

Fedora 12

Virtualisierungshandbuch

Das definitive Handbuch zur Virtualisierung unter Fedora



Christoph Curran

Fedora 12 Virtualisierungshandbuch

Das definitive Handbuch zur Virtualisierung unter Fedora

Ausgabe 1

Autor

Christoph Curran

ccurran@redhat.com

Copyright © 2009 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at <http://creativecommons.org/licenses/by-sa/3.0/>. The original authors of this document, and Red Hat, designate the Fedora Project as the "Attribution Party" for purposes of CC-BY-SA. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

For guidelines on the permitted uses of the Fedora trademarks, refer to https://fedoraproject.org/wiki/Legal:Trademark_guidelines.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

All other trademarks are the property of their respective owners.

Das Fedora 12 Virtualisierungshandbuch beinhaltet Informationen rund um Installation, Konfiguration, Verwaltung, Tipps und Tricks sowie Suche und Beseitigung von Fehlern (Troubleshooting) von Virtualisierungstechnologien, die in Fedora 12 zur Anwendung kommen.

Vorwort	vii
1. Über dieses Buch	vii
2. Dokumentkonventionen	vii
2.1. Typografische Konventionen	vii
2.2. Pull-Quote-Konventionen	ix
2.3. Anmerkungen und Warnungen	x
3. Wir freuen uns auf Ihr Feedback!	x
I. Installation	1
1. Installation der Virtualisierungspakete	3
1.1. Installation von KVM während einer neuen Fedora-Installation	3
1.2. Installation von KVM-Paketen auf einem vorhandenen Fedora-System	5
2. Überblick über die Installation virtualisierter Gäste	7
2.1. Erzeugen von Gästen mit virt-install	7
2.2. Erzeugen von Gästen mit virt-manager	8
2.3. Installation von Gästen mit PXE	16
3. Installationsverfahren für Gastbetriebssysteme	23
3.1. Installation von Red Hat Enterprise Linux 5 als paravirtualisierter Gast	23
3.2. Installation von Red Hat Enterprise Linux als voll virtualisierter Gast	65
3.3. Installation von Windows XP als voll virtualisierter Gast	74
3.4. Installation von Windows Server 2003 als voll virtualisierter Gast	93
3.5. Installation von Windows Server 2008 als voll virtualisierter Gast	96
II. Configuration	109
4. Virtualisierte Blockgeräte	111
4.1. Anlegen eines virtualisierten Floppy-Disk-Controllers	111
4.2. Hinzufügen von Speichergeräten zum Gast	112
4.3. Konfiguration von persistentem Speicher	115
4.4. Hinzufügen eines virtualisierten CD-ROM- oder DVD-Laufwerks zu einem Gast.....	118
5. Gemeinsam verwendeter Speicher und Virtualisierung	119
5.1. Verwenden von iSCSI zur Speicherung von Gästen	119
5.2. Verwenden von NFS zur Speicherung von Gästen	119
5.3. Verwenden von GFS2 zur Speicherung von Gästen	119
6. Beste Verfahren für Server	121
7. Sicherheit für Virtualisierung	123
7.1. SELinux und Virtualisierung	123
7.2. Hinweise in Zusammenhang mit SELinux	124
8. Netzwerkkonfiguration	127
8.1. Network Address Translation (NAT) mit libvirt	127
8.2. Bridged-Netzwerk mit libvirt	128
9. KVM paravirtualisierte Treiber	131
9.1. Installation der KVM Windows paravirtualisierten Treiber	131
III. Administration	141
10. Gästeverwaltung mit Hilfe von xend	143

11. Zeitverwaltung bei KVM-Gästen	145
12. KVM Live-Migration	149
12.1. Voraussetzungen der Live-Migration	149
12.2. Beispiel für gemeinsam genutzten Speicher: NFS für eine einfache Migration	150
12.3. KVM Live-Migration mit virsh	151
12.4. Migration mit virt-manager	152
13. Remote-Verwaltung virtualisierter Gäste	165
13.1. Remote-Verwaltung mit SSH	165
13.2. Remote-Verwaltung über TLS und SSL	166
13.3. Transportmodi	167
IV. Referenzhandbuch zur Virtualisierung	173
14. Virtualisierungs-Tools	175
15. Das Verwalten von Gästen mit virsh	179
16. Das Verwalten von Gästen mit dem Virtual Machine Manager (virt-manager)	189
16.1. Das Fenster "Verbindung öffnen"	189
16.2. Das Hauptfenster des Virtual Machine Managers	190
16.3. Das Detail-Fenster des Virtual Machine Managers	191
16.4. Die grafische Konsole der virtuellen Maschine	192
16.5. Starting virt-manager	193
16.6. Wiederherstellen einer gespeicherten Maschine	194
16.7. Anzeigen von Gastdetails	195
16.8. Überwachen des Status	200
16.9. Anzeigen der Gast-Identifizierung	202
16.10. Anzeigen des Gaststatus	203
16.11. Anzeigen virtueller CPUs	204
16.12. Anzeigen der CPU-Auslastung	205
16.13. Anzeigen des Speicherverbrauchs	207
16.14. Verwalten eines virtuellen Netzwerks	208
16.15. Erstellen eines virtuellen Netzwerks	209
V. Tips and Tricks	219
17. Tipps und Tricks	221
17.1. Gäste automatisch starten	221
17.2. Wechseln zwischen dem KVM- und Xen-Hypervisor	221
17.2.1. Xen zu KVM	221
17.2.2. KVM zu Xen	223
17.3. Verwenden von qemu-img	224
17.4. Overcommitting mit KVM	226
17.5. Modifizieren von /etc/grub.conf	228
17.6. Überprüfen der Virtualisierungserweiterungen	229
17.7. Identifizieren des Gasttyps und der Implementierung	230
17.8. Generieren einer neuen, eindeutigen MAC-Adresse	230
17.9. Very Secure ftpd	231
17.10. Konfiguration von LUN-Persistenz	232
17.11. Abschalten der SMART-Disk Überwachung von Gästen	234
17.12. Klonen von Gastkonfigurationsdateien	234
17.13. Kopieren eines existierenden Gasts und seiner Konfigurationsdatei	234

18. Erstellung angepasster libvirt-Skripte	237
18.1. Benutzung von XML-Konfigurationsdateien mit virsh	237
VI. Troubleshooting	239
19. Troubleshooting	241
19.1. Fehler bei Loop-Gerät	241
19.2. Aktivieren der Intel VT und AMD-V Virtualisierungs-Hardware-Erweiterungen im BIOS	241
A. Zusätzliche Informationsquellen	243
A.1. Online-Informationsquellen	243
A.2. Installierte Dokumentation	243
B. Versionsgeschichte	245
C. Kolophon	247
Glossar	249

Vorwort

Dieses Fedora 12 Virtualisierungshandbuch behandelt alle Aspekte der Verwendung und Verwaltung von Virtualisierungstechnologien unter Fedora 12.

1. Über dieses Buch

Dieses Buch ist in sieben Teile aufgeteilt:

- Systemvoraussetzungen
- Installation
- Configuration
- Administration
- Glossar
- Tips and Tricks
- Troubleshooting

2. Dokumentkonventionen

Dieses Handbuch verwendet mehrere Konventionen, um bestimmte Wörter und Phrasen hervorzuheben und Aufmerksamkeit auf spezifische Teile von Informationen zu lenken.

In PDF- und Papierausgaben verwendet dieses Handbuch Schriftbilder des *Liberation-Fonts*¹-Sets. Das Liberation-Fonts-Set wird auch für HTML-Ausgaben verwendet, falls es auf Ihrem System installiert ist. Falls nicht, werden alternative, aber äquivalente Schriftbilder angezeigt. Beachten Sie: Red Hat Enterprise Linux 5 und die nachfolgende Versionen beinhalten das Liberation-Fonts-Set standardmäßig.

2.1. Typografische Konventionen

Es werden vier typografische Konventionen verwendet, um die Aufmerksamkeit auf spezifische Wörter und Phrasen zu lenken. Diese Konventionen und ihre Umstände, unter denen sie auftreten, sind folgende:

Mono-spaced Bold

Dies wird verwendet, um Systemeingaben hervorzuheben, einschließlich Shell-Befehle, Dateinamen und Pfade. Es wird ebenfalls zum Hervorheben von Tasten und Tastenkombinationen verwendet. Zum Beispiel:

To see the contents of the file **my_next_bestselling_novel** in your current working directory, enter the **cat my_next_bestselling_novel** command at the shell prompt and press **Enter** to execute the command.

The above includes a file name, a shell command and a key cap, all presented in Mono-spaced Bold and all distinguishable thanks to context.

¹ <https://fedorahosted.org/liberation-fonts/>

Key-combinations can be distinguished from key caps by the hyphen connecting each part of a key-combination. For example:

Press **Enter** to execute the command.

Press **Ctrl+Alt+F1** to switch to the first virtual terminal. Press **Ctrl+Alt+F7** to return to your X-Windows session.

The first sentence highlights the particular key cap to press. The second highlights two sets of three key caps, each set pressed simultaneously.

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in **Mono-spaced Bold**. For example:

File-related classes include **filesystem** for file systems, **file** for files, and **dir** for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialogue box text; labelled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

Choose **System > Preferences > Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, click the **Left-handed mouse** check box and click **Close** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).

To insert a special character into a **gedit** file, choose **Applications > Accessories > Character Map** from the main menu bar. Next, choose **Search > Find...** from the **Character Map** menu bar, type the name of the character in the **Search** field and click **Next**. The character you sought will be highlighted in the **Character Table**. Double-click this highlighted character to place it in the **Text to copy** field and then click the **Copy** button. Now switch back to your document and choose **Edit > Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in Proportional Bold and all distinguishable by context.

Note the **>** shorthand used to indicate traversal through a menu and its sub-menus. This is to avoid the difficult-to-follow 'Select **Mouse** from the **Preferences** sub-menu in the **System** menu of the main menu bar' approach.

Mono-spaced Bold Italic or ***Proportional Bold Italic***

Whether Mono-spaced Bold or Proportional Bold, the addition of Italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

To connect to a remote machine using ssh, type **ssh *username@domain.name*** at a shell prompt. If the remote machine is **example.com** and your username on that machine is john, type **ssh *john@example.com***.

The **mount -o remount *file-system*** command remounts the named file system. For example, to remount the **/home** file system, the command is **mount -o remount /home**.

To see the version of a currently installed package, use the **rpm -q *package*** command. It will return a result as follows: ***package-version-release***.

Note the words in bold italics above — username, domain.name, file-system, package, version and release. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

When the Apache HTTP Server accepts requests, it dispatches child processes or threads to handle them. This group of child processes or threads is known as a *server-pool*. Under Apache HTTP Server 2.0, the responsibility for creating and maintaining these server-pools has been abstracted to a group of modules called *Multi-Processing Modules (MPMs)*. Unlike other modules, only one module from the MPM group can be loaded by the Apache HTTP Server.

2.2. Pull-Quote-Konventionen

Zwei, im Allgemeinen mehrzeilige, Daten-Typen werden visuell vom umliegenden Text hervorgehoben.

Eine Ausgabe, die an das Terminal gesendet wird, wird durch Mono-spaced Roman definiert und daher wie folgt präsentiert:

```
books      Desktop  documentation  drafts  mss    photos  stuff  svn
books_tests Desktop1  downloads      images  notes  scripts svgs
```

Quellcode-Auflistungen werden auch durch Mono-spaced Roman definiert. Sie werden wie folgt präsentiert und hervorgehoben:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
    public static void main(String args[])
        throws Exception
    {
        InitialContext iniCtx = new InitialContext();
        Object          ref    = iniCtx.lookup("EchoBean");
        EchoHome        home   = (EchoHome) ref;
        Echo             echo   = home.create();

        System.out.println("Created Echo");

        System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
    }
}
```

}

2.3. Anmerkungen und Warnungen

Abschließend verwenden wir drei visuelle Stile, um die Aufmerksamkeit auf Informationen zu lenken die anderenfalls vielleicht übersehen werden könnten.



Anmerkung

Eine Anmerkung ist ein Tipp oder ein Shortcut oder ein alternativer Ansatz für die vorliegende Funktion. Das Ignorieren von Anmerkungen sollte keine negativen Auswirkungen haben, aber Sie verpassen so vielleicht einen Trick der Ihnen das Leben vereinfacht.



Wichtig

Die Wichtig-Schaukästen lenken die Aufmerksamkeit auf Dinge die sonst leicht übersehen werden können: Konfigurationsänderungen die nur für die aktuelle Sitzung gelten oder Dienste für die ein Neustart nötig ist, bevor ein Update gültig ist. Das Ignorieren von Wichtig-Schaukästen würde keinen Datenverlust verursachen, aber unter Umständen zu Irritation und Frustration führen.



Warnung

Eine Warnung sollte nicht ignoriert werden. Das Ignorieren von Warnungen führt mit hoher Wahrscheinlichkeit zu Datenverlust.

3. Wir freuen uns auf Ihr Feedback!

Wenn Sie einen Fehler in diesem Handbuch finden oder eine Idee haben, wie dieses verbessert werden könnte, freuen wir uns über Ihr Feedback! Reichen Sie einen Fehlerbericht für die Komponente **Fedora Documentation**. in Bugzilla unter <http://bugzilla.redhat.com/bugzilla/> ein.

Vergewissern Sie sich beim Einreichen eines Fehlerberichts die Kennung des Handbuchs mit anzugeben: *Virtualization_Guide*

Falls Sie uns einen Vorschlag zur Verbesserung der Dokumentation senden möchten, sollten Sie hierzu möglichst genaue Angaben machen. Wenn Sie einen Fehler gefunden haben, geben Sie bitte die Nummer des Abschnitts und einen Ausschnitt des Textes an, damit wir diesen leicht finden können.

Teil I. Installation

Themen der Virtualisierungsinstallation

Diese Kapitel beschreiben das Einrichten des Hosts und die Installation virtualisierter Gäste mit Fedora. Es wird empfohlen, diese Kapitel sorgfältig durchzulesen, um eine erfolgreiche Installation von virtualisierten Gastbetriebssystemen zu gewährleisten.

Installation der Virtualisierungspakete

1.1. Installation von KVM während einer neuen Fedora-Installation

Dieser Abschnitt behandelt die Installation der Virtualisierungs-Tools und des KVM-Pakets als Teil einer neuen Fedora 12 Installation.



Benötigen Sie Hilfe bei der Installation?

Das *Fedora 12 Installationshandbuch* (verfügbar unter <http://docs.fedoraproject.org>) behandelt die Fedora 12 Installation im Detail.

1. Starten Sie eine interaktive Fedora-Installation von der Fedora 12 Installations-CD-ROM, DVD oder PXE.
2. Führen Sie alle weiteren Schritte aus, bis Sie an den Schritt zur Paketauswahl gelangen.

RED HAT ENTERPRISE LINUX 5

The default installation of Red Hat Enterprise Linux Server includes a set of software applicable for general internet usage. What additional tasks would you like your system to include support for?

- Clustering
- Software Development
- Storage Clustering
- Virtualization
- Web server

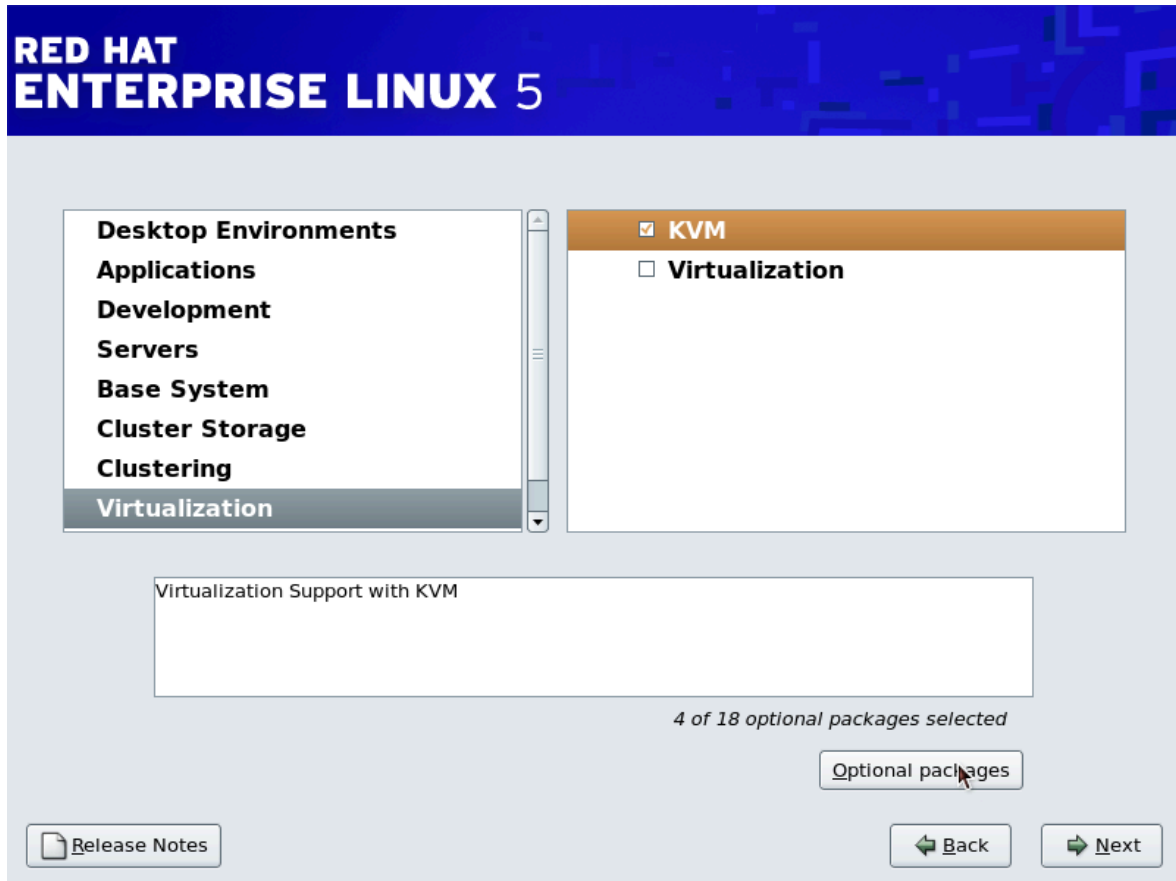
You can further customize the software selection now, or after install via the software management application.

Customize later Customize now

[Release Notes](#) [Back](#) [Next](#)

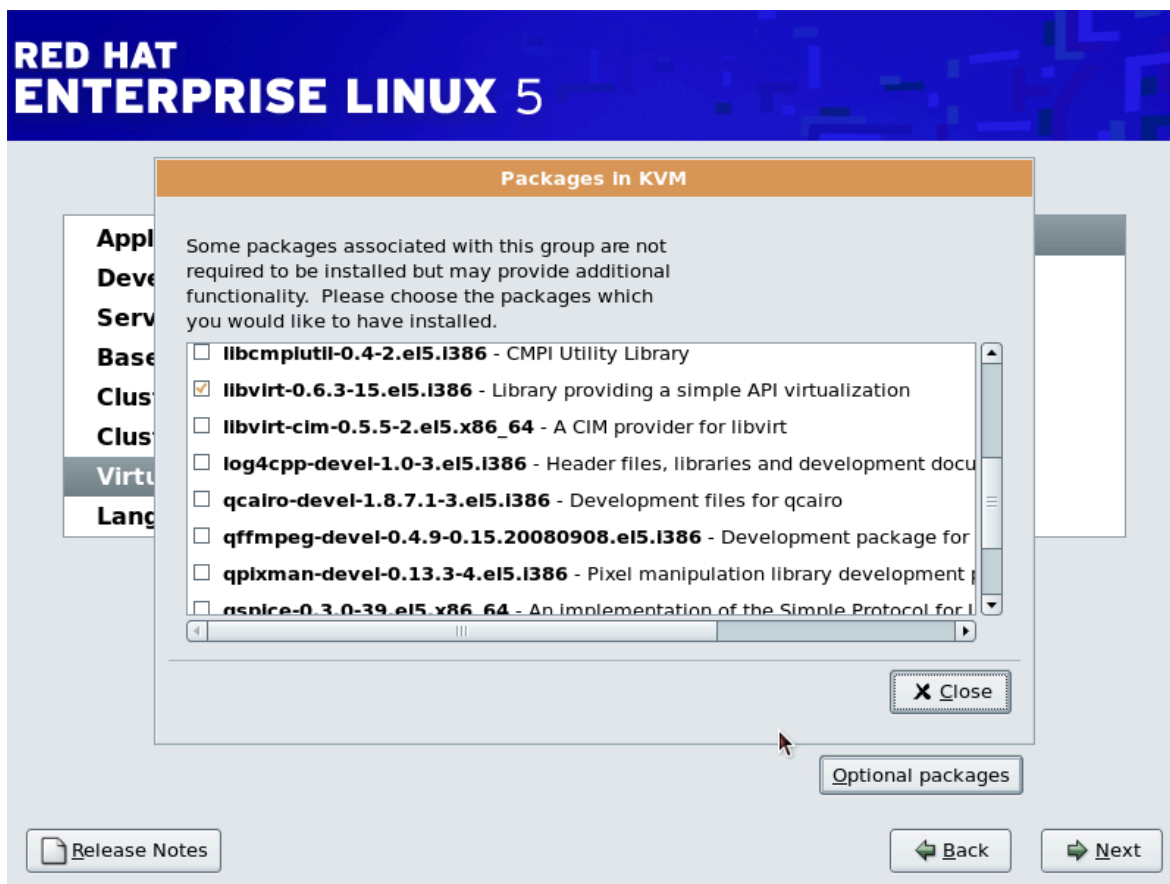
Wählen Sie die Paketgruppe **Virtualisierung** und die Option **Jetzt anpassen**.

3. Wählen Sie die **KVM**-Paketgruppe. Heben Sie die Auswahl der Paketgruppe **Virtualisierung** auf. Dadurch wird der KVM-Hypervisor sowie **virt-manager**, **libvirt** und **virt-viewer** zur Installation ausgewählt.



4. **Anpassen der Pakete (falls nötig)**

Passen Sie die Gruppe **Virtualisierung** an, falls Sie andere Virtualisierungspakete benötigen.



Klicken Sie auf **Schließen**, dann **Weiter**, um mit der Installation fortzufahren.

Installation von KVM-Paketen mit Kickstart-Dateien

In diesem Abschnitt wird beschrieben, wie Sie Kickstart-Dateien dazu verwenden können, Fedora mit den KVM-Hypervisor-Paketen zu installieren. Kickstart-Dateien erlauben umfangreiche, automatisierte Installationen, ohne dass jedes einzelne System manuell installiert werden muss. Die Schritte in diesem Abschnitt sollen Ihnen dabei helfen, unter Verwendung einer Kickstart-Datei Fedora mit den Virtualisierungspaketen zu installieren.

Fügen Sie im %packages-Abschnitt Ihrer Kickstart-Datei die folgende Paketgruppe hinzu:

