4x Xeon X3210	2133 MHz	Intel D975XBX2
9193	8x Xeon L5320	1866 MHz	Intel S5000VCL
8085	4x Xeon 5140	2333 MHz	Intel S5000VSA
7898	2x Athlon64 X2 Black ...	3200 MHz	MSI K9N SLI Platinum
6455	2x Athlon64 X2 4000+	2100 MHz	ASRock ALiveNF7G-HDrea...
6364	Core 2 Duo	2179 MHz	VirtualBox Virtual Platfo...
6051	2x Pentium EE 955 HT	3466 MHz	Intel D955XBK
5897	4x Phenom 9500	2200 MHz	Asus M3A
5263	2x Core Duo T2500	2000 MHz	Asus N4L-VM DH
5089	2x Core 2 Duo T5600	1833 MHz	Asus F3000Jc Notebook
4624	P4EE HT	3733 MHz	Intel SE7230NH1LX

Abbildung 5.38 Test mit einer CPU

Bei zwei Prozessoren wird sich der Wert nicht verdoppeln – eine zweite CPU bringt nicht die doppelte Geschwindigkeit. Aber dennoch sollte es eine Verbesserung geben. Im zweiten Test (Abbildung 5.39) ist dies auch der Fall; hier ist das Ergebnis 8409.

CPU		CPU Takt	Motherboard
25587	8x Xeon E5462	2800 MHz	Intel S5400SF
13578	4x Core 2 Extreme QX...	2666 MHz	Gigabyte GA-EP35C-DS3R
11950	2x Core 2 Duo E6700	2666 MHz	Abit AB9
10990	4x Xeon X3210	2133 MHz	Intel D975XBX2
9193	8x Xeon L5320	1866 MHz	Intel S5000VCL
8409	2x Core 2 Duo	2200 MHz	VirtualBox Virtual Platfo...
8085	4x Xeon 5140	2333 MHz	Intel S5000VSA
7898	2x Athlon64 X2 Black ...	3200 MHz	MSI K9N SLI Platinum
6455	2x Athlon64 X2 4000+	2100 MHz	ASRock ALiveNF7G-HDrea...
6051	2x Pentium EE 955 HT	3466 MHz	Intel D955XBK
5897	4x Phenom 9500	2200 MHz	Asus M3A
5263	2x Core Duo T2500	2000 MHz	Asus N4L-VM DH
5089	2x Core 2 Duo T5600	1833 MHz	Asus F3000Jc Notebook
4624	P4EE HT	3733 MHz	Intel SE7230NH1LX

Abbildung 5.39 Verbesserung durch zwei CPUs

Bei Gastsystemen und Anwendungen, die SMP unterstützen, können Sie gegebenenfalls auch selbst einen Geschwindigkeitszuwachs bemerken. Bei vielen Anwendungen wiederum bringt die zweite CPU keine Verbesserung. Überlegen Sie

sich daher genau, wann Sie diese Funktion einsetzen. Denken Sie immer daran, dass dadurch auch die Leistung des Hostsystems beeinträchtigt wird.

## 5.10 Serielle Schnittstellen

Mit einer virtuellen Maschine können auch Peripheriegeräte angesprochen oder von diesen Informationen bezogen werden. Dazu bietet VirtualBox an, eine virtuelle serielle Schnittstelle (auch COM-Port genannt) zu verwenden. Dabei greift das Programm entweder auf eine vorhandene Schnittstelle des Hostsystems zu oder leitet die Daten um (Pipe). Aktuelle PCs bieten seriellen Schnittstellen häufig nicht mehr an, da sie inzwischen veraltet sind. Allerdings verwenden viele Steuerungseinrichtungen von Maschinen, Temperaturüberwachungen oder Ähnliches noch diese Ports. Für den Fall, dass Sie die serielle Schnittstelle benötigen und in Ihrem Rechner keine eingebaut ist, sollten Sie eine entsprechende PCI-Karte erwerben. Es gibt auch USB-Seriell- oder PCMCIA-Seriell-Adapter. Mit diesen hatte ich (und viele andere Benutzer) jedoch Kompatibilitätsprobleme.

### 5.10.1 Serielle Schnittstelle konfigurieren

Einer Maschine können Sie in der grafischen Oberfläche zwei Schnittstellen zuordnen. Dabei nehmen Sie die folgenden Einstellungen vor:

- ▶ PORTNUMMER
- ▶ IRQ UND I/O-PORT
- ▶ PORTMODUS
- ▶ PORTPFAD



Abbildung 5.40 Konfiguration der seriellen Schnittstellen

Bei der Vergabe der PORTNUMMER bietet VirtualBox die für COM1 bis COM4 üblichen Konfigurationen an:

- ▶ COM1: IRQ 4, 0x3F8
- ▶ COM2: IRQ 3, 0x2F8
- ▶ COM3: IRQ 4, 0x3F8
- ▶ COM4: IRQ 3, 0x2F8

Bei der Verwendung von COM1 bis COM4 werden der IRQ und die Adresse übernommen. Bei BENUTZERDEFINIERT können Sie diese selbst festlegen. Letzteres ist jedoch nur selten notwendig.

Der nächste wichtige Parameter ist der PORTMODUS. Das gleichnamige Drop-down-Menü bietet drei Optionen:

- ▶ NICHT VERBUNDEN
- ▶ HOST-SCHNITTSTELLE
- ▶ HOST-PIPE

NICHT VERBUNDEN bedeutet, dass dem Gastssystem zwar ein virtueller Port zur Verfügung steht, an diesen jedoch nichts angeschlossen ist – einfach so, als wäre auch an einem PC kein Gerät mit der Schnittstelle verbunden.

Wenn eine serielle Schnittstelle benötigt wird, werden Sie in den meisten Fällen den Portmodus HOST-SCHNITTSTELLE verwenden. Mit diesem wird der virtuellen eine physikalische Schnittstelle zugeordnet. In diesem Fall tragen Sie bei PORT-PFAD den Namen der Schnittstelle ein:

Unter Linux:

- ▶ /dev/ttyS0
- ▶ /dev/ttyS1

Unter Windows:

- ▶ COM1
- ▶ COM2

### 5.10.2 Host-Pipe

Die Einstellung HOST-PIPE dient dazu, den Datenstrom über eine sogenannte Pipe zu leiten. Damit können zum Beispiel die Daten in einer Datei gespeichert oder an eine andere Anwendung umgelenkt werden. Tauchen bei dieser Methode Probleme auf, liegt dies häufig nicht an einer fehlerhaften Konfiguration der Maschine, sondern an der falschen Anwendung der Pipe-Methoden.

In Abbildung 5.41 sehen Sie die Konfiguration einer seriellen Schnittstelle mit dem Portmodus HOST-PIPE. Der Datenstrom wird dabei in die Datei */tmp/com1.tmp* umgelenkt.





**Abbildung 5.41** Host-Pipe mit Umlenkung in die Datei /tmp/com1.tmp

Wichtig ist die Aktivierung der Checkbox ERZEUGE PIPE, damit beim Start die unter PORTPFAD angegebene Pipe-Datei erstellt wird. Diese Datei ist eine sogenannte Socket-Datei. Eine eventuelle Ausgabe wird dann nicht in diese Datei geschrieben, sondern dorthin »umgelenkt«. Eine andere Anwendung muss diesen Datenstrom abfangen und weiterverarbeiten. Unter Linux-Hostsystemen eignet sich hierzu zum Beispiel die Anwendung `socat`. Diese kann den Datenstrom einer Socket-Datei weiterverarbeiten.

In Abbildung 5.42 sehen Sie die Ausgabe einer Socket-Datei. Im Gastsystem wurde mit `echo` ein »Hallo« an die serielle Schnittstelle gesendet. Im Vordergrund erkennen Sie die Ausgabe von `socat`.

*Gastsystem:*

```
echo Hallo > COM1
```

*Hostsystem:*

```
socat /tmp/com1.tmp stdout
```

Im nächsten Beispiel werden die Daten in eine Logdatei geschrieben:

*Hostsystem:*

```
socat -u /tmp/com1.tmp GOPEN:/tmp/com1.log
```

`socat` kann aber nicht nur die Daten empfangen, sondern diese auch an die Schnittstelle senden. Dazu müssen Sie unter dem Windows-Gastsystem die Anwendung HyperTerminal starten und eine Verbindung mit COM1 aufbauen. Auf dem Hostsystem können dann die Daten folgendermaßen gesendet werden:

```
socat -U /tmp/com1.tmp STDIN
```

STDIN bedeutet, dass die Daten von der Standardeingabe kommen sollen. Die Standardeingabe ist Ihre Tastatur. In Abbildung 5.43 erkennen Sie im HyperTerminal die Ausgabe der vom Hostsystem gesendeten Daten.

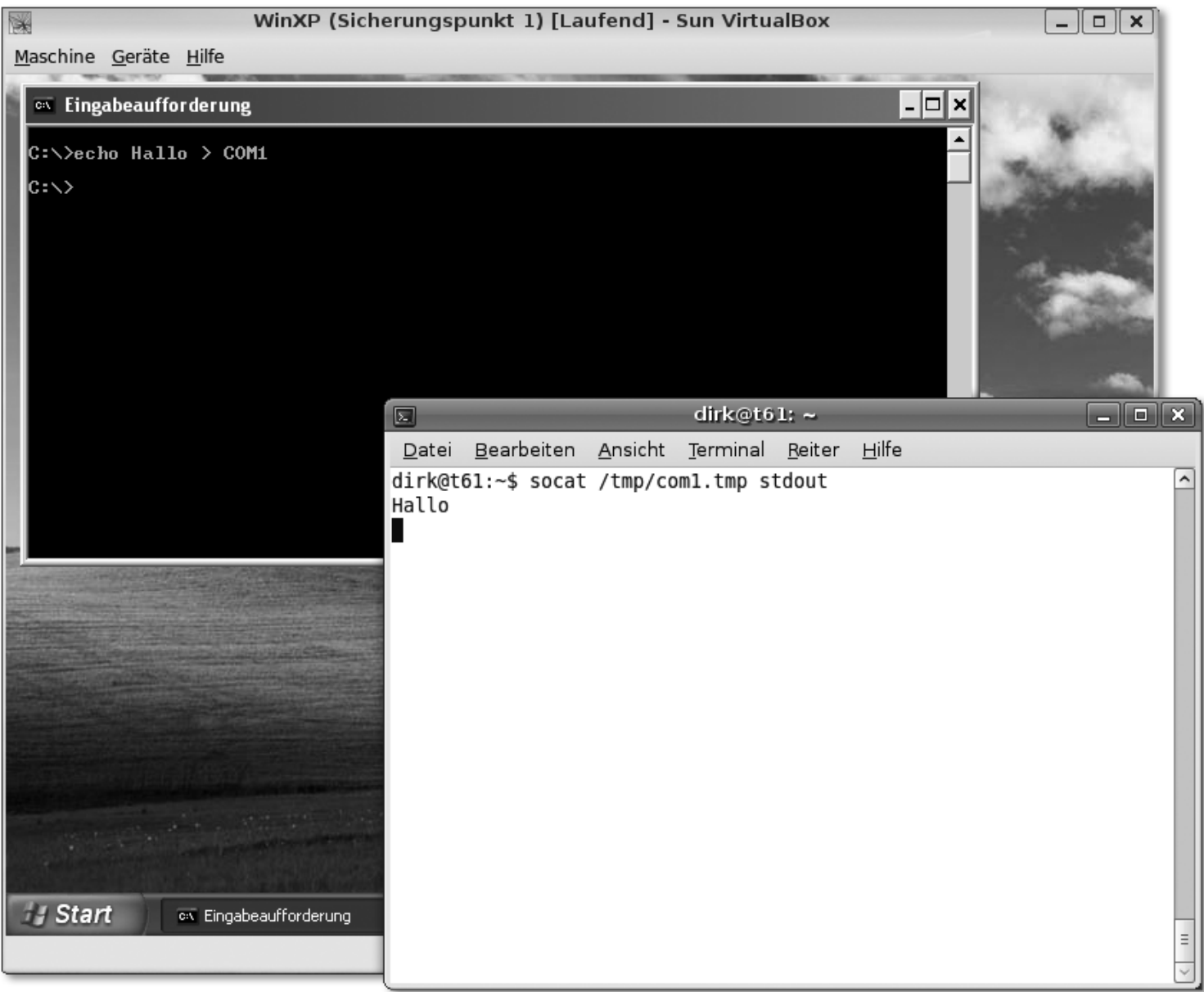


Abbildung 5.42 Ausgabe einer Socket-Datei



Abbildung 5.43 Ausgabe im Programm HyperTerminal

Weiter will ich hier nicht auf die Pipe-Methode eingehen, da ich davon ausgehe, dass Leser, die diese benötigen, im Umgang mit dieser Methode bereits sehr versiert sein dürften.

## 5.11 USB

Eines der VirtualBox-»Highlights« ist die Möglichkeit, den virtuellen Maschinen ein USB-Gerät zu übergeben. War es bisher nur möglich, auf die von dem Hypervisor angebotene, virtuelle Hardware zuzugreifen (abgesehen von den wenigen Möglichkeiten der seriellen Schnittstelle), so ist diese Schranke hiermit gefallen. Sie können den Anwendungsbereich der Gastsysteme dadurch enorm erweitern. Es gibt zwar noch USB-Hardware, die nicht funktioniert, die von mir getesteten Geräte konnte ich jedoch alle erfolgreich verwenden:

- ▶ USB-Sticks
- ▶ externe Festplatten
- ▶ Drucker
- ▶ Web- und Digitalkamera
- ▶ verschiedene Handys

Ein großer Vorteil ist, dass Sie auch Geräte verwenden können, die von dem Hostsystem nicht erkannt werden. Meine (billige) Webcam ist zum Beispiel unter Linux nicht lauffähig. Mit einem Windows-Gastsystem unter VirtualBox konnte ich sie jedoch verwenden, und mit dem nahtlosen Modus fällt dieser »Trick« noch nicht einmal auf.

Beim Einsatz von USB-Geräten sollten Sie jedoch nicht vergessen, dass es sich um eine virtuelle Maschine handelt. Diese sind für Videobearbeitung oder Fernsehen über USB nicht sonderlich geeignet. Häufig kommt es hierbei zu Problemen bei der Ansicht, oder die TV-Ausgabe ruckelt bzw. der Ton ist nicht synchron.

### Probleme mit Linux-Hostsystemen

Wenn es im Folgenden zu Problemen unter Linux-Hostsystemen kommt, lesen Sie den Abschnitt 7.2, »Linux-Hostsysteme«. Dort finden Sie den entsprechenden Abschnitt »Probleme mit USB oder Rechteprobleme«.

[+]

### 5.11.1 USB konfigurieren

Um USB überhaupt nutzen zu können, muss diese Anschlussart zunächst in den Eigenschaften der virtuellen Maschine aktiviert werden (siehe Abbildung 5.44).

Je nachdem, welches Gastsystem Sie verwenden, hat der Assistent dies gegebenenfalls bereits erledigt. Ich verwende im nächsten Beispiel eine virtuelle Maschine mit Windows XP als Gastsystem.



Abbildung 5.44 USB aktivieren

Über die Checkbox USB-2.0-CONTROLLER AKTIVIEREN können Sie festlegen, ob ein USB-1.1- oder eben ein -2.0-Controller verwendet werden soll. Bei USB 2.0 müssen Sie darauf achten, dass Ihr Gastsystem dies auch unterstützt (Windows XP benötigt dazu mindestens ServicePack 1). Das Feld FILTER FÜR USB-GERÄTE dient dazu, USB-Hardware einzutragen, die dann automatisch übernommen wird. Wir nutzen diese Filter jedoch erst später.

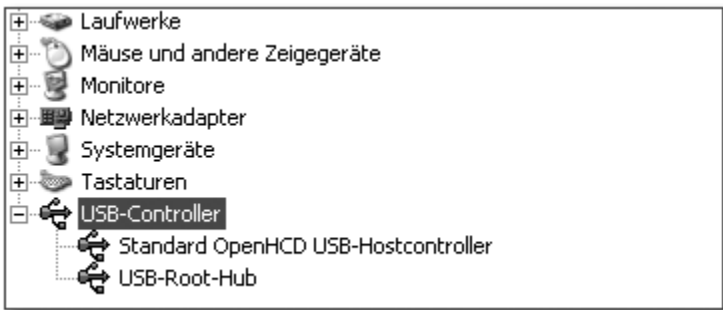


Abbildung 5.45 USB 1.1: Standard OpenHCD USB-Hostcontroller

**VBoxManage – USB 1.1 aktivieren**

```
VBoxManage modifyvm WinXP --usb on --usbehci off
```

USB 2.0 verwendet das *Enhanced Host Controller Interface* (EHCI, erweiterter Host-Controller) und den *Standard OpenHCD USB-Hostcontroller*. Dadurch können USB-1.1 und -2.0-Geräte eingebunden werden.



Abbildung 5.46 USB 2.0: erweiterter Host-Controller

**VBoxManage – USB 2.0 aktivieren**

```
VBoxManage modifyvm WinXP --usb on --usbehci on
```

**5.11.2 USB-Stick einbinden**

Nun binden wir unser erstes USB-Device ein. Fangen wir dazu zunächst einmal mit einem USB-Stick an. Die Zuordnung erfolgt im Menü GERÄTE • USB-GERÄTE. Dort erscheint eine Auflistung aller zur Verfügung stehenden USB-Geräte. Abbildung 5.47 zeigt die Auswahl eines TransMemory-Sticks.

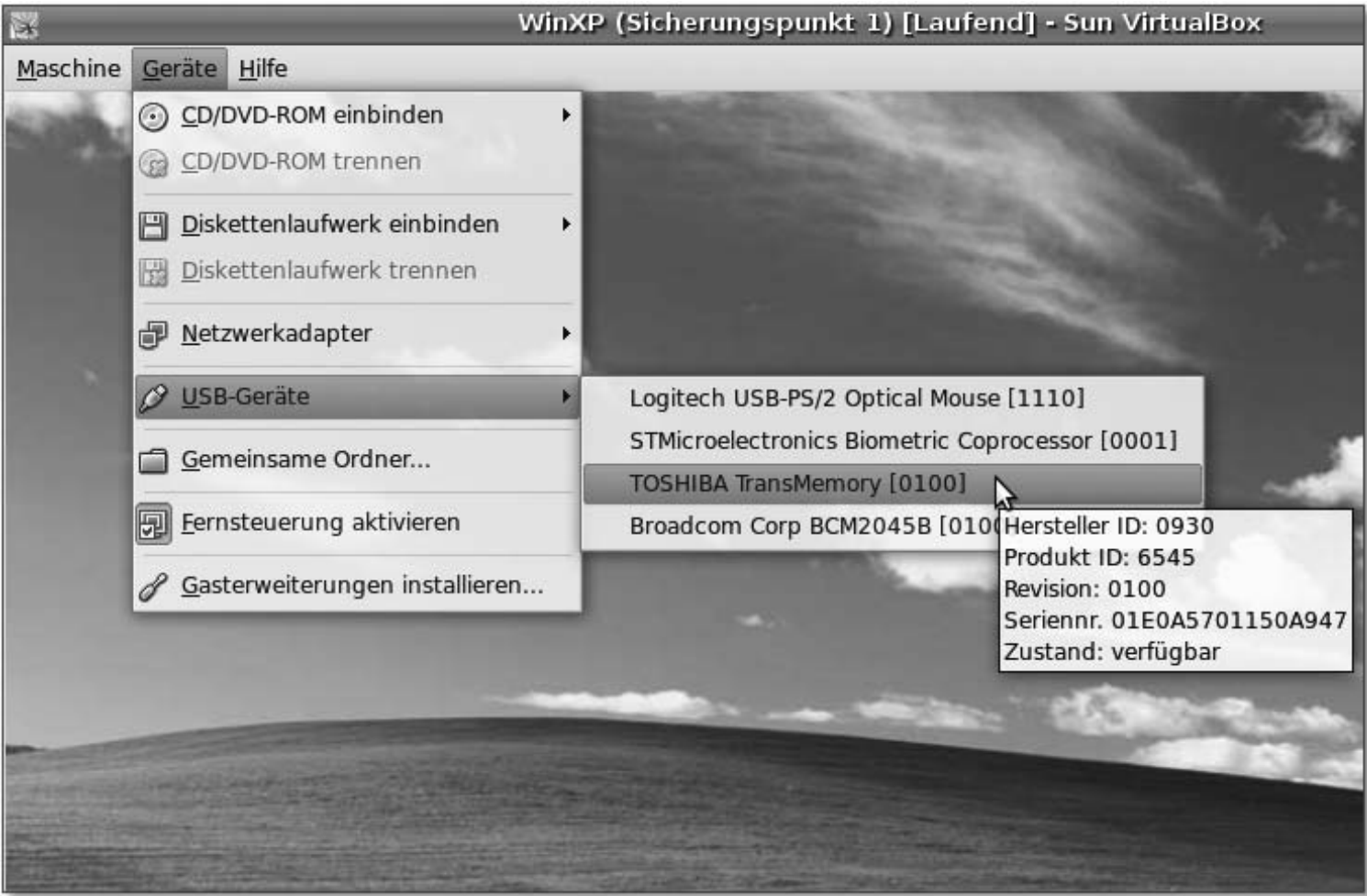


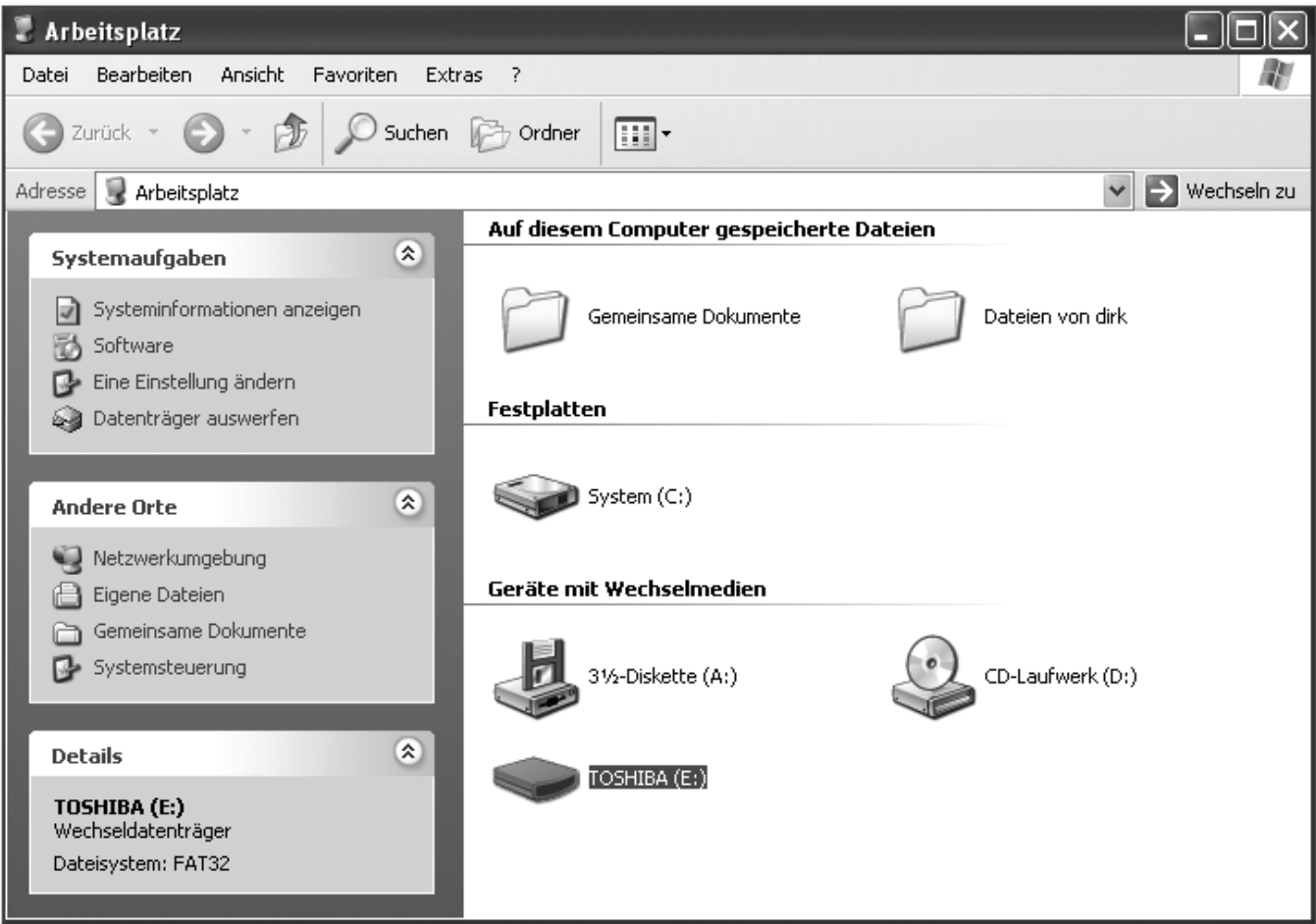
Abbildung 5.47 Einen USB-Stick einbinden



**USB-Geräte ausschließlich für Gastsystem zur Verfügung stellen**

Das Gerät, das dem Gastsystem zur Verfügung gestellt werden soll, sollte sich nicht »in Benutzung« befinden. Das bedeutet, dass Sie zum Beispiel nicht gerade Dateien vom Hostsystem auf den Stick kopieren dürfen.