```
%packages
@kvm
```

Mehr Informationen über Kickstart-Dateien finden Sie auf der Red Hat Website <http://docs.fedoraproject.org> im *Fedora 12 Installationshandbuch*.

1.2. Installation von KVM-Paketen auf einem vorhandenen Fedora-System

Dieser Abschnitt beschreibt die Schritte zur Installation des KVM-Hypervisors auf einem laufenden Fedora 12 System oder höher.

Installation des KVM-Hypervisors mit yum

Um Virtualisierung unter Fedora einzusetzen, benötigen Sie das **kvm**-Paket. Das **kvm**-Paket beinhaltet das KVM-Kernel-Modul, das den KVM-Hypervisor auf dem standardmäßigen Linux-Kernel liefert.

Um das **kvm**-Paket zu installieren, führen Sie Folgendes aus:

```
# yum install kvm
```

Installieren Sie nun die zusätzlichen Virtualisierungspakete zur Verwaltung.

Empfohlene Virtualisierungspakete:

python-virtinst

Liefert den **virt-install**-Befehl für die Erzeugung von virtuellen Maschinen.

libvirt

libvirt ist eine API-Bibliothek zur Interaktion mit Hypervisoren. **libvirt** verwendet das **xm**-Virtualisierungs-Framework und das **virsh**-Befehlszeilen-Tool, um virtuelle Maschinen zu verwalten und zu steuern.

libvirt-python

Das **libvirt-python**-Paket beinhaltet ein Modul, das in Python programmierten Anwendungen die Möglichkeit gibt, die von der **libvirt**-Bibliothek gelieferte Schnittstelle zu verwenden.

virt-manager

virt-manager, auch **Virtual Machine Manager** genannt, bietet ein grafisches Tool zur Verwaltung virtueller Maschinen. Es verwendet die **libvirt**-Bibliothek als Management-API.

Installieren Sie die anderen empfohlenen Virtualisierungspakete:

```
# yum install virt-manager libvirt libvirt-python python-virtinst
```

Überblick über die Installation virtualisierter Gäste

Nachdem Sie die Virtualisierungspakete auf dem Host-System installiert haben, können Sie Gastbetriebssysteme installieren. Dieses Kapitel beschreibt die allgemeinen Verfahren zur Installation von Gastbetriebssystemen auf virtuellen Maschinen. Sie können Gäste erzeugen durch Klick auf die **Neu**-Schaltfläche im **virt-manager** oder mit Hilfe der Befehlszeilenschnittstelle **virt-install**. Beide Methoden werden in diesem Kapitel behandelt.

Für spezifische Versionen von Fedora, anderen Linux-Distributionen, Solaris und Windows stehen detaillierte Installationsanweisungen zur Verfügung. Siehe [Kapitel 3, Installationsverfahren für Gastbetriebssysteme](#) für diese Verfahren.

2.1. Erzeugen von Gästen mit virt-install

Sie können den **virt-install**-Befehl verwenden, um virtualisierte Gäste von der Befehlszeile aus zu erzeugen. **virt-install** kann entweder interaktiv verwendet werden oder in einem Skript, um die Erstellung virtueller Maschinen zu automatisieren. Wenn **virt-install** zusammen mit Kickstart-Dateien verwendet wird, können virtuelle Maschinen unbeaufsichtigt installiert werden.

Das **virt-install**-Tool bietet eine Reihe von Optionen, die in der Befehlszeile übergeben werden können. Um eine vollständige Auflistung dieser Optionen zu sehen, geben Sie ein:

```
$ virt-install --help
```

Auch die **virt-install**-Handbuchseite dokumentiert jede Befehlszeilenoption sowie wichtige Variablen.

qemu-img ist ein zugehöriger Befehl, der vor **virt-install** dazu verwendet werden kann, Speicheroptionen zu konfigurieren.

Eine wichtige Option ist die **--vnc**-Option, mit der grafisches Fenster zur Gastinstallation geöffnet wird.

Dieses Beispiel erzeugt einen Red Hat Enterprise Linux 3 Gast namens *rhel3support* von einer CD-ROM, mit virtuellem Netzwerk und einem 5 GB dateibasiertem Blockgerätabbild. Dieses Beispiel verwendet den KVM-Hypervisor.

```
# virt-install --accelerate --hvm --connect qemu:///system \  
  --network network:default \  
  --name rhel3support --ram=756\  
  --file=/var/lib/libvirt/images/rhel3support.img \  
  --file-size=6 --vnc --cdrom=/dev/sr0
```

[Beispiel 2.1. Verwenden von virt-install mit KVM, um einen Red Hat Enterprise Linux 3 Gast zu erzeugen](#)

```
# virt-install --name Fedora11 --ram 512 --file=/var/lib/libvirt/images/  
Fedora11.img \  
    --file-size=3 --vnc --cdrom=/var/lib/libvirt/images/Fedora11.iso
```

Beispiel 2.2. Verwenden von virt-install, um einen Fedora 11 Gast zu erzeugen

2.2. Erzeugen von Gästen mit virt-manager

virt-manager, auch als Virtual Machine Manager bekannt, ist ein grafisches Tool zur Erstellung und Verwaltung virtualisierter Gäste.

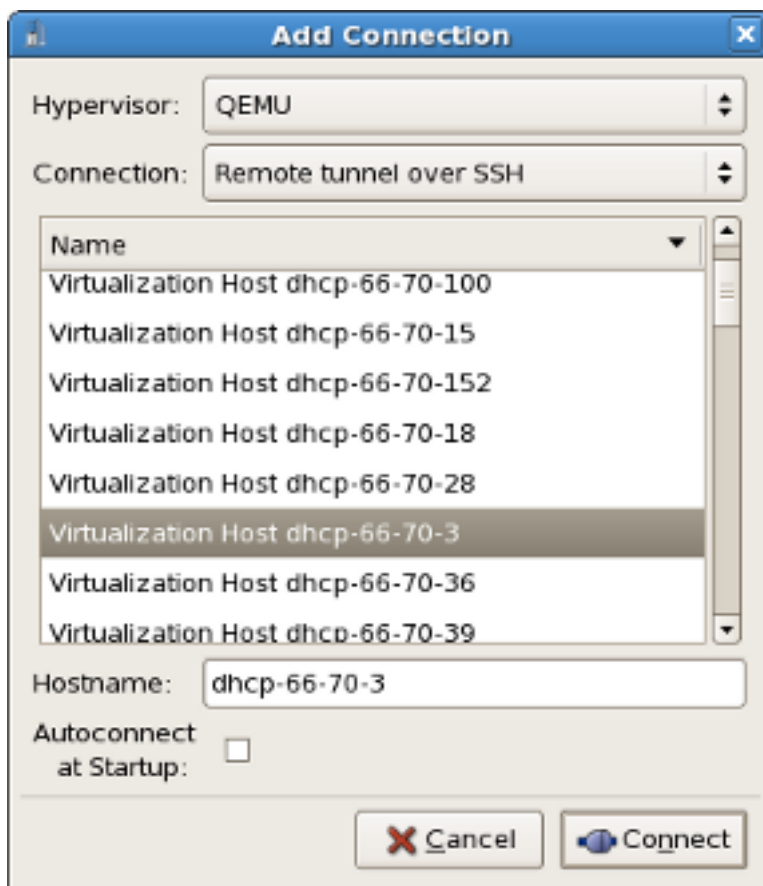
Prozedur 2.1. Erzeugen eines virtualisierten Gasts mit virt-manager

1. Um **virt-manager** zu starten, führen Sie als Root den folgenden Befehl aus:

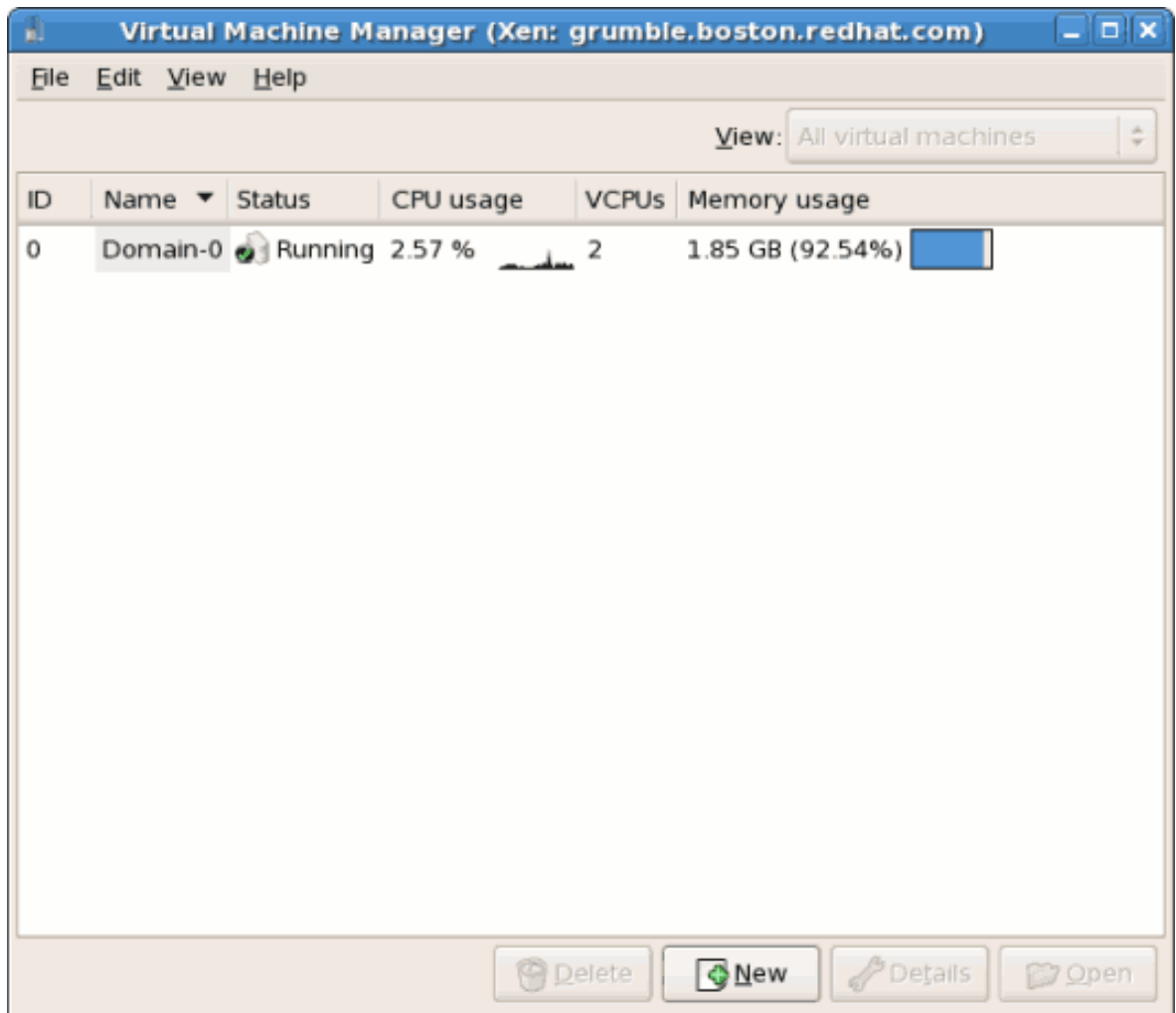
```
# virt-manager &
```

Der **virt-manager**-Befehl öffnet eine grafische Benutzeroberfläche. Ohne Root-Rechte oder konfiguriertem **sudo** stehen Ihnen eine Reihe von Funktionen, u. a. die Schaltfläche **Neu**, nicht zur Verfügung und Sie können keinen neuen virtualisierten Gast erzeugen.

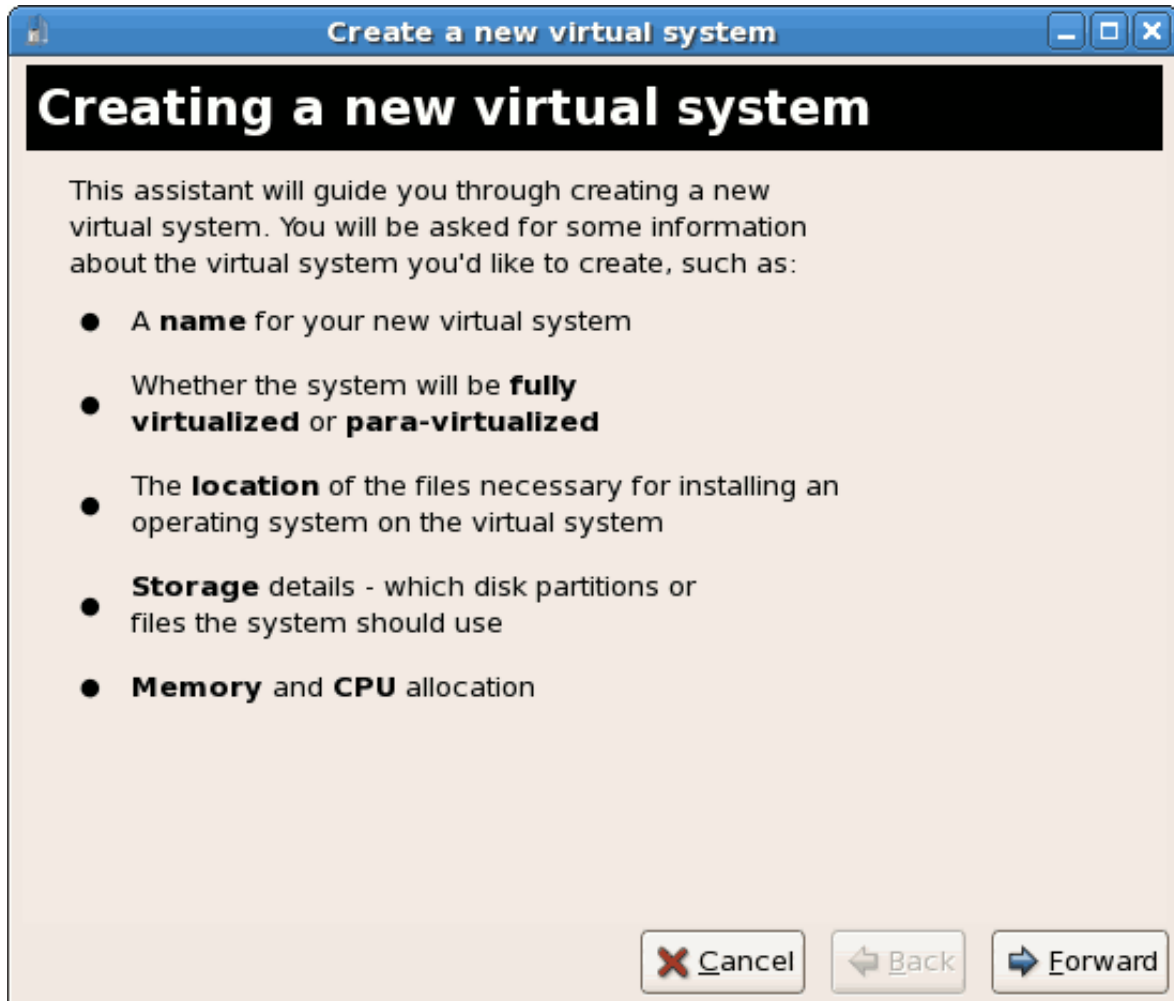
2. Wählen Sie **Datei -> Verbindung öffnen**. Das Dialogfenster unten erscheint. Wählen Sie einen Hypervisor und klicken Sie auf die Schaltfläche **Verbinden**:



3. Im **virt-manager**-Fenster können Sie eine neue virtuelle Maschine erstellen. Klicken Sie dazu auf die **Neu**-Schaltfläche. Dies öffnet den im nachfolgenden Screenshot gezeigten Assistenten.



4. Das Fenster **Erstellen eines neuen virtuellen Systems** enthält eine Zusammenfassung der Informationen, die Sie angeben müssen, um eine virtuelle Maschine zu erzeugen:

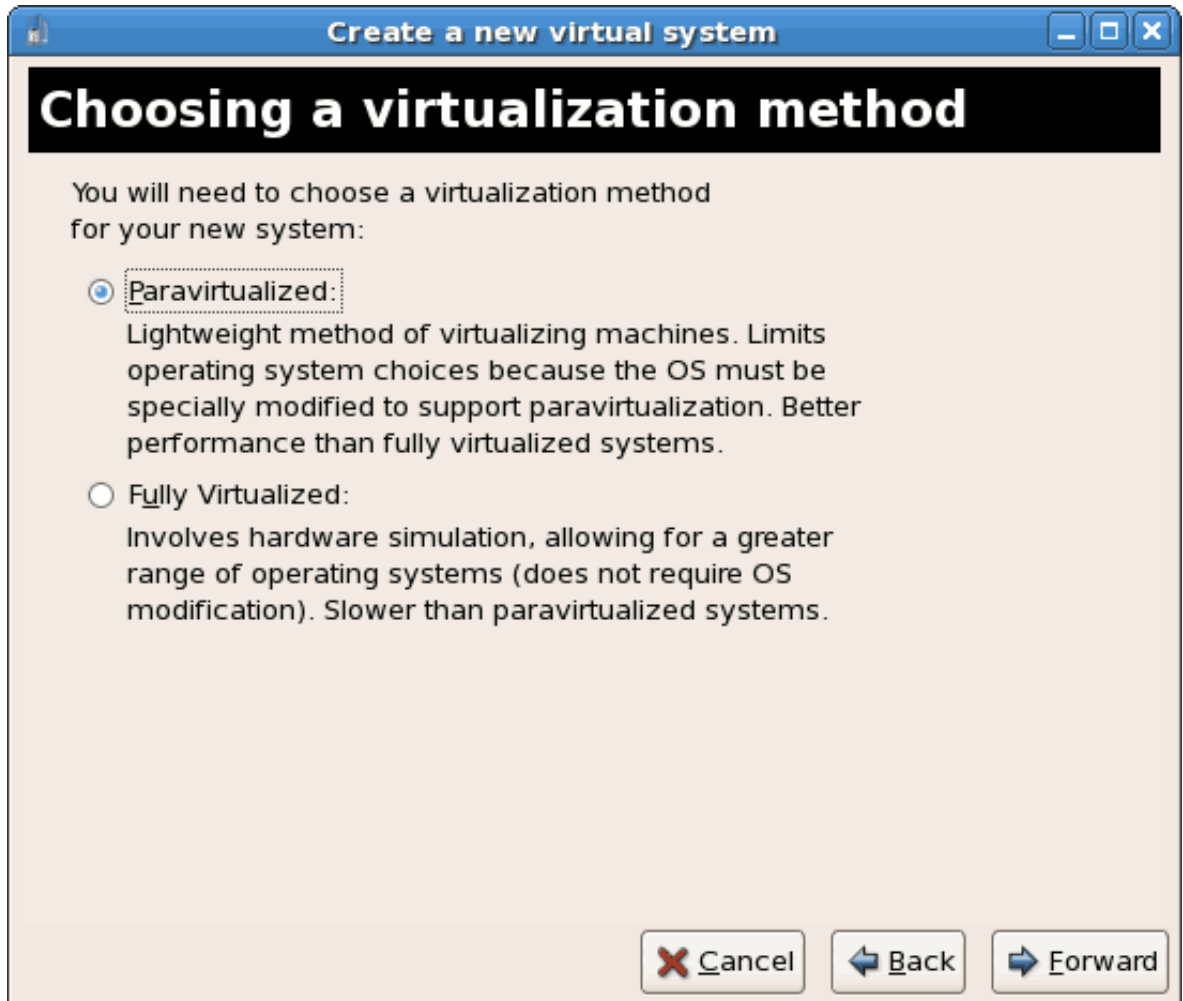


Überprüfen Sie die angegebenen Informationen für Ihre Installation und klicken anschließend auf **Weiter**.

5. Es erscheint nun das Fenster **Wählen einer Virtualisierungsmethode**. Wählen Sie zwischen **Paravirtualisiert** oder **Voll virtualisiert**.

Volle Virtualisierung setzt ein System mit Intel® VT oder AMD-V Prozessor voraus. Falls die Virtualisierungserweiterungen nicht vorhanden sind, sind die Optionen **Voll virtualisiert** oder **Kernel/Hardware-Beschleunigung aktivieren** nicht auswählbar. Die Option **Paravirtualisiert** ist grau hinterlegt, falls der **kerne1-xen** nicht der aktuell laufende Kernel ist.

Falls Sie mit einem KVM-Hypervisor verbunden sind, steht nur die volle Virtualisierung zur Auswahl.

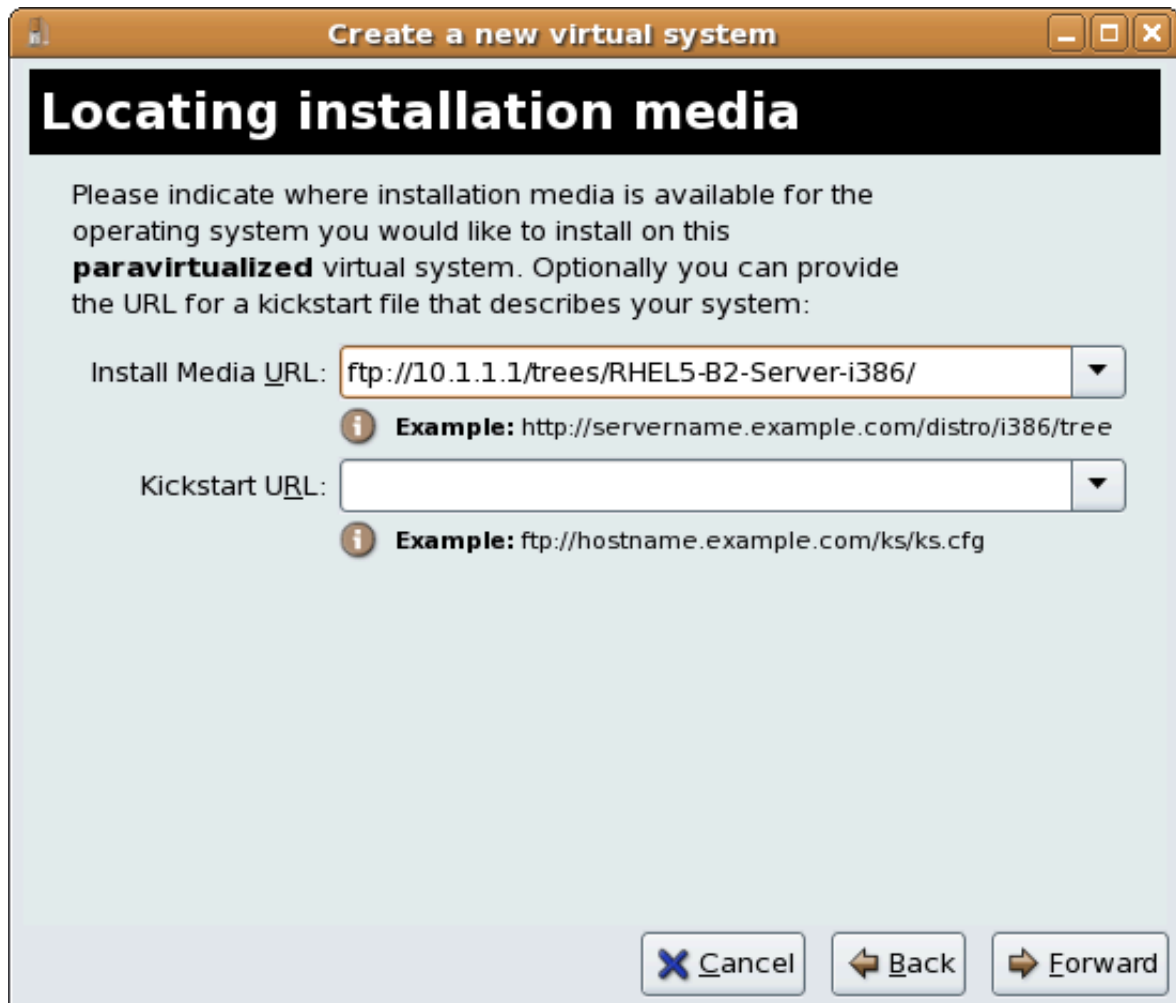


Wählen Sie den Virtualisierungstyp und klicken auf **Weiter**.

6. Der Prompt **Installationsmedium suchen** fragt nach dem Installationsmedium für den von Ihnen ausgewählten Installationstyp. Der Inhalt dieses Fensters hängt von der Auswahl im vorangegangenen Schritt ab.
 - a. Die paravirtualisierte Installation erfordert einen Installationsbaum, auf den via einer der folgenden Netzwerkprotokolle zugegriffen werden kann: HTTP, FTP oder NFS. Die URL des Installationsmediums muss einen Fedora Installationsbaum enthalten. Dieser Baum wird gehostet mittels NFS, FTP or HTTP. Die Netzwerkdienste und Dateien können gehostet werden unter Verwendung der Netzwerkdienste auf dem Host oder einem anderen Spiegelserver.

Bei der Verwendung eine CD-ROM oder DVD-Abbilds (als **.iso**-Datei gekennzeichnet) hängen Sie das CD-ROM-Abbild ein und hosten die eingehängten Dateien mit einem der genannten Protokolle.

Alternativ können Sie den Installationsbaum von einem Fedora-Spiegelserver kopieren.



- b. Eine voll virtualisierte Gastinstallation benötigt bootbare Installations-DVDs, CD-ROMs oder lokale Abbilder von bootbaren Installations-DVDs oder CD-ROMs (mit dem Dateityp .iso oder .img). Windows-Installationen verwenden DVDs, CD-ROMs oder .iso-Dateien. Viele Linux- oder UNIX-ähnliche Betriebssysteme nutzen eine .iso-Datei zur Installation des Basissystems, um anschließend die Installation mit einem netzwerkbasieren Installationsbaum zu vervollständigen.



Nachdem Sie das passende Installationsmedium ausgewählt haben, klicken Sie auf **Weiter**.

7. The **Assigning storage space** window displays. Choose a disk partition, LUN or create a file based image for the guest storage.


Gemäß Konvention für dateibasierte Abbilder in Fedora sollten alle dateibasierten Gastabbilder im `/var/lib/xen/images/`-Verzeichnis abgelegt werden. Andere Speicherorte für dateibasierte Abbilder werden von SELinux verweigert. Falls Sie SELinux im Enforcing-Modus ausführen, werfen Sie einen Blick auf [Abschnitt 7.1, „SELinux und Virtualisierung“](#) für weitere Informationen über die Installation von Gästen.

Your guest storage image should be larger than the size of the installation, any additional packages and applications, and the size of the guests swap file. The installation process will choose the size of the guest's swap file based on size of the RAM allocated to the guest.

Allocate extra space if the guest needs additional space for applications or other data. For example, web servers require additional space for log files.



Choose the appropriate size for the guest on your selected storage type and click the **Forward** button.

 **Anmerkung**

Es wird empfohlen, das Standardverzeichnis für die virtuellen Maschinenabbilder zu verwenden, also `/var/lib/xen/images/`. Falls Sie einen anderen Speicherort verwenden (wie z. B. `/xen/images/` in diesem Beispiel), stellen Sie sicher, dass Sie ihn in der SELinux-Richtlinie hinzugefügt und neu gekennzeichnet haben, bevor Sie mit der Installation fortfahren (an späterer Stelle im Dokument finden Sie Informationen darüber, wie Sie Ihre SELinux-Richtlinie anpassen).

8. The Allocate memory and CPU window displays. Choose appropriate values for the virtualized CPUs and RAM allocation. These values affect the host's and guest's performance.

Gäste benötigen ausreichend physischen Arbeitsspeicher (RAM), um effektiv und effizient zu arbeiten. Wählen Sie einen Wert für den Speicher, der am Besten Ihrem Gastbetriebssystem und den Anforderungen der Anwendungen gerecht wird. Die meisten Betriebssysteme benötigen mindestens 512 MB RAM, um effizient zu arbeiten. Bedenken Sie, dass Gäste physischen RAM verbrauchen. Falls zu viele Gäste ausgeführt werden oder nicht genügend Arbeitsspeicher für das Host-System verbleibt, wird verstärkt virtueller Speicher genutzt. Virtueller Speicher hat

jedoch deutlich langsamere Zugriffszeiten, infolgedessen sinkt die Leistung und Reaktionszeit des Systems. Stellen Sie daher sicher, dass Sie ausreichend Speicher zuweisen, damit sowohl alle Gäste als auch der Host effektiv arbeiten können.

Assign sufficient virtual CPUs for the virtualized guest. If the guest runs a multithreaded application assign the number of virtualized CPUs it requires to run most efficiently. Do not assign more virtual CPUs than there are physical processors (or hyper-threads) available on the host system. It is possible to over allocate virtual processors, however, over allocating has a significant, negative affect on guest and host performance due to processor context switching overheads.

Create a new virtual system

Allocate memory and CPU

Memory:
Please enter the memory configuration for this VM. You can specify the maximum amount of memory the VM should be able to use, and optionally a lower amount to grab on startup.

Total memory on host machine: 2046 GB

VM Max Memory: 500

VM Startup Memory: 500

CPUs:
Please enter the number of virtual CPUs this VM should start up with.

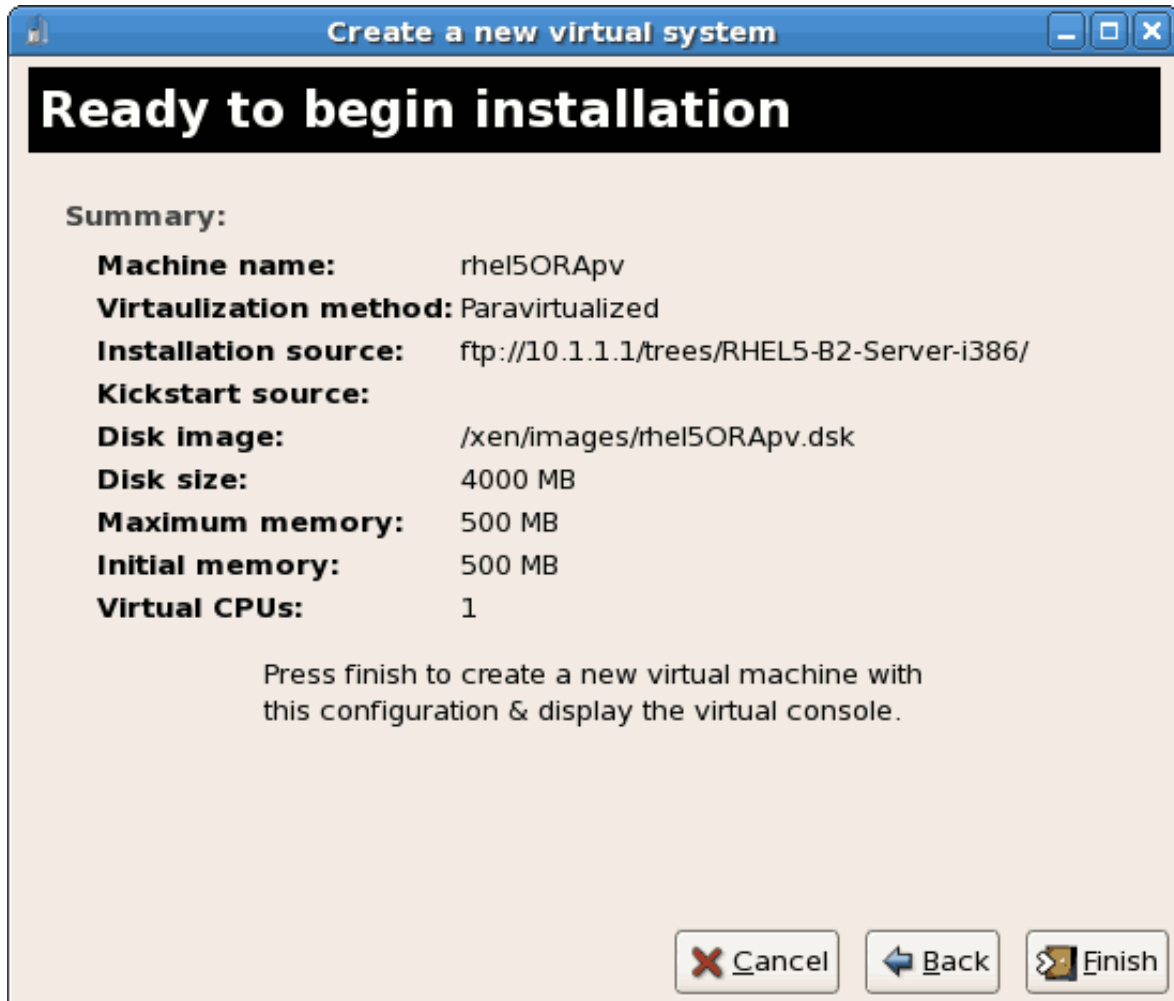
Logical host CPUs: 2

VCPUs: 1

i **Tip:** For best performance, the number of virtual CPUs should be less than (or equal to) the number of logical CPUs on the host system.

Cancel Back Forward

- Der nun folgende Bildschirm "Bereit zur Installation" zeigt eine Zusammenfassung aller von Ihnen eingegebenen Konfigurationsinformationen. Überprüfen Sie die angezeigten Informationen und nehmen ggf. nötige Änderungen über die **Zurück**-Schaltfläche vor. Wenn Sie mit den eingegebenen Daten zufrieden sind, klicken Sie auf **Fertigstellen**, um den Installationsprozess zu beginnen.



Ein VNC-Fenster öffnet sich und zeigt den Beginn des Installationsprozesses für das Gastbetriebssystem an.

Damit ist der allgemeine Prozess für die Gästeerstellung mit **virt-manager** abgeschlossen. [Kapitel 3, Installationsverfahren für Gastbetriebssysteme](#) enthält schrittweise Anleitungen zur Installation einer Vielzahl gebräuchlicher Betriebssysteme.

2.3. Installation von Gästen mit PXE

Dieser Abschnitt behandelt die nötigen Schritte zur Installation von Gästen mit PXE. Die PXE-Gästeinstallation setzt ein gemeinsam verwendetes Netzwerkgerät, auch Netzwerk-Bridge genannt, voraus. Das nachfolgend beschriebene Verfahren umfasst die Erstellung einer Bridge sowie die nötigen Schritte, um diese Bridge für die PXE-Installation zu nutzen.

1. **Neue Bridge erstellen**
 - a. Erzeugen Sie eine neue Netzwerkskriptdatei im `/etc/sysconfig/network-scripts/`-Verzeichnis. Dieses Beispiel erstellt eine Datei namens `ifcfg-installation`, die eine Bridge namens `installation` erzeugt.

```
# cd /etc/sysconfig/network-scripts/  
# vim ifcfg-installation
```

```
DEVICE=installation
TYPE=Bridge
BOOTPROTO=dhcp
ONBOOT=yes
```



Warning

The line, `TYPE=Bridge`, is case-sensitive. It must have uppercase 'B' and lower case 'ridge'.

- b. Starten Sie die neue Bridge.

```
# ifup installation
```

- c. Es sind bislang noch keine Schnittstellen zur neuen Bridge hinzugefügt. Verwenden Sie den **brctl show**-Befehl, um Einzelheiten der Netzwerk-Bridges auf dem System einzusehen.

```
# brctl show
bridge name      bridge id                STP enabled    interfaces
installation     8000.0000000000000000    no
virbr0           8000.0000000000000000    yes
```

Die **virbr0**-Bridge ist die Standard-Bridge, die von **libvirt** für Network Address Translation (NAT) auf dem Standard-Ethernet-Gerät verwendet wird.

2. Eine Schnittstelle zur neuen Bridge hinzufügen

Bearbeiten Sie die Konfigurationsdatei der Schnittstelle. Fügen Sie den **BRIDGE**-Parameter mit dem Namen der Bridge, die im vorangegangenen Schritt erzeugt wurde, zur Konfigurationsdatei hinzu.

```
# Intel Corporation Gigabit Network Connection
DEVICE=eth1
BRIDGE=installation
BOOTPROTO=dhcp
HWADDR=00:13:20:F7:6E:8E
ONBOOT=yes
```

Nachdem Sie die Konfigurationsdatei bearbeitet haben, starten Sie das Netzwerk oder das ganze System neu.

```
# service network restart
```

Überprüfen Sie mit Hilfe des **brctl show**-Befehls, dass die Schnittstelle nun verknüpft ist:

```
# brctl show
bridge name      bridge id                STP enabled    interfaces
installation     8000.001320f76e8e       no              eth1
virbr0           8000.0000000000000000    yes
```

3. Sicherheitskonfiguration

Configure **iptables** to allow all traffic to be forwarded across the bridge.

```
# iptables -I FORWARD -m physdev --physdev-is-bridged -j ACCEPT
# service iptables save
# service iptables restart
```



Disable iptables on bridges

Alternatively, prevent bridged traffic from being processed by **iptables** rules. In `/etc/sysctl.conf` append the following lines:

```
net.bridge.bridge-nf-call-ip6tables = 0
net.bridge.bridge-nf-call-iptables = 0
net.bridge.bridge-nf-call-arptables = 0
```

Reload the kernel parameters configured with **sysctl**

```
# sysctl -p /etc/sysctl.conf
```

4. libvirt vor der Installation neustarten

Restart the **libvirt** daemon.

```
# service libvirtd reload
```

Die Bridge ist nun konfiguriert, Sie können jetzt mit einer Installation beginnen.

PXE-Installation mit virt-install

Fügen Sie für **virt-install** den `--network=bridge:BRIDGENAME`-Installationsparameter an, wobei "installation" der Name Ihrer Bridge ist. Verwenden Sie für PXE-Installationen den `--pxe`-Parameter.