Kurz nach dem Einbinden erscheint der Stick auf dem Windows-Arbeitsplatz. Wenn er zum ersten Mal verwendet wird, startet außerdem die automatische Hardwareerkennung von Windows XP.



**Abbildung 5.48** Der USB-Stick (Laufwerk E:) auf dem Arbeitsplatz

**VBoxManage – USB-Stick einbinden**

- Information über die Hardware ermitteln:  
VBoxManage list usbhost  
UUID: a77780e7-ca6c-409f-adb0-f82efdd03766  
VendorId: 0x0930 (0930)  
ProductId: 0x6545 (6545)  
Revision: 1.0 (0100)  
Manufacturer: TOSHIBA  
Product: TransMemory  
SerialNumber: 01E0A5701150A947  
Address: sysfs:/sys/devices/pci0000:00/\0000:00:1d.7/usb7/7-2//device:/dev/bus/usb/007/009  
Current State: Busy

- USB-Gerät einbinden:
- VBoxManage controlvm WinXP **usbattach\**  
a77780e7-ca6c-409f-adb0-f82efdd03766

### 5.11.3 USB-Stick freigeben

Für die Verwendung von USB-Medien gelten unter VirtualBox die gleichen Richtlinien wie für die »normale« Benutzung. Der Stick muss also vor dem Freigeben sauber aus dem Dateisystem ausgehängt werden, da es sonst zu Datenverlusten kommen kann.

Um den USB-Stick zu deaktivieren, öffnen Sie wieder das Menü GERÄTE • USB-GERÄTE (Abbildung 5.49). An dem Häkchen vor dem USB-Stick (TOSHIBA TRANSMEMORY) erkennen Sie, dass dieser zurzeit eingebunden ist (ZUSTAND: GEFANGEN).



Abbildung 5.49 Einen USB-Stick wieder aushängen

Nach dem Aushängen aus dem Gastsystem erkennt das Hostsystem das USB-Gerät, als wenn Sie es gerade eingesteckt hätten.

### VBoxManage – USB-Stick freigeben

VBoxManage controlvm WinXP **usbdetach\**  
a77780e7-ca6c-409f-adb0-f82efdd03766

### 5.11.4 USB-Filter

Bei der Aktivierung von USB hatten wir im FELD FILTER FÜR USB-GERÄTE nichts angegeben. Das USB-Gerät wurde manuell eingebunden und wieder freigegeben.

VirtualBox bietet Ihnen die Möglichkeit, spezielle USB-Geräte als Filter einzutragen, um sie automatisch der virtuellen Maschine zuzuweisen.

Um einen neuen Filter zu erstellen, gibt es zwei Möglichkeiten:

- ▶ Sie übernehmen die Informationen von einem angeschlossenen Gerät.
- ▶ Sie erstellen einen leeren Filter und geben die Daten manuell ein.

Im Kontextmenü zu den USB-Filtern können Sie die gewünschte Methode auswählen.

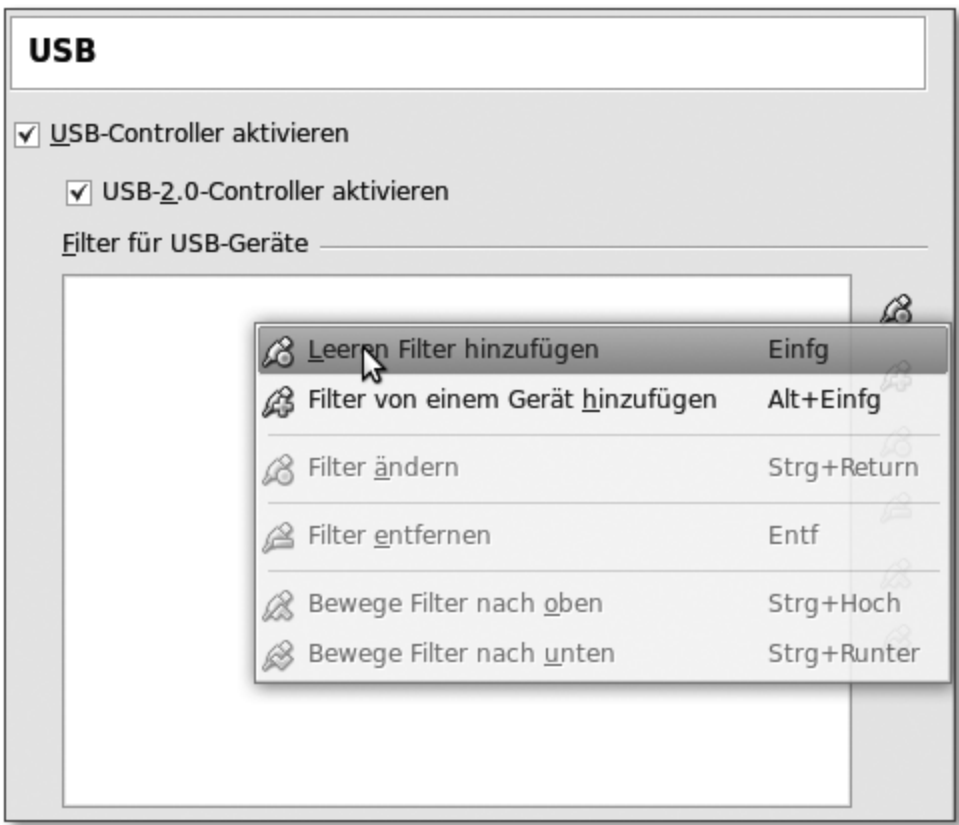


Abbildung 5.50 Einen neuen Filter für USB-Geräte anlegen

**Filter von einem Gerät hinzufügen**

Für diese Methode muss das gewünschte USB-Gerät mit dem PC verbunden sein. VirtualBox übernimmt dann automatisch die notwendigen Daten dieser Hardware.

**Leeren Filter hinzufügen**

Für den leeren Filter sollten Sie zunächst ein paar Informationen über das gewünschte Gerät ermitteln. Dazu verwenden wir den VBoxManage-Befehl `list`.

**VBoxManage – USB-Geräte ermitteln**

```
VBoxManage list usbhost
UUID:                a77780e7-ca6c-409f-adb0-f82efdd03766
VendorId:             0x0930 (0930)
ProductId:            0x6545 (6545)
```

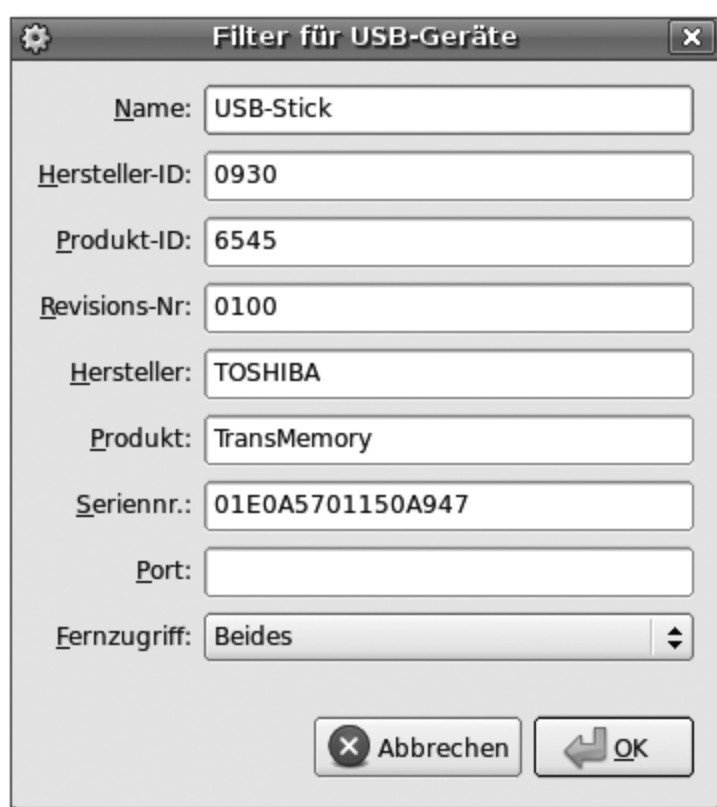


```

Revision:          1.0 (0100)
Manufacturer:      TOSHIBA
Product:           TransMemory
SerialNumber:      01E0A5701150A947
Address:           sysfs:/sys/devices/pci0000:00/\
                  0000:00:1d.7/usb7/7-2//device:/dev/bus/usb/007/009
Current State:     Busy

```

Anhand dieser Angaben können Sie nun einen neuen Filter anlegen. Alle Felder, die Sie nicht ausfüllen, gelten allgemeingültig für sämtliche USB-Geräte. Daher sollten Sie aufpassen, um nicht ungewollt Ihrem Hostsystem eventuell wichtige Komponenten zu entziehen.



**Abbildung 5.51** Dialog zum Anlegen neuer Filter

Unter NAME können Sie eine Bezeichnung für den Filter vergeben. Die weiteren Felder können Sie anhand der Ausgabe von `list usbhost` ausfüllen (verwenden Sie jedoch nur die numerischen Angaben in den Klammern). Bei PORT kann noch ein spezieller USB-Port definiert werden. Dadurch würde der Filter nur aktiv, wenn das Gerät auch wirklich in den entsprechenden Port eingesteckt wird. Der Parameter FERNZUGRIFF bestimmt, ob der Filter auch über eine RDP-Verbindung verwendet werden kann. Dabei stehen Ihnen folgende Werte zur Verfügung:

- ▶ JA: Der Filter gilt nur für Fernzugriffe.
- ▶ NEIN: Der Filter gilt nur für lokale Geräte.
- ▶ BEIDES: Der Filter gilt sowohl für Fernzugriffe als auch für lokale Geräte.

In Version 2.2 funktionierte die Einstellung JA im Dropdown-Menü FERNZUGRIFF allerdings nicht; der Filter wurde auch für lokale USB-Geräte verwendet – dies dürfte nicht sein.



USB-Filter optimal anlegen

- ▶ Nur NAME und FERNZUGRIFF müssen angegeben werden.
- ▶ Mit den Feldern können Sie den Filter präzisieren.
- ▶ Leer gelassene Felder bedeuten, dass das entsprechende Feld ignoriert wird.
- ▶ Je mehr Felder bzw. Daten Sie angeben, desto präziser ist der Filter.

5.11.5 USB über RDP

VirtualBox bietet die Möglichkeit, über einen Fernzugriff USB-Geräte einzubinden. Dies funktioniert jedoch nur mit Linux-Clients und dem Kommandozeilen-tool `rdesktop-vrdp`. Das Hostsystem, auf dem die virtuelle Maschine läuft, und deren Gastsystem spielen dabei keine Rolle.



Hinweis zur Verwendung des Kommandozeilentools `rdesktop-vrdp`

Falls diese Methode nicht funktioniert, lesen Sie den Abschnitt 7.2, »Linux-Hostsysteme«. Dort finden Sie weitere Informationen unter der Überschrift »Probleme mit USB oder Rechteprobleme«. Es liegt meist am fehlenden Eintrag in der Datei `/etc/fstab`.

Der Ablauf ist dabei wie folgt: In der Konfiguration der virtuellen Maschine richten Sie einen USB-Filter und den RDP-Zugriff ein. Nach dem Start der virtuellen Maschine bauen Sie mit dem Linux-Client die RDP-Verbindung auf und verbinden das USB-Gerät mit dem Client.

Die ersten drei Schritte kennen Sie bereits. Das Einrichten des Filters ist dabei Pflicht, da das Gerät nicht über das Menü eingefügt werden kann. Danach stellen Sie die Verbindung mit dem Kommandozeilentool `rdesktop-vrdp` her.

Fernzugriff auf USB

```
rdesktop-vrdp -r usb 192.168.0.254:3400
```

Wenn Sie nach dem Aufbau der Verbindung das USB-Gerät einstecken, wird dieses von `rdesktop-vrdp` übernommen und mit dem Gastsystem verbunden. Beim Beenden der Sitzung wird das Gerät wieder freigegeben. Daher denken Sie auch hier daran, das Gerät vorher im Gastsystem auszuhängen, um einem Datenverlust vorzubeugen.

### Fernzugriff auf USB

- ▶ USB über RDP kann nur mit Hilfe der USB-Filter verwendet werden.
- ▶ Der Zugriff ist nur von einem Linux-System mit `rdesktop-vrdp` möglich.

[!]

## 5.12 Port-Forwarding mit setextradata

Ich habe bereits mehrfach erwähnt, dass der Netzwerkmodus NAT in den meisten Fällen ausreichend sein dürfte (siehe hierzu auch Abschnitt 3.9, »Netzwerk«). Allerdings ist bei NAT kein direkter Zugriff auf eventuelle Netzwerkdienste eines Gastsystems möglich. Daher müssten Sie, um zum Beispiel einen Remote-Zugriff auf ein Windows-XP-Gastsystem zu ermöglichen, eine Bridge-Verbindung einsetzen. Mit ein wenig Netzwerkkennntnissen geht es jedoch auch anders. Mit dem `VBoxManage`-Befehl `setextradata` können Sie spezielle Parameter einer virtuellen Maschine setzen. Wir nutzen diese Fähigkeit nun, um Port-Forwarding einzurichten.

### Implementierung des Port-Forwarding eine Frage der Zeit

In neueren Programmversionen bietet die grafische VirtualBox-Oberfläche immer mehr Funktionen, die in Vorgängerversionen bislang nur mit `setextradata` gesetzt werden konnten. Prüfen Sie daher gegebenenfalls, ob nicht auch diese Funktion inzwischen mit aufgenommen wurde.

[+]

Im Folgenden beschreibe ich einen RDP-Zugriff auf eine virtuellen Maschine mit Windows XP als Gastsystem. In diesem Beispiel werden Anfragen auf Port 3500 des Hostsystems an den Port 3389 des Gastsystems umgeleitet.

### VBoxManage – Remote-Zugriff auf ein Windows-XP-Gastsystem

```
VBoxManage setextradata WinXP "VBoxInternal/Devices/pcnet\
    /0/LUN#0/Config/RDPZugriff/Protocol" TCP
VBoxManage setextradata WinXP "VBoxInternal/Devices/pcnet\
    /0/LUN#0/Config/RDPZugriff/HostPort" 3500
VBoxManage setextradata WinXP "VBoxInternal/Devices/pcnet\
    /0/LUN#0/Config/RDPZugriff/GuestPort" 3389
```

Es werden hier drei Parameter gesetzt: `Protocol` (TCP), `HostPort` (3500) und `GuestPort` (3389). Die Bezeichnung `RDPZugriff` ist frei wählbar; Sie können sie und auch ändern. Die Angabe `pcnet` bedeutet, dass die Einstellungen für den virtuellen AMD-PCnet-Adapter gelten. Wenn die Maschine einen Adapter der Intel-Pro/1000-Familie verwendet, ersetzen Sie den Wert durch `e1000`. Die darauf folgende `0` (Null) verweist auf die Nummer der Karte (beginnend bei 0). Wenn Sie

mehrere Adapter zugeordnet hätten, müsste hier der Wert des gewünschten eingetragen werden.

Bei einem Zugriff auf das Gastsystem verwenden Sie nun die Netzwerkadresse des Hostrechners und den entsprechenden Port.



**Abbildung 5.52** RDP-Zugriff von Windows aus

Im nächsten Beispiel gehe ich davon aus, dass ein Linux-Gastsystem auf einem Windows-Hostsystem läuft. Der virtuellen Maschine wurden zwei Intel-Pro-Adapter zugewiesen. Nun sollen SSH-Anfragen (Port 22) auf Ihr Hostsystem an den zweiten Adapter des Gastsystems weitergeleitet werden.

### **VBoxManage – SSH-Zugriff auf ein Linux-Gastsystem**

```
VBoxManage setextradata Ubuntu "VBoxInternal/Devices/e1000\
/1/LUN#0/Config/SSH/Protocol" TCP
VBoxManage setextradata Ubuntu "VBoxInternal/Devices/e1000\
/1/LUN#0/Config/SSH/HostPort" 22
VBoxManage setextradata Ubuntu "VBoxInternal/Devices/e1000\
/1/LUN#0/Config/SSH/GuestPort" 22
```

Beachten Sie, dass Sie bei `HostPort` nicht mehrmals den gleichen Port übergeben. Bei `GuestPort` ist dies natürlich nicht wichtig – solange es sich um unterschiedliche Maschinen handelt.

## 5.13 RPD-Authentisierung

Bei unserem ersten Einsatz des in VirtualBox integrierten VRDP-Servers (vgl. Abschnitt 3.6.4, »rdesktop-vrdp«) haben wir die Standard-Authentisierungsmethode NULL (also keine Authentisierung) verwendet. Dadurch kann jeder Netzwerkteilnehmer ohne Passwort auf eine virtuelle Maschine zugreifen. Für ein privates Netzwerk ist dies natürlich völlig ausreichend. Wenn jedoch ein höherer Zugriffsschutz benötigt wird, bietet VirtualBox hierzu noch zwei weitere Methoden:

- ▶ Extern
- ▶ Gast

Die jeweilige Methode aktivieren Sie in den Eigenschaften einer Maschine im Bereich FERNSTEUERUNG (Abbildung 5.53).

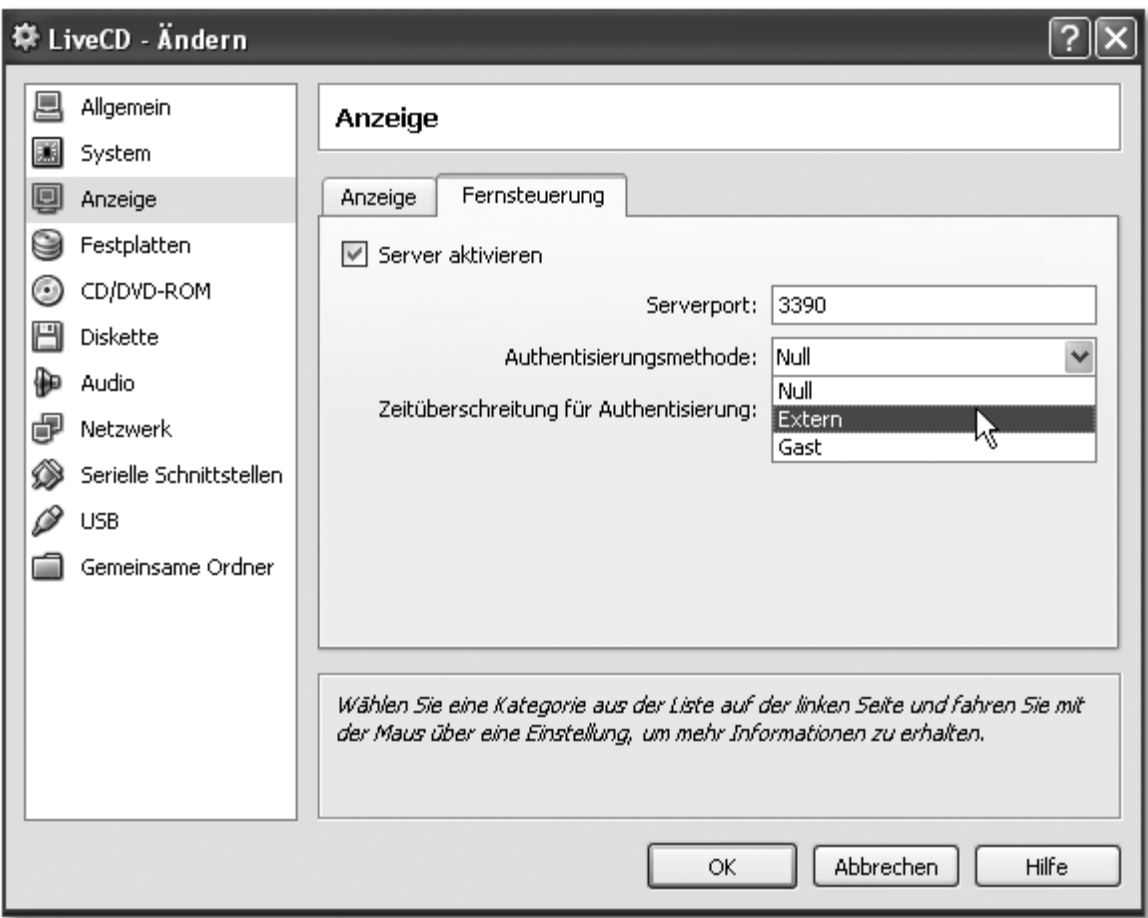
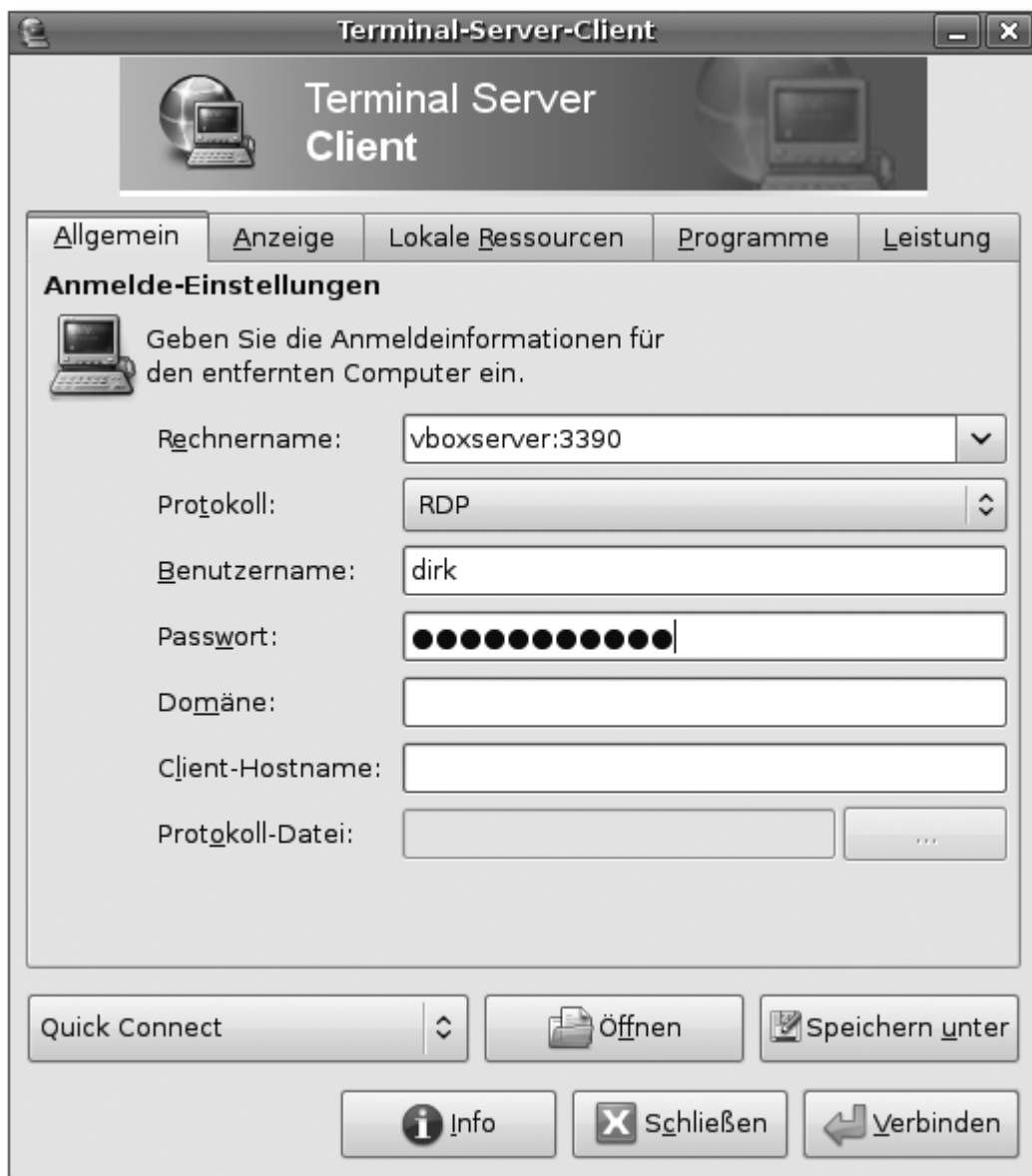


Abbildung 5.53 Authentisierung für die Fernsteuerung

### 5.13.1 Extern

Die Authentisierungsmethode EXTERN ermöglicht es, für den RDP-Zugriff auf eine virtuelle Maschine einen Benutzernamen und ein Passwort zu setzen. Diese Benutzerangaben werden dabei standardmäßig mit den Accounts des jeweiligen Hostsystems abgeglichen. Diese Art der Authentisierung wird nur unter Linux- und Windows-Hostsystemen unterstützt.

Wenn Sie die Methode EXTERN aktiviert haben, muss im RDP-Client ein Benutzer und sein Passwort angegeben werden. Im Gegensatz zu einer »normalen« Windows-RDP-Sitzung öffnet sich kein Anmeldefenster, in dem das Passwort abgefragt wird. Sie müssen es also bereits bei der Konfiguration des Clients eintragen.



**Abbildung 5.54** Den RDP-Client unter Ubuntu einrichten

Danach baut der Client die Verbindung zu VirtualBox auf, übergibt die Benutzerdaten und öffnet, bei erfolgreicher Anmeldung, das Fenster für die Fernsteuerung. Es bietet sich an, einen speziellen Benutzer für VirtualBox zu erstellen, der eventuell eingeschränkte Rechte auf dem Hostsystem besitzt.