```
# virt-install --accelerate --hvm --connect qemu:///system \
  --network=bridge:installation --pxe \
  --name EL10 --ram=756 \
  --vcpus=4
  --os-type=linux --os-variant=rhel5
  --file=/var/lib/libvirt/images/EL10.img \
```


Beispiel 2.3. PXE-Installation mit virt-install

PXE-Installation mit virt-manager

Die nachfolgenden Schritte unterscheiden sich von jenen des standardmäßigen Installationsverfahrens mit virt-manager. Die Standardinstallation finden Sie unter [Kapitel 3, Installationsverfahren für Gastbetriebssysteme](#).

1. PXE auswählen

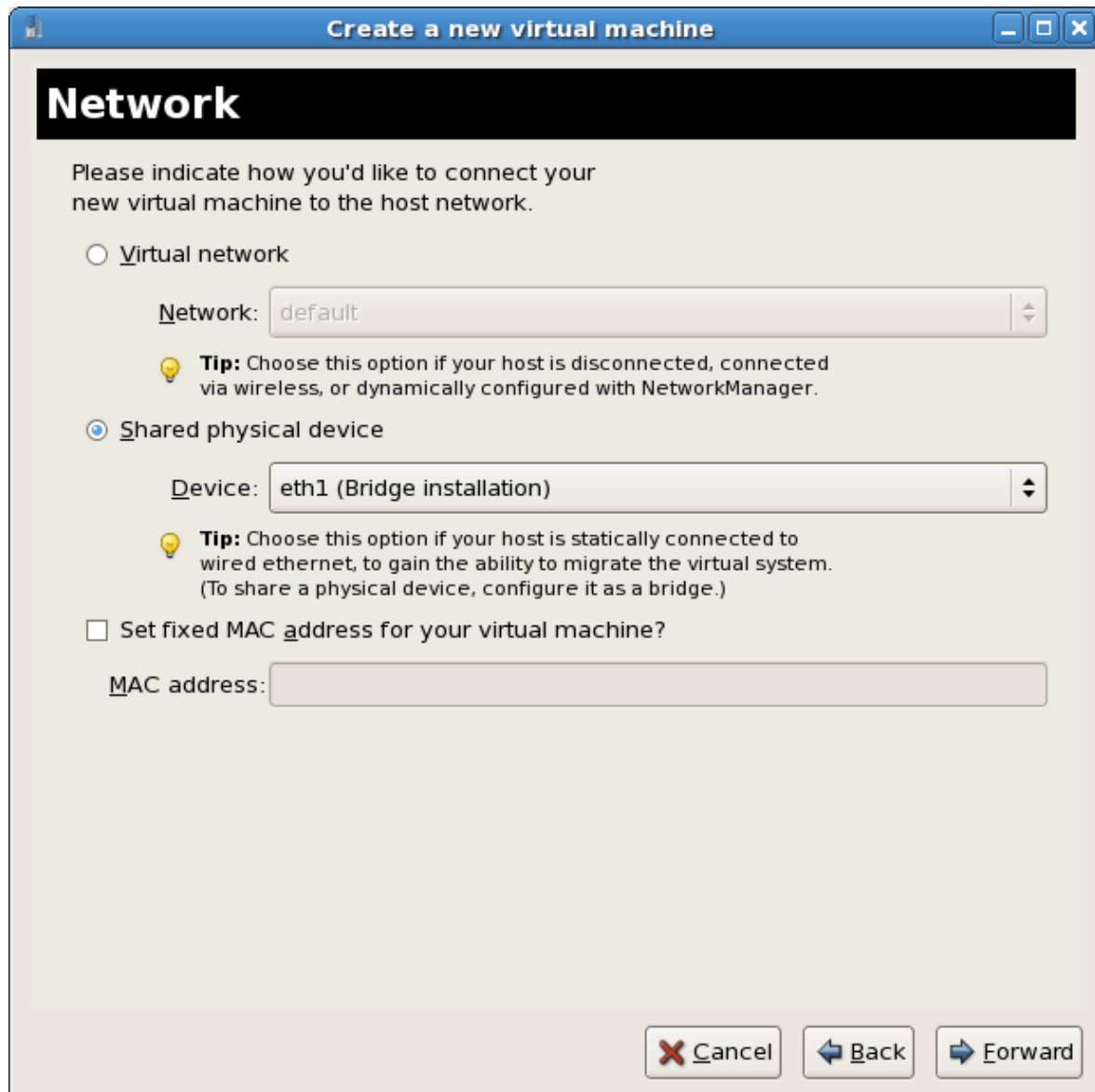
Wählen Sie PXE als Installationsmethode.



The screenshot shows a window titled "Create a new virtual machine" with a sub-header "Installation Method". The main text asks the user to indicate where installation media is available. Three radio buttons are present: "Local install media (ISO image or CDRROM)", "Network install tree (HTTP, FTP, or NFS)", and "Network boot (PXE)", with the last one selected. Below this, the user is asked to choose the operating system. Two dropdown menus are shown: "OS Type" set to "Linux" and "OS Variant" set to "Red Hat Enterprise Linux 5". A warning message with a lightbulb icon states: "Not all operating system choices are supported by Red Hat. Please see the link below for supported configurations:" followed by a blue hyperlink: "[Red Hat Enterprise Linux 5 virtualization support](#)". At the bottom right, there are three buttons: "Cancel", "Back", and "Forward".

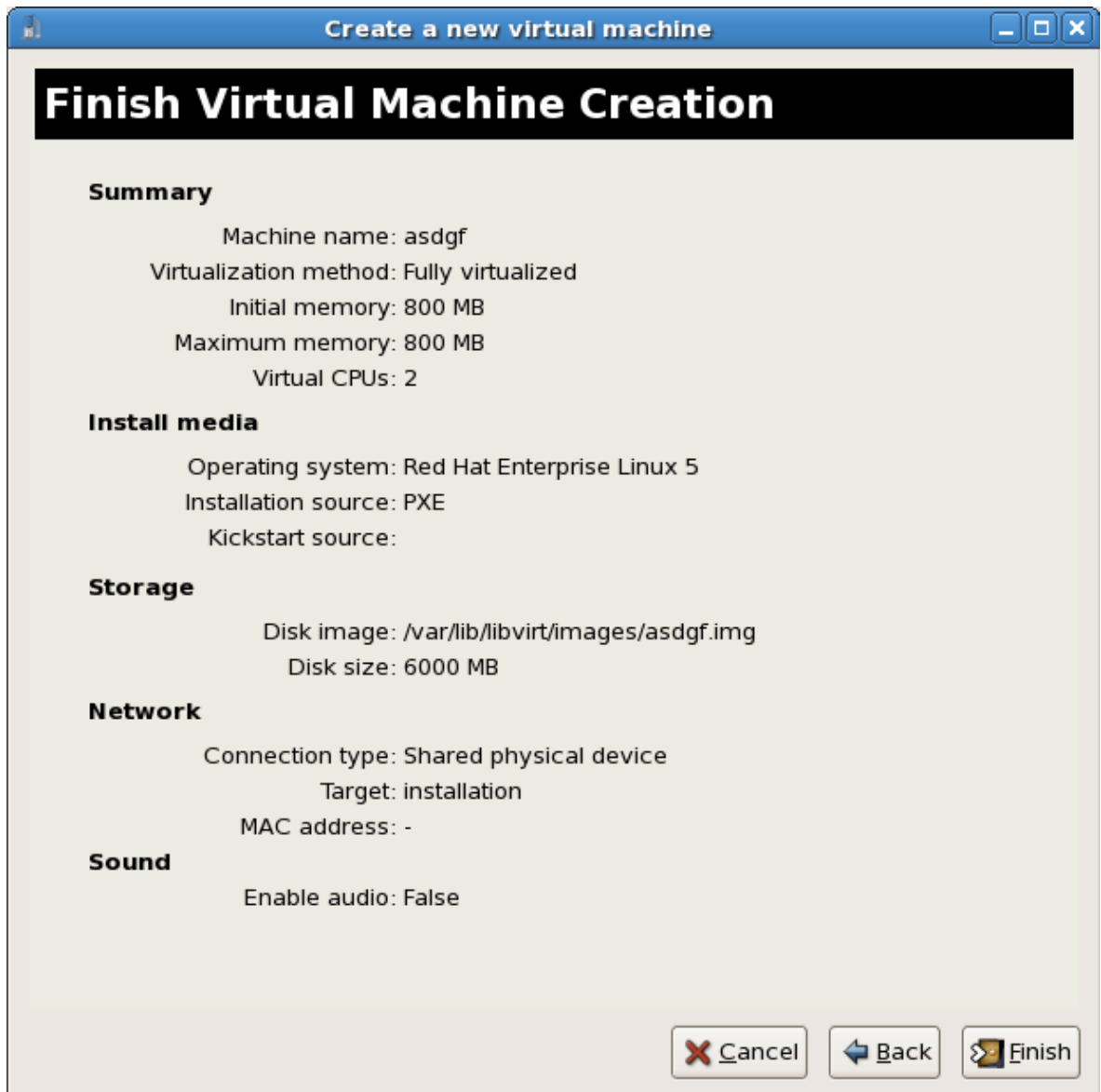
2. Bridge auswählen

Wählen Sie **Gemeinsam verwendetes physisches Gerät** und wählen die Bridge, die in den vorangegangenen Schritten erstellt wurde.



3. **Installation starten**

Die Installation ist bereit zum Start.



Eine DHCP-Anfrage wird gesendet, und falls ein gültiger PXE-Server gefunden wird, beginnt der Gastinstallationsprozess.

Installationsverfahren für Gastbetriebssysteme

Dieses Kapitel beschreibt die Installation verschiedener Gastbetriebssysteme in einer virtualisierten Umgebung unter Fedora. Werfen Sie zum besseren Verständnis der grundlegenden Verfahren einen Blick auf [Kapitel 2, Überblick über die Installation virtualisierter Gäste](#).

3.1. Installation von Red Hat Enterprise Linux 5 als paravirtualisierter Gast

Dieser Abschnitt beschreibt die Installation eines paravirtualisierten Red Hat Enterprise Linux 5 Gasts. Paravirtualisierung ist schneller als volle Virtualisierung, unterstützt dabei jedoch sämtliche Vorzüge der vollen Virtualisierung. Paravirtualisierung erfordert einen speziellen, unterstützten Kernel, den **kernel-xen**-Kernel.



Wichtige Anmerkung zur Paravirtualisierung

Paravirtualisierung funktioniert nur mit dem Xen-Hypervisor. Paravirtualisierung funktioniert nicht mit dem KVM-Hypervisor.

Vergewissern Sie sich, dass Sie über Root-Rechte verfügen, bevor Sie mit der Installation beginnen.

Dieses Verfahren installiert Red Hat Enterprise Linux von einem Remote-Server. Die in diesem Abschnitt beschriebenen Installationsanweisungen ähneln denen für die Installation von der Live-CD-ROM für Minimalinstallation.

Erzeugen Sie paravirtualisierte Red Hat Enterprise Linux 5 Gäste mittels `virt-manager` oder `virt-install`. Anleitungen zur Verwendung von **virt-manager** finden Sie unter [Abschnitt 2.2, „Erzeugen von Gästen mit virt-manager“](#).

Erzeugen Sie einen paravirtualisierten Gast mit dem Befehlszeilen-Tool **virt-install**. Durch die Option `--vnc` wird die grafische Installation gestartet. Der Name des Gasts lautet in diesem Beispiel `rhe15PV`, die Abbilddatei ist `rhe15PV.dsk` und ein lokaler Spiegelserver des Red Hat Enterprise Linux 5 Installationsbaums ist `ftp://10.1.1.1/trees/CentOS5-B2-Server-i386/`. Ersetzen Sie diese Werte passend für Ihr System und Ihr Netzwerk.

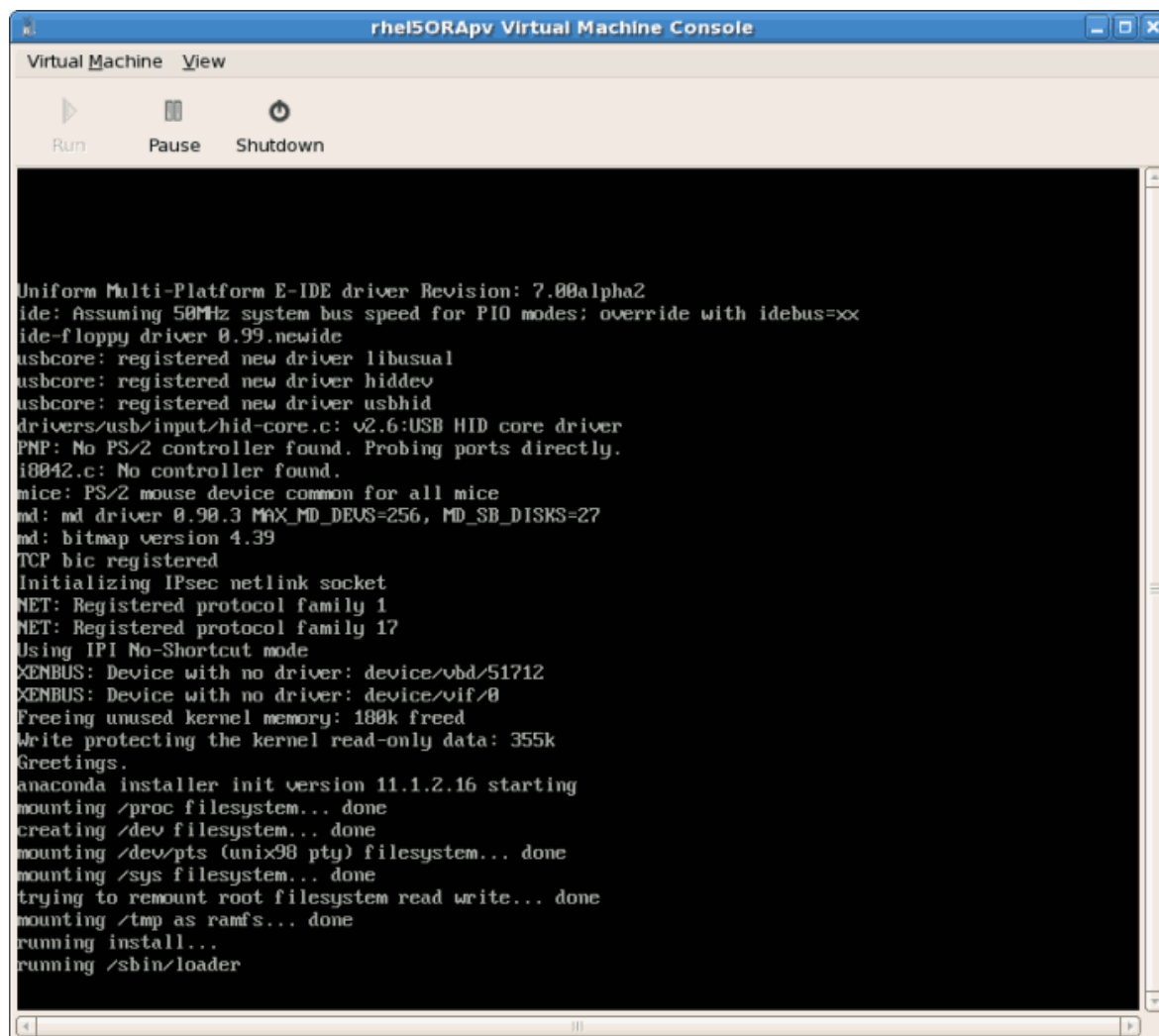
```
# virt-install -n rhe15PV -r 500 \  
-f /var/lib/libvirt/images/rhe15PV.dsk -s 3 --vnc -p \  
-l ftp://10.1.1.1/trees/CentOS5-B2-Server-i386/
```



Automatische Installation

Red Hat Enterprise Linux kann ohne grafische Oberfläche oder manuelle Eingaben installiert werden. Verwenden Sie Kickstart-Dateien, falls Sie die Installation automatisieren möchten.

Unabhängig von der gewählten Methode öffnet sich nun das folgende Fenster und zeigt die ersten Boot-Phasen Ihres Gasts:



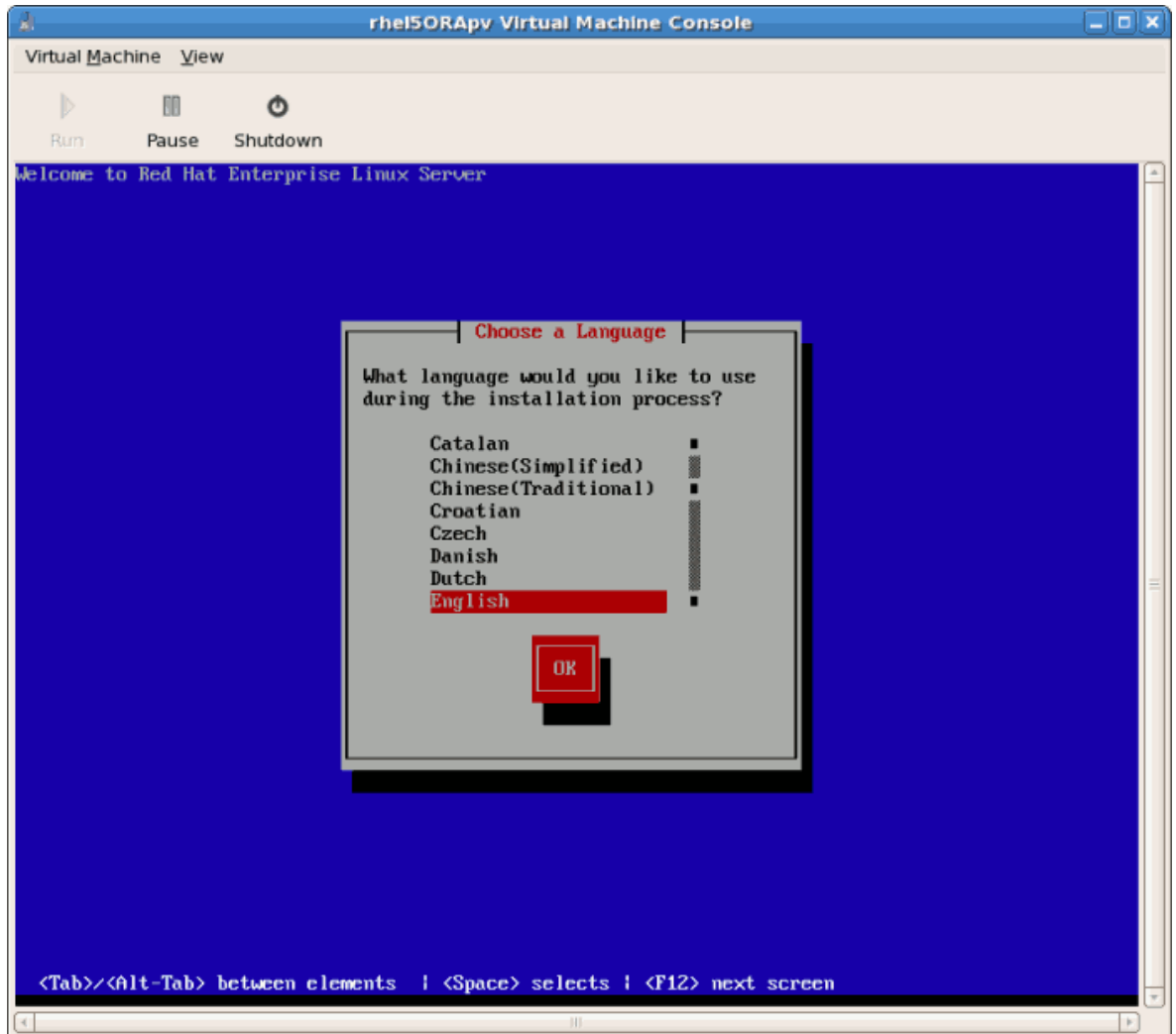
```
Virtual Machine View
Run Pause Shutdown

Uniform Multi-Platform E-IDE driver Revision: 7.00alpha2
ide: Assuming 50MHz system bus speed for PIO modes; override with idebus=xx
ide-floppy driver 0.99.newide
usbcore: registered new driver libusual
usbcore: registered new driver hiddev
usbcore: registered new driver usbhid
drivers/usb/input/hid-core.c: v2.6:USB HID core driver
PNP: No PS/2 controller found. Probing ports directly.
i8042.c: No controller found.
mouse: PS/2 mouse device common for all mice
md: md driver 0.90.3 MAX_MD_DEVS=256, MD_SB_DISKS=27
md: bitmap version 4.39
TCP bic registered
Initializing IPsec netlink socket
NET: Registered protocol family 1
NET: Registered protocol family 17
Using IPI No-Shortcut mode
XENBUS: Device with no driver: device/vbd/51712
XENBUS: Device with no driver: device/vif/0
Freeing unused kernel memory: 180k freed
Write protecting the kernel read-only data: 355k
Greetings.
anaconda installer init version 11.1.2.16 starting
mounting /proc filesystem... done
creating /dev filesystem... done
mounting /dev/pts (unix98 pty) filesystem... done
mounting /sys filesystem... done
trying to remount root filesystem read write... done
mounting /tmp as ramfs... done
running install...
running /sbin/loader
```

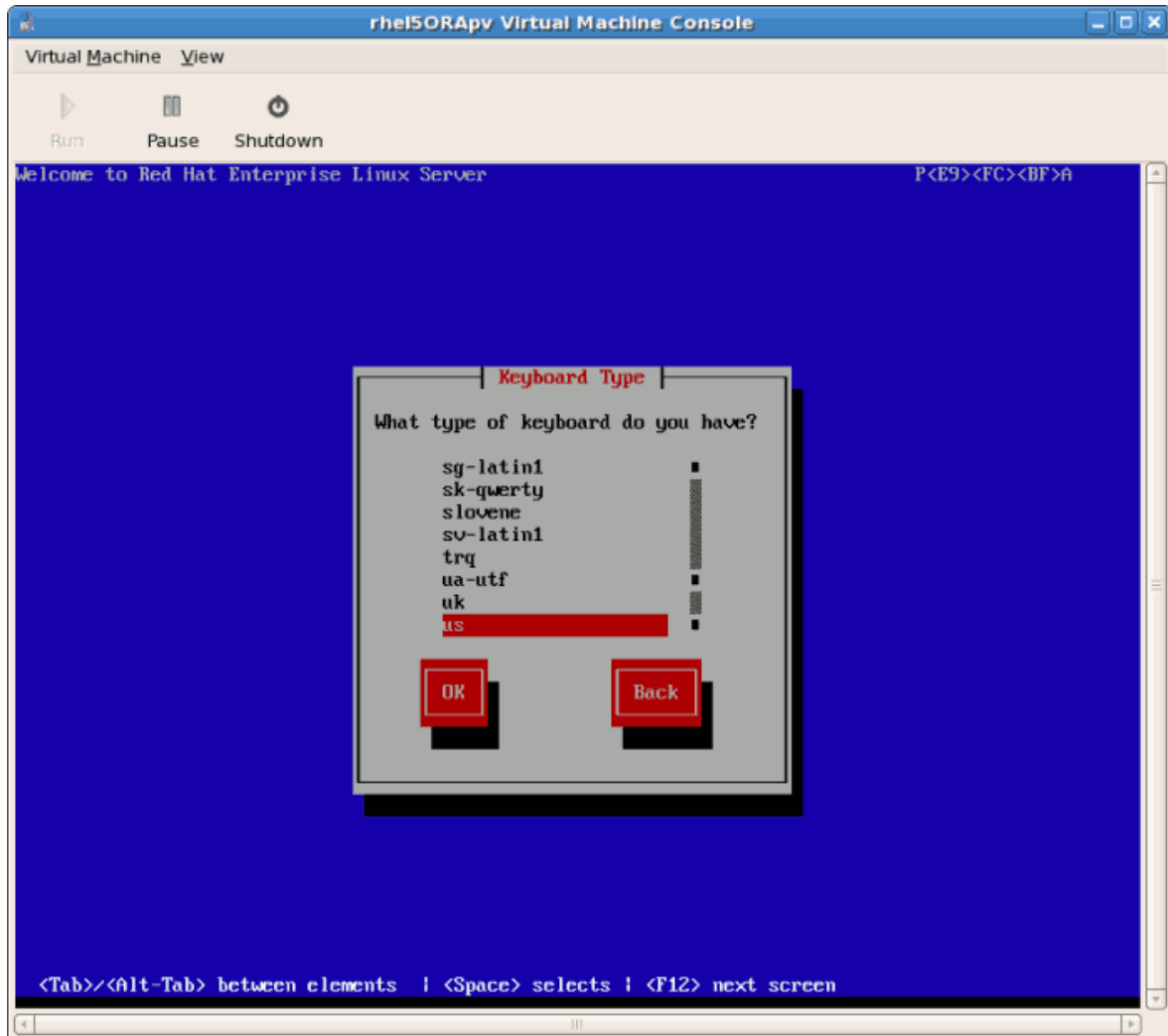
Nachdem Ihr Gast den initialen Boot abgeschlossen hat, beginnt nun der Standardinstallationsprozess für Red Hat Enterprise Linux. Für die meisten Installationen können dabei die Standardantworten übernommen werden.

Prozedur 3.1. Installationsprozess für paravirtualisierten Red Hat Enterprise Linux Gast

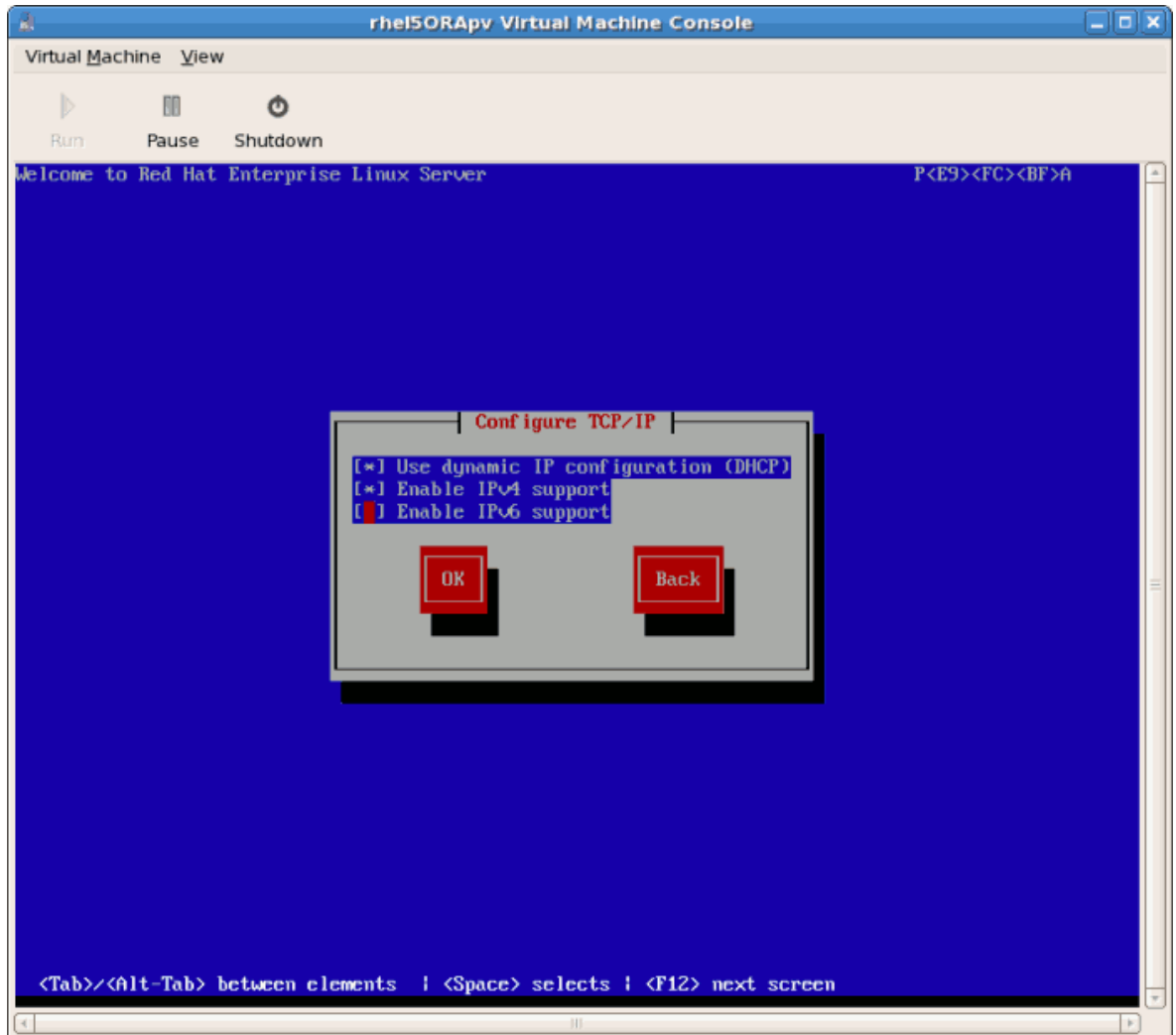
1. Wählen Sie die gewünschte Sprache und klicken Sie auf **OK**.



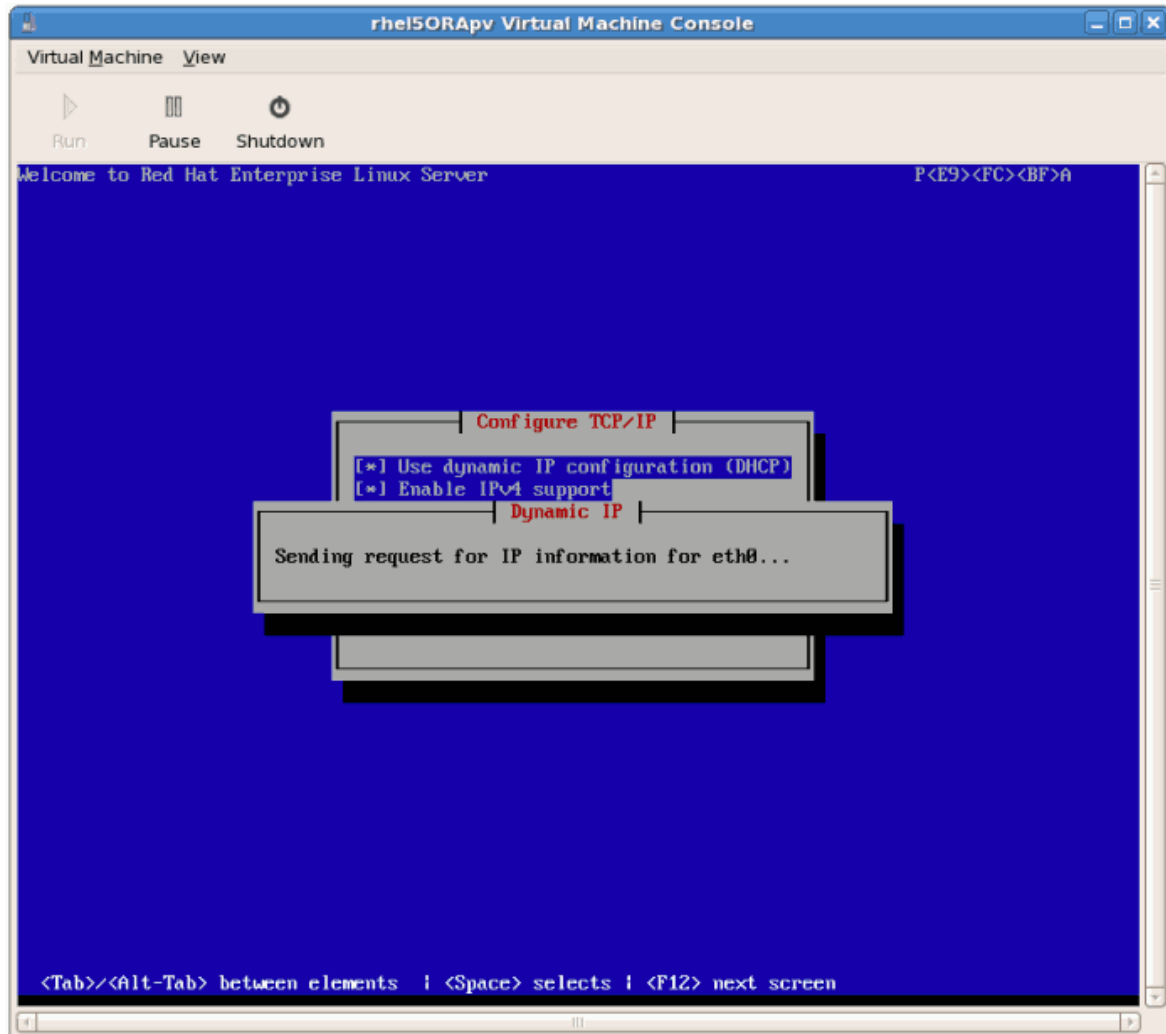
2. Wählen Sie das gewünschte Tastatur-Layout und klicken Sie auf **OK**.



3. Weisen Sie dem Gast eine Netzwerkadresse zu. Sie können zwischen DHCP (wie unten gezeigt) oder einer statischen Adresse wählen:

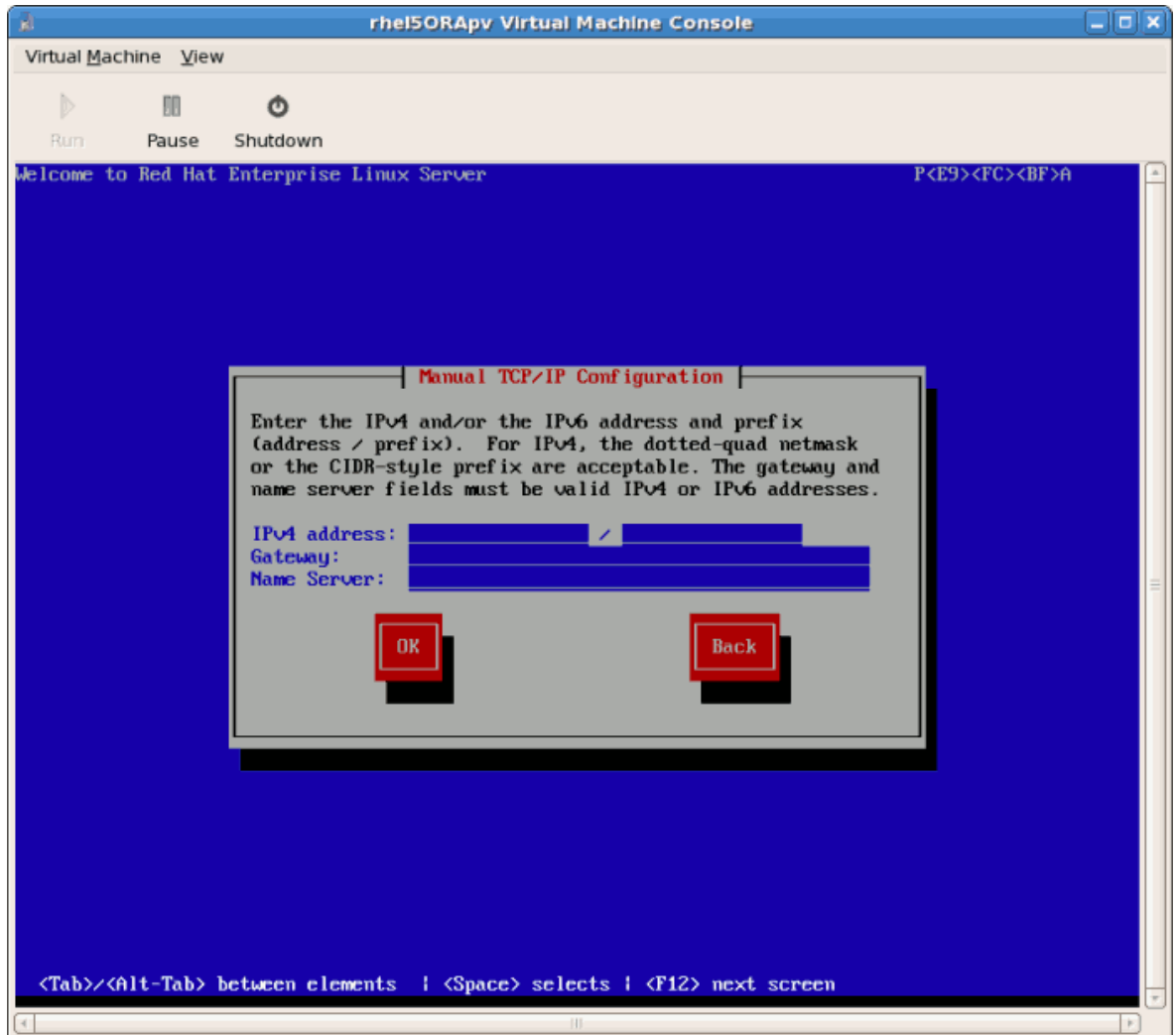


4. Falls Sie DHCP für Ihren Gast gewählt haben, wird der Installationsprozess nun versuchen, eine IP-Adresse für Ihren Gast zu beziehen:

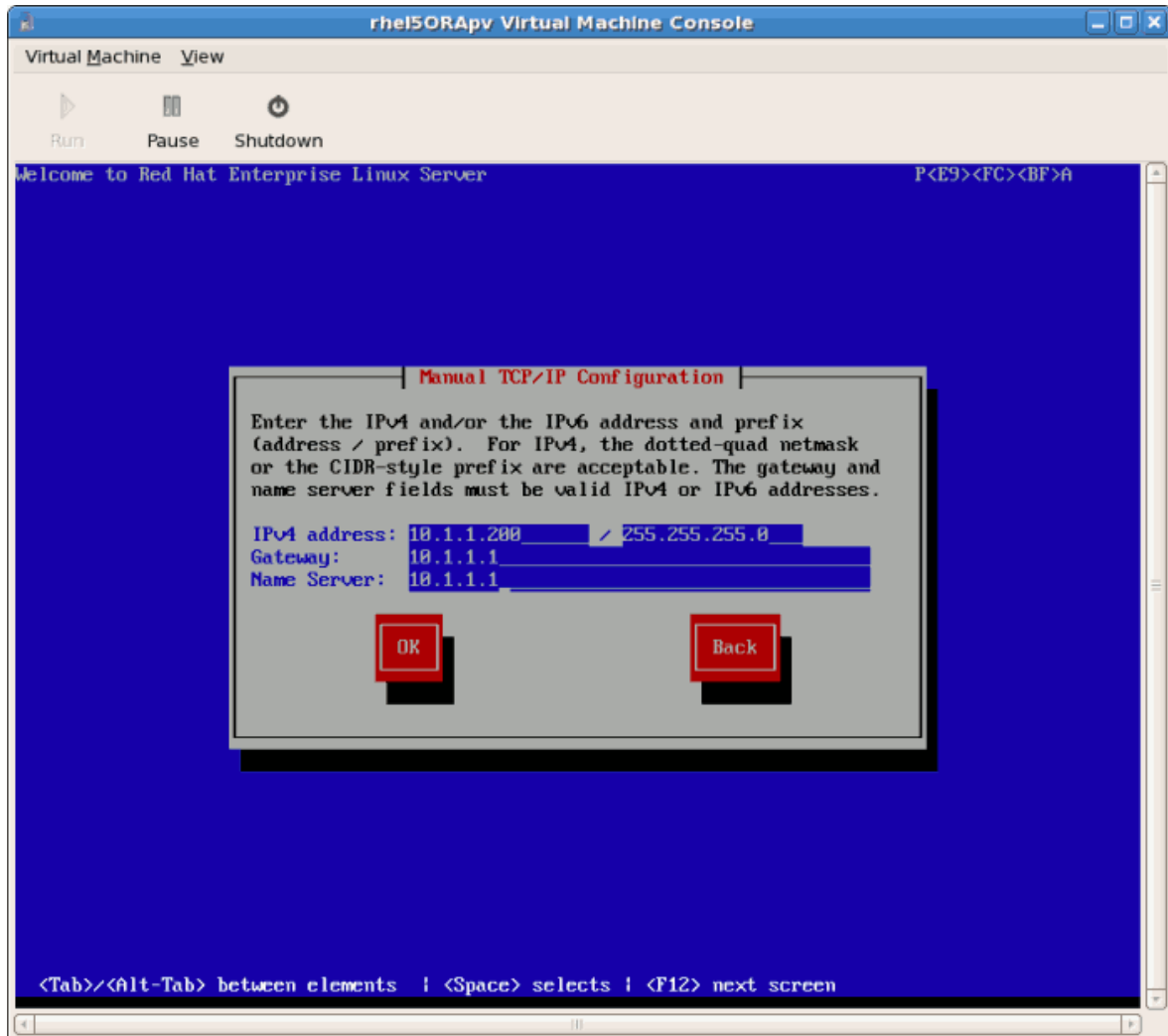


5. Falls Sie eine statische IP-Adresse für Ihren Gast gewählt haben, erscheint die nachfolgend gezeigte Eingabeaufforderung. Geben Sie hier die Details der Gast-Netzwerkconfiguration ein:
 - a. Geben Sie eine gültige IP-Adresse ein. Stellen Sie sicher, dass die eingegebene IP-Adresse den Installations-Server mit dem Installationsbaum erreichen kann.
 - b. Geben Sie eine gültige Subnetzmaske, das Standard-Gateway und die Name-Server-Adresse ein.

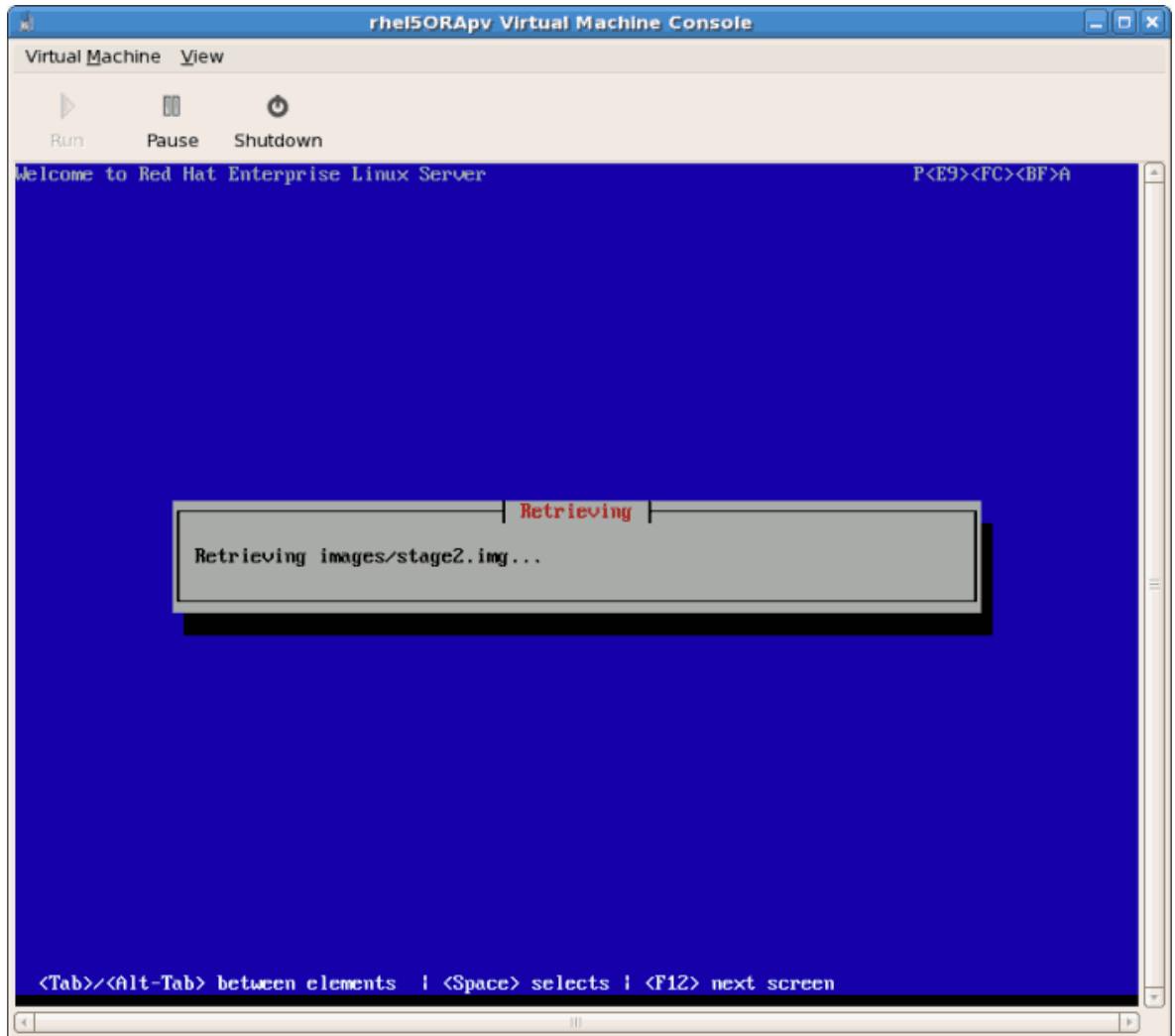
Wählen Sie die gewünschte Sprache und klicken Sie auf **OK**.



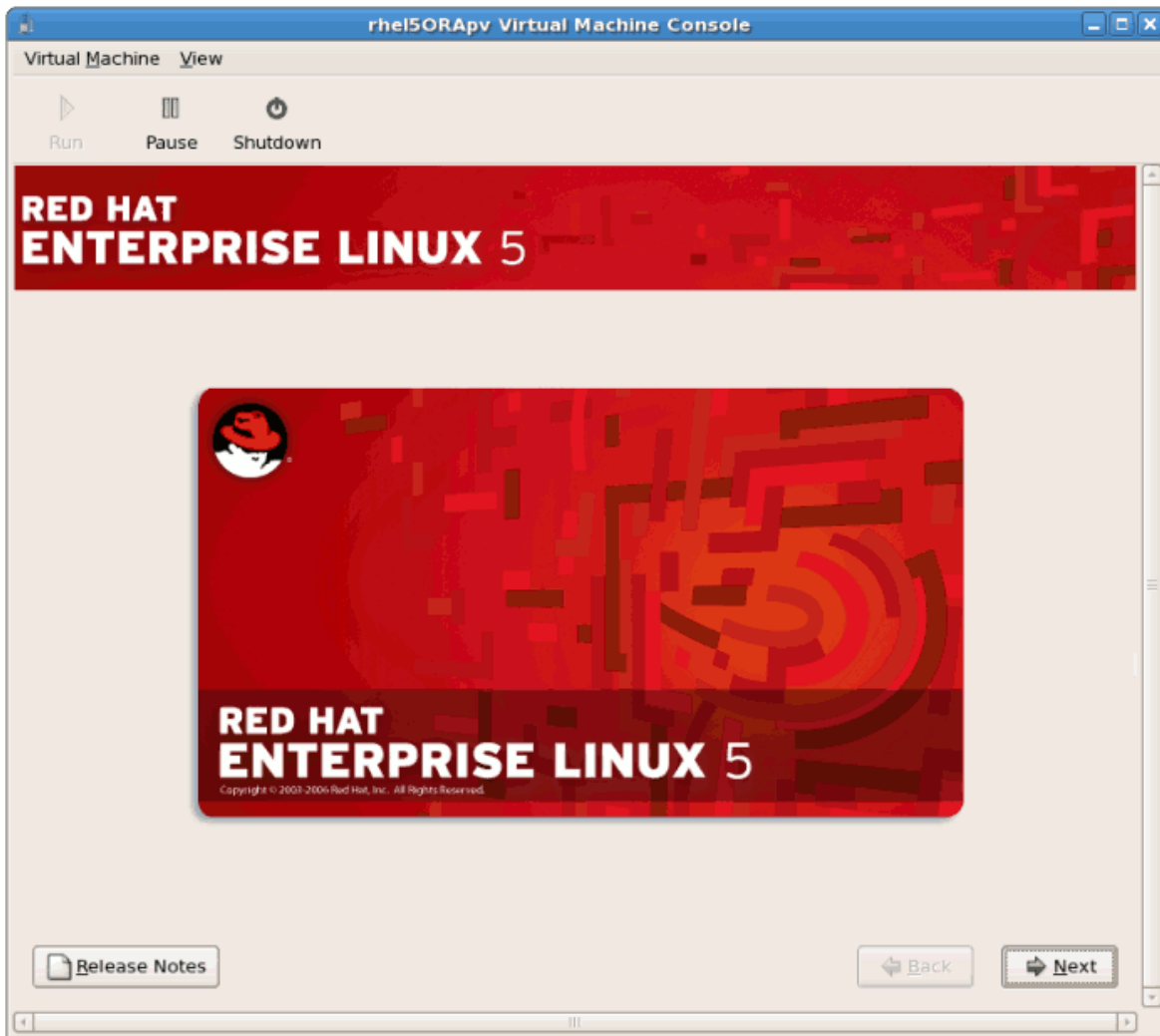
6. Sehen Sie nachfolgend ein Beispiel für eine Konfiguration mit statischer IP-Adresse:



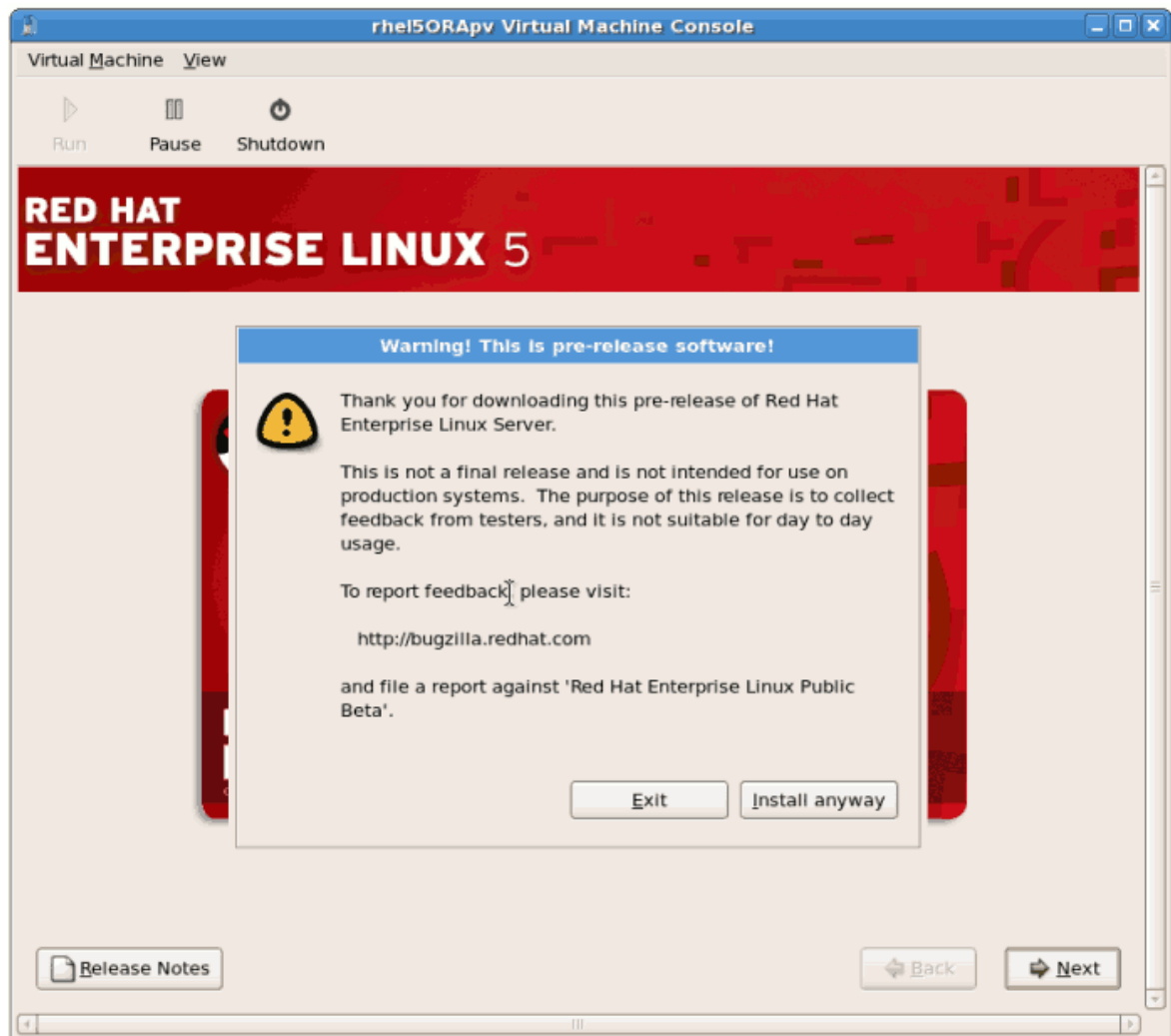
7. Der Installationsprozess ruft nun die benötigten Dateien vom Server ab:



Sobald die ersten Schritte fertiggestellt sind, startet der grafische Installationsprozess.

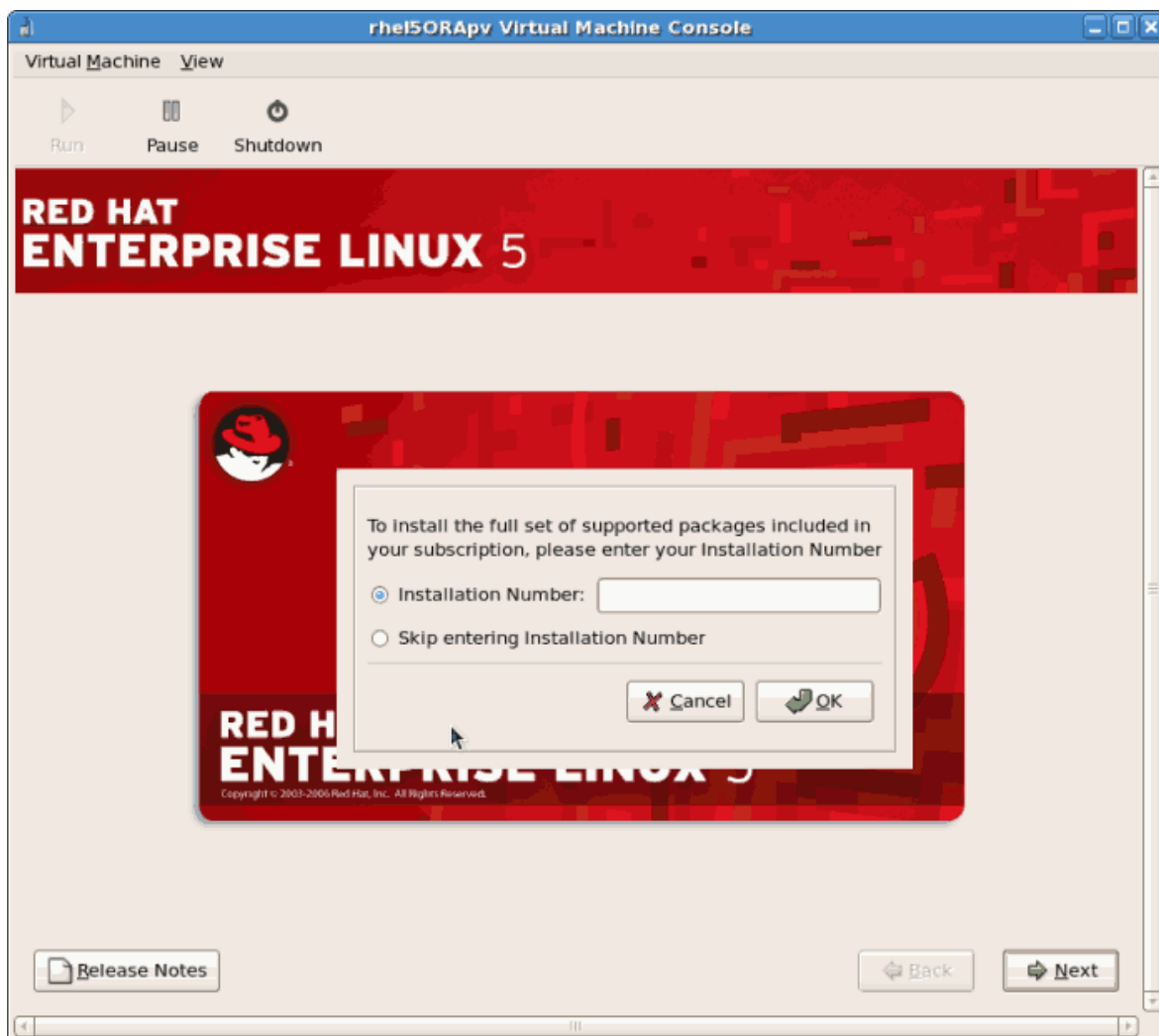