VirtualBox verwendet für diese Anmeldung die Bibliothek *VRDPAuth*. Sie können jedoch auch eine eigene erstellen und dadurch andere Benutzerverwaltungen verwenden. Näheres hierzu finden Sie in der englischen Dokumentation von VirtualBox.

### **VBoxManage – Authentisierungsmethode setzen (Extern)**

```
VBoxManage.exe modifyvm "Windows 7" --vrdpauthtype external
```

### 5.13.2 Gast

Diese Methode ermöglicht den RDP-Login in Verbindung mit den Gasterweiterungen über die Benutzer des Gastsystems. In Version 3.0 befand sich dieses Feature jedoch noch im Teststadium und funktionierte auf allen meinen getesteten Gastsystemen (Windows XP, Windows 7 und Ubuntu 9.04) nicht.

#### **VBoxManage – Authentisierungsmethode setzen (Gast)**

```
VBoxManage.exe modifyvm "Windows 7" --vrdpauthtype guest
```

## 5.14 Virtuelle Maschinen an- und abmelden

Es gibt die Möglichkeit, nicht benötigte Maschinen abzumelden, ohne sie zu löschen. Diese können dann später wieder registriert und verwendet werden. Diese Registrierung ist bei VirtualBox jedoch leider nicht so einfach wie unter VMware. Um eine vorhandene Maschine wieder anzumelden, darf nicht bereits eine Maschine mit der gleichen UUID vorhanden sein; eine einfache Kopie ist somit nicht möglich.

Ein Abmelden über die grafische Oberfläche ist nicht möglich. Dort kann eine Maschine nur gelöscht werden. Daher verwenden Sie das Kommandozeilentool VBoxManage mit dem Befehl `unregistervm`.

#### **VBoxManage – virtuelle Maschine abmelden**

```
VBoxManage unregistervm Test
```

oder per UUID:

```
VBoxManage unregistervm 8cb70952-3ed5-41ef-bd0a-ee4dd3295c89
```

Voraussetzung für eine erfolgreiche Abmeldung mit VBoxManage ist jedoch, dass keine Medien (Festplatten-Images, DVDs usw.) und Sicherungspunkte in der Konfigurationsdatei enthalten sind. Diese müssen Sie zuerst entfernen.

Das erneute Anmelden erfolgt mit `registervm`. Hierbei ist jedoch die komplette Pfadangabe notwendig.

#### **VBoxManage – virtuelle Maschine registrieren**

- unter Windows:

```
VBoxManage registervm E:\VBox\VMS\Test\Test.xml
```

- unter Linux:

```
VBoxManage registervm /media/vms/Test/Test.xml
```

Wie auch beim Abmelden, können nur Maschinen registriert werden, die keine Images oder Sicherungspunkte enthalten.



#### Beachten Sie beim An- und Abmelden

- ▶ Ein Entfernen über die grafische Oberfläche löscht die Konfigurationsdatei.
- ▶ Vor dem Abmelden einer virtuellen Maschine müssen registrierte Images und Sicherungspunkte aus der Konfiguration entfernt werden.
- ▶ Es können nur »leere« Konfigurationsdateien registriert werden.

## 5.15 Virtuelle Maschine löschen

In den Praxisbeispielen haben wir immer nur neue virtuelle Maschinen erstellt oder vorhandene für Tests verwendet. Allerdings wird es auch vorkommen, dass Sie nicht mehr benötigte Maschinen löschen wollen. Dabei sollten Sie die folgenden Punkte beachten:

- ▶ Zuerst müssen Sie die Sicherungspunkte entfernen.
- ▶ Der Ordner der Maschine wird meist nicht komplett gelöscht.

Das Entfernen können Sie über die grafische Oberfläche vornehmen. Zuvor entfernen Sie jedoch die Sicherungspunkte. Beim Löschen über die grafische Oberfläche werden danach nicht alle Daten entfernt. Der Ordner der Maschine bleibt häufig noch erhalten, da sich Backups von Updates oder Ähnliches in ihm befinden. Daher löschen Sie das Verzeichnis von Hand.

Auch das Festplatten-Image der Maschine entfernen Sie gesondert mit dem Manager für virtuelle Medien.

Bitte achten Sie auf diese Punkte, da sich ansonsten schnell sehr viele unnötige Dateien auf der Partition der virtuellen Maschinen ansammeln.

## 5.16 Automatische Installation der Gasterweiterungen unter einem Windows-Gastsystem

Die Gasterweiterungen bieten unter Windows-Gastsystemen mehrere Optionen, um die Installation zu beeinflussen. Wenn Sie die Installationsdatei zusammen mit `/?` aufrufen, zeigt ein Dialogfenster diese an.





Abbildung 5.55 Die Parameter der Gasterweiterungen

Durch den Aufruf mit `/S` werden die Gasterweiterungen ohne Bildschirmhinweise (*silent install*) installiert. Ein Manko hierbei ist jedoch die Treibersignierung. Diese meldet weiterhin, dass Treiber installiert werden, die den Windows-Logo-Test nicht bestanden haben. Um die Treiber also wirklich im Hintergrund zu installieren, muss die Treibersignierung in den Systemeigenschaften im Register `HARDWARE` deaktiviert werden.



Abbildung 5.56 Die Treibersignierung deaktivieren

Des Weiteren benötigt der Benutzer auch Administrationsrechte. Wenn dies nicht erwünscht ist, müssen Sie die Gruppenrichtlinien verwenden. Auf diese will ich in diesem Buch jedoch nicht weiter eingehen.

Nun können Sie die Installation im Hintergrund starten. Dadurch ist es möglich, Windows-Gastsysteme immer auf dem aktuellen Stand zu halten.

## 5.17 Virtuelle Maschine klonen

VirtualBox bietet keine Funktion, um eine komplette Maschine inklusive der Konfigurationsdatei oder sogar mit Sicherungspunkten zu klonen. Die einzige Möglichkeit ist, das Festplatten-Image zu klonen und dieses dann in eine neue Konfigurationsdatei einzubinden (siehe Abschnitt 5.2.3, »Kopieren (Klonen)«).

## 5.18 VirtualBox sichern

VirtualBox und die virtuellen Maschinen können Sie auf verschiedene Arten sichern. Dabei unterscheide ich grob zwischen folgenden Vorgehensweisen:

- ▶ Sichern einzelner Festplatten-Images
- ▶ Sichern einer virtuellen Maschine
- ▶ Komplettsicherung von VirtualBox (alle Maschinen und Einstellungen)

Die Wiederherstellung eines Backups ist dabei immer abhängig davon, ob das Original noch vorhanden ist oder nicht. Wenn zum Beispiel eine virtuelle Maschine noch vorhanden ist, kann die gesicherte nicht einfach wieder registriert werden, da deren UUID ja bereits vorhanden ist.

### 5.18.1 Festplatten-Image sichern

Diese Methode verwende ich vorwiegend zum Sichern von Datenmedien oder Systemständen (zum Beispiel vor der Übernahme von Sicherungspunkten). Bei einem Systemausfall geht jedoch die Konfiguration der virtuellen Maschine verloren und muss daher neu erstellt werden.

Das Image muss zum Sichern nur auf das entsprechende Medium kopiert werden. Für den Fall, dass Sicherungspunkte vorhanden sind, werden die dort enthaltenen Änderungen natürlich nicht mit gesichert. Daher müssen Sie zunächst entscheiden, ob die Sicherungspunkte vorher übernommen werden oder ob sie nicht wichtig sind.

Beim Wiederherstellen sollten Sie prüfen, ob das Original-Image noch vorhanden ist oder nicht. Wenn es noch vorhanden ist, müssen eventuelle Sicherungspunkte

vorher entfernt werden. Bei Verwendung auf einem anderen System muss das Image nach dem Einspielen erneut registriert werden.

#### Richtige Verwendung der Sicherung von Festplatten-Images

- ▶ Diese Sicherungsmethode ist am besten für Datenmedien geeignet.
- ▶ Die Änderungen in Sicherungspunkten sind nicht enthalten.
- ▶ Vor dem Wiederherstellen müssen Sicherungspunkte entfernt werden.

[+]

### 5.18.2 Virtuelle Maschine sichern

Das Sichern einer kompletten virtuellen Maschine (Konfiguration, Festplatten-Images und Sicherungspunkte) ist leider nicht wirklich möglich, da Maschinen mit Images und Sicherungspunkten nicht wieder registriert werden können. Ein möglicher Ausweg ist das Anpassen der globalen Konfigurationsdatei. Da dies aber XML-Erfahrung voraussetzt und keine wirklich sinnvolle Sicherungsmethode ist, möchte ich nicht weiter darauf eingehen.

### 5.18.3 Komplettsicherung

Das komplette Backup von VirtualBox benötigt natürlich auch den meisten Speicherplatz auf dem Sicherungsmedium. Sie sichern bei dieser Methode die VirtualBox-Konfigurationsdatei, sämtliche Maschinen und deren Festplatten-Images sichern. Die Komplettsicherung eignet sich am besten dazu, Ihre Umgebung vor einer eventuellen Neuinstallation bzw. vor einem Umzug des Hostsystems zu sichern oder diese nach einem Totalausfall wiederherzustellen.

In der Datei *VirtualBox.xml* sind, wie in Abschnitt 3.5.2, »Globale Einstellungen«, beschrieben, die globalen Einstellungen, die Registrierungen aller virtuellen Maschinen, Festplatten-Images und CD/DVD-Abbilder enthalten. Vor einer Wiederherstellung sollten Sie zunächst die gleiche VirtualBox-Version installieren, die auch zum Zeitpunkt der Sicherung verwendet wurde. Dadurch vermeiden Sie eventuelle Kompatibilitätsprobleme. Danach ersetzen Sie die neue, leere Datei *VirtualBox.xml* durch die gesicherte.

Zur Erinnerung – die Konfigurationsdatei befindet sich in den folgenden Verzeichnissen:

- ▶ Unter Linux:  
*/home/BENUTZER/.VirtualBox/VirtualBox.xml*
- ▶ Unter Windows:  
*C:\Dokumente und Einstellungen\BENUTZER\.VirtualBox\VirtualBox.xml*

- Unter Windows Vista/7:  
`C:\Users\BENUTZER\VirtualBox`

Außerdem sollten sich die virtuellen Maschinen und Festplatten-Images an demselben Ort bzw. auf demselben Laufwerk befinden wie vorher. Wenn dies nicht der Fall ist, können Sie die Konfigurationsdatei manuell anpassen. Dazu öffnen Sie sie in einem geeigneten Editor (unter Windows zum Beispiel mit Notepad++) und ändern die Pfadangaben:

```
<MachineEntry uuid="{07a4350c-2285-478a-489b-46949ce3d970}"\  
  src="/media/vms/vbox/VM/LiveCD/LiveCD.xml"/>
```

Die UUIDs müssen und dürfen Sie hierbei natürlich nicht verändern.

[+]

**Pfade von CD/DVD-Abbildern**  
Meist reicht ein Suchen und Ersetzen für die Pfade der Maschinen und Images aus. Nur bei den CD/DVD-Abbildern sollten Sie aufpassen, da diese sich an verschiedenen Orten befinden können.

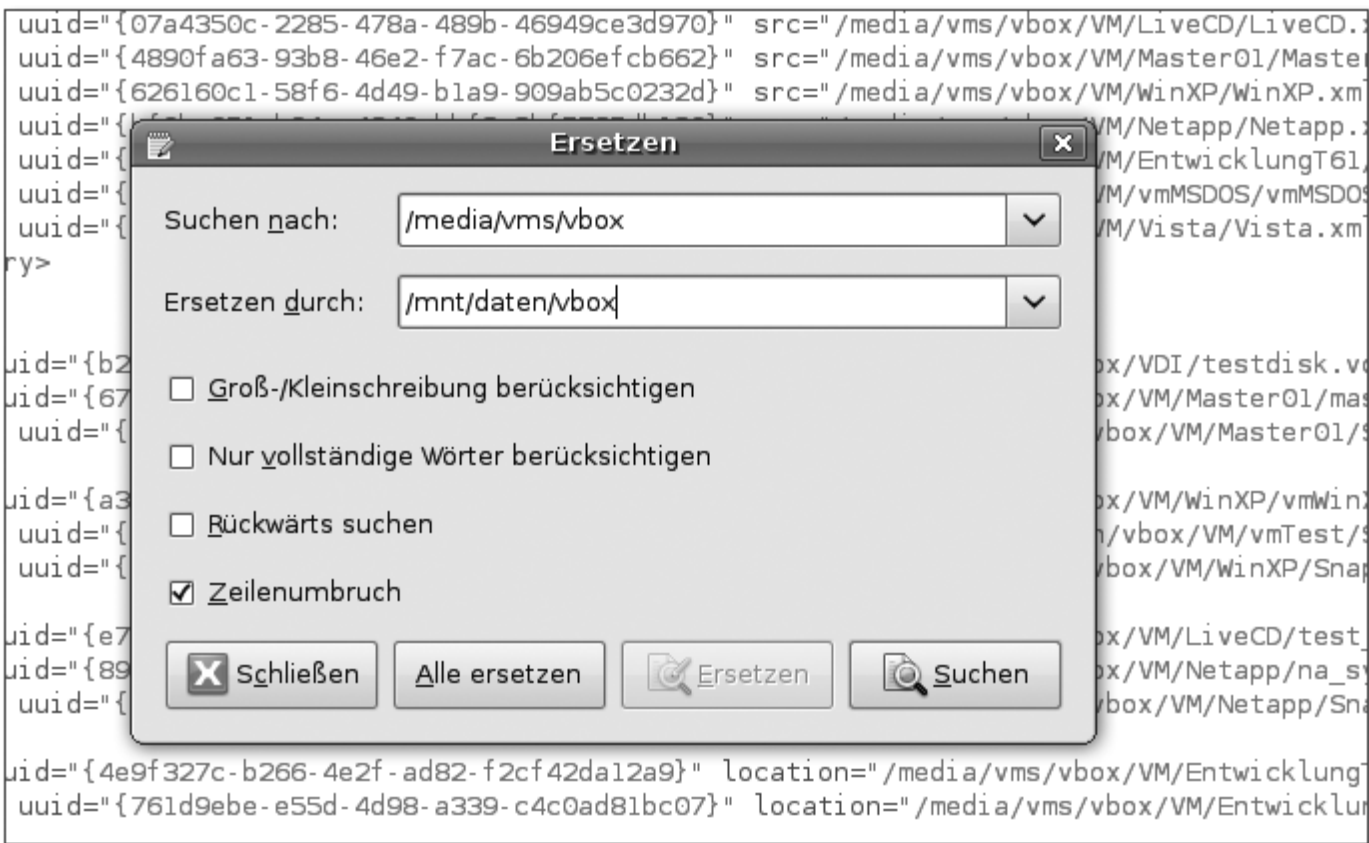


Abbildung 5.57 Suchen und Ersetzen im Texteditor

[+]

**Richtige Verwendung der Komplettsicherung**

- Diese Sicherungsmethode eignet sich auch zum Umzug der Maschinen auf ein anderes Hostsystem.
- Wenn die virtuellen Maschinen oder Festplatten-Images an einem anderen Ort wiederhergestellt wurden, müssen Sie die Datei *VirtualBox.xml* anpassen.

## 5.19 Tastenkombinationen

Achten Sie auf die korrekte Anwendung von Tastenkombinationen. Ein häufiger Fehler ist zum Beispiel, in einer virtuellen Maschine die Tastenkombination `Strg` + `Alt` + `Entf` oder `Strg` + `Alt` + `←` zu verwenden. Diese Shortcuts fängt dann das Hostsystem ab und verarbeitet sie. Die erste Tastenkombination startet unter Windows-Hostsystemen den Task-Manager. Die zweite ist unter Linux-Hostsystemen tragischer. Sie startet in verschiedenen Distributionen den X-Server neu, beendet also die grafische Arbeitsoberfläche. Für diese Tastenkombinationen gibt es daher in Verbindung mit der Host-Taste spezielle Shortcuts:

- ▶ Host-Taste + `Entf` ersetzt `Strg` + `Alt` + `Entf`.
- ▶ Host-Taste + `←` ersetzt `Strg` + `Alt` + `←`.



*Sie finden hier die Antworten auf Fragen zu Grundlagen der komplexen Themen »Migration« und »Import«. Sie können nach der Lektüre eine Migration Schritt für Schritt planen und sicher durchführen. Darüber hinaus kennen Sie die häufigsten Ursachen für Fehler und wissen, wie Sie diese beheben.*

## 6 Migration und Import

Migration wird häufig im gleichen Atemzug mit Virtualisierung genannt. Dabei ist Migration eigentlich ein viel weitreichenderer Begriff. In der virtuellen Welt ist hiermit aber meist die Virtualisierung eines vorhandenen Rechners oder Servers mit dessen Betriebssystem und allen installierten Programmen gemeint. Ein weiterer Bereich ist der Import von virtuellen Maschinen, die mit einer anderen Virtualisierungssoftware erstellt wurden. VirtualBox bietet zwei Import-Möglichkeiten an:

- ▶ Einbinden fremder Festplatten-Images
- ▶ Import von virtuellen Maschinen im Open Virtualization Format

Welche der beiden Varianten Sie nutzen, hängt von der Quelle der Maschinen ab. Bei der ersten Möglichkeit müssen Sie sich selbst um die korrekte Konfiguration der virtuellen Hardware kümmern. Wenn Ihnen jedoch eine Maschine im Open Virtualization Format angeboten wird, ist diese Methode daher vorzuziehen.

### Weiterführende Informationen zur Migration

Bei der Migration von Systemen sollten Sie Erfahrung mit Hardware und der Installation von Treibern mitbringen und natürlich die entsprechenden Systeme kennen. Als Laie auf diesem Gebiet werden Sie schnell an Ihre Grenzen stoßen. Ich gebe in den folgenden Abschnitten nur grundsätzliche Informationen zur Vorgehensweise und gehe nicht auf die Treiberinstallation oder Migrationsprobleme sämtlicher Betriebssysteme ein – dies würde den Umfang des Buchs sprengen. Weitere Hilfe finden Sie auch im VirtualBox-Forum (<http://forums.virtualbox.org/>).

[+]

## 6.1 Planung

Bevor Sie mit einer Migration beginnen, sollten Sie überlegen, ob diese auch wirklich sinnvoll ist. Die Dauer einer Migration kann bei Problemen schnell die Dauer einer Neuinstallation überschreiten.

Ein Windows-System, auf dem bereits viele Anwendungen installiert (und wieder deinstalliert) wurden, wird bekanntermaßen »schwerfällig« und in manchen Fällen auch instabil. In diesem Fall lohnt sich gegebenenfalls eine Neuinstallation unter VirtualBox. Linux-Systeme sind in dieser Hinsicht weniger problematisch. Sie sollten jedoch prüfen, ob nicht bereits eine neuere Version der entsprechenden Distribution vorhanden ist. Auch in diesem Fall lohnt sich eventuell eine Neuinstallation, denn ein Upgrade birgt gelegentlich Probleme.

Weiterhin sollten Sie sich mit dem Betriebssystem auskennen, das migriert werden soll. Nach einer Migration, egal ob von einem PC oder einer fremden virtuellen Maschine, sind zumindest Treiberanpassungen notwendig. Wenn Sie mit dem Betriebssystem nicht vertraut sind, kann sich dies zum »Stolperstein« entwickeln. Ein häufiges Problem sind dabei zum Beispiel die IDE-Treiber. Spezielle Treiber können dazu führen, dass das Gastsystem überhaupt nicht bootet.

Wenn Sie eine virtuelle Maschine von Fremdsystemen migrieren wollen, müssen Sie vorher die Gasterweiterungen der entsprechenden Virtualisierungssoftware entfernen. Diese können sonst unter VirtualBox zu Komplikationen führen. Das am einfachsten zu lösende Problem, das hier neben anderen schwerwiegenden Problemen auftreten kann, ist, dass die Erweiterungen nicht mehr starten und auch nicht mehr deinstalliert werden können.

Bei physikalischen Rechnern können Sie ein wenig Vorarbeit leisten. Nicht vorhandene IDE-Treiber, die unter VirtualBox notwendig sind, sollten installiert oder spezielle Linux-Kernel durch Standard-Kernel ausgetauscht werden.

Ein weiterer Punkt bei der Migration von Windows-Systemen, mit den entsprechenden Office-Anwendungen, ist die Aktivierung. Nach der Migration müssen Sie mit ziemlicher Sicherheit das Gastsystem und verschiedene Anwendungen wieder aktivieren. Der Grund ist, dass sich die Hardware geändert hat und diese Programme das überprüfen. Das Problem ist nicht weiter tragisch, wenn es nur selten vorkommt. Bei vielen (Migrations-)Tests kann es Ihnen jedoch passieren, dass Microsoft Ihre Lizenzschlüssel sperrt. In dem Fall müssen Sie sich mit der Firma in Verbindung setzen, um dies zu klären und die Schlüssel wieder verwenden zu können.



**VirtualBox-Wiki**

Weitere Infos, viele HOWTOs und Tutorials finden Sie auch im VirtualBox-Wiki ([http://www.virtualbox.org/wiki/User\\_HOWTOS](http://www.virtualbox.org/wiki/User_HOWTOS)).

**[+]**

## 6.2 Open Virtualization Format

Der Import fertiger Umgebungen (Virtual Appliances) ist durch die Einführung des Open Virtualization Formats stark vereinfacht worden. Komplizierter ist es zurzeit eher, eine Maschine in diesem Format zu finden. Ich fand kein passendes Beispiel und habe daher selbst eine Virtual Appliance erstellt. Leider kann ich Ihnen daher auch keine Quellen mit Beispielen zum Download angeben. Ich gehe aber davon aus, dass Ihnen beim wirklichen Einsatz des Imports von Maschinen im OV-Format auch eine entsprechende Maschine vorliegt.

### 6.2.1 Aufbau von Virtual Appliances

Eine Virtual Appliance im Open Virtualization Format besteht ähnlich wie eine normale virtuelle Maschine aus einer Konfigurationsdatei (mit der Endung *.ovf*) und den notwendigen Festplatten-Images. Ich habe in diesem Beispiel eine Windows-XP-Umgebung von einem VMware Server in das Open Virtualization Format exportiert. Die Konfigurationsdatei wurde dabei im XML-Format erstellt und kann mit einem Editor eingesehen und, wenn notwendig, bearbeitet werden.

Im folgenden Listing sehen Sie den Auszug einer solchen Datei (ich habe einige Bereiche entfernt):

```
<?xml version="1.0" encoding="UTF-8"?>
[...]
<References>
  <File ovf:href="WinXP-disk1.vmdk" ovf:id="file1"
    ovf:size="1935508992"/>
</References>
<Section xsi:type="ovf:DiskSection_Type">
  <Info>Meta-information about the virtual disks</Info>
  <Disk ovf:capacity="4294967296" ovf:diskId="vmdisk1"
    ovf:fileRef="file1"
    ovf:format="http://www.vmware.com/specifications/vmdk.html#sparse"/>
</Section>
<Section xsi:type="ovf:NetworkSection_Type">
  <Info>The list of logical networks</Info>
  <Network ovf:name="Bridged">
    <Description>The Bridged network</Description>
```

```
</Network>
</Section>
<Content ovf:id="WinXP" xsi:type="ovf:VirtualSystem_Type">
  <Info>A virtual machine</Info>
[...]
```

```
    <rasd:Caption>392MB of memory</rasd:Caption>
    <rasd:Description>Memory Size</rasd:Description>
    <rasd:InstanceId>2</rasd:InstanceId>
    <rasd:ResourceType>4</rasd:ResourceType>
    <rasd:AllocationUnits>MegaBytes</rasd:AllocationUnits>
    <rasd:VirtualQuantity>392</rasd:VirtualQuantity>
  </Item>
  <Item>
    <rasd:Caption>ideController0</rasd:Caption>
    <rasd:Description>IDE Controller</rasd:Description>
    <rasd:InstanceId>4</rasd:InstanceId>
    <rasd:ResourceType>5</rasd:ResourceType>
    <rasd:Address>0</rasd:Address>
    <rasd:BusNumber>0</rasd:BusNumber>
  </Item>
[...]
```

```
  <Item>
    <rasd:Caption>disk1</rasd:Caption>
    <rasd:InstanceId>6</rasd:InstanceId>
    <rasd:ResourceType>17</rasd:ResourceType>
    <rasd:HostResource>/disk/vmdisk1</rasd:HostResource>
    <rasd:Parent>4</rasd:Parent>
    <rasd:AddressOnParent>0</rasd:AddressOnParent>
  </Item>
[...]
```

```
  <Item>
    <rasd:Caption>ethernet0</rasd:Caption>
    <rasd:Description>PCNet32 ethernet adapter on "Bridged"
  </rasd:Description>
    <rasd:InstanceId>8</rasd:InstanceId>
    <rasd:ResourceType>10</rasd:ResourceType>
    <rasd:ResourceSubType>PCNet32</rasd:ResourceSubType>
    <rasd:AutomaticAllocation>false</rasd:AutomaticAllocation>
    <rasd:Connection>Bridged</rasd:Connection>
    <rasd:AddressOnParent>1</rasd:AddressOnParent>
  </Item>
[...]
```

**Listing 6.1** Konfigurationsdatei WinXP.ovf

### 6.2.2 Import

Den Import starten Sie in der grafischen Oberfläche über das Menü DATEI • APPLIANCE IMPORTIEREN. Als Erstes wählen Sie die Konfigurationsdatei aus (Abbildung 6.1).