Falls Sie eine Beta- oder Vorabversion installieren, müssen Sie bestätigen, dass Sie das Betriebssystem wirklich installieren wollen. Klicken Sie **Trotzdem installieren** und klicken anschließend auf **OK**:



Prozedur 3.2. Der grafische Installationsprozess

1. Geben Sie einen gültigen Registrierungsschlüssel ein. Falls Sie einen gültigen RHN-Abonnementsschlüssel besitzen, geben Sie diesen im Feld Installationsnummer ein:

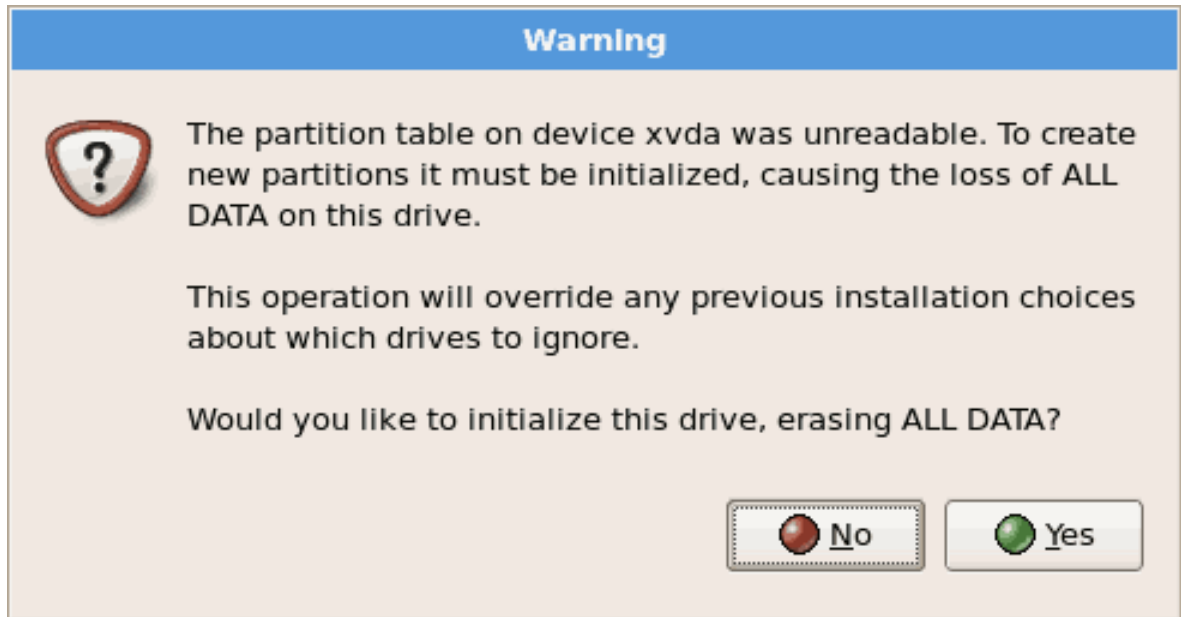


Note

Falls Sie den Schritt zur Registrierung überspringen, können Sie nach abgeschlossener Installation Ihre Fedora Account-Details mit Hilfe des `rhn_register`-Befehls bestätigen. Der `rhn_register`-Befehl erfordert Root-Rechte.

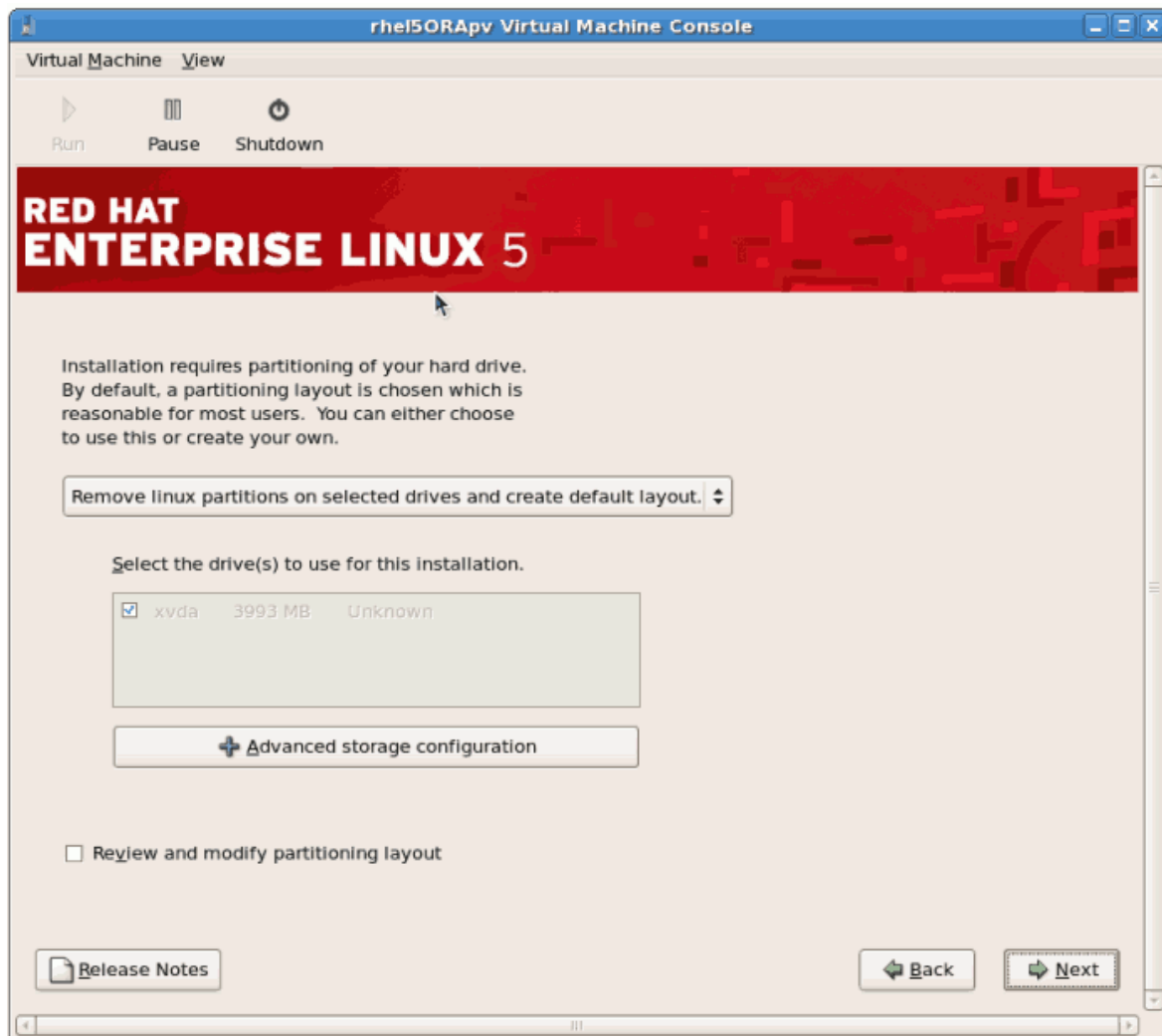
```
# rhn_register
```

2. Die Installation fordert Sie nun auf zu bestätigen, dass alle Daten in dem Speicher, den Sie für die Installation ausgewählt haben, gelöscht werden sollen:



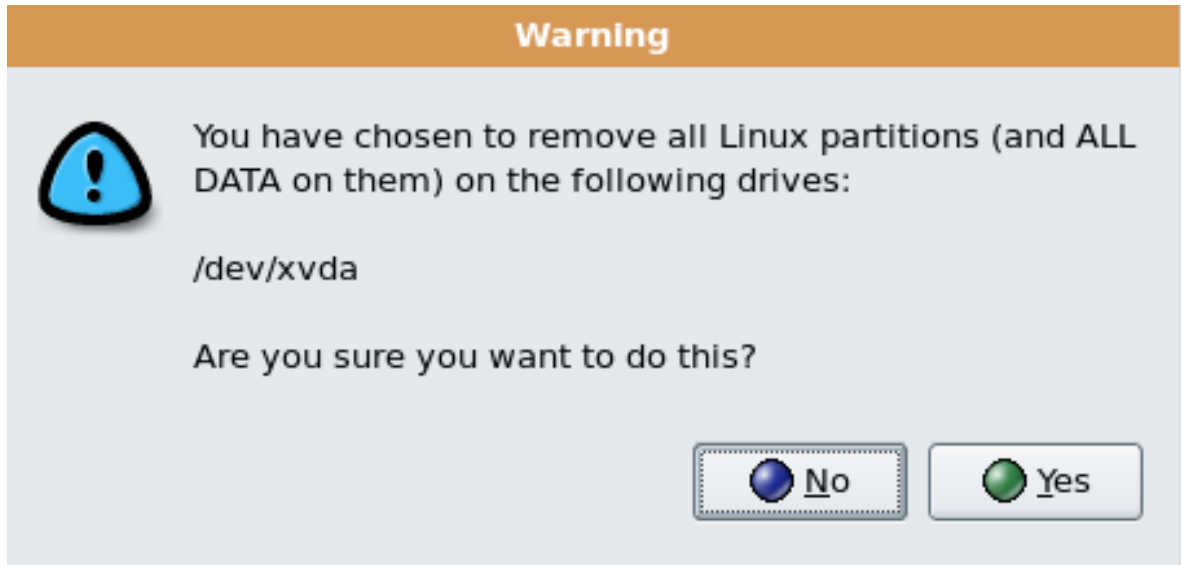
Klicken Sie auf **Ja**, um fortzufahren.

3. Überprüfen Sie noch einmal die Speicherkonfiguration und das Partitions-Layout. Sie können ebenfalls die fortgeschrittene Speicherkonfiguration auswählen, falls Sie iSCSI für Gastspeicher verwenden wollen:



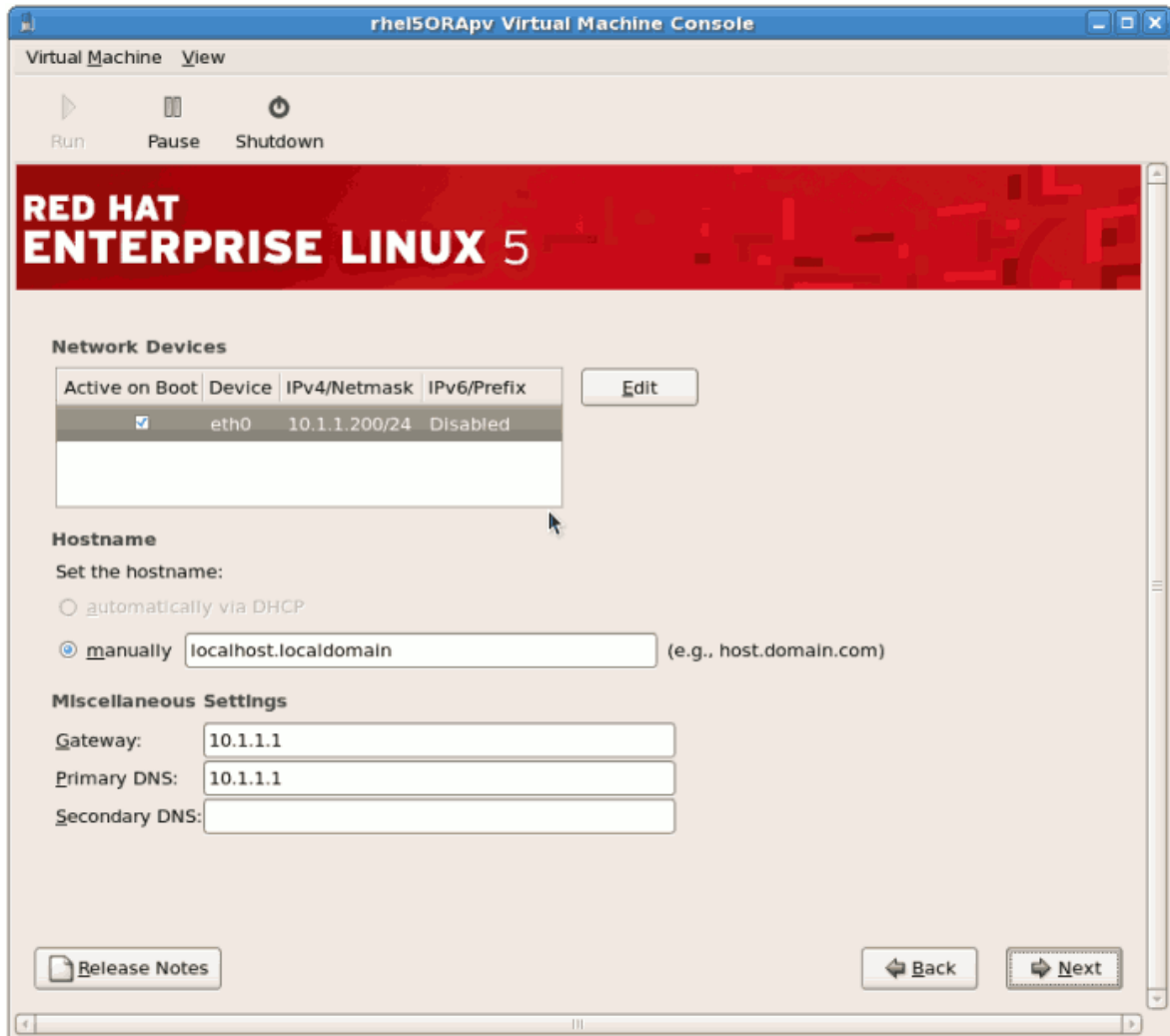
Treffen Sie Ihre Auswahl und klicken auf **Weiter**.

4. Bestätigen Sie den ausgewählten Speicher für die Installation.



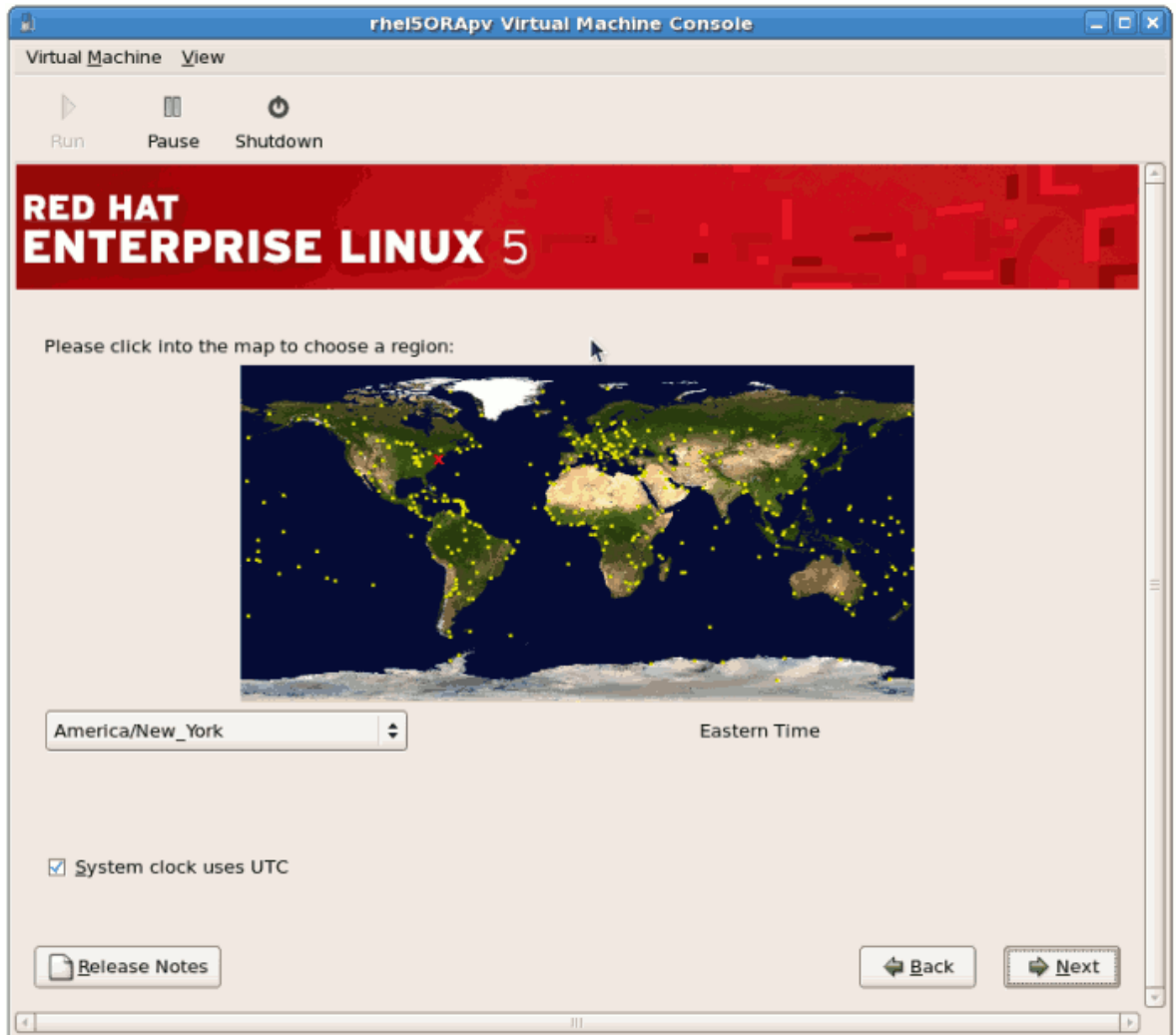
Klicken Sie auf **Ja**, um fortzufahren.

5. Konfigurieren Sie als Nächstes das Netzwerk und den Host-Namen. Diese Einstellungen werden mit den Daten, die Sie zuvor im Installationsprozess eingegeben haben, vorausgefüllt. Sie können diese Einstellungen jedoch ändern, falls notwendig.

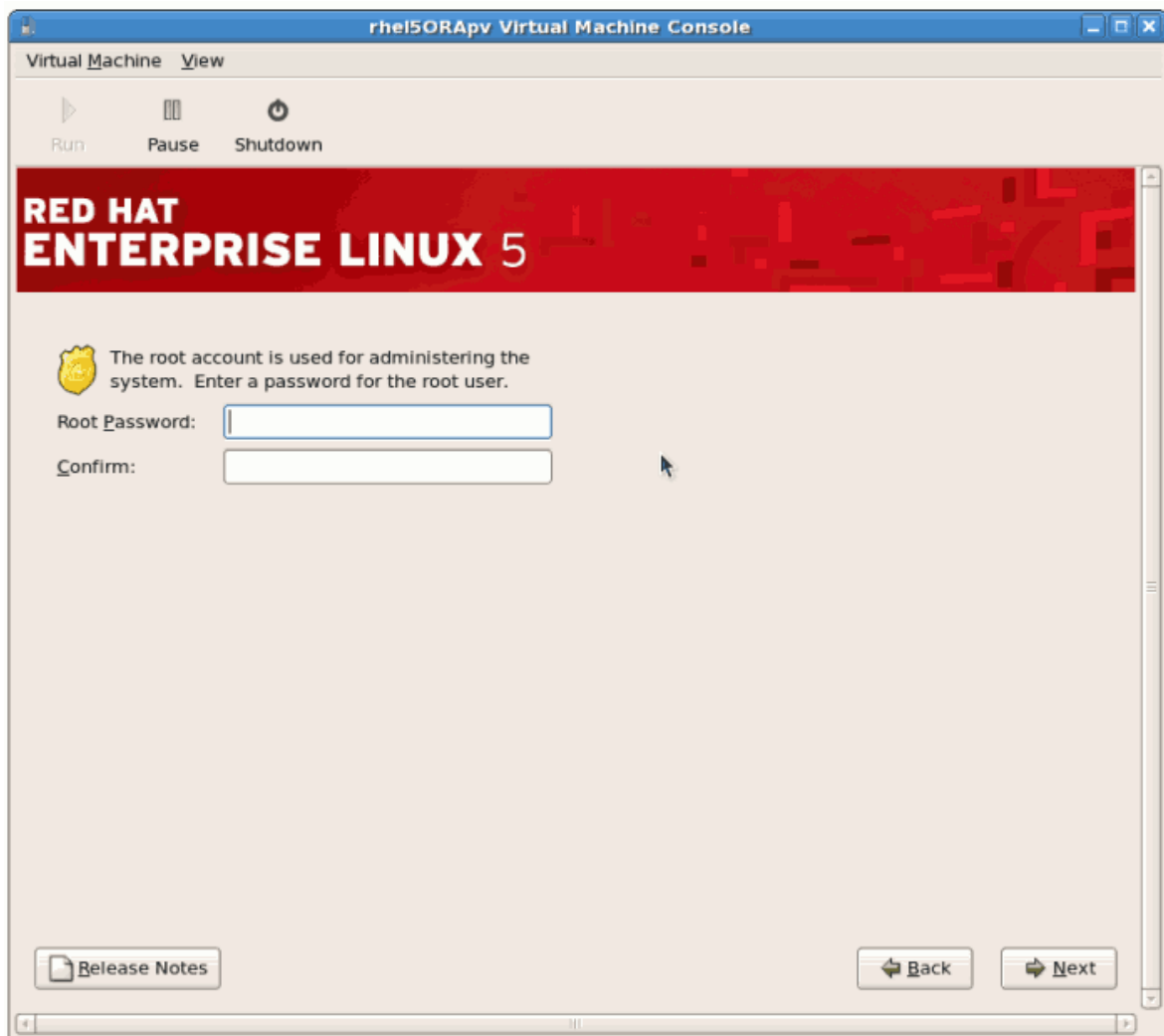


Klicken Sie auf **OK**, um fortzufahren.

6. Wählen Sie die richtige Zeitzone für Ihren Standort aus:

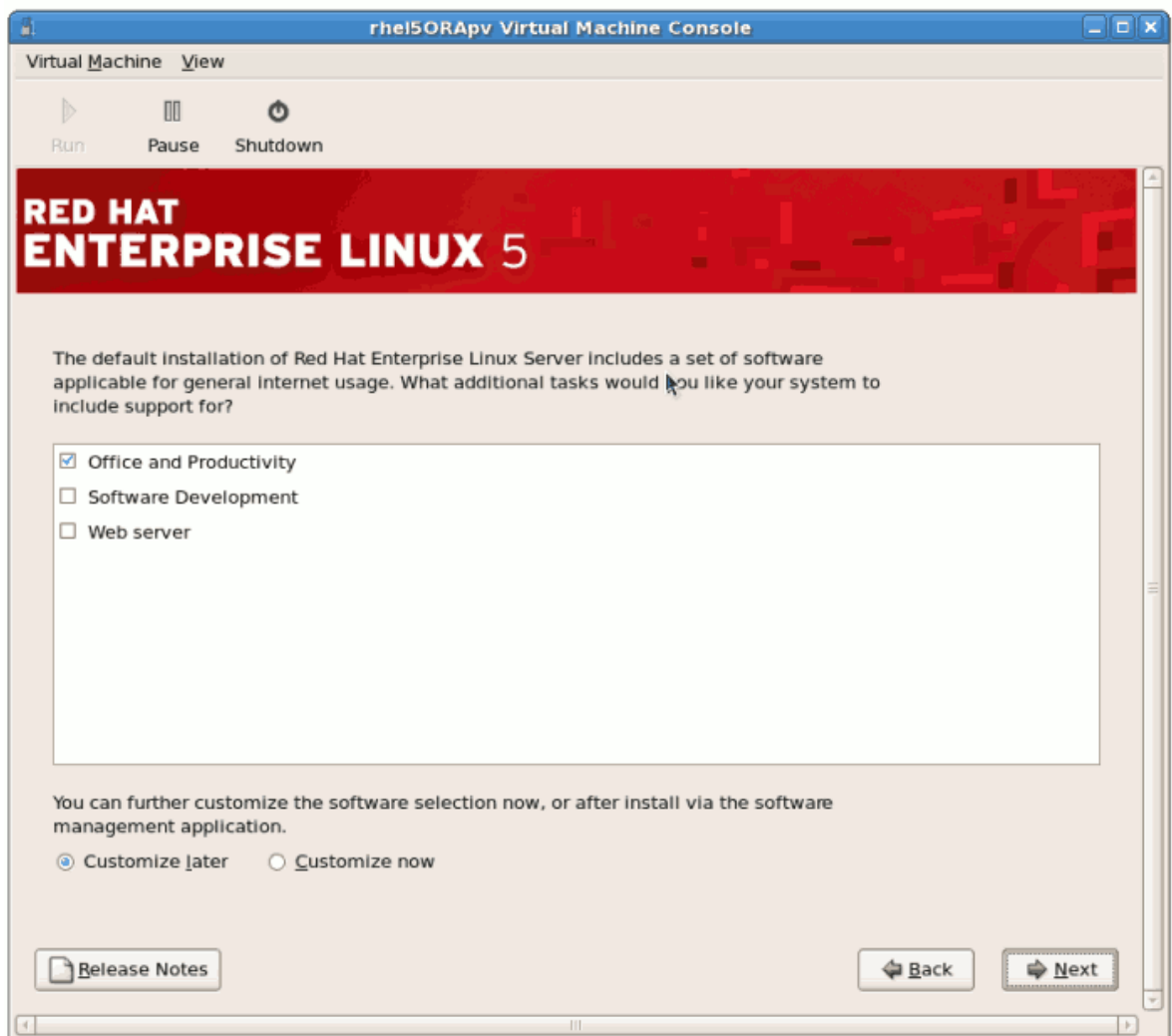


7. Wählen Sie ein Root-Passwort für Ihren Gast.



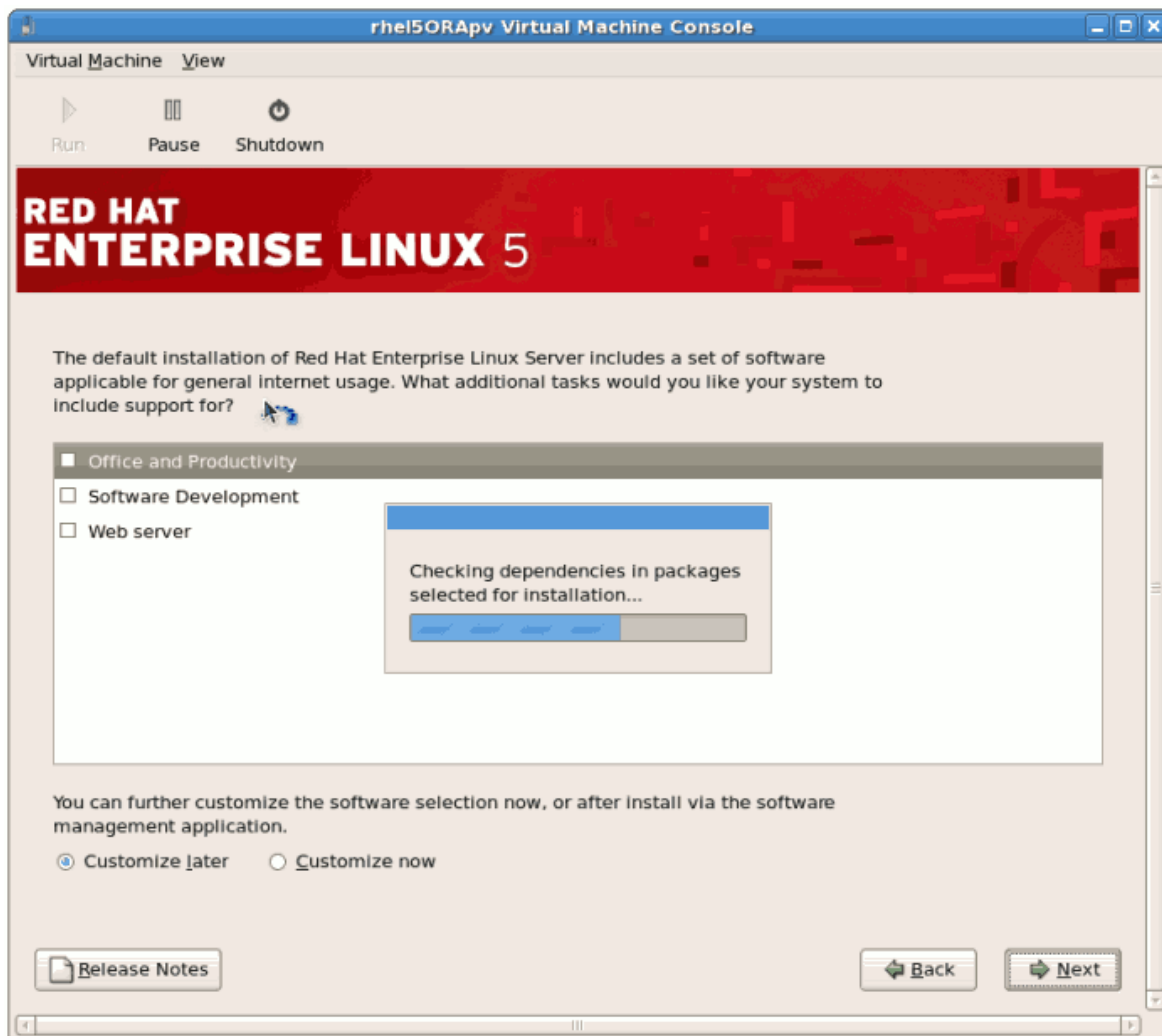
Klicken Sie auf **Weiter**, um fortzufahren.

8. Wählen Sie die zu installierenden Software-Pakete. Wählen Sie die Schaltfläche **Jetzt anpassen**. Sie müssen das **kernel-xen**-Paket im **System**-Verzeichnis installieren. Das **kernel-xen**-Paket ist Voraussetzung für die Paravirtualisierung.

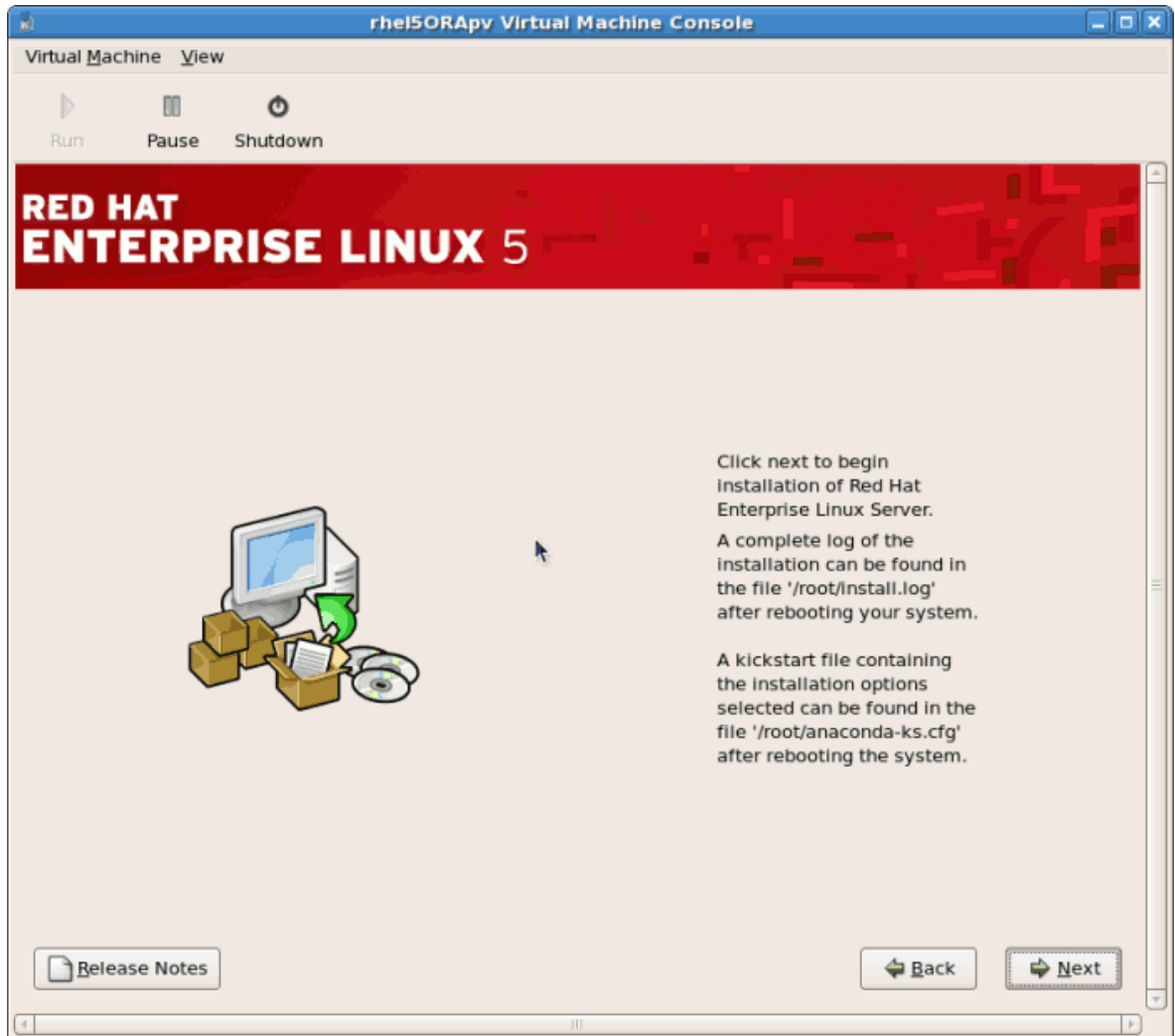


Klicken Sie auf **Weiter**.

9. Abhängigkeiten und Speicherplatzvoraussetzungen werden ermittelt.



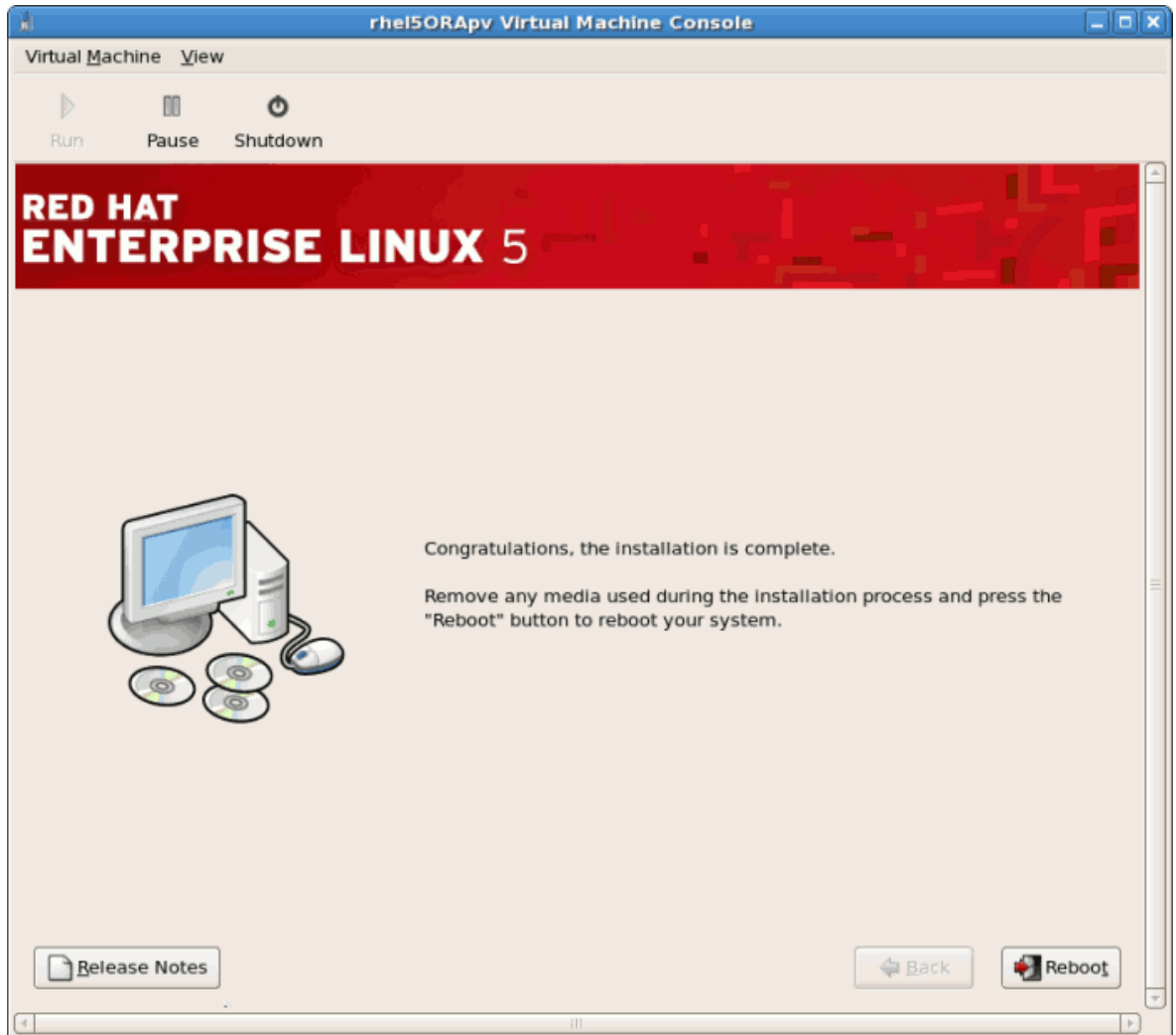
10. Nachdem die Installationsabhängigkeiten und Speicherplatzvoraussetzungen überprüft wurden, klicken Sie auf **Weiter**, um die eigentliche Installation zu starten.



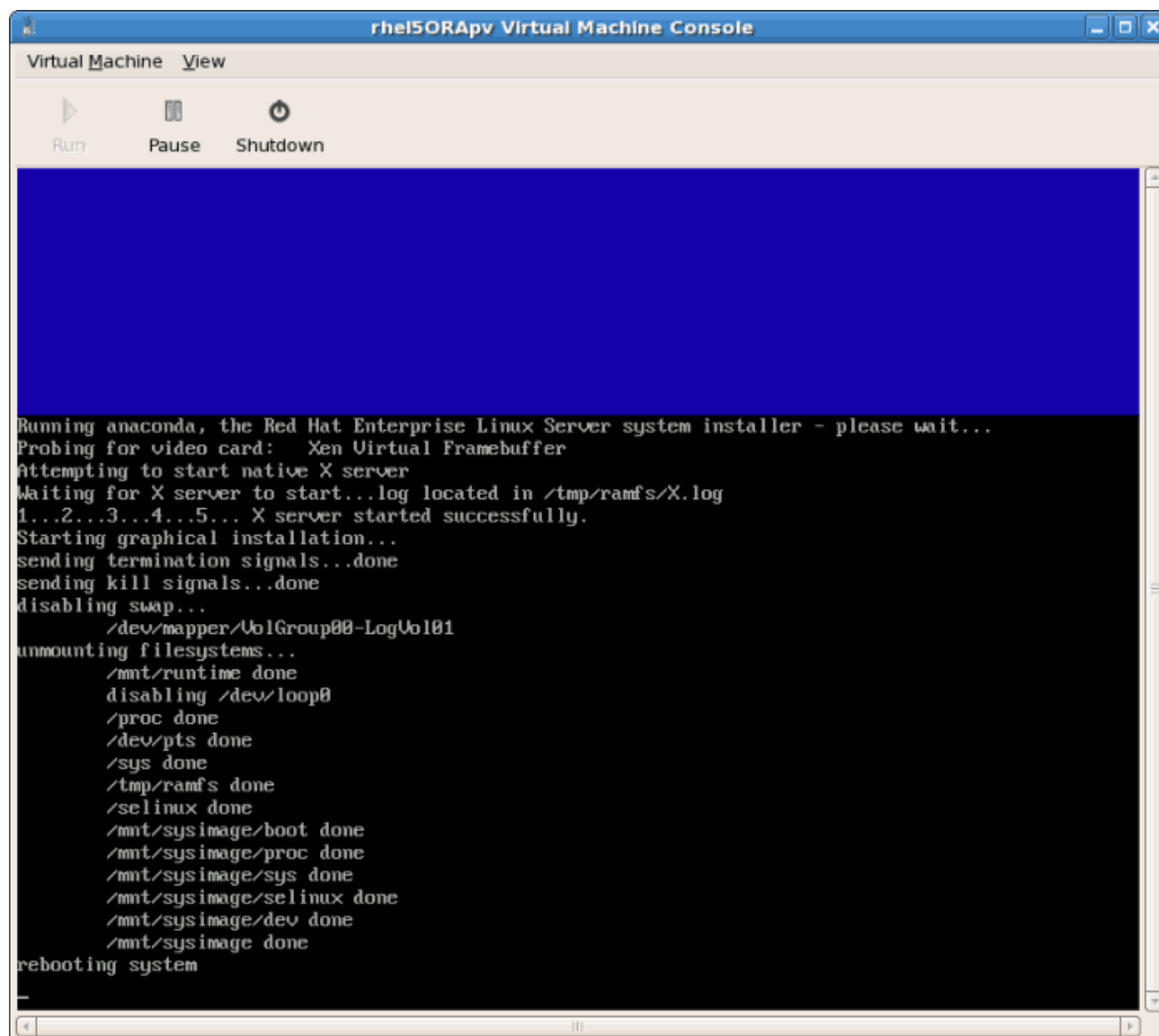
11. Alle ausgewählten Software-Pakete werden automatisch installiert.



12. Nachdem die Installation abgeschlossen ist, müssen Sie den Gast neu starten:



13. Der Gast wird nicht neu starten, sondern herunterfahren.



14. Starten Sie den Gast. Der Gastname wurde ausgewählt, als Sie **virt-install** in [Abschnitt 3.1, „Installation von Red Hat Enterprise Linux 5 als paravirtualisierter Gast“](#) angewendet haben. Falls Sie das Standardbeispiel übernommen haben, lautet der Name *rhe15PV*.

Geben Sie ein:

```
virsh reboot rhe15PV
```

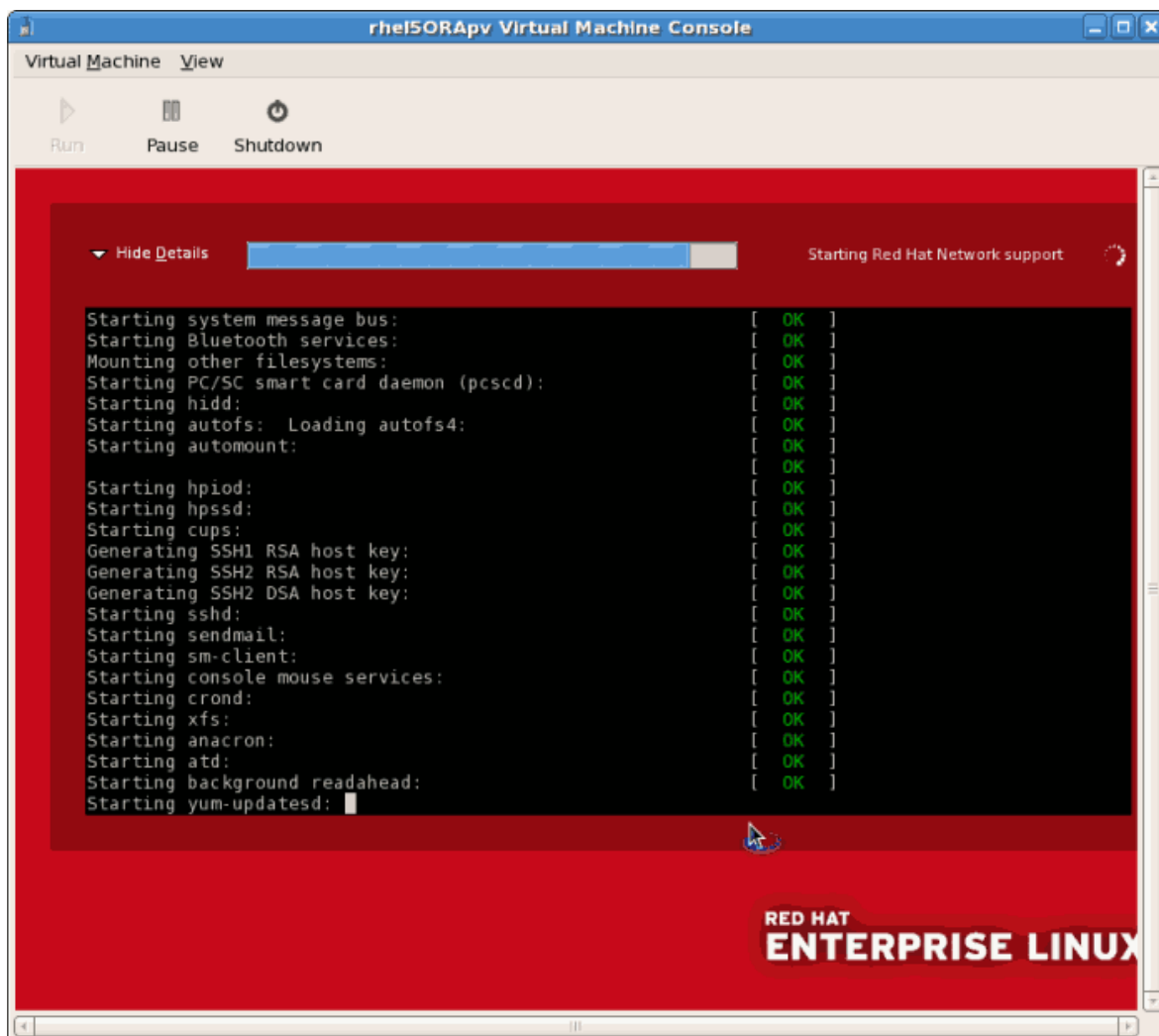
Alternativ können Sie **virt-manager** öffnen, den Namen Ihres Gasts auswählen, auf **Öffnen** und danach auf **Ausführen** klicken.

Es öffnet sich daraufhin ein **VNC-Fenster**, das die Boot-Prozesse des Gasts anzeigt.