**Abbildung 6.1** Die Konfigurationsdatei auswählen

Im nächsten Schritt sehen Sie eine Zusammenfassung der Konfiguration (Abbildung 6.2). In dieser können Sie verschiedene Anpassungen vornehmen. Dabei sollten Sie jedoch beachten, dass das Gastsystem der virtuellen Maschine auf diese Hardware eingestellt wurde. Größere Änderungen führen daher eventuell zu Problemen.

Die angebotenen Einstellungen müssen hier nicht vollständig mit denen der exportierten Maschine übereinstimmen. Das Open Virtualization Format ist eine gemeinsame Grundlage zum Austausch von virtuellen Maschinen. Spezifische Einstellungen können daher »verloren« gehen. Selbst wenn Sie in VirtualBox eine Umgebung exportieren und dann wieder in VirtualBox importieren, wird es Unterschiede geben.



Abbildung 6.2 Überblick über die Konfiguration der Maschine

[+]

**Bei nicht kompatibler Konfiguration stoppt VirtualBox Importe**

Wenn die Konfiguration der virtuellen Maschine nicht mit VirtualBox kompatibel ist, kann der Import nicht gestartet werden, und es erscheint eine entsprechende Fehlermeldung.

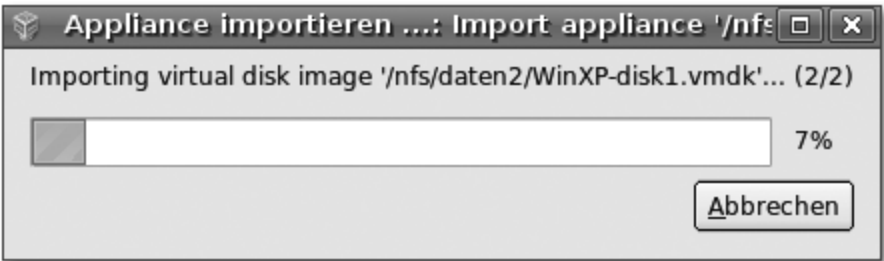


Abbildung 6.3 Nach dem Start des Imports informiert ein Dialogfenster über den Fortschritt.

6.2.3 Erster Start

Nachdem der Import abgeschlossen ist, befindet sich die Maschine in der gewohnten Auflistung der grafischen Oberfläche. Bevor Sie diese nun starten, sollten Sie einen Sicherungspunkt erstellen. Beim ersten Start wird vom Gastsystem häufig neue Hardware gefunden, und Treiber müssen eingebunden werden.

Durch den Sicherungspunkt sind Sie bei Problemen in der Lage, schnell wieder den Zustand nach dem Import herzustellen. Ein neuer Import dauert, je nach Größe des Festplatten-Images, bedeutend länger als das Verwerfen der Änderungen.

Das Verhalten der Maschine ist nun, wie in Abschnitt 6.1, »Planung«, bereits erwähnt, abhängig vom Gastsystem. Die Tastatur und Maus reagieren eventuell die ersten Minuten nicht – also warten Sie ein wenig mit einem Reset oder Zurücksetzen der Maschine. Weiterhin erkennt Windows XP kurz nach dem Start neue Hardware und verlangt nach Treibern. Ich empfehle, als Erstes die Gasterweiterungen zu installieren und das Gastsystem danach neu zu starten.

### 6.3 Migration von VMware-Maschinen

Wie erwähnt setze ich selbst gelegentlich VMware Server ein, da für VMware mehr Maschinen mit laufzeitbegrenzten Testumgebungen (Windows-Server, verschiedene Software) oder Linux-Systeme zum Download angeboten werden. Für VirtualBox gibt es leider noch sehr wenige solcher Quellen. Es bietet sich daher an, diese Virtual Appliance im VMware-Format in VirtualBox zu migrieren.

Virtual Appliances für VirtualBox beschaffen oder selbst erstellen

Prüfen Sie gegebenenfalls, ob die Umgebung auch im VirtualBox- oder im Open Virtualization Format angeboten wird. Eine weitere Möglichkeit ist der Export aus VMware heraus. Dazu benötigen Sie das kostenlose Programm VMware OVF Tool (<http://www.vmware.com/appliances/learn/ovf.html>).  
Virtual Appliances im VMware-Format finden Sie unter anderem unter <http://vmplanet.net/> und <http://www.vmware.com/appliances>.

[+]

Die weiteren Schritte zur Migration von VMware-Maschinen haben Sie bereits in verschiedenen Abschnitten kennen gelernt. Sie können nun VMDK-Festplatten-Images in eine neue virtuelle Maschine einbinden und direkt verwenden oder ein VMDK-Image in ein VDI-Festplatten-Image konvertieren (siehe Abschnitt 5.2.4, »Einbinden von fremden Festplatten-Images«, und Abschnitt 5.2.5, »Festplatten-Images konvertieren«). Wenn das Image auch vergrößert werden soll, können Sie dies wie in Abschnitt 5.2.1, »Festplatten-Image vergrößern«, beschrieben erledigen.

## 6.4 Migration eines PCs

[+]

### Ablauf abhängig von Betriebssystem

Der Ablauf einer PC-Migration hängt natürlich von dessen Betriebssystem ab. Ich gehe im Folgenden nicht ins Detail, da dies den Umfang des Buchs sprengen würde.

Bevor Sie einen physikalischen Rechner migrieren, sollten Sie die bereits in Abschnitt 6.1, »Planung«, angesprochenen Punkte überdenken:

- ▶ Lohnt sich die Migration, oder ist eine Neuinstallation sinnvoller?
- ▶ Ist spezielle Hardware notwendig?
- ▶ Sind die Ressourcen (Speicher, Festplattenspeicher) auf dem Hostsystem ausreichend?

Die Migration kann in folgenden Schritten ablaufen:

- ▶ Betriebssystem des PCs sichern.
- ▶ Eventuelle Standardtreiber installieren und gegebenenfalls das System aktualisieren (Servicepacks usw.).
- ▶ Image mit Partimage auf ein Sicherungsmedium (externe USB-Festplatte oder Ähnliches) erstellen.
- ▶ Neue virtuelle Maschine mit entsprechendem Festplatten-Image anlegen.
- ▶ USB-Festplatte bei der virtuellen Maschine einbinden.
- ▶ Die virtuelle Maschine mit der Live-CD Parted Magic starten.
- ▶ Das Festplatten-Image der virtuellen Maschine mit GParted einrichten.
- ▶ Das Sicherungs-Image mit Partimage von der USB-Festplatte auf die virtuelle Maschine übernehmen.

Verwechseln Sie das Image, das im dritten Punkt erstellt wird, nicht mit einem Festplatten-Image einer virtuellen Maschine. Dieses Image ist eine Abbilddatei des Inhalts der entsprechenden Partition.

### 6.4.1 System vorbereiten

Als Erstes sollten Sie die Daten des PCs sichern. Danach prüfen Sie, ob ein IDE-Standardtreiber vorhanden ist und spielen diesen gegebenenfalls ein, falls er fehlt. Wenn zum Beispiel ein spezieller SCSI-Controller verwendet wird und keine IDE-Treiber installiert sind, kann das System in einer virtuellen Maschine möglicherweise nicht booten. Der Treiber kann dann zwar meist nachträglich über eine Live-CD eingespielt werden – dies ist aber bedeutend umständlicher.

Windows-Systeme sollten sich am besten auf dem aktuellen Stand befinden. Gerade Windows XP unterstützt zum Beispiel ohne Servicepack keinen SATA-Controller.

### 6.4.2 Festplatten-Image erstellen

Nach dieser Vorarbeit erstellen wir eine Sicherung des PC-Festplatteninhalts. Ich verwende hierzu wieder die Live-CD Parted Magic. Diese enthält auch das Programm Partimage, mit dem Images von verschiedenen Dateisystemen erstellt werden können (siehe auch Abschnitt 6.4.5, »Alternativen zu Partimage«). Dieses Image kann dann in einer virtuellen Maschine eingespielt werden. Das Zielmedium für dieses Image könnte eine externe USB-Festplatte oder ein Ziel im Netzwerk sein. Ich gehe in diesem Beispiel von einer USB-Festplatte aus. Diese kann dann ins Dateisystem der virtuellen Maschine eingehängt werden, um die Sicherung zu übernehmen.

Mounten Sie nach dem Start zunächst die USB-Festplatte. Die zu sichernden Partitionen dürfen nicht gemountet sein. Danach starten Sie in der Konsole das Tool Partimage über den gleichnamigen Befehl `partimage`. Im ersten Schritt legen Sie fest, für welche Partition ein Sicherungs-Image erstellt werden soll (siehe Abbildung 6.4).

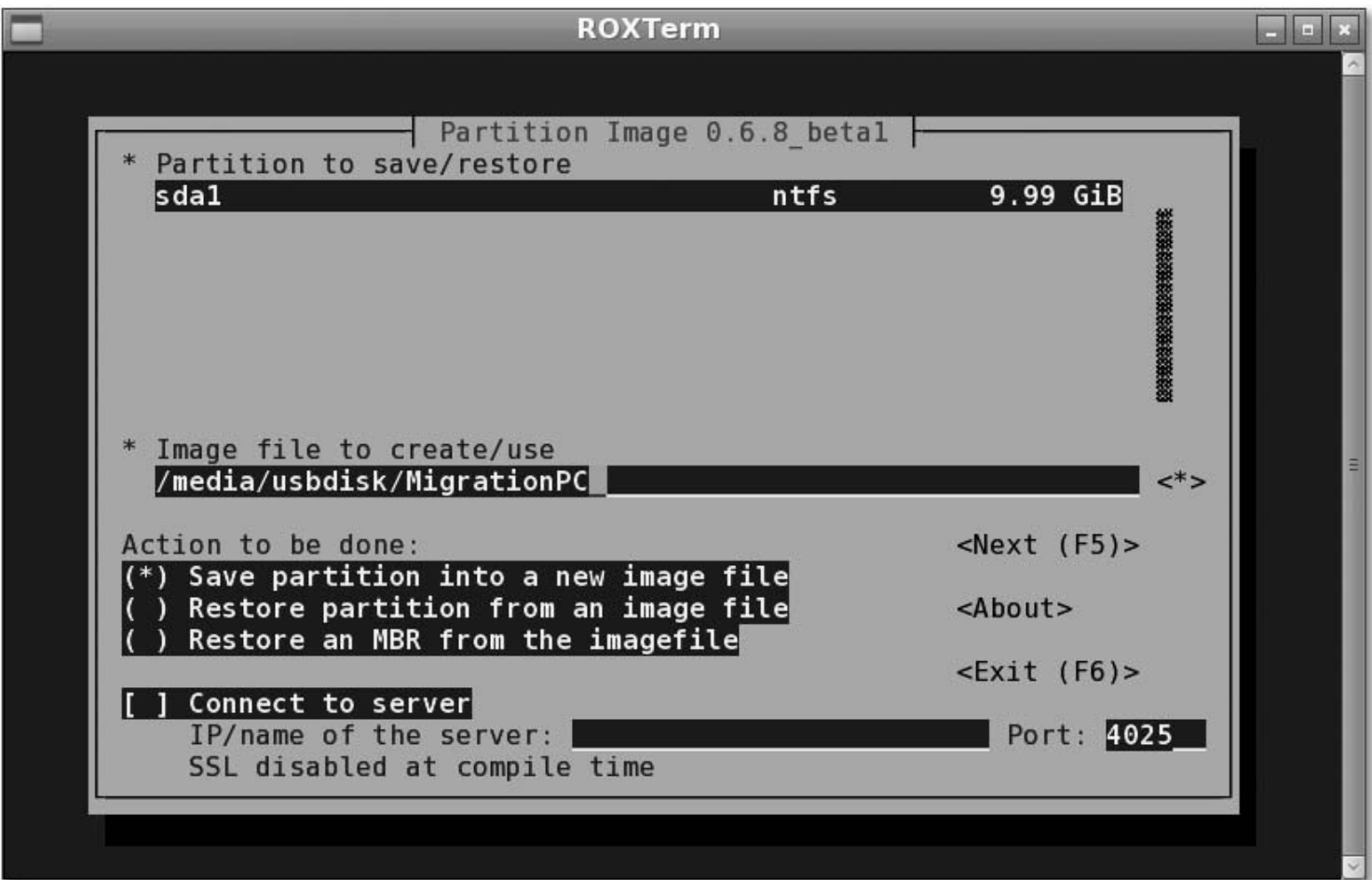


Abbildung 6.4 Partimage im Einsatz

Achten Sie darauf, dass der Punkt SAVE PARTITION INTO A NEW IMAGE FILE aktiviert und unter IMAGE FILE TO CREATE/USE der korrekte Einhängpunkt des Sicherungsziels mit einem Dateinamen für die Sicherung angegeben ist. Mehrere Partitionen sichern Sie nacheinander in verschiedenen Images.

Durch Betätigen der Taste **[F5]** gelangen Sie zum nächsten Schritt; in diesem Dialog können Sie weitere Optionen auswählen. Wenn Sie mit Partimage noch nicht vertraut sind, sollten Sie die Standardeinstellungen am besten übernehmen (Abbildung 6.5).

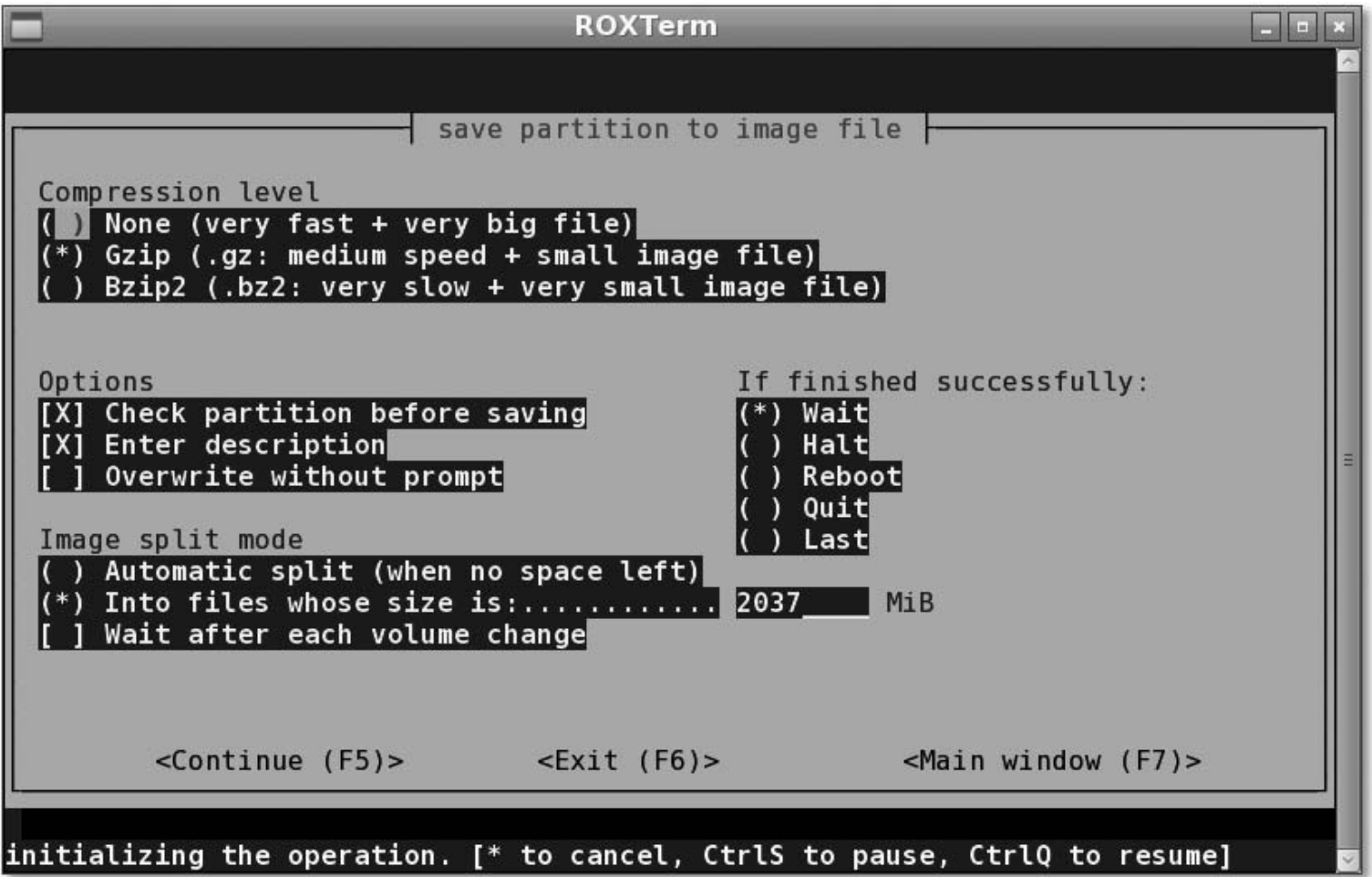


Abbildung 6.5 Weitere Partimage-Optionen

Nach einem erneuten Druck auf **[F5]** können Sie noch eine Beschreibung der Sicherung angeben. Diese Funktion sollten Sie gerade bei mehreren Sicherungen nutzen. Nach der Eingabe der Beschreibung erscheint noch eine Zusammenfassung, und das Backup wird gestartet.

[+]

NTFS-Unterstützung

Bei NTFS-Dateisystemen erscheint eine Warnung, dass die Unterstützung des Schreibzugriffs auf dieses Dateisystem sich noch im experimentellen Status befindet, was Sie an dieser Stelle getrost ignorieren können. In meinen Tests hatte ich ebenfalls keine Probleme damit.



### 6.4.3 Neue virtuelle Maschine

Als Nächstes erstellen Sie eine neue virtuelle Maschine mit den notwendigen Konfigurationseinstellungen und Festplatten-Images. Zusätzlich sollten Sie einen USB-Filter für das Sicherungsmedium einrichten und die Live-CD einbinden.

Nach dem Start von Parted Magic bearbeiten Sie dann mit GParted die notwendigen Partitionen. Diese sollten zumindest die gleiche Größe wie die des PCs besitzen. Eine Vergrößerung ist ebenfalls möglich. Nach diesem Schritt spielen Sie das Sicherungsimagen wieder mit Partimage ein. In Abbildung 6.6 sehen Sie, dass wieder eine 10-GByte-Partition erstellt wurde.

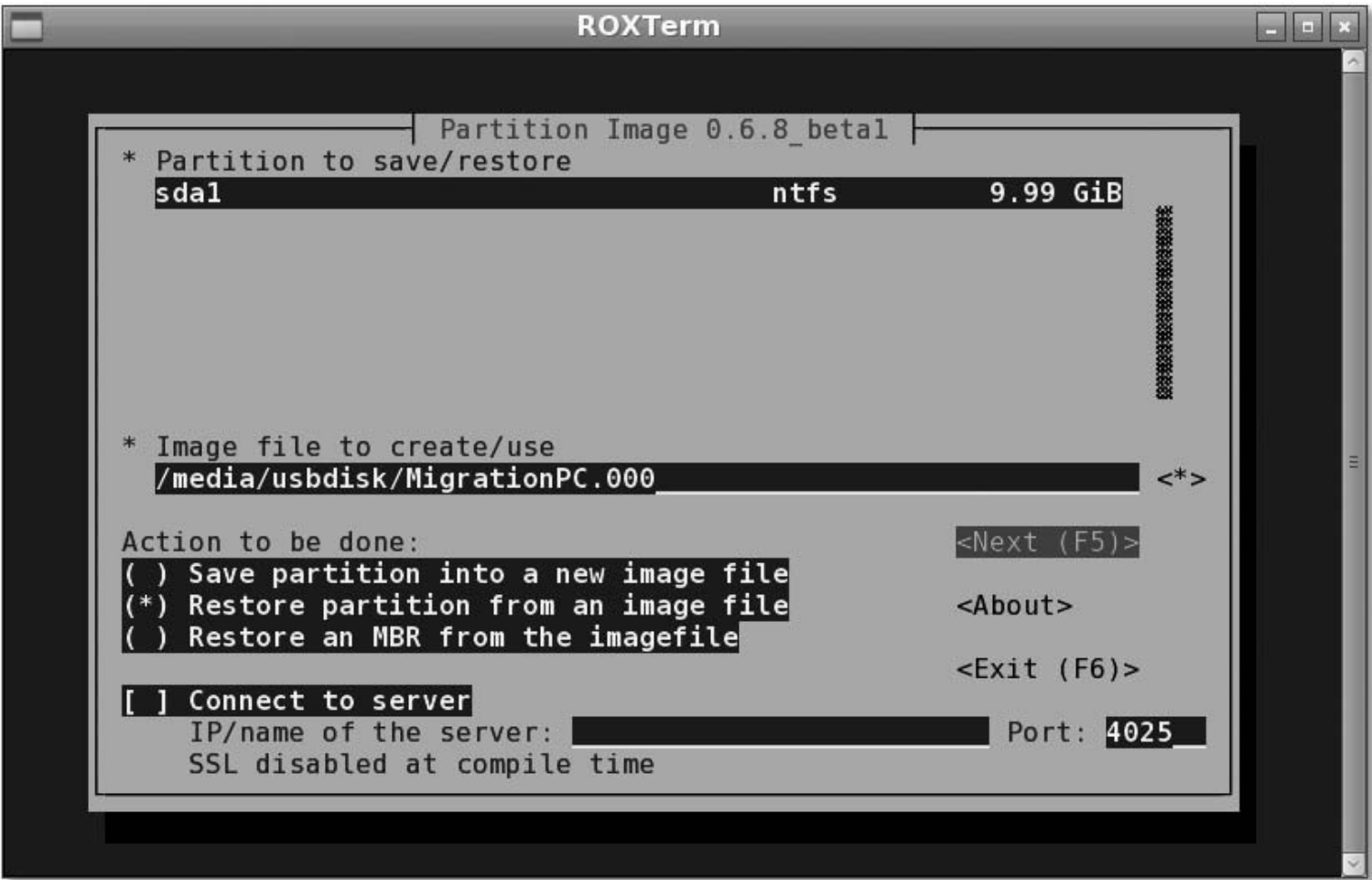


Abbildung 6.6 Ein Backup mit Partimage zurückspielen

### 6.4.4 Erster Start

Nach dem Einspielen des Sicherungs-Images können Sie das neue Gastsystem starten. Allerdings sollten Sie vorher unbedingt einen Sicherungspunkt erstellen und diesen erst nach erfolgreichem Abschluss verwerfen.

Die weitere Migration läuft dann ähnlich ab wie die einer Virtual Appliance (siehe Abschnitt 6.2.3, »Erster Start«). Ihr Gastsystem wird abhängig vom Hersteller neue Hardware erkennen, und Sie müssen die Gasterweiterungen und Treiber installieren.

### 6.4.5 Alternative zu Partimage

Statt Partimage können Sie auch das Programm `dd` auf der Kommandozeile verwenden. Mit `dd` können Sie eine komplette Festplatte in ein Image bzw. eine Rohdatei schreiben. Dieses Image konvertieren Sie dann mit dem Befehl `VboxManage convertfromraw` in eine VDI-Datei. Allerdings ist das mit `dd` erstellte Image bedeutend größer als das von Partimage, da hier auf Bit-Ebene auch freier, ungenutzter Speicher gesichert wird. Partimage komprimiert die Sicherungsdateien außerdem im ZIP-Format. Der Vorteil an `dd` ist jedoch, dass das Programm alle Partitionen einer Festplatte in »einem Rutsch« kopiert.

Auch in diesem Szenario starten Sie den PC mit Parted Magic, und Sie benötigen ein Sicherungsmedium wie zum Beispiel eine USB-Festplatte.

So erstellen Sie ein Image mit `dd`:

*Rohdatei erstellen mit der Live-CD*

```
dd if=/dev/sda of=/media/usbdisk/pc.img
```

*Auf dem Hostsystem die Rohdatei in ein Festplatten-Image konvertieren*

```
VBoxManage convertfromraw /media/usbdisk/pc.img  
    /media/virtuelle Maschinen/vdi/pc_system.vdi  
    --format VDI --variant Standard
```

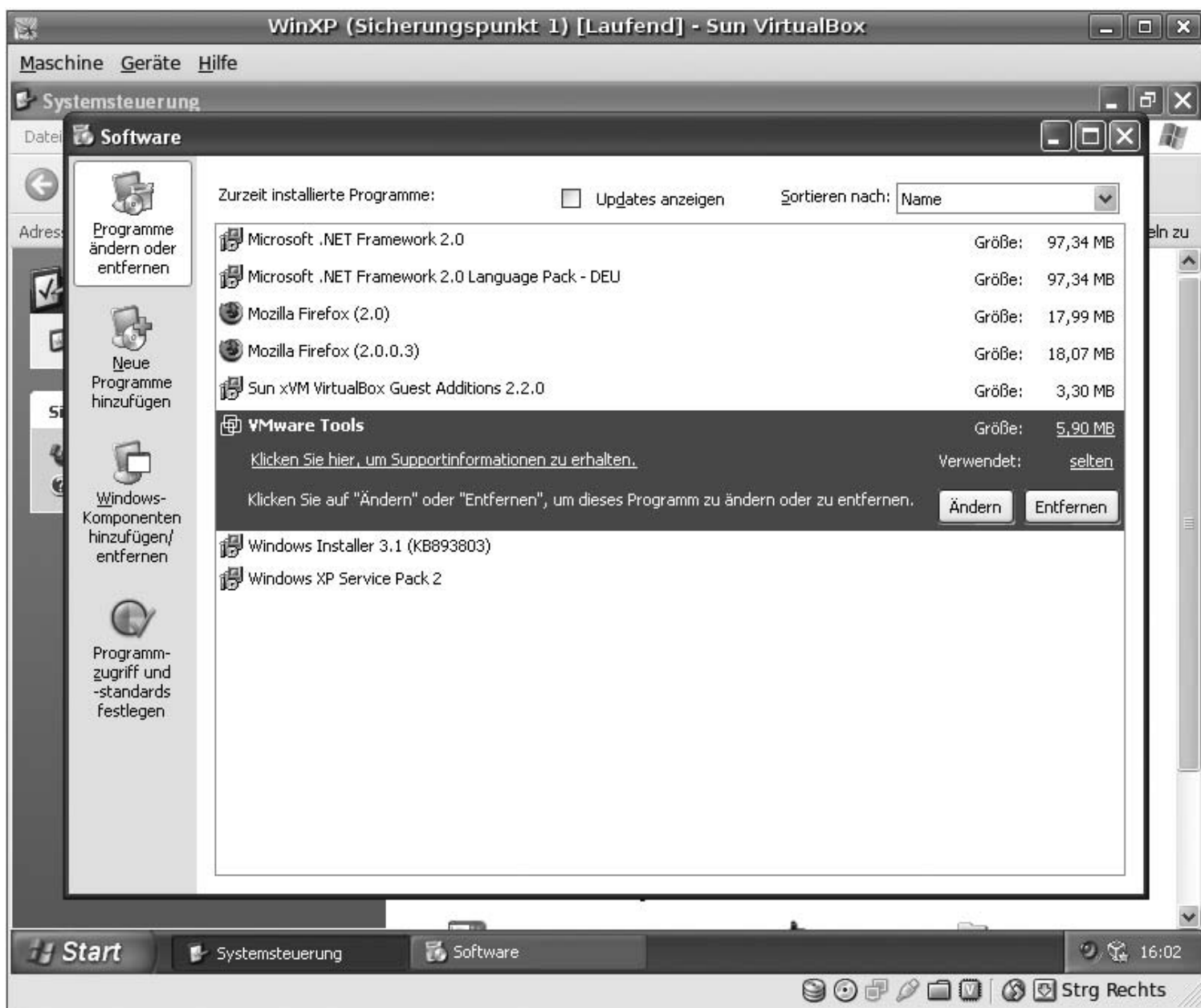
Nach dem Konvertieren erstellen Sie die entsprechende virtuelle Maschine und binden dort das Festplatten-Image ein. Danach fahren Sie wie im Abschnitt 6.4.4, »Erster Start«, fort.

## 6.5 Migrationsprobleme

Falls es bei der Migration zu Problemen kommt, finden Sie in den nächsten Abschnitten häufige Problemfälle und mögliche Lösungen. Falls Ihr Problem hier nicht aufgeführt ist, lesen Sie auch Kapitel 7, »Probleme und Fehlersuche«, oder werfen Sie einen Blick ins VirtualBox-Forum (<http://forums.virtualbox.org/>).

### 6.5.1 Gasterweiterungen des Fremdsystems stören

Wenn Sie vergessen haben, die Gasterweiterungen des Fremdsystems vor der Migration zu deinstallieren, können diese die Verwendung unter VirtualBox stören (Mausprobleme durch falsche Treiber usw.). Diese Probleme fallen meist bei Windows-Systemen an. In diesem Fall sollten Sie zunächst versuchen, die Erweiterungen über `SYSTEMSTEUERUNG • SOFTWARE` zu deinstallieren.



**Abbildung 6.7** VMware-Tools unter VirtualBox

Die nachträgliche Deinstallation der VMware-Tools hat auf meinen Testsystemen jedoch noch nie funktioniert. Eine etwas »unsaubere« Lösung wäre, den Ordner, in dem sich die VMware-Tools befinden, zu löschen und (manuell oder mit einem entsprechenden Tool) die Registrierungsdatenbank von VMware-Einträgen zu säubern.

Unter Linux-Gastsystemen ist dieses Problem weniger tragisch. Prüfen Sie hier, ob sich die entsprechenden Tools unter `/etc/init.d` befinden, und entfernen Sie diesen Eintrag (sowie gegebenenfalls die Links in den Ordnern `/etc/rcX`).

### 6.5.2 Geräte funktionieren unter Windows-Systemen nicht

Gelegentlich sind trotz Deinstallation noch Treiber der Gasterweiterungen vorhanden. Windows versucht in diesem Fall, bei gleicher Hardware (wie beispielsweise beim AMD-PCNet-Netzwerkadapter), die des Fremdsystems zu laden – und dies meist erfolglos. Aktualisieren Sie in diesem Fall die Treiber von dem CD-Ab-

bild der VirtualBox-Gasterweiterungen. Erst danach werden die Treiber von VirtualBox verwendet.

6.5.3 Volle CPU-Auslastung

Bei der Migration von Windows XP-Systemen kann es passieren, dass diese ständig 100 % CPU-Auslastung anzeigen, obwohl keine Programme laufen. In diesem Fall muss zumeist der HAL (*Hardware Abstraction Layer*) neu installiert werden.

[+]

**Vor Änderung des Hardware Abstraction Layers Sicherungspunkt erstellen**

Ein falscher HAL kann dazu führen, dass das Gastsystem nicht mehr startet. Erstellen Sie daher unbedingt vor der Änderung einen Sicherungspunkt.

Welche HAL-Version zurzeit installiert ist, erfahren Sie im *Gerätemanager* unter dem Eintrag COMPUTER. Ändern können Sie den HAL über TREIBER AKTUALISIEREN. Wählen Sie dort die Punkte SOFTWARE VON EINER LISTE... und danach NICHT SUCHEN, SONDERN DEN ZU INSTALLIERENDEN TREIBER SELBST WÄHLEN. Nun zeigt der Dialog die zur Auswahl stehenden Komponenten an (Abbildung 6.8).

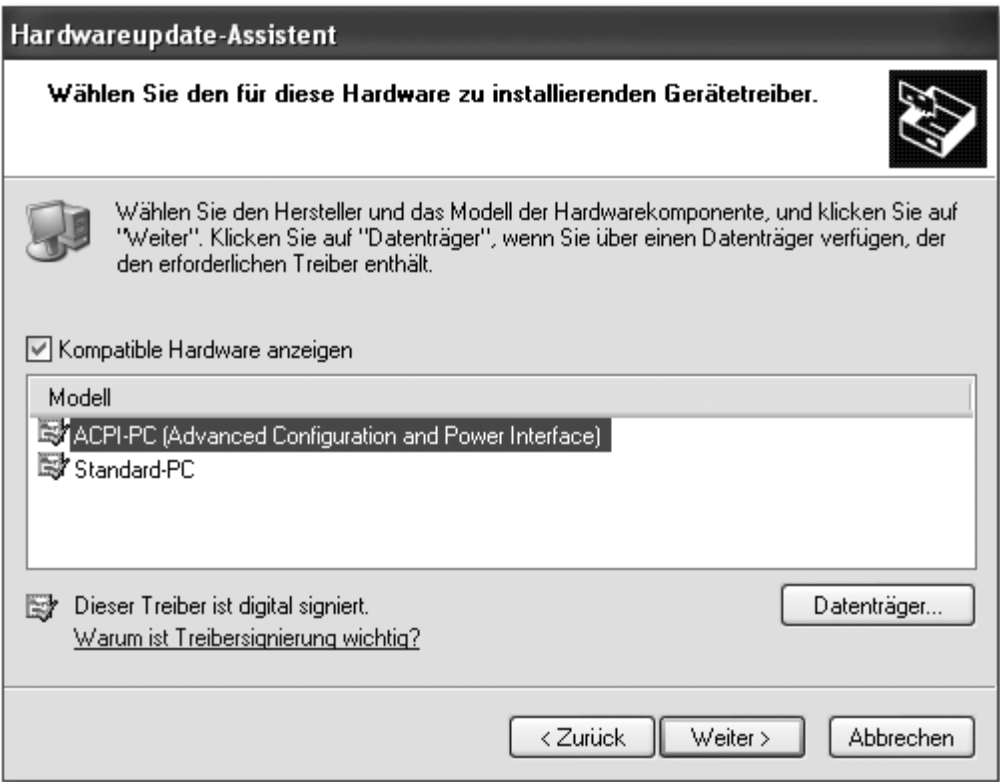


Abbildung 6.8 ACPI-PC-Treiber installieren

Das VirtualBox-Team empfiehlt ACPI-PC (ADVANCED CONFIGURATION AND POWER INTERFACE) und die Aktivierung von ACPI in der Konfiguration der virtuellen Maschine. Nach der Installation starten Sie das Gastsystem neu.

*Leider ist auch VirtualBox kein »perfektes« Programm und enthält einige Stolpersteine und Fehler. In den folgenden Abschnitten habe ich daher einige Probleme, die auf meinen Testsystemen auftraten oder auf die ich bei Recherchen gestoßen bin, zusammen mit Lösungsvorschlägen aufgelistet.*

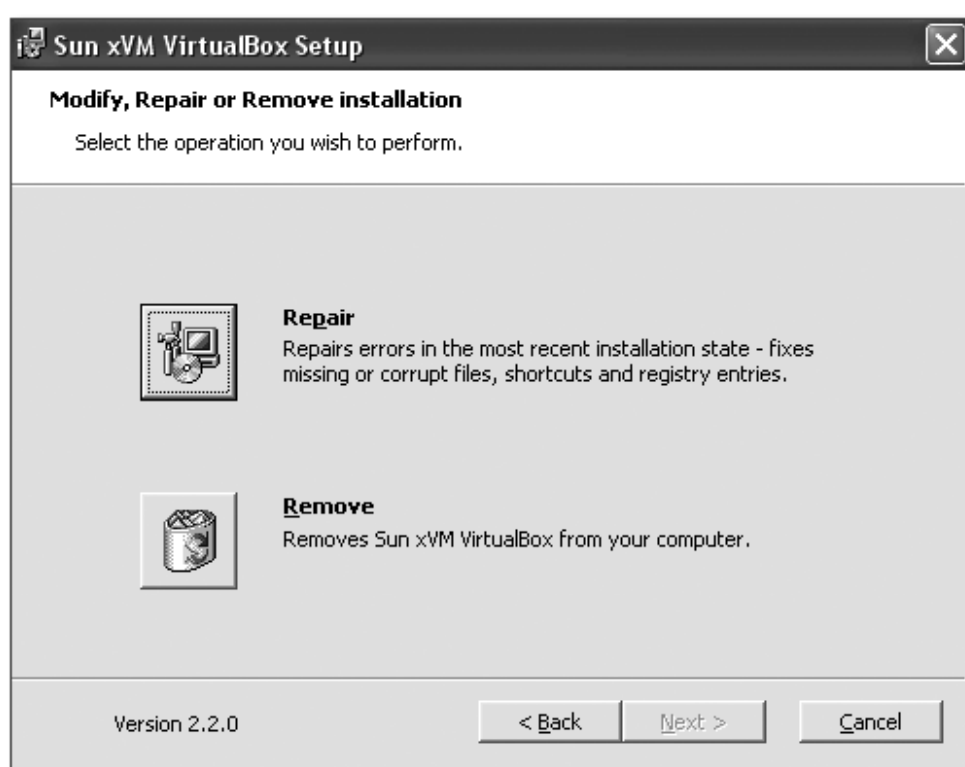
## 7 Probleme und Fehlersuche

Schauen Sie im Zweifelsfall auch immer in die englische Dokumentation von VirtualBox. Dort finden Sie stets aktuelle Problemlösungen.

### 7.1 Windows-Hostsysteme

#### Installationsprobleme

Wenn VirtualBox sich allgemein »eigenartig« verhält (virtuelle Maschinen starten nicht, die grafische Oberfläche wird nicht geöffnet oder Ähnliches) oder Sie aus Versehen wichtige VirtualBox-Dateien entfernt haben, können Sie das Programm mit der Installationsroutine reparieren (siehe Abschnitt 3.3.2, »xVM VirtualBox unter Windows«).



**Abbildung 7.1** Der Setup-Dialog unter Windows

### Kein Adapter bei Netzwerkbrücke

Wenn bei der Auswahl für den Adapter der Netzwerkbrücke keine Hardware erscheint, wurde der entsprechende Treiber nicht korrekt installiert. Dies kommt meist bei einem Update von einer vorherigen Version vor. Reparieren Sie nun, wie im vorherigen Abschnitt beschrieben, die Installation und achten Sie darauf, dass die VirtualBox-Bridged-Networking-Treiber installiert werden (Stichwort »Windows-Logo-Test«).



Abbildung 7.2 Einstellungen für VirtualBox Bridged Networking

## 7.2 Linux-Hostsysteme

### VirtualBox startet nach Kernel-Update (Systemupdate) nicht

Wenn ein neuer Kernel installiert wurde, müssen die Module von VirtualBox ebenfalls aktualisiert (bzw. neu kompiliert) werden. Wiederholen Sie in diesem Fall entweder die Installation von VirtualBox, oder kompilieren Sie die Treiber mit Root-Rechten selbst:

```
sudo /etc/init.d/vboxdrv setup
```

Gelegentlich kommt es auch zu Problemen mit der Gerätedatei `/dev/vboxdrv` (vgl. in diesem Abschnitt die Informationen unter der Überschrift »Probleme mit USB oder Rechteprobleme«).

### Falsche Tastatur bei VBoxSDL

Bei der Verwendung von VBoxSDL wird häufig die Tastatur nicht korrekt an das Gastsystem weitergereicht. Sondertasten oder deutsche Umlaute funktionieren dann nicht.

Verwenden Sie in einem solchen Fall den Parameter `--evdevkeymap` (siehe Abschnitt 8.3, »VBoxSDL«).

```
VBoxSDL --startvm LiveCD --evdevkeymap --termacpi
```

### Probleme mit USB oder Rechteprobleme

Um USB- oder auch andere Geräten zu verwenden, benötigen Sie meist besondere Zugriffsrechte. Unter Linux muss der VirtualBox-Benutzer daher Mitglied in der Gruppe `vboxusers` sein.

Im Folgenden finden Sie wichtige Angaben zum Umgang mit der Gruppe `vboxusers`:

- *Die Gruppe erstellen (falls sie bereits existiert, erscheint eine entsprechende Meldung)*

```
sudo groupadd vboxusers
```

- *Den Benutzer der Gruppe zuordnen*

```
sudo usermod -G vboxusers dirk
```

Des Weiteren muss meist noch ein Eintrag in der Datei `/etc/fstab` vorgenommen werden. Ermitteln Sie dazu die ID der neuen Gruppe `vboxusers` (125 in meinem Beispiel), und passen Sie den Eintrag in `/etc/fstab` an:

- *Die ID der Gruppe ermitteln*

```
cat /etc/group | grep vboxusers
```

Ausgabe: `vboxusers:x:125:dirk`

- */etc/fstab zum Bearbeiten öffnen*

```
sudo gedit /etc/fstab
```

- *Folgenden Eintrag anfügen*

```
# USB
```

```
none /proc/bus/usb usbfs devgid=125,devmode=664 0 0
```

Falls der Editor Gedit nicht installiert ist, verwenden Sie einen anderen Texteditor. Achten Sie darauf, das Programm mit Root-Rechten (`sudo`) zu verwenden.

Auf manchen Systemen müssen Sie zusätzlich die Rechte des `vboxdrv`-Device anpassen:

```
sudo chmod 660 /dev/vboxdrv
```

```
sudo chgrp vboxusers /dev/vboxdrv
```

## 7.3 Allgemeine Probleme

### VirtualBox startet nicht mehr

Durch Systemabstürze oder Ähnliches kann es vorkommen, dass die Konfigurationsdatei von VirtualBox zerstört wird. Eine Neuinstallation ist in diesem Fall keine Lösung, da dabei eine vorhandene Konfigurationsdatei nicht ersetzt wird.



**Abbildung 7.3** Fehlerhafte Konfigurationsdatei

Nun haben Sie verschiedene Möglichkeiten. Die erste ist natürlich das Einspielen einer (hoffentlich vorhandenen) Sicherungskopie. Als zweite Möglichkeit können Leser mit XML-Kenntnissen die Datei in einem Editor bearbeiten und fehlerhafte Einträge entfernen.

Alternativ bleibt nur das Löschen der defekten Datei. VirtualBox legt dann beim Start eine neue, leere Datei an. Dies hat jedoch zur Folge, dass Sie sämtliche virtuelle Maschinen und Festplatten-Images neu registrieren müssen.

### Virtuelle Maschine startet nach Änderungen der Einstellungen nicht

Wenn das Gastsystem einer Maschine nach Änderungen an den Einstellungen nicht mehr startet, kann dies natürlich viele Gründe haben. Daher kann ich Ihnen hier auch keine »allgemeingültige Lösung« vorstellen. Allerdings gibt es weniger Probleme, wenn Sie sich an einige grundsätzliche Regeln halten:

- ▶ Achten Sie darauf, dass immer die aktuelle Version der Gasterweiterungen installiert ist.
- ▶ Erstellen Sie vor den Änderungen einen Sicherungspunkt, um ein defektes System zu vermeiden.



- ▶ Ändern Sie nicht zu viele Einstellungen auf einmal, sondern lieber eine nach der anderen. Starten Sie zwischendurch das Gastsystem, um die Neuerungen einzurichten.
- ▶ Prüfen Sie vor einer Änderung, ob das Gastsystem die Funktion überhaupt unterstützt und aktualisieren Sie, wenn möglich, die Treiber bereits vorher.

### Linux-Gastsysteme und Klonen

Die meisten Linux-Systeme verwenden den Bootmanager GRUB. Dieser wiederum verwendet beim Booten keinen Partitionsnamen, sondern die UUIDs der Festplatten. Diese UUIDs haben nichts mit denen von VirtualBox zu tun – auch wenn sie das gleiche Format besitzen. Die Einträge dazu finden Sie in der Datei */boot/grub/menu.lst*, zum Beispiel:

```
kernel /boot/vmlinuz-2.6.27-14-generic\
  root=UUID=f6501175-0e40-4a97-9a09-f9b28fe775c0\
  ro quiet splash
```

Wenn nun die Systemfestplatte geklont wurde, erhält diese auch vom Gastsystem eine neue UUID. Daher müssen Sie in diesem Fall die Datei *menu.lst* anpassen. Hierzu booten Sie die Maschine mit Parted Magic und ermitteln die neue UUID mit dem Kommandozeilentool *blkid*:

Ausgabe *blkid*:

```
/dev/sda1: LABEL="system" \
  UUID="f6501175-0e40-4a97-9a09-f9b28fe775c0" TYPE="ext3"
/dev/sda5: LABEL="swap" \
  UUID="6fa1f1cb-f408-4f04-943c-4dc5bbd9323d" TYPE="swap"
```

Danach ändern Sie mit einem Editor die entsprechenden Einträge im Bootmenü (*/boot/grub/menu.lst*). Meist ist *sda1* oder *hda1* die Bootpartition.

Auf die gleiche Weise passen Sie auch Einträge in der Datei */etc/fstab* an. Diese Datei legt die Einhängepunkte (Mount Points) der einzelnen Partitionen und Dateisysteme fest. Auch hier werden neuerdings die UUIDs verwendet.

### CPU-Temperatur

Es kann vorkommen, dass die CPU-Temperatur des Hostsystems zu hoch ist, obwohl das Gastsystem keine hohe Aktivität zeigt. Dies liegt meist an fehlenden Gasterweiterungen. Installieren Sie diese, und das Problem ist in der Regel behoben.

Sicherungspunkt kann nicht entfernt werden

Wenn Sie nach einem Sicherungspunkt neue Festplatten-Images hinzugefügt und wieder entfernt haben, kann es zu Problemen kommen, wenn dieser Sicherungspunkt auch entfernt werden soll. Es kann vorkommen, dass mehr als eine Sicherungsdatei vorhanden ist (siehe Abbildung 7.4).



Abbildung 7.4 Fehler beim Entfernen eines Snapshots («has more than one child hard disk«)

Dieser Fehler ist bei mir schon öfter aufgetreten. In diesem Beispiel hat die Maschine zwei Sicherungspunkte. Der zweite Sicherungspunkt hat plötzlich zwei Sicherungsdateien (*child hard disks*). Um das Problem zu lösen, entfernen Sie nun das überflüssige Image im Manager für virtuelle Medien.

In Abbildung 7.5 sehen Sie das Festplatten-Image *vmWinXP.vdi* und die zwei Ebenen der Sicherungspunkte. Die zweite Ebene enthält dabei zwei Festplatten-Images. Das erste Abbild besitzt den Status *nicht angeschlossen* und kann entfernt werden.

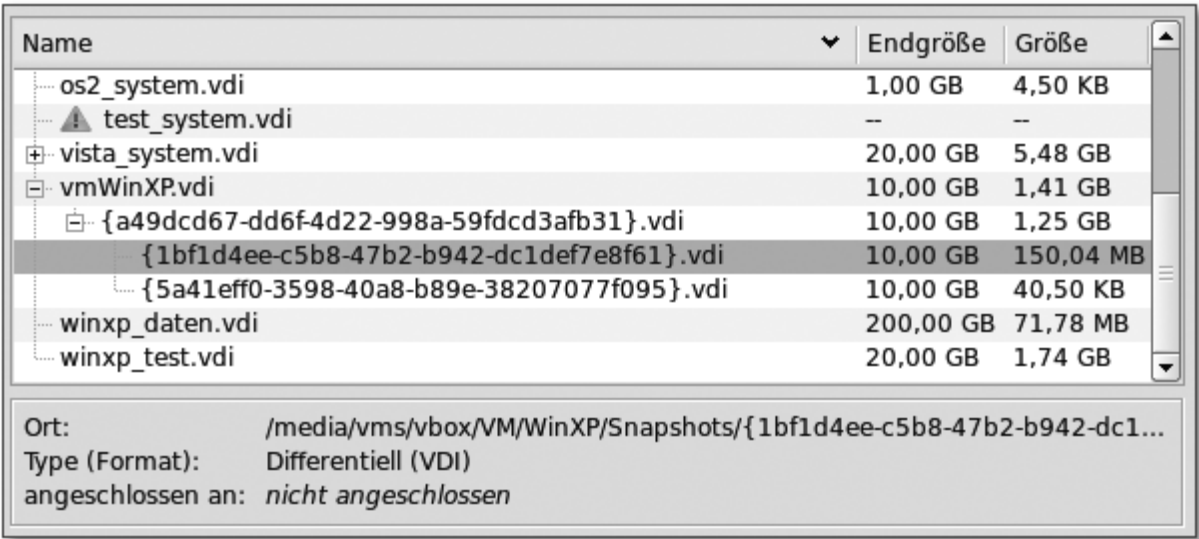
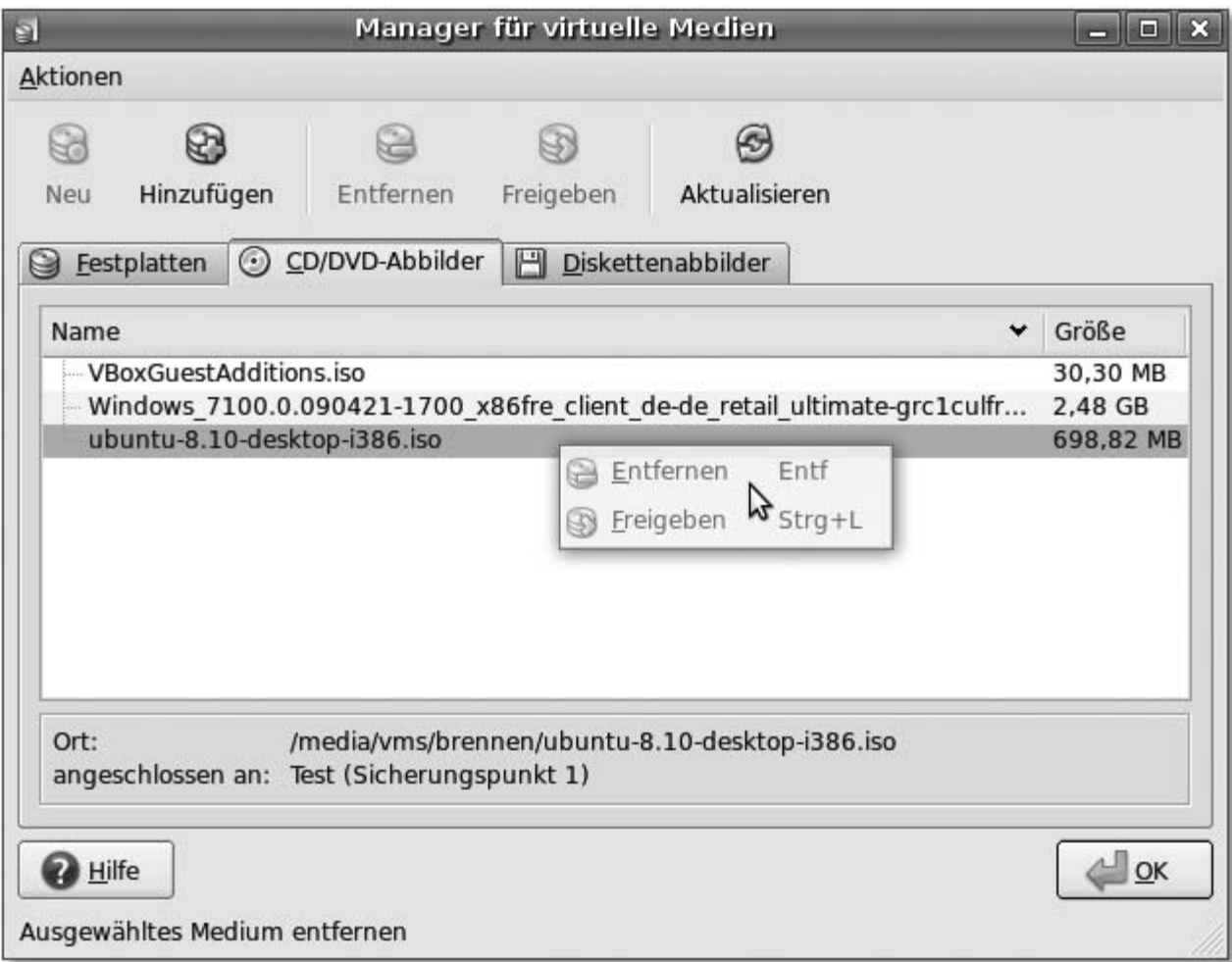


Abbildung 7.5 Nicht angeschlossenenes Image

Danach können Sie auch Sicherungspunkt 2 problemlos entfernen.

**Image kann nicht entfernt werden**

Bietet der Manager für virtuelle Medien beim Entfernen von CD/DVD-Abbildern oder Festplatten-Images die Funktionen ENTFERNEN oder FREIGEBEN nicht an, ist das Image gesperrt (siehe Abbildung 7.6).



**Abbildung 7.6** Ein gesperrtes Image

Des Rätsels Lösung ist ganz einfach: Das Image ist in einem Sicherungspunkt enthalten. In diesem Fall werden Images gesperrt, denn wenn sie entfernt würden, könnte eine virtuelle Maschine nicht zu diesem Sicherungspunkt zurückkehren. Belassen Sie das Image einfach im Manager für virtuelle Medien, oder verwerfen Sie den im Infobereich angezeigten Sicherungspunkt.

**Nahtloser Modus kann nicht zurückgesetzt werden**

Gelegentlich hat man den Eindruck, dass die Tastenkombination Host-Taste + L den nahtlosen Modus nur sporadisch zurücksetzt und nicht immer reagiert. Dies liegt einfach daran, dass dazu ein Objekt des Gastsystems den Fokus besitzen muss. Wenn dagegen ein Fenster des Hostsystems diesen hat, wird die Tastenkombination auch an dieses Fenster gesendet, und VirtualBox bekommt davon nichts mit.

**Nahtloser Modus und 3-D-Beschleunigung**

Wenn Sie die 3-D-Beschleunigung aktiviert haben, kommt es zu Problemen mit dem nahtlosen Modus. Sie müssen sich hier leider für eine der beiden Funktionen entscheiden.

Wollen Sie den nahtlosen Modus verwenden, müssen Sie die 3-D-Beschleunigung deaktivieren, und wenn Sie Wert auf 3-D legen, müssen Sie leider auf den nahtlosen Modus verzichten. Beides zusammen funktioniert (zurzeit) leider nicht korrekt.

**Im nahtlosen Modus wird der Desktop des Hostsystems nicht angezeigt**

Unter Windows XP, Vista und Windows 7 als Gastssysteme hatte ich Anzeigeprobleme, wenn keine Anwendung geöffnet war. Der Desktop des Hostsystems wurde dann nicht korrekt dargestellt. Öffnen Sie in diesem Fall eine Anwendung im Gastsystem (zum Beispiel den Explorer), und die Anzeige ist wieder in Ordnung.

**Grafikprobleme mit Windows-Gastsystemen**

Gelegentlich kommt es nach der Aktualisierung der Gasterweiterungen und dem damit verbundenen Treiberupdate vor, dass Windows-Gastsysteme keine korrekte Auflösung mehr und nur noch 4-Bit Farbtiefe haben. In der Systemsteuerung wird jedoch der korrekte Treiber angezeigt. Neustarts und andere Änderungen bringen keine Lösung.

In diesem Fall ändern Sie einfach die Größe des Fensters der virtuellen Maschine. Dabei wird die virtuelle Grafikkarte neu initialisiert, und das Problem ist verschwunden.

**Das Gastsystem kann keine CD/DVD brennen**

Sie haben bei der Konfiguration die Option PASSTHROUGH nicht eingeschaltet. Erst durch diese Aktivierung erhält das Gastsystem die Möglichkeit, die entsprechenden Befehle an das CD/DVD-Laufwerk zu senden.



**Abbildung 7.7** Aktivieren Sie »Passthrough«, um CD/DVD-Laufwerke des Hostsystems im Gast zu nutzen.

### SATA-Festplatte wird nicht erkannt

Wenn Ihr Gastsystem das Image der SATA-Festplatte nicht erkennt, kann dies daran liegen, dass das System noch kein SATA unterstützt. Entweder muss dann der entsprechende Intel-Treiber installiert werden, oder der SATA-Controller wird auf IDE-kompatibel geschaltet. Letztere Vorgehensweise ist sinnvoll bei der Installation des Gastsystems.

```
VBoxManage modifyvm WinXP --sataideemulation 1 3
```

Im Listing steht die 1 für die Nummer des Controllers und die 3 für die Anschlussnummer des Festplatten-Images. Nach der Installation können Sie sich dann den aktuellen Treiber von <http://www.intel.com/products/chipsets> herunterladen und installieren (siehe auch Abschnitt 3.8, »Hardware«).

### Windows XP auf SATA umstellen

Wenn Sie Ihr Windows-XP-Gastsystem nachträglich auf SATA umstellen wollen, gehen Sie wie folgt vor:

- ▶ Laden Sie den Treiber von <http://www.intel.com/products/chipsets> herunter.
- ▶ Aktivieren Sie SATA, aber belassen Sie die Festplatten-Images noch an IDE.

- ▶ Starten Sie die Maschine, und installieren Sie die Treiber.
- ▶ Beenden Sie die Maschine, und stellen Sie die Festplatten-Images auf SATA um (eventuell auch den IDE-Controller auf ICH6).

*VirtualBox kann sehr effektiv über die Kommandozeile gesteuert werden. Einige Funktionen stehen lediglich so zur Verfügung. Lesen Sie hier, welche das sind und wie sie bedient werden.*

## 8 Kommandozeilentools

Die Kommandozeilentools verwenden viele Befehle (auch Optionen genannt) zum Konfigurieren der Maschinen. Der einfache Aufruf des Tools ohne weitere Befehle listet diese alle auf (siehe auch Abschnitt 3.6.1, »VBoxManage«). Diese Befehle wiederum benötigen häufig einen oder mehrere Parameter. Viele Parameter sind hierbei optional, oder es kann nur einer der möglichen übergeben werden. Bei der Hilfeanzeige verwendet VirtualBox ein Format, das auch bei Linux-Befehlen zum Einsatz kommt. Der senkrechte Strich (<uuid>|<name>) zwischen Befehlen oder Parametern bedeutet, dass entweder nur der eine oder der andere verwendet werden kann. Ein in spitze Klammern eingebetteter Parameter (<uuid>) bedeutet stattdessen, dass es sich nur um einen Platzhalter handelt, den Sie durch den entsprechenden Wert ersetzen. Parameter in eckigen Klammern sind dagegen optional – sie müssen nicht verwendet werden.

Beispiel:

```
VBoxManage unregistervm <uuid>|<name> [--delete]
```

Dabei ist der Aufbau wie folgt:

- ▶ **VBoxManage:** das Kommandozeilentool
- ▶ **unregistervm:** der Befehl
- ▶ **<uuid>|<name>:** die möglichen Parameter, also entweder eine UUID oder der Name in diesem Fall – nicht beides zusammen
- ▶ **[--delete]:** ein weiterer Parameter, der jedoch optional ist

Den Befehl können Sie also auf verschiedene Weisen verwenden. Ersetzen Sie beispielsweise <uuid> durch die UUI der virtuellen Maschine:

```
VBoxManage unregistervm 626160c1-58f6-4d49-b1a9-909ab5c0232d
```

Alternativ geben Sie den Namen (<name>) an, zum Beispiel:

```
VBoxManage unregistervm LiveCD
```

Den optionalen Parameter setzen Sie beispielsweise so ein:

```
VBoxManage unregistervm LiveCD --delete
```

oder

```
VBoxManage unregistervm 626160c1-58f6-4d49-b1a9-909ab5c0232d\
--delete
```

Die meisten Parameter besitzen eine Kurz- und Langform. Die Kurzform beginnt mit einem und die Langform mit zwei Minuszeichen.

- ▶ *Langform:*  
VBoxManage **--nologo** list **--long** runningvms
- ▶ *Kurzform:*  
VBoxManage **-q** list **-l** runningvms

[+]

#### Hinweis zur richtigen Schreibweise für Kommandozeilentools

Bei Kommandozeilentools unter Linux ist diese Schreibweise üblich (ein Minus für die Kurzform, zwei Minuszeichen für die Langform). VirtualBox akzeptiert häufig jedoch auch ein einzelnes Minuszeichen beim Langformat. Dies ist aber nur noch aus Gründen der Abwärtskompatibilität möglich und sollte nicht mehr angewandt werden.

Wenn ich in den Beispielen einen Backslash am Zeilenende verwende, bedeutet dies, dass die folgende Zeile und die aktuelle als eine einzige Zeile eingegeben werden muss:

```
VBoxManage unregistervm\
626160c1-58f6-4d49-b1a9-909ab5c0232d
```

ist gleich

```
VBoxManage unregistervm 626160c1-58f6-4d49-b1a9-909ab5c0232d
```

Ich verwende bei den folgenden Beschreibungen dieselbe Reihenfolge wie die Ausgabe der Hilfe (zum Beispiel `VboxManage --help`).

[+]

#### Änderungen bei künftigen VirtualBox-Versionen berücksichtigen

Die folgenden Beschreibungen beziehen sich auf VirtualBox 3.0. Achten Sie unbedingt auf Änderungen bei neueren Versionen!

## 8.1 VirtualBox

Normalerweise startet der Befehl `VirtualBox` in der Eingabeaufforderung bzw. Konsole »nur« die grafische Oberfläche. Allerdings bietet auch dieser Befehl einige weitere Parameter.



**--startvm <vmname|UUID>**

Startet eine virtuelle Maschine, ohne die grafische Oberfläche zu öffnen.

**Parameter für --startvm**

Die folgenden Parameter können nur zusammen mit --startvm verwendet werden.

Parameter	Funktion des Parameters
--dbg	Aktiviert das Menü DEBUG.
--debug	Aktiviert das Menü DEBUG und öffnet sofort die Debug-Ausgabe.
--debug-command-line	Aktiviert die Debug-Ausgabe in der Kommandozeile.
--no-debug	Deaktiviert das Debug-Menü und die -Ausgabe.

**Tabelle 8.1** Parameter zur ausschließlichen Verwendung mit »--startvm«

Beispiel für die Verwendung des Parameters --dbg:

```
VBoxManage --startvm LiveCD --dbg
```

## 8.2 VBoxManage

VBoxManage ist wohl das wichtigste Tool zum Konfigurieren und Verwalten von virtuellen Maschinen und VirtualBox. VBoxManage bietet auch Befehle und Einstellungsmöglichkeiten, die in der grafischen Oberfläche nicht vorhanden sind.

**-v|--version**

Gibt die Informationen zur installierten VirtualBox-Version aus.

Beispiel:

- *Kurzform:*  
VBoxManage -v
- *Langform:*  
VBoxManage --version

**-q|--nologo**

Blendet die VirtualBox- und Sun-Informationen bei der Ausgabe aus.

Beispiel:

```
dirk@t61:~$ VBoxManage list vms
VirtualBox Command Line Management Interface Version 2.2.4
(C) 2005-2009 Sun Microsystems, Inc.
```

```
All rights reserved.
"LiveCD" {07a4350c-2285-478a-489b-46949ce3d970}
"WinXP" {626160c1-58f6-4d49-b1a9-909ab5c0232d}
"Test" {b3fc68ea-09d3-4d26-b872-ae4823bcd3b2}
dirk@t61:~$ VBoxManage --nologo list vms
"LiveCD" {07a4350c-2285-478a-489b-46949ce3d970}
"WinXP" {626160c1-58f6-4d49-b1a9-909ab5c0232d}
"Test" {b3fc68ea-09d3-4d26-b872-ae4823bcd3b2}
```

list

```
list [--long|-l]
    vms|runningvms|ostypes|hostdvds|hostfloppies|
    bridgedifs|hostonlyifs|dhcpservers|hostinfo|
    hddbackends|hdds|dvds|floppies|
    usbhost|usbfilters|systemproperties
```

Der Befehl `list` muss in Verbindung mit einem weiteren Parameter aufgerufen werden. Er gibt dann entsprechende Informationen zurück. Standardmäßig ist die Ausgabe »gekürzt«. Mit dem Parameter `--long` erfolgt eine ausführliche Anzeige.

Parameter	Funktion des Parameters
vms	Gibt die registrierten virtuellen Maschinen aus.
runningvms	Gibt die UUIDs der aktuell gestarteten virtuellen Maschinen aus.
ostypes	Listet alle unterstützten Gastsysteme auf.
hostdvds	Zeigt die CD/DVD-Laufwerke des Hostsystems an.
hostfloppies	Zeigt die Diskettenlaufwerke des Hostsystems an.
bridgedifs	Zeigt die Netzwerkadapter des Hostsystems an, die als Netzwerkbrücke verwendet werden können.
hostonlyifs	Zeigt die Adapter für Host-only Netzwerke an.
dhcpservers	Zeigt die DHCP-Einstellungen für Host-only Netzwerke an.
hostinfo	Gibt Informationen über die CPUs des Hosts zurück.
hddbackends	Listet Informationen zur Einbindung von Datenträgern auf (Festplatten-Images von VirtualBox oder VMware, iSCSI usw.).
hdds	Zeigt die registrierten Festplatten-Images mit ihrer UUID an.
dvds	Wie hdds, gibt aber die CD- bzw. DVD-Images zurück.
floppies	Ebenfalls wie hdds, zeigt aber registrierte Disketten-Images an.
usbhost	Gibt Informationen zu eingesteckten USB-Geräten zurück (zum Beispiel die UUID, Product-Id usw.).

Tabelle 8.2 Parameter zur Verwendung mit dem Befehl »list«

Parameter	Funktion des Parameters
usbfilters	Zeigt selbstdefinierte USB-Filter an.
systemproperties	Listet verschiedene Systemeinstellungen auf.

**Tabelle 8.2** Parameter zur Verwendung mit dem Befehl »list« (Forts.)

Einige dieser Parameter liefern seitenweise Informationen, die aus der Eingabeaufforderung bzw. der Konsole herausscrollen. Wenn Sie die Ausgabe über ein Pipe-Zeichen (|) an das Programm `more` (Windows und Linux) oder `less` (Linux) übergeben, können Sie in der Ausgabe blättern.

**Beispiel:**

```
VBoxManage list vms | more
VBoxManage list ostypes | less
VBoxManage -q list hostinfo
```

Host Information:

```
Host time: 2008-09-14T11:01:31.801000000
Processor online count: 4
Processor count: 4
Processor#0 speed: 3065 MHz
Processor#1 speed: 3065 MHz
Processor#2 speed: 3065 MHz
Processor#3 speed: 3065 MHz
```

**showvminfo**

```
showvminfo <uuid>|<name> [--details] [--statistics]
                [--machinereadable]
```

Der Befehl `showvminfo` gibt Details über die Konfiguration einer virtuellen Maschine zurück.

Parameter	Funktion des Parameters
--details	Gibt die Informationen im Detail zurück (Standard).
--statistics	Die gleiche Ausgabe wie bei <code>details</code> , aber mit einer Statistik ( <i>guest statistics</i> ).
--machinereadable	Die Ausgabe erfolgt in einem anderen Format.

**Tabelle 8.3** Parameter für den Befehl »showvminfo«

**registervm <filename>**

Diese Option registriert eine virtuelle Maschine. Sie übergeben zusätzlich den Dateinamen der Konfigurationsdatei mit vollem Pfad. Die Konfigurationsdatei darf keine registrierten Festplatten-Images, CD/DVD-Abbilder oder Sicherungspunkte enthalten.

Beispiel:

```
VboxManage registervm D:\vbox\vmtest\vmtest01.xml
```

**unregistervm <uuid>|<name> [--delete]**

Entfernt eine virtuelle Maschine aus der Registrierung, ohne sie zu löschen. Wenn der optionale Parameter `--delete` mit übergeben wird, löscht VBoxManage auch die Dateien.

**createvm**

```
createvm --name <name>
        [--ostype <ostype>]
        [--register]
        [--basefolder <path> | --settingsfile <path>]
        [--uuid <uuid>]
```

Erstellt eine neue virtuelle Maschine. Der Parameter `--name` ist dabei eine Pflichtangabe, die weiteren sind optional.

Parameter	Funktion des Parameters
--name	Name der Maschine
--ostype	Typ des Gastsystems (siehe auch <code>VBoxManage list ostypes</code> )
--register	Registriert die virtuelle Maschine nach der Erstellung.
--basefolder	Hauptordner der virtuellen Maschine
--settingsfile	Pfad zur Konfigurationsdatei, nicht in Verbindung mit <code>--basefolder</code> verwendbar
--uuid	Die zu verwendende UUID (diese wird normalerweise automatisch vergeben und muss unbedingt eindeutig sein)

**Tabelle 8.4** Parameter für den Befehl »createvm«

Beispiel:

```
VboxManage --name LiveCD --ostype Linux64 --register
```

## modifyvm

```

modifyvm <uuid|name> [--name <name>] [--ostype <ostype>]
    [--memory <memorysize in MB>] [--vram <vramsize in MB>] [--acpi on|off]
    [--ioapic on|off] [--pae on|off] [--hwvrtex on|off|default] [--nestedpaging on|off]
    [--vtxvpid on|off] [--cpus <number>]
    [--monitorcount <number>] [--accelerate3d <on|off>]
    [--bioslogofadein on|off] [--bioslogofadeout on|off]
    [--bioslogodisplaytime <msec>] [--bioslogoimagepath <imagepath>]
    [--biosbootmenu disabled|menuonly|messageandmenu]
    [--biossystemtimeoffset <msec>] [--biospxedebg on|off]
    [--boot<1-4> none|floppy|dvd|disk|net>] [--hd<a|b|d> none|<uuid>|<filename>]
    [--idecontroller PIIX3|PIIX4] [--sata on|off] [--sataportcount <1-30>]
    [--sataport<1-30> none|<uuid>|<filename>] [--sataideemulation<1-4> <1-30>]
    [--scsi on|off] [--scsiport<1-16> none|<uuid>|<filename>]
    [--scsitype LsiLogic|BusLogic] [--dvd none|<uuid>|<filename>|host:<drive>]
    [--dvdpassthrough on|off]
    [--floppy disabled|empty|<uuid>|<filename>|host:<drive>]
    [--nic<1-N> none|null|nat|bridged|intnet|hostonly]
    [--nictype<1-N> Am79C970A|Am79C973|82540EM|82543GC|82545EM]
    [--cableconnected<1-N> on|off] [--nictrace<1-N> on|off]
    [--nictracefile<1-N> <filename>] [--nicspeed<1-N> <kbps>]
    [--bridgeadapter<1-N> none|<devicename>]
    [--hostonlyadapter<1-N> none|<devicename>] [--intnet<1-N> <network name>]
    [--natnet<1-N> <network>|default] [--macaddress<1-N> auto|<mac>]
    [--uart<1-N> off|<I/O base> <IRQ>]
    [--uartmode<1-N> disconnected|server <pipe>|client <pipe>|
        file <file>|<devicename>]
    [--gueststatisticsinterval <seconds>]
    [--audio none|null|oss|alsa|pulse] [--audiocontroller ac97|sb16]
    [--clipboard disabled|hosttoguest|guesttohost|bidirectional]
    [--vrdp on|off] [--vrdpport default|<port>] [--vrdpaddress <host>]
    [--vrdpauthtype null|external|guest] [--vrdpmulticon on|off]
    [--vrdpreusecon on|off] [--usb on|off] [--usbehci on|off]
    [--snapshotfolder default|<path>]

```

Mit `modifyvm` nehmen Sie Änderungen an der Konfiguration der virtuellen Maschinen vor. Die meisten Parameter dürften selbsterklärend sein. Achten Sie auf die korrekte Schreibweise einiger Optionen. So ist zum Beispiel bei `--hd<a|b|d>` absichtlich kein Leerzeichen vorhanden, da auch keines verwendet werden darf. Die richtige Schreibweise ist also `--hda` oder `--hdb` usw.

Sie können mit `modifyvm` auch Optionen setzen, die in der grafischen Oberfläche nicht vorhanden sind.

Parameter	Funktion des Parameters
--name	Ändert den Namen der Maschine.
--ostype	Setzt den Typ des Gastsystems (siehe auch <code>VBoxManage list ostypes</code> ).
--cpus	Bestimmt die Anzahl der Prozessoren.
--monitorcount	Setzt die Anzahl der virtuellen Monitore.
--bioslogodisplaytime <msec>	Setzt die Anzeigedauer des BIOS-Logos in Millisekunden.
--idecontroller PIIX3 PIIX4	Legt den zu verwendenden IDE-Controller fest.
--scsi on off	Aktiviert (on) oder deaktiviert (off) den SCSI-Controller.
--scsitype LsiLogic BusLogic	Setzt den Adapter-Typ des SCSI-Controllers.
--clipboard disabled disabled hosttoguest guesttohost bidirectional	Setzt das Verhalten der Zwischenablage: kein Datenaustausch, nur vom Host- zum Gastsystem, vom Gast- zum Hostsystem, Datenaustausch in beide Richtungen.
--vrdpmulticon on off	Erlaubt (on) mehrere RDP-Zugriffe gleichzeitig oder verhindert sie (off).

Tabelle 8.5 Parameter für den Befehl »modifyvm«

Beispiel:

```
VBoxManage modifyvm WinXP \
    --hda 626160c1-58f6-4d49-b1a9-909ab5c0232d
VBoxManage modifyvm WinXP --name InternetPC\
    --vrdp on --vrdpport 3400
```

**import <ovf> [--dry-run|-n] [more options]**

Importiert eine Virtual Appliance im Open Virtualization Format. Dieser Import ist sehr umfangreich (more options). Ich empfehle Ihnen, hierzu die grafische Oberfläche zu verwenden (siehe Abschnitt 6.2, »Open Virtualization Format«).

**export <machines>**

```
export <machines> --output|-o <ovf>
    [--legacy09]
    [--vsys <number of virtual system>]
        [--product <product name>] [--producturl <product url>]
        [--vendor <vendor name>] [--vendorurl <vendor url>]
```

```
[--version <version info>]
[--eula <license text>] [--eulafile <filename>]
```

Exportiert eine virtuelle Maschine im Open Virtualization Format. Der Export ist im Gegensatz zum Import eine einfache Angelegenheit. Sie geben nur die virtuelle Maschine und das Ziel an.

Parameter	Funktion des Parameters
--output	Zieldatei (am besten mit der Endung .ovf).
--legacy09	Erstellt die Datei in einem vorherigen Format.

Tabelle 8.6 Parameter für den Befehl »export«

Die weiteren optionalen Parameter übergeben Produkt- und Versionsinformationen und gegebenenfalls Lizenzverträge. Sie dürften selbsterklärend sein.

Beispiel:

```
VBoxManage export WindowsXP --output /media/daten/WindowsXP.ovf
VBoxManage export WindowsXP --output E:\WindowsXP.ovf\
--legacy09
```

**startvm <uuid>|<name> [--type gui|vrdp|headless]**

Der Befehl startet eine virtuelle Maschine. Mit der optionalen Angabe --type geben Sie an, ob die Maschine mit einer grafischen Oberfläche (gui) oder RDP-Zugriff (vrdp) gestartet werden soll.

Hinweis

Statt der Verwendung von --type vrdp sollten Sie das Programm VBoxHeadless einsetzen (siehe Abschnitt 8.4, »VBoxHeadless«).

[+]

Beispiel:

```
VBoxManage startvm WindowsXP
```

**controlvm**

```
controlvm <uuid>|<name>
    pause|resume|reset|poweroff|savestate|
    acpipowerbutton|acpisleepbutton|
    keyboardputscancode <hex> [<hex> ...]|
    injectnmi|
    setlinkstate<1-N> on|off |
    usbattach <uuid>|<address> | usbdetach <uuid>|<address> |
    dvdattach none|<uuid>|<filename>|host:<drive> |
    floppyattach none|<uuid>|<filename>|host:<drive> |
```

```
vrdp on|off] |
setvideomodehint <xres> <yres> <bpp> [display]|
setcredentials <username> <password> <domain>
[--allowlocallogon <yes|no>]
```

Mit dem Befehl `controlvm` kontrollieren Sie eine laufende Maschine. So können Sie zum Beispiel einen Reset auslösen, die Maschine beenden, DVDs einbinden usw. Es darf hierbei immer nur ein Parameter übergeben werden.

Parameter	Funktion des Parameters
pause	Hält die Maschine an.
resume	Führt eine mit <code>pause</code> angehaltene Maschine fort.
reset	Sendet ein Reset-Signal.
poweroff	Schaltet die virtuelle Maschine aus.
savestate	Speichert den aktuellen Zustand der Maschine und beendet diese. Der Zustand wird im Snapshot-Ordner der Maschine gespeichert. Die Datei hat die Endung <code>.sav</code> .
acpipowerbutton	Sendet das ACPI-Signal zum Herunterfahren (muss vom Gast-system unterstützt werden).
acpisleepbutton	Sendet das ACPI-Signal zum Wechseln in den Stromsparmodus (muss vom Gastsystem unterstützt werden).
setlinkstate	Legt fest, ob das Netzkabel des entsprechenden virtuellen Netzwerkadapters eingesteckt ist oder nicht.
usbattach / usbdetach	Übergibt bzw. trennt ein USB-Gerät.
dvdattach	Übergibt bzw. trennt (mit <code>none</code> ) ein CD/DVD-Laufwerk oder -Abbild.
setvideomodehint	Ändert die Auflösung einer virtuellen Maschine (die Gasterweiterungen müssen installiert sein).
setcredentials	Erlaubt den automatischen Login bei Windows-XP-Gastsyste-men (benötigt die DLL <code>VBoxGina.dll</code> ).

**Tabelle 8.7** Parameter zum Befehl »controlvm«

Beispiele:

*Maschine anhalten:*

```
VBoxManage controlvm LiveCD pause
```

*Netzkabel von Adapter 1 trennen:*

```
VBoxManage controlvm LiveCD setlinkstate1 off
```



*CD/DVD trennen:*

```
VBoxManage controlvm WindowsXP dvdattach none
```

*Auflösung in 800 x 600 mit 32 Bit Farbtiefe ändern:*

```
VBoxManage controlvm WindowsXP setvideomodehint 800 600 32
```

**discardstate <uuid>|<name>**

Verwirft die Informationen einer Maschine, die mit `controlvm savestate` angehalten wurde. Achtung: Diese Methode kann zu Datenverlusten führen.

**adoptstate <uuid>|<name> <state\_file>**

Wenn Sie den aktuellen Zustand einer Maschine mit `savestate` gespeichert haben, kann diese Datei gesichert werden. Mit `adoptstate` können Sie dann eine solche Sicherung der Maschine wieder zuweisen. Achten Sie hierbei darauf, dass die Statusdatei auch »passt«.

**Beispiele:**

```
VBoxManage adoptstate LiveCD /media/vms/vbox/State/Test1.sav
VBoxManage adoptstate LiveCD D:\Ubuntu.sav
```

**snapshot**

```
snapshot <uuid>|<name>
    take <name> [--description <desc>] |
    discard <uuid>|<name> |
    discardcurrent --state|--all |
    edit <uuid>|<name>|--current [--name <name>] [--description <desc>] |
    showvminfo <uuid>|<name>
```

Erstellt, bearbeitet und verwirft die Sicherungspunkte einer Maschine (siehe auch Abschnitt 5.3, »Arbeiten mit Sicherungspunkten (Snapshots)«).

Parameter	Funktion des Parameters
take	Erstellt einen Sicherungspunkt. Optional kann noch eine Bemerkung ( <code>--description</code> ) vergeben werden.
discard	Verwirft einen bestimmten Sicherungspunkt.
discardcurrent	Verwirft den aktuellen Status.
edit	Bearbeitet den Namen und die Beschreibung eines Sicherungspunkts.
showvminfo	Zeigt die Infos zu dem Sicherungspunkt an.

**Tabelle 8.8** Parameter zur Verwendung mit »snapshot«

Beispiele:

*Erstellt einen Sicherungspunkt:*

```
VBoxManage snapshot Ubuntu take "Sicherung 1"\  
    --description "Vor Upgrade auf XY"
```

*Zeigt Infos zu einem Sicherungspunkt an:*

```
VBoxManage snapshot Ubuntu showvminfo "Sicherung 1"
```

**openmedium**

```
openmedium disk|dvd|floppy <filename>  
    [--type normal|immutable|writethrough]
```

Registriert vorhandene Image-Dateien (von Festplatten, CD/DVDs oder Disketten) in VirtualBox. Als Parameter sind die Art des Images, der Pfad und Dateiname sowie die Art der Registrierung wichtig.

[+]

**Hinweis**

Dieser Befehl ersetzt den Befehl `registerimage`.

Parameter	Funktion des Parameters
disk dvd floppy	Legt fest, welche Art von Image registriert wird.
--type	Hiermit bestimmen Sie, ob ein Festplatten-Image den Typ normal (Standard), immutable oder writethrough erhalten soll. Dieser Parameter kann bei DVDs und Disketten nicht verwendet werden.
normal	Das Image wird »normal« eingebunden. Änderungen werden gespeichert, und es wird bei Sicherungspunkten berücksichtigt.
immutable	Die Änderungen werden nur temporär gespeichert. Beim Beenden der virtuellen Maschine werden sie wieder verworfen.
writethrough	Änderungen werden immer gespeichert. Bei Sicherungspunkten wird das Image nicht berücksichtigt. Dieser Typ ist sinnvoll für Datenlaufwerke.

**Tabelle 8.9** Parameter zur Verwendung mit »registerimage«

Beispiele:

*Disketten-Image einbinden:*

```
VBoxManage openmedium floppy /media/vms/floppy.img
```

*Festplatten-Image »immun« einbinden:*

```
VBoxManage openmedium disk E:\vdi\Internet.vdi\
--type immutable
```

### **closemedium disk|dvd|floppy <uuid>|<filename>**

Meldet ein Image ab. Voraussetzung ist jedoch, dass das entsprechende Image in keine virtuelle Maschine eingebunden ist.

*Beispiel:*

```
VBoxManage closemedium dvd /media/brennen/ubuntu-desktop-i386.iso
```

#### **Hinweis**

Dieser Befehl ersetzt den Befehl `unregisterimage`.

[+]

### **showhdinfo <uuid>|<filename>**

Listet Informationen zu einem Festplatten-Image auf.

*Beispiel:*

```
dirk@t61:~$ VBoxManage -q showhdinfo /media/VDI/Ubuntu_system.vdi
UUID:                ee51d579-b377-49f4-90f4-f42df219e398
Accessible:          yes
Logical size:         20480 MBytes
Current size on disk: 2935 MBytes
Type:                 normal (base)
Storage format:       VDI
In use by VMs:        Ubuntu_9 (UUID: cf9a2af6-3d4...443b49e02fd)
Location:             / media/VDI /Ubuntu_system.vdi
```

### **createhd**

```
createhd --filename <filename>
--size <megabytes>
[--format VDI|VMDK|VHD]
[--variant Standard,Fixed,Split2G,Stream,ESX]
[--type normal|writethrough]
[--comment <comment>]
[--remember]
```

Erstellt ein Festplatten-Image unter dem mit `--filename` übergebenen Namen. Die Größe (`--size`) wird in MByte angegeben. Standardmäßig wird ein neu erstelltes Image nicht registriert. Dies muss explizit mit dem Parameter `--remember` angegeben oder mit `openmedium` nachgeholt werden.

Parameter	Funktion des Parameters
--filename	Pfad und Dateiname des Images
--size	Größe in MByte
--format	Format des Images
VDI	VirtualBox-Image (Standard)
VMDK	VMware-Image
VHD	MS-VirtualPC-Image
--variant	Legt die »Variante« des Images fest. Diese ist jedoch auch abhängig vom --format.
Standard	Das Image wird bei Bedarf vergrößert (dynamisch wachsendes Medium).
Fixed	Das Image belegt sofort den benötigten Speicher.
--type	Der Typ des Images (siehe auch openmedium).
normal	Das Image wird »normal« eingebunden. Änderungen werden gespeichert, und es wird bei Sicherungspunkten berücksichtigt.
writethrough	Änderungen werden immer gespeichert. Bei Sicherungspunkten wird das Image nicht berücksichtigt.
--comment	Fügt eine Beschreibung zum Festplatten-Image hinzu.
--remember	Registriert das Image bei VirtualBox.

**Listing 8.1** Parameter zur Verwendung mit dem Befehl »createhd«

Beispiel:  
*Image mit einer festen Größe von 10 GByte erstellen und registrieren:*

```
VBoxManage createhd --filename /media/vbox/VDI/daten.vdi \
    --size 10000 --variant fixed --remember
```

**modifyhd**

```
modifyhd <uuid>|<filename>
    [--type normal|writethrough|immutable]
    [--autoreset on|off]
    [--compact]
```

Mit diesem Befehl ändern Sie ein Festplatten-Image.

Parameter	Funktion des Parameters
--type	Hiermit kann der Typ festgelegt werden.
normal	Das Image wird »normal« behandelt. Änderungen werden gespeichert, und es wird bei Sicherungspunkten berücksichtigt.
writethrough	Änderungen werden immer gespeichert. Bei Sicherungspunkten wird das Image nicht berücksichtigt. Dieser Typ ist sinnvoll für Datenlaufwerke.
immutable	Die Änderungen werden nur temporär gespeichert. Beim Beenden der virtuelle Maschine werden sie wieder verworfen.
--autoreset on off	Aktiviert oder deaktiviert das automatische Verwerfen der Änderungen bei einem Festplatten-Image vom Typ immutable.
--compact	Komprimiert ein dynamisches Image.

**Tabelle 8.10** Parameter für den Befehl »modifyhd«

Beispiel:

*Image komprimieren:*

```
VBoxManage modifyhd ubuntu_system --compact
```

**clonehd**

```
clonehd <uuid>|<filename> <outputfile>  
    [--format VDI|VMDK|VHD|RAW|<other>]  
    [--variant Standard,Fixed,Split2G,Stream,ESX]  
    [--type normal|writethrough|immutable]  
    [--remember]
```

Wegen der eindeutigen UUIDs ist ein »normaler« Kopiervorgang und erneutes Registrieren eines Festplatten-Images nicht möglich. Stattdessen verwenden Sie clonehd. Der Befehl erstellt eine Kopie eines vorhandenen Festplatten-Images und gibt diesem eine neue UUID. Dabei können auch das Format und andere Eigenschaften des Images geändert werden. clonehd hat die gleichen Parameter wie createhd.

Beispiel:

*Erstellt eine unveränderliche Kopie des XP-Images und registriert diese:*

```
VBoxManage clonehd winxp_system.vdi E:\vbox\vdi\xp2.vdi  
    --variant Standard --type immutable --remember
```

*Erstellt eine Rohdatei (RAW):*

```
VBoxManage clonehd ubuntu_system.vdi E:\ubuntu.img
--format RAW
```

### **convertfromraw**

```
convertfromraw
    <filename> <outputfile>
    [--format VDI|VMDK|VHD]
    [--variant Standard,Fixed,Split2G,Stream,ESX]

    stdin <outputfile> <bytes>
    [--format VDI|VMDK|VHD]
    [--variant Standard,Fixed,Split2G,Stream,ESX]
```

Erstellt aus einer Rohdatei oder einem per Pipe übergebenen Datenstrom ein Festplatten-Image im entsprechenden Format. Das Quell-Image kann unter anderem mit dem Programm dd auf der Kommandozeile erstellt werden.

Beispiel:

*Konvertiert eine Rohdatei ins VDI-Format:*

```
VBoxManage convertfromraw
    /media/daten/winxp.img
    /media/virtuelle Maschinen/vdi/winxp.vdi
    --variant Standard
```

### **addiscsidisk**

```
addiscsidisk --server <name>|<ip>
    --target <target>
    [--port <port>]
    [--lun <lun>]
    [--encodedlun <lun>]
    [--username <username>]
    [--password <password>]
    [--type normal|writethrough|immutable]
    [--comment <comment>]
    [--intnet]
```

Mit addiscsidisk richten Sie für VirtualBox den Zugriff auf ein iSCSI-Volume ein.

**getextradata**

```
getextradata global|<uuid>|<name>
               <key>|enumerate
```

Mit dem Befehl `getextradata` können Informationen bzw. Keys die mit `setextradata` gesetzt wurden, abgefragt werden. Der Parameter `enumerate` listet alle vorhandenen Keys auf.

Beispiele:

*Alle Informationen abrufen:*

```
VBoxManage getextradata LiveCD enumerate
```

*Einzelne Informationen abrufen:*

```
VBoxManage getextradata WindowsXP Benutzer
```

*Alle globalen Informationen:*

```
VBoxManage getextradata global enumerate
```

**setextradata**

```
setextradata global|<uuid>|<name>
               <key>
               [<value>] (no value deletes key)
```

Setzt zu einer virtuellen Maschine oder global zusätzliche Informationen. Werte mit Leerstellen müssen in Anführungszeichen gesetzt werden. Weiterhin dient der Befehl zum Einstellen spezieller Parameter (siehe Abschnitt 5.12, »Port-Forwarding mit `setextradata`«).

Beispiele:

*Globale Information setzen:*

```
VBoxManage setextradata global Hostsystem "System für VBox-Buch"
```

*Einzelne Informationen setzen:*

```
VBoxManage setextradata WindowsXP Benutzer "Dirk Becker"
```

**setproperty**

```
setproperty hdfolder default|<folder> |
               machinefolder default|<folder> |
               vrdpauthlibrary default|<library> |
               webrvauthlibrary default|null|<library> |
               hwvrtexenabled yes|no
               loghistorycount <value>
```

Mit diesem Befehl können Sie die globalen Einstellungen konfigurieren.

Parameter	Funktion des Parameters
hdfolder	Zielverzeichnis für die Festplatten-Images
machinefolder	Zielverzeichnis für die virtuellen Maschinen
vrdpauthlibrary	Art der Authentisierung für den RDP-Zugriff auf virtuelle Maschinen
loghistorycount	Anzahl der Logbücher

**Tabelle 8.11** Parameter für »setproperty«

**usbfilter add**

```
usbfilter remove <index,0-N>
    --target <uuid>|<name>|global
```

Fügt einen neuen USB-Filter hinzu (siehe Abschnitt 5.11.4, »USB-Filter«).

**usbfilter modify**

```
usbfilter modify <index,0-N>
    --target <uuid>|<name>|global
    [--name <string>]
    [--action ignore|hold] (global filters only)
    [--active yes|no]
    [--vendorid <XXXX>|'']
    [--productid <XXXX>|'']
    [--revision <IIFF>|'']
    [--manufacturer <string>|'']
    [--product <string>|'']
    [--remote yes|no] (null, VM filters only)
    [--serialnumber <string>|'']
    [--maskedinterfaces <XXXXXXXX>]
```

Verändert einen vorhandenen USB-Filter (siehe Abschnitt 5.11.4, »USB-Filter«).

**usbfilter remove**

```
usbfilter remove <index,0-N>
    --target <uuid>|<name>|global
```

Entfernt einen vorhandenen USB-Filter (siehe Abschnitt 5.11.4, »USB-Filter«).



### sharedfolder add

```
sharedfolder add <vmname>|<uuid>
    --name <name> --hostpath <hostpath>
    [--transient] [--readonly]
```

Fügt einer virtuellen Maschine einen gemeinsamen Ordner hinzu. Dieser Befehl kann auch bei einer laufenden Maschine angewendet werden.

Parameter	Funktion des Parameters
--name	Name des gemeinsamen Ordners in der virtuellen Maschine
--hostpath	Pfad zu dem Ordner auf dem Hostsystem
--transient	Übergibt den gemeinsamen Ordner temporär, das heißt, er ist nach dem Beenden nicht mehr zugewiesen.
--readonly	Erlaubt nur Lesezugriffe auf den Ordner.

**Tabelle 8.12** Parameter zur Verwendung mit »sharedfolder add«

#### Beispiele:

```
VBoxManage sharedfolder add WinXP --name GemeinsameDaten\
    --hostpath /media/daten/gemeinsam
```

### sharedfolder remove

```
sharedfolder remove <vmname>|<uuid>
    --name <name> [--transient]
```

Entfernt einen gemeinsamen Ordner. Dieser Befehl kann auch bei einer laufenden Maschine angewendet werden.

Parameter	Funktion des Parameters
--name	Name des gemeinsamen Ordners in der virtuellen Maschine
--transient	Entfernt einen Ordner vom Typ <i>transient</i> .

**Tabelle 8.13** »sharedfolder remove« und seine Parameter

### vmstatistics

```
vmstatistics <vmname>|<uuid> [--reset]
    [--pattern <pattern>] [--descriptions]
```

Gibt Informationen zu einer laufenden Maschine zurück.

Parameter	Funktion des Paramenters
--reset	Setzt die Angaben zurück.
--pattern	Begrenzt die Ausgabe.
--descriptions	Fügt der Ausgabe noch eine kurze Beschreibung hinzu.

Tabelle 8.14 Parameter zur Steuerung von »vmstatistics«

Beispiele:

*Alle Informationen ausgeben:*

```
VBoxManage vmstatistics WinXP
```

*Spezielle Information ausgeben:*

```
VBoxManage vmstatistics WinXP\  
    --pattern /TM/VirtualSync/CurrentOffset
```

guestproperty

```
guestproperty get <vmname>|<uuid> <property> [--verbose]  
guestproperty set <vmname>|<uuid> <property> [<value> [--flags <flags>]]  
guestproperty enumerate <vmname>|<uuid> [--patterns <patterns>]  
guestproperty wait <vmname>|<uuid> <patterns> [--timeout <timeout>]
```

Seit Version 2.1 ist es mit dem Befehl `guestproperty` möglich, verschiedene Einstellungen des Gastsystems einer laufenden Maschine abzufragen oder zu ändern. Um den Befehl zu nutzen, müssen die Gasterweiterungen installiert sein.

Parameter	Funktion des Paramenters
get	Gibt eine Eigenschaft zurück.
set	Setzt eine vorhandene oder neue Eigenschaft.
enumerate	Gibt alle vorhandenen Eigenschaften aus.
wait	Wartet auf die Änderung einer Eigenschaft.

Tabelle 8.15 Parameter zur Verwendung mit dem Befehl »guestproperty«

Im Gastsystem kann das Kommandozeilentool `VBoxControl`, das mit den Gasterweiterungen installiert wird, verwendet werden.

[+]

Hinweis

Dieser Befehl kann zurzeit (Version 3.0) nur mit Windows-Gastsystemen verwendet werden.

Beispiele im Hostsystem:

*Alle Informationen ausgeben:*

```
VBoxManage guestproperty enumerate WinXP
```

*Ausgabe filtern:*

```
VBoxManage guestproperty enumerate WinXP -patterns *Infos*
```

*Aktueller Benutzer:*

```
VBoxManage guestproperty get WinXP\  
    /VirtualBox/GuestInfo/OS/LoggedInUsersList
```

*Eigenes Property setzen:*

```
VBoxManage guestproperty set WinXP\  
    /VirtualBox/Infos/Ersteller "Dirk Becker"
```

*Warten, bis sich die Eigenschaft Pause geändert hat (maximal 10 s):*

```
VBoxManage guestproperty wait WinXP "/VirtualBox/Infos/Pause"\  
    --timeout 10000
```

Beispiele im Gastsystem (in der Eingabeaufforderung):

*Alle Informationen ausgeben:*

```
VBoxControl guestproperty enumerate
```

*Spezielle Information ausgeben:*

```
VBoxControl guestproperty get /VirtualBox/Infos/Ersteller
```

*Eigenschaft »Pause« ändern:*

```
VBoxControl guestproperty set "/VirtualBox/Infos/Pause" 0
```

**metrics**

```
metrics list [*|host|<vmname> [<metric_list>]]  
metrics setup [--period <seconds>] [--samples <count>] [--list]\  
    [*|host|<vmname> [<metric_list>]]  
  
metrics query [*|host|<vmname> [<metric_list>]]  
metrics collect [--period <seconds>] [--samples <count>]  
    [--list] [--detach]  
    [*|host|<vmname> [<metric_list>]]
```

Der Befehl `metrics` ermöglicht die Überwachung einer laufenden virtuellen Maschine und des Hostsystems.

Parameter	Funktion des Paramenters
list	Listet die zur Verfügung stehenden Messpunkte auf.
setup	Konfiguriert einen Messpunkt.

**Tabelle 8.16** Mit diesen Parametern den Befehl »metrics« verwenden

Parameter	Funktion des Paramenters
query	Fragt Messpunkte ab.
collect	Fragt Messpunkte in Intervallen ab.

**Tabelle 8.16** Mit diesen Parametern den Befehl »metrics« verwenden (Forts.)

Beispiele:

*Alle Informationen ausgeben:*

```
VBoxManage metrics list LiveCD
```

*Speicherverbrauch:*

```
VBoxManage metrics query WinXP RAM/Usage/Used
```

*Alle 5 s die CPU-Auslastung ausgeben:*

```
VBoxManage metrics collect --period 5 LiveCD CPU*
```

**hostonlyif**

```
hostonlyif ipconfig <name>
    [--dhcp |
    --ip<ipv4> [--netmask<ipv4> (def: 255.255.255.0)] |
    --ipv6<ipv6> [--netmasklengthv6<length> (def: 64)]]
```

Ermöglicht die Konfiguration des Adapters für Host-only Netzwerke.

**dhcpserver**

```
dhcpserver add|modify --netname <network_name> |
    --ifname <hostonly_if_name>
    [--ip <ip_address>
    --netmask <network_mask>
    --lowerip <lower_ip>
    --upperip <upper_ip>]
    [--enable | --disable]
```

Ermöglicht die Konfiguration oder das Hinzufügen von DHCP-Servern. Diese können für bestimmte interne oder Host-only Netzwerke eingesetzt werden.

Parameter	Funktion des Paramenters
--netname	Name des internen Netzwerks (nicht in Verbindung mit --ifname)
--ifname	Name des Host-only Adapters
--ip	IP-Adresse des DHCP-Servers

**Tabelle 8.17** Parameter zur Verwendung mit »dhcpserver«

Parameter	Funktion des Paramenters
--netmask	Netzmaske
--lowerip	Die niedrigste IP-Adresse, die vergeben wird
--upperip	Die höchste IP-Adresse, die vergeben wird
--enable   --disable	Aktiviert oder deaktiviert den Server

**Tabelle 8.17** Parameter zur Verwendung mit »dhcpserver« (Forts.)

Beispiele:

*Einen neuen DHCP-Server erstellen:*

```
VBoxManage dhcpserver add --netname Netzwerk1 --ip 192.168.51.1\  
    --netmask 255.255.255.0  
    --lowerip 192.168.51.10 --upperip 192.168.51.100
```

*Server deaktivieren:*

```
VBoxManage dhcpserver modify --netname Netzwerk1 --disable
```

**dhcpserver remove**

```
dhcpserver remove --netname <network_name> |  
    --ifname <hostonly_if_name>
```

Entfernt einen DHCP-Server anhand des Netzwerk- oder Adapternamens.

**8.3 VBoxSDL**

Mit VBoxSDL können Sie virtuelle Maschinen starten und verwenden. Dabei können Sie verschiedene Einstellungen auch temporär ändern. Diese Änderungen werden nach dem Beenden der Maschine wieder verworfen.

**--startvm <uuid|name>**

Startet die virtuelle Maschine.

**--hda <file>**

Ersetzt das primäre Festplatten-Image (meist das Bootlaufwerk).

Beispiel:

```
VBoxSDL --startvm WinXP --hda E:\Vbox\VDI\winxp2_system.vdi
```

**--fda <file>**

Ersetzt das erste Disketten-Image bzw. -laufwerk.

**--cdrom <file>**

Ersetzt das CD/DVD-Laufwerk. Geben Sie `none` an, um die CD/DVD zu entfernen.

Beispiel:

```
VBoxSDL --startvm LiveCD --cdrom E:\Images\debian.iso
```

oder

```
VBoxSDL --startvm WinXP --cdrom none
```

**--boot <a|c|d|n>**

Ändert die Bootreihenfolge:

- ▶ a: Diskette
- ▶ c: erstes Festplatten-Image
- ▶ d: CD/DVD
- ▶ n: Netzwerkboot

Beispiel:

```
VBoxSDL --startvm WinXP --cdrom E:\Images\pmagic.iso\
--boot d
```

**--memory <size>**

Ändert die Größe des Hauptspeichers. Die Größe geben Sie in MByte an.

**--vram <size>**

Ändert die Größe des Grafikkartenspeichers. Die Größe geben Sie in MByte an.

**--fullscreen**

Startet die virtuelle Maschine im Vollbildmodus und behält dabei die Auflösung des Gastsystems bei.

**--fullscreenresize**

Startet die virtuelle Maschine im Vollbildmodus und passt dabei die Auflösung des Gastsystems an.

**--fixedmode <w> <h> <bpp>**

Startet die virtuelle Maschine in der angegebenen Auflösung.

Beispiel:

```
VBoxSDL --startvm WinXP --fixedmode 800 600 32
```

**--nofstoggle**

Verhindert, dass die Maschine in den Vollbildmodus wechselt. Die Funktion der Host-Taste wird dabei deaktiviert.

**--noresize**

Verhindert, dass die Fenstergröße der Maschine geändert wird.

**--nohostkey**

Deaktiviert die Funktionen der Host-Taste. Diesen Parameter sollten Sie mit Bedacht verwenden, es ist nämlich nicht mehr möglich, eine eingefangene Maus und Tastatur freizugeben, wenn keine Gasterweiterungen installiert sind.

**--nogradonclick**

Verhindert das Einfangen der Maus und Tastatur bei Systemen ohne Gasterweiterungen.

**--detecthostkey**

Gibt Informationen zu einer Tastenkombination mit der Host-Taste aus. Dabei muss keine virtuelle Maschine gestartet werden.

Beispiel:

```
VBoxSDL --detecthostkey
```

**--termacpi**

Sendet beim Schließen des Fensters das ACPI-Signal zum Herunterfahren des Betriebssystems. Dadurch wird ein Gastsystem, das dieses unterstützt, auch korrekt beendet. Ohne diesen Parameter wird die virtuelle Maschine beim Schließen des Fensters einfach »ausgeschaltet«.

Beispiel:

```
VBoxSDL --startvm WinXP --termacpi
```

**--evdevkeymap**

Nur für Linux-Hostsysteme: Übernimmt Tastatureinstellungen, die mit dem *evdev*-Protokoll ermittelt wurden (*evdev* erkennt und konfiguriert automatisch verschiedene Eingabegeräte).

**--vrdp <port>**

Mit dieser Option legen Sie einen anderen Port für den RDP-Zugriff fest.

**--discardstate**

Falls sich die Maschine im Ruhezustand befindet, wird mit diesem Parameter die Statusdatei gelöscht und das Gastsystem neu gestartet. Hierbei kann es zu Datenverlust kommen.

**--[no]rawr0**

Dieser Parameter aktiviert bzw. deaktiviert den Zugriff auf Ring 0 des Prozessors, auf dem normalerweise der Kernel läuft. Dadurch kann festgelegt werden, ob das Gastsystem auf diesen Ring Zugriff erhält. Ob dies wirklich funktioniert, kann ich nicht genau sagen, da meine Testsysteme unter anderem unterschiedliche Taktfrequenzen der CPU anzeigen.

**--[no]rawr3**

siehe `--rawr0`

**--[no]hvwrtex**

Aktiviert oder deaktiviert die Intel-VT-x- bzw. AMD-V-Funktionen.

Beispiel:

*Aktivieren:*

```
VBoxSDL --startvm OS2 --hvwrtex
```

*Deaktivieren:*

```
VBoxSDL --startvm LiveCD --nohvwrtex
```

**--convertSettings**

Falls ein Update von VirtualBox eingespielt wurde, aktualisiert dieser Parameter automatisch die Konfigurationsdatei der virtuellen Maschine.

**--convertSettingsBackup**

Wie `--convertSettings`; zusätzlich wird eine Sicherungskopie der Originaldatei erstellt.

## 8.4 VboxHeadless

Das Kommandozeilentool VBoxHeadless startet virtuelle Maschinen ohne die Verwendung einer grafischen Oberfläche. Der Zugriff auf diese Maschinen kann dann über den RDP-Zugriff erfolgen.



**--startvm | -s <name|uuid>**

Startet eine virtuelle Maschine unter Angabe des Namens oder der UUID.

Beispiel:

```
VBoxSDL --startvm LiveCD
```

*oder in Kurzform:*

```
VBoxSDL -s LiveCD
```

**--vrdp | -v on|off|config**

Bestimmt, ob die virtuelle Maschine RDP-Zugriff erhalten darf oder nicht:

- ▶ **on:** Aktiviert den VRDP-Server (Standard).
- ▶ **off:** Deaktiviert den VRDP-Server.
- ▶ **config:** Übernimmt die Einstellungen aus der Konfiguration der Maschine.

**--vrdpport | -p <port>**

Legt den Port für den RDP-Zugriff fest.

Beispiel:

```
VBoxSDL --startvm LiveCD --vrdp on --vrdpport 3400
```

**--vrdpaddress | -a <ip>**

Legt die Adresse für den RDP-Zugriff fest. Dies ist nur wichtig, wenn das Hostsystem über mehrere Netzwerkadapter mit unterschiedlichen IP-Adressen verfügt.

Beispiel:

```
VBoxSDL --startvm LiveCD --vrdp on --vrdpport 3400\
  --vrdpaddress 192.168.0.50
```

**--capture | -c**

Startet die Videoaufnahme, zeichnet also die Ausgabe der virtuellen Maschine auf (siehe auch Abschnitt 4.6.3, »Sitzung aufzeichnen«).

**--width | -w**

Bestimmt die Breite des Videos in Pixel. Kann nur in Verbindung mit `--capture` verwendet werden.

**--height | -h**

Bestimmt die Höhe des Videos in Pixel. Kann nur in Verbindung mit `--capture` verwendet werden.

**--bitrate | -r**

Bestimmt die Bitrate des Videos in Bytes. Kann nur in Verbindung mit `--capture` verwendet werden.

**--filename | -f**

Bestimmt den Dateinamen des Videos. Kann nur in Verbindung mit `--capture` verwendet werden.

## 8.5 rdesktop-vrdp

Das Kommandozeilentool `rdesktop-vrdp` ist eine speziell für VirtualBox erweiterte Version des Open-Source-Tools `rdesktop` und läuft nur unter Linux-Hostsystemen. `rdesktop-vrdp` baut Verbindungen mit dem Remote Desktop Protocol (RDP) auf. Die Gegenstelle kann dabei ein MS-Server mit Terminal-Service, ein PC mit Windows XP/Vista oder eben eine VirtualBox-Umgebung sein. Die wichtigste Erweiterung ist die Möglichkeit, USB-Geräte zu übergeben.

Ich erläutere im Folgenden nur die wichtigsten Parameter für den Zugriff auf eine virtuelle Maschine.

**rdesktop-vrdp <server>[:<port>]**

Mit `--server` übergeben Sie den Netzwerknamen oder die IP-Adresse der Gegenstelle. Optional geben Sie den an. Beim Zugriff auf eine virtuelle Maschine darf als Name nicht der Netzwerkname des Gastsystems angegeben werden, sondern der des Hostsystems.

Beispiel:

*Zugriff auf einen Windows-Rechner:*

```
rdesktop-vrdp Fibu1
```

*Zugriff auf eine virtuelle Maschine:*

```
rdesktop-vrdp VMServer:3401
```

**-u <user name >**

Wenn die Authentisierung mit Benutzername und Passwort aktiviert ist, übergeben Sie mit diesem Parameter den Benutzername.

Beispiel:

```
rdesktop-vrdp -u dirk -p 12345 VMServer:3401
```

**-p <password>**

Wenn die Authentisierung aktiviert ist, übergeben Sie mit diesem Parameter das Passwort. Alternativ tippen Sie statt des Kennworts ein Minuszeichen; das Passwort wird dann beim Verbindungsaufbau abgefragt.

Beispiel:

*Passwort im Klartext übergeben:*

```
rdesktop-vrdp -u dirk -p 12345 VMServer:3401
```

*Passwort wird abgefragt:*

```
rdesktop-vrdp -u dirk -p - VMServer:3401
```

**-k <keyboard layout>**

Mit dieser Option legen Sie das gewünschte Tastaturlayout fest, wobei `de` für eine deutsche Tastatur steht.

Beispiel:

```
rdesktop-vrdp -k de VMServer:3401
```

**-f**

Startet die Verbindung im Vollbildmodus. Im Gegensatz zu Verbindungen mit Windows XP oder Vista, wird dabei jedoch die Auflösung des Gastsystems nicht angepasst. Dies nehmen Sie dann im Gastsystem selbst vor.

**-T <window title>**

Mit diesem Parameter können Sie einen eigenen Namen für die Titelzeile des Fensters setzen.

Beispiele:

```
rdesktop-vrdp -k de -T LiveCD VMServer:3401
```

```
rdesktop-vrdp -k de -T "Windows XP" VMServer:3401
```

**-N**

Wenn das numerische Tastenfeld Ihrer Tastatur nicht verwendet werden kann, können Sie das Problem mit diesem Parameter beheben.

**-r <option>**

Mit diesem Parameter aktivieren Sie die Umlenkung von Geräten (*redirection*). Sie können die Option auch mehrfach übergeben.

Beispiel:

```
rdesktop-vrdp -k de -T LiveCD -r usb VMServer:3401
```



## 9 Glossar

**DHCP** Das *Dynamic Host Configuration Protocol* (DHCP) dient dazu, einem Betriebssystem seine Netzwerkeinstellungen automatisch von einem Server aus zuzuweisen. VirtualBox bietet eigene DHCP-Server an, die bei der Netzwerkeinstellung NAT und internen Netzwerken zum Einsatz kommen.

**DNS** Das *Domain Name System* (DNS) dient dazu, Netzwerknamen in IP-Adressen aufzulösen. Beim Zugriff auf einen Rechner im Internet über dessen Namen (zum Beispiel *www.64-bit.de*) wird der Name im Hintergrund in dessen IP-Adresse aufgelöst. Der eigentliche Zugriff erfolgt dann über diese Adresse.

**Festplatten-Image** Siehe *Virtuelle Festplatte*

**Gastsystem (auch: Gastbetriebssystem)** Mit einem Gastsystem bezeichnet man bei der Virtualisierung das Betriebssystem, das in einer virtuellen Maschine (als »Gast«) installiert wird.

**Gateway** Ein Gateway verbindet zwei oder mehrere Netzwerke miteinander. Diese Netzwerke sind dabei meist absolut unterschiedlich bzw. inkompatibel zueinander. Häufig (aber fälschlicherweise) werden Gateways mit Routern gleichgesetzt.

**Hostsystem** Das Hostsystem ist das Betriebssystem, auf dem VirtualBox installiert wird.

**I/O-APIC** Der *Input/Output Advanced Programmable Interrupt Controller* (I/O-APIC) ist ein von Intel speziell für Multiprozessorsysteme entwickelter Control-

ler. Er verbessert unter anderem die Interrupt-Verteilung. Dieser Controller ist inzwischen in allen gängigen Chipsätzen, selbst bei Einzelprozessorsystemen, enthalten.

**Kernel** Der Kernel ist der zentrale »Kern« eines Betriebssystems. Er befindet sich meist im Arbeitsspeicher des Computers.

**Kontextmenü** Ein Kontextmenü ist ein Menü, das sich beim Rechtsklick mit der Maus auf ein Objekt öffnet. Sein Aufbau ist dabei abhängig von dem entsprechenden Objekt.

**Live-CD** Bei einer Live-CD handelt es sich um eine bootfähige CD oder DVD. Auf dieser befindet sich ein Betriebssystem, das nach dem Startvorgang wie ein installiertes System verwendet werden kann. Eine Besonderheit gegenüber Systemen, die von einer Festplatte oder einem USB-Stick starten, ist hierbei, dass eine CD standardmäßig keinen Schreibzugriff auf die Festplatte bietet.

**Namensauflösung** Siehe *DNS*

**Netzwerkmaske** Eine Netzwerkmaske beschreibt den Aufbau der IP-Adressen der Netzwerkteilnehmer eines Netzwerks. Dabei müssen Teilnehmer, die sich finden sollen, die gleiche Netzwerkmaske erhalten. Ansonsten können diese nur über *Router* miteinander kommunizieren.

**OEM-Software** Als OEM-Software (*Original Equipment Manufacture*) bezeichnet man meist Programme oder Betriebssysteme, die mit PCs ausgeliefert wurden. Dabei sind diese meist mit Einschränkungen verbunden (keine CD, nicht alle Funk-

tionen wie die Vollversion, begrenzte Laufzeit usw.).

**Peripheriegeräte** Mit Peripheriegeräten sind umgangssprachlich Geräte gemeint, die sich außerhalb eines Computers befinden. Streng genommen wird noch zwischen *internen* und *externen* Peripheriegeräten unterschieden. Ein internes Gerät ist zum Beispiel eine Netzwerkkarte und ein externes ein Drucker oder Scanner.

**Router** Wenn sich das Ziel eines Datenpakets nicht im lokalen Netzwerk befindet, wird dieses an einen Router übergeben. Dieser leitet das Paket weiter oder – je nach Konfiguration – blockiert es.

**SMP** Bei einem *symmetrischen Multiprocessorsystem* (SMP) werden die Prozesse dynamisch auf die vorhandenen Prozessoren verteilt, wenn das Betriebssystem dies unterstützt.

**Southbridge** Die Southbridge steuert den Datenverkehr zwischen internen Peripheriegeräten und stellt hierzu selbst verschiedene Schnittstellen (zum Beispiel USB) zur Verfügung.

**Virtuelle Festplatte** Eine virtuelle Festplatte ist eine Datei, die einer virtuellen Maschine als Festplatte zugeteilt werden kann. Häufig werden diese auch *Festplatten-Image* oder einfach nur *Image* genannt. Unter VirtualBox haben diese Dateien die Endung *.vdi*.

# Index

/etc/fstab 275  
3-D-Beschleunigung 123, 171, 226, 280

## A

---

abmelden 251  
Accelerated Graphics Port 27  
ACPI-Event 146  
    *acpipowerbutton* 292  
    *acpisleepbutton* 292  
Adaptertyp 124  
addiscsidisk 298  
adoptstate 220, 293  
Adresseinheit 30  
Advanced Host Controller Interface 122  
Aero 171  
AGP 27  
AHCI 122  
Aktivierung 260  
ALU (Arithmetic Logic Unit) 30  
AMD Virtualization 30  
AMD-V 30, 45, 232  
Android 42  
Anschlussart 125  
Anwenderfehler 110  
Anwendungsvirtualisierung 46  
Apple 24  
Arbeitsspeicher 31  
Assistent 84  
Audio-Adapter 123  
Audio-Controller 172  
Aufbau 115  
Auflistung 90  
aufzeichnen 196  
Authentisierung 249  
autoreset 297

## B

---

Backslash 22, 284  
Backup 254  
basefolder 288  
Basic Input Output System (BIOS) 28  
Beenden 145  
Befehle 283

Befehlsdecoder 30  
Befehlssatzerweiterungen 45  
Benutzermenü 90  
Benutzername 249  
Beschreibung 99  
Betriebssystem 137  
Bibliotheken 78  
bidirectional 290  
Binärversion 69  
BIOS 25  
bitrate 310  
blkid 277  
Bochs 43  
boot 306  
brennen 280  
Brenner 141  
Bridged Networking 128  
bridgedifs 286  
Build instructions 81  
BusLogic 123

## C

---

Cache 30  
capture 309  
CD/DVD-Abbild 65, 96, 152, 204  
cdrom 306  
chgrp 275  
Chipsatz 121  
chmod 275  
Citrix XenCenter 51  
Citrix XenServer 50  
clipboard 290  
clonehd 297  
closed source 69  
closemedium 295  
COM 235  
comment 296  
compact 297  
Composite 226  
configure 83  
Connectix 63  
controlvm 291  
convertfromraw 270, 298  
converthd 210

convertSettings 308  
convertSettingsBackup 308  
CPU 25, 29  
CPU-Sockel 25  
CPU-Temperatur 277  
cpus 290  
createhd 295  
createvm 288  
Creative Soundblaster 16 123

**D**

---

Datenspeicher 25, 32  
dbg 285  
dd 270, 298  
Debian 79  
delete 288  
details 287  
detecthostkey 307  
Device 141  
DHCP 313  
    *DHCP-Server* 126, 130  
    *dhcpcserver* 304  
    *dhcpcservers* 286  
Dienste 45  
Direct3D 66, 171, 227  
discard 214, 293  
discardcurrent 215, 216, 293  
discardstate 219, 293, 308  
Disketten 32  
    *-abbilder* 97  
    *-Images* 205  
DNS 313  
Domain Name System 313  
Download 71  
Dualprozessor 53  
dvdattach 292  
dvds 286  
Dynamic Host Configuration Protocol 313

**E**

---

e1000 247  
edit 293  
EHCI 241  
Eingabe 91  
Eingabeaufforderung 102  
Eingabegeräte 25, 33

Einsatzgebiet 65  
Emulation 39  
Emulatoren 40  
Enhanced Host Controller Interface 241  
erweiterter Host-Controller 241  
Erzeuge Pipe 237  
evdevkeymap 111, 275, 307  
export 290  
Extern 249

**F**

---

fda 305  
Fehlermeldungen 109  
Fernsteuerung 146  
feste Größe 201  
Festplattenimage 149, 200, 206, 313, 314  
filename 310  
Filter 243  
Fixed 296  
fixedmode 306  
floppies 286  
Floppy-Images 65  
Fokus 279  
format 296  
fullscreen 306  
fullscreenresize 306

**G**

---

G++ 166  
Gast 249  
Gastbetriebssystem 313  
Gasterweiterungen 132, 155, 165, 177  
Gastsystem 45, 66, 313  
Gateway 313  
GCC 80  
Gemeinsame Ordner 65, 178, 221  
getextradata 299  
Globale Einstellungen 90  
Glossar 313  
GNU Compiler Collection 80, 166  
GParted 56  
Grafikkarte 25, 32, 280  
Grafikprobleme 280  
Grafische Oberfläche 88  
Graphical User Interface 88  
Großrechner 23  
groupadd 80, 275



GRUB 277  
 Grundkenntnisse 19  
 Gruppe 80  
 GuestPort 247  
 guesttohost 290  
 GUI 88

## H

---

Hardware 60, 119  
 Hardwareemulator 39  
 Hauptplatine 25  
 Hauptspeicher 25, 31  
 hda 305  
 hddbackends 286  
 hdds 286  
 hdfolder 300  
 height 309  
 Heimcomputer 23  
 High Availability 49  
 Hilfe 101  
 Hinweise 21  
 Hochverfügbarkeit 49  
 Homecomputer 23  
 hostdvs 286  
 hostfloppies 286  
 hostinfo 286  
 Host-only Netzwerk 130  
 hostonlyif 304  
 hostonlyifs 286  
 hostpath 301  
 Host-Pipe 236  
 HostPort 247  
 Host-Schnittstelle 236  
 Hostsystem 36, 45, 52, 66, 313  
 Host-Taste 91, 113  
 hosttguest 290  
 hwvrtex 308  
 HyperTerminal 237  
 Hypervisor 45

## I

---

I/O-APIC 232, 313  
 IBM 24  
 IDE 26, 122  
 idecontroller 290  
 IDE-kompatibel 281  
 ifname 304

immutable 294, 297  
 Import 259  
 import 290  
 Infobereich 90  
 innotek GmbH 63  
 Intel 24  
   AC97 123  
   PRO/1000 125  
   Virtualization Technology 30  
   VT 30, 45, 232  
 Internes Netzwerk 129, 185  
 ip 304  
 ipconfig 184  
 ISO 204  
 ISO-Image 152

## K

---

Kernel 120, 274, 313  
   -Header 80, 166  
   -Sourcen 166  
   -Update 274  
 klonen 254, 297  
 kmk 83  
 Kommandozeilentools 102, 283  
 Kompilierung 81  
 Komplettsicherung 255  
 Konfiguration 97  
 Konfigurationsdatei 93, 116, 276  
 Konkurrenten 47  
 Konsole 102  
 Kontextmenü 313  
 konvertieren 210  
 kopieren 254  
 Kurzform 284

## L

---

Langform 284  
 Lastverteilung 49  
 Laufzeit-Attribute 166  
 legacy09 291  
 List 107  
 list 286  
 Live-CD 135, 158, 313  
 Lizenzen 46  
 Load Balancing 49  
 loghistorycount 300  
 löschen 252

lowerip 305  
LSI-Logic 123

**M**

---

machinefolder 300  
machinereadable 287  
Magnetscheiben 32  
Mainboard 25  
make 80, 166  
Manager für virtuelle Medien 95, 152, 199  
memory 306  
menu.lst 277  
metrics list 303  
Microsoft Virtual PC 52  
Migration 210, 259  
Mikrocomputer 23  
Mindestanforderungen 53  
MIPS 29  
modifyhd 296  
modifyvm 289  
Monitor 229  
monitorcount 230, 290  
Monochrome Display Adapter (MDA) 32  
mount 224  
MSI 72  
msiexec 73  
Multi-Head 229  
Multiprozessorsystem 232

**N**

---

Nahtloser Modus 224, 279  
Namensauflösung 313  
NAT 126, 158, 247  
net 179  
netmask 305  
netname 304  
Network Address Translation 126  
Netzwerk 59, 123  
Netzwerkadapter 33, 124  
Netzwerkbrücke 128, 182, 274  
Netzwerkkarte 33, 124  
Netzwerkmaske 313  
Netzwerkname 129, 188  
Netzwerkvirtualisierung 46  
no-debug 285  
nofstoggle 307

nograbonclick 307  
nohostkey 307  
nohwvrtex 308  
nologo 285  
norawr0 308  
norawr3 308  
noresize 307  
Notepad++ 93  
NTFS 268  
null 249  
Nur Lesbar 178

**O**

---

OEM-Software 313  
Open Source Edition 68  
Open Virtualization Format 66, 133, 261  
OpenGL 66, 123, 226  
openmedium 294  
Oracle 65  
Ordner-Name 178  
Ordner-Pfad 178  
Orginal Equipment Manufacture 313  
OSE 68  
ostype 288, 290  
ostypes 286  
output 291  
OVF 133, 261

**P**

---

Parameter 283  
Paravirtualisierung 46  
Parted Magic 55  
Partimage 267  
Partition 55  
Passthrough 141, 280  
Passwort 249  
Path-Variable 103  
pattern 302  
pause 292  
PC 266  
PCI (Peripheral Component Interconnect) 27  
PCIe (Peripheral Component Interconnect Express) 27  
pcnet 247  
PCnet-FAST III 125  
PCnet-PCI II 125

Peripherie 27  
 Peripheriegeräte 25, 34, 314  
     *externe* 34  
     *interne* 34  
 Personal Computer 22, 24  
 Personal Use and Evaluation License 69  
 Pipe 236  
 Planung 52  
 Platzhalter 283  
 Port-Forwarding 247  
 POST (Power On Self Test) 29  
 poweroff 292  
 Praxisbeispiele 135  
 Pro und Kontra 38  
 Protocol 247  
 Prozessor 25, 29, 53, 232  
 PUEL 69

## R

---

rawr0 308  
 rawr3 308  
 rdesktop-vrdp 114, 246, 310  
 RDP 66  
 RDP-Client 114  
 RDP-Zugriff 146  
 readonly 301  
 Recheneinheit 30  
 Rechteprobleme 275  
 redirection 311  
 Register 30  
 register 288  
 registerimage 294  
 registervm 251, 288  
 registrieren 251  
 Registrierung 77  
 remember 296  
 Remote Desktop Protocol 66  
 Remotezugriff 146  
 reparieren 273  
 reset 292  
 resize 206  
 Ressourcen 45, 60  
 resume 292  
 Router 314  
 RPD-Authentisierung 249

Ruhezustand 146, 168, 216  
 runningvms 286

## S

---

SATA (Serial Advanced Technology Attachment) 26, 122, 281  
 sataideemulation 281  
 sav 217  
 savestate 217, 292  
 scsi 290  
 SCSI (Small Computer System Interface) 26, 123  
 scsitype 290  
 sdelete 208  
 Seamless Windows 224  
 Serial ATA 122  
 Serielle Schnittstellen 235  
 Serverport 147  
 Servervirtualisierung 38  
 Session-Information 166  
 setcredentials 292  
 setextradata 247, 299  
 setlinkstate 292  
 setproperty 299  
 settingsfile 288  
 setvideomodehint 292  
 Shared Folders 221  
 sharedfolder 222  
 sharedfolder add 301  
 sharedfolder remove 301  
 showhdinfo 295  
 showvminfo 213, 287, 293  
 Sichern 254  
 Sicherungsimagen 267  
 Sicherungspunkt 192  
 Sicherungspunkte 99, 186, 211, 278  
 silent install 253  
 size 296  
 SMP 232, 314  
 snapshot 293  
 Snapshots 99, 186, 211  
 socat 237  
 Sockel 29  
 Socket-Datei 237  
 Softwareemulator 39  
 Sondertasten 274  
 Soundblaster 123  
 Soundkarte 123, 172

source 83  
sources.list 79  
Southbridge 314  
Speicher 54  
Speicherbausteine 26  
Speicherriegel 26, 31  
Sprache 92  
SSH 248  
Standardtreiber 266  
startvm 111, 285, 291, 305, 309  
statistics 287  
Steckplatz 29  
Straftat 47  
Sun 64  
Symbolleiste 90  
Symmetrisches Multiprozessorsystem 232, 314  
Syntaxfehler 109  
systemproperties 287  
Systemupdate 274

T

---

take 213, 293  
Tastatur 274  
Tastenkombination 257, 279  
Temperatur 277  
Temporäre Änderungen 113  
termacpi 113, 307  
Terminal 23  
transient 222, 301  
Treiber 132, 280, 281  
tsclient 148  
Typ des Gastbetriebssystems 137  
type 294, 296

U

---

Umgebungsvariablen 83, 103  
Universally Unique Identifier 107, 117  
unregisterimage 295  
unregistervm 251, 288  
Update 92  
upperip 305  
USB (Universal Serial Bus) 27, 59, 239, 275  
    über RDP 246  
    USB 2.0 240

USB (Universal Serial Bus) (Forts.)  
    usbattach 243, 292  
    usbdetach 243, 292  
    usbehci 240  
    USB-Filter 243, 287  
    usbfilter modify 300  
    usbfilter remove 300  
    usbfilters 287  
    usbhost 242, 286  
usermod 275  
UUID 107, 117, 277  
uuid 288

V

---

Vanderpool 30  
variant 296  
vboxdrv 274  
VBoxHeadless 113, 195, 229  
VboxHeadless 308  
VBoxManage 106, 285  
VBoxSDL 110, 274, 305  
vboxsf 224  
vboxsvr 224  
vboxusers 80, 275  
VDI 64, 65, 296  
VDI Software 64  
vergrößern 206  
verkleinern 208  
Version 137  
version 285  
Versionen 67  
VHD 296  
VICE (Versatile Commodore Emulator) 40  
Video 196  
Virtual Appliances 133, 261  
Virtual Center 49  
Virtual Desktop Infrastructure 64  
Virtual Disk Image 64, 65  
Virtual Machine Monitor 45  
Virtual PC 52, 210  
VirtualBox 63, 65, 284  
    VirtualBox.xml 93  
    Command Line Management Interface 106  
    Open Source Edition 68  
    OSE 80  
    Shared Folders 223  
Virtualisierung 39, 43