```

Virtual Machine  View
Run  Pause  Shutdown

Probing IDE interface ide3...
Probing IDE interface ide4...
Probing IDE interface ide5...
ide-floppy driver 0.99.newide
usbcore: registered new driver libusual
usbcore: registered new driver hiddev
usbcore: registered new driver usbhid
drivers/usb/input/hid-core.c: v2.6:USB HID core driver
PNP: No PS/2 controller found. Probing ports directly.
i8042.c: No controller found.
mice: PS/2 mouse device common for all mice
md: md driver 0.90.3 MAX_MD_DEVS=256, MD_SB_DISKS=27
md: bitmap version 4.39
TCP bic registered
Initializing IPsec netlink socket
NET: Registered protocol family 1
NET: Registered protocol family 17
Using IPI No-Shortcut mode
XENBUS: Device with no driver: device/vbd/51712
XENBUS: Device with no driver: device/vif/0
Freeing unused kernel memory: 180k freed
Write protecting the kernel read-only data: 355k
Red Hat nash version 5.1.19.1 starting
USB Universal Host Controller Interface driver v3.0
ohci_hcd: 2005 April 22 USB 1.1 'Open' Host Controller (OHCI) Driver (PCI)
Registering block device major 202
xuda:<6>device-mapper: ioctl: 4.11.0-ioctl (2006-09-14) initialised: dm-devel@redhat.com
  Reading all physical volumes.  This may take a while...
xuda1 xuda2
  No volume groups found
  Volume group "VolGroup00" not found
  
```



15. Beim Hochfahren des Gasts startet der *First Boot*-Konfigurationsbildschirm. Dieser Assistent wird Sie durch einige grundlegende Konfigurationsschritte für Ihren Gast führen:



16. Zuerst müssen Sie die Lizenzvereinbarung lesen und dieser zustimmen.



Klicken Sie im Fenster der Lizenzvereinbarung auf **Weiter**.

17. Konfigurieren Sie als Nächstes die Firewall.

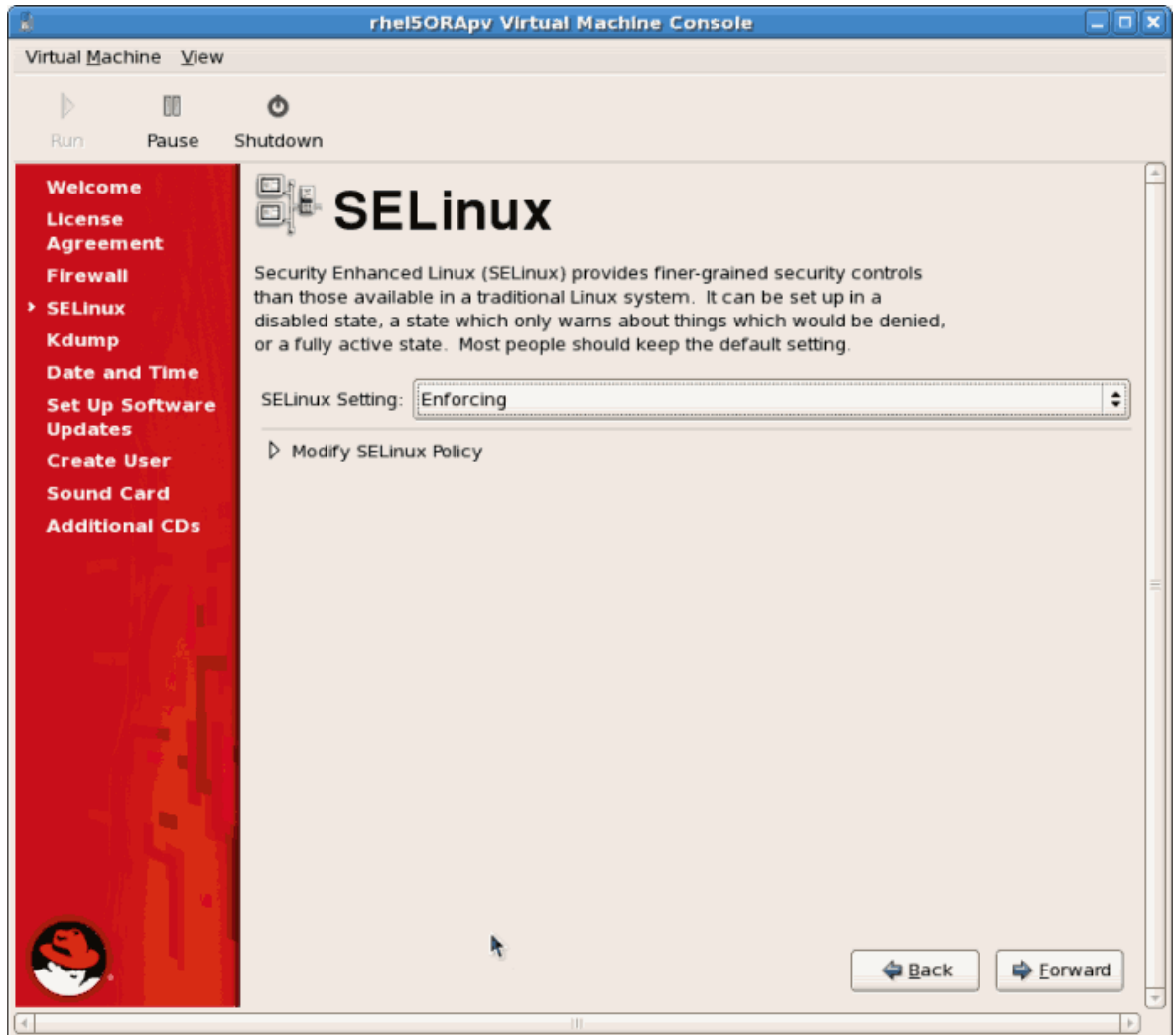


Click **Forward** to continue.

- Falls Sie die Firewall deaktivieren, müssen Sie diese Entscheidung noch einmal bestätigen. Klicken Sie auf **Ja**, um zu bestätigen und fortzufahren.



18. Konfigurieren Sie als Nächstes SELinux. Es wird dringend empfohlen, SELinux im **Enforcing-Modus** auszuführen. Sie können SELinux jedoch auch im Permissive-Modus ausführen oder vollständig deaktivieren.



Click **Forward** to continue.

- Falls Sie SELinux deaktivieren, wird diese Warnung angezeigt. Klicken Sie **Ja**, um SELinux zu deaktivieren.

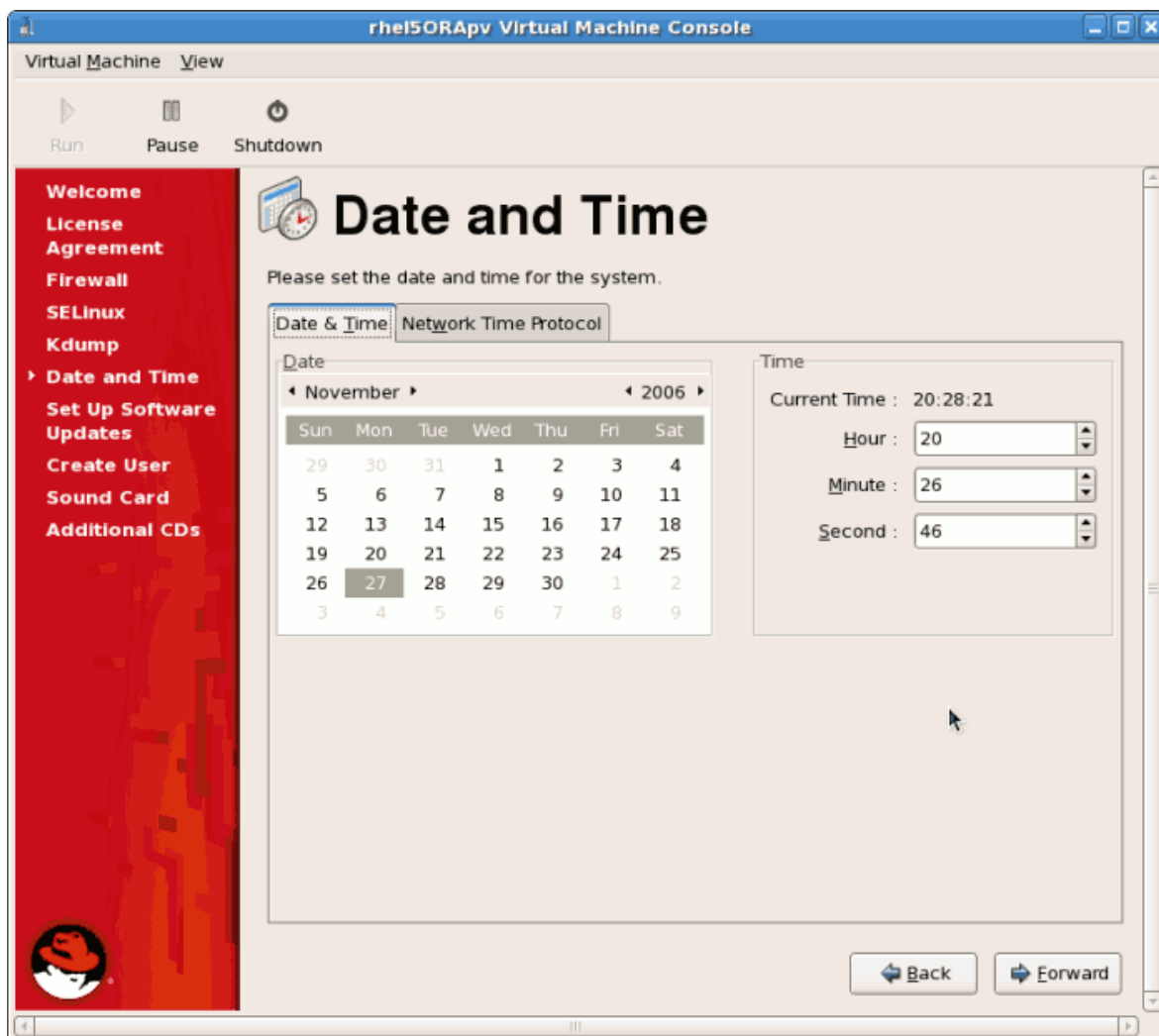


19. Aktivieren Sie **kdump**, falls nötig.



Click **Forward** to continue.

20. Bestätigen Sie, dass Uhrzeit und Datum korrekt für den Gast eingestellt wurden. Wenn Sie einen paravirtualisierten Gast installieren, sollten Uhrzeit und Datum synchron mit dem Hypervisor sein.



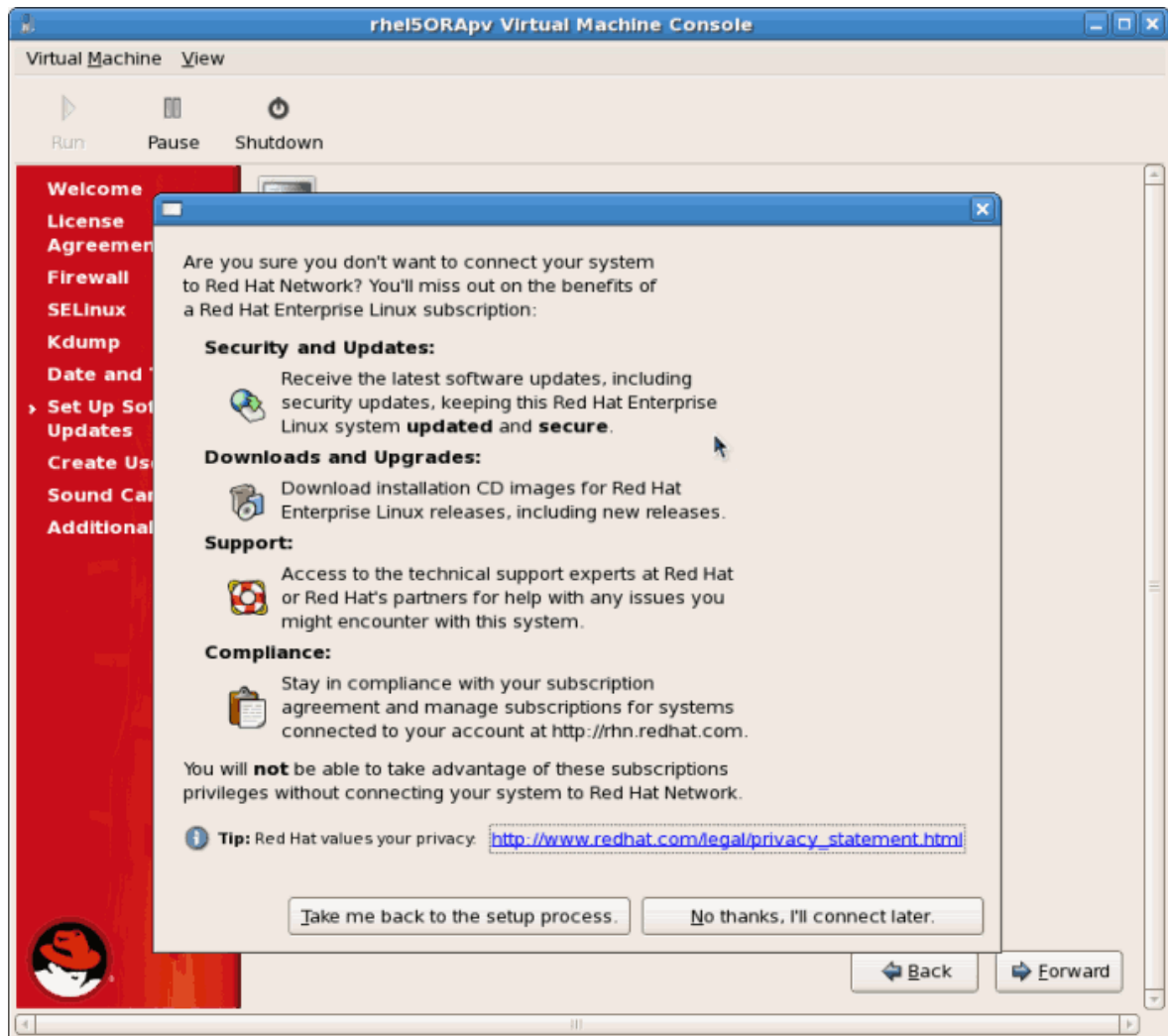
Click **Forward** to continue.

21. Richten Sie Software-Aktualisierungen ein. Falls Sie ein Fedora Network Abonnement haben oder eine Testversion ausprobieren möchten, können Sie auf dem unten gezeigten Bildschirm Ihren neu installierten Gast in RHN anmelden:

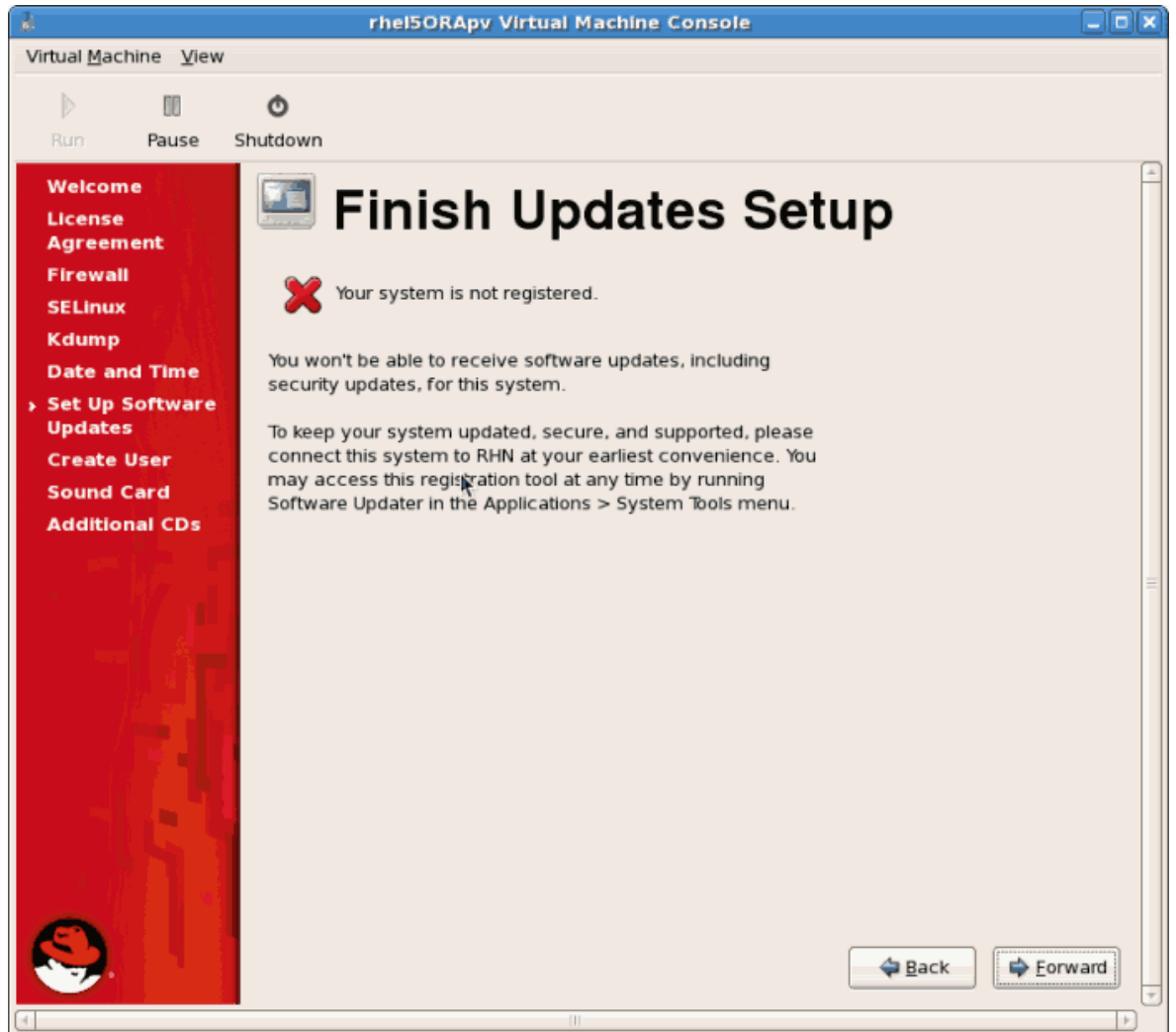


Click **Forward** to continue.

- a. Bestätigen Sie Ihre Auswahl für RHN.

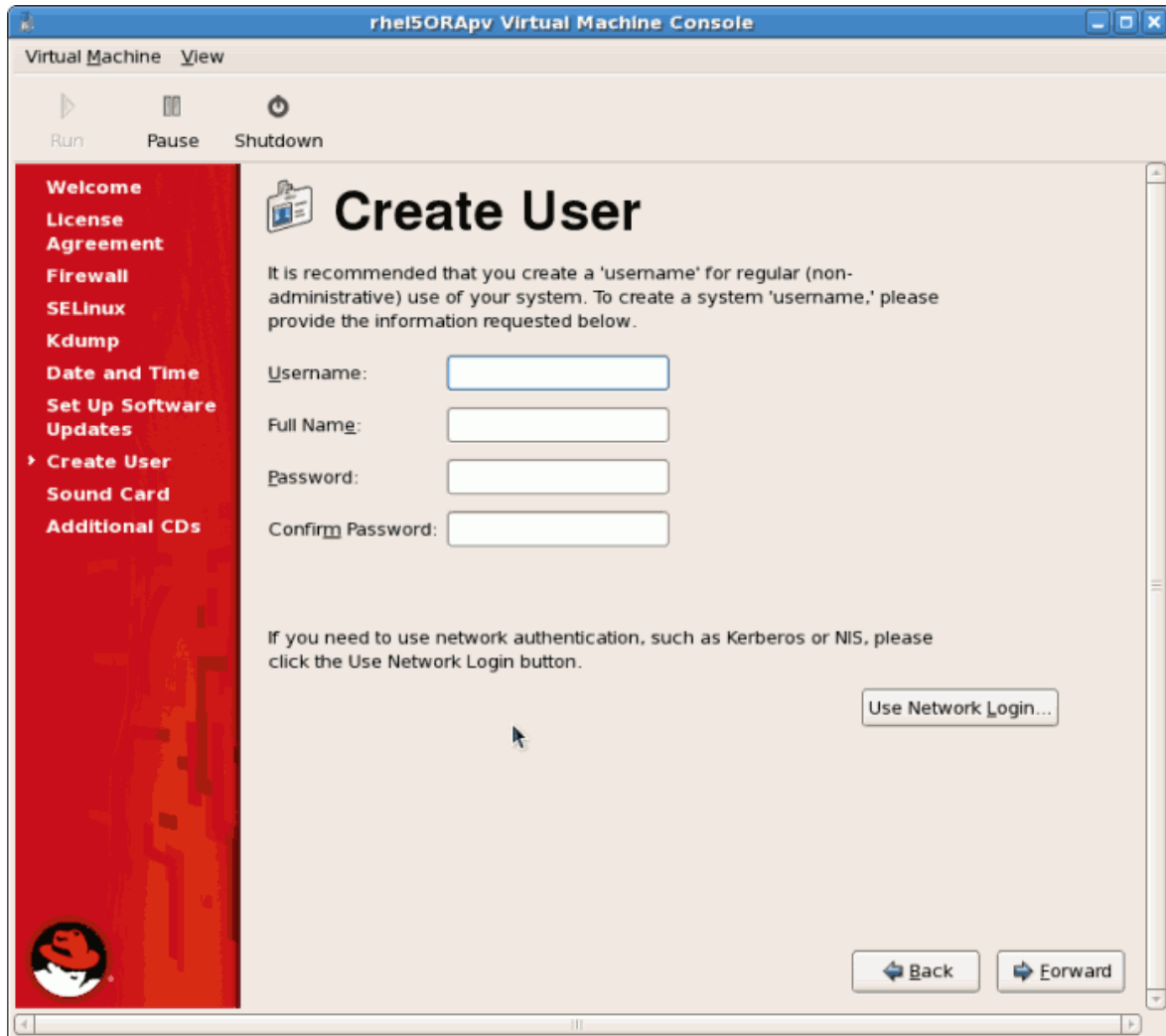


- b. Wenn das Setup abgeschlossen ist, sehen Sie ggf. noch einen weiteren Bildschirm, wenn Sie sich zu diesem Zeitpunkt gegen RHN entschieden haben. Sie werden keine Software-Aktualisierungen erhalten.



Klicken Sie auf **Weiter**.

22. Erstellen Sie ein nicht privilegiertes (nicht-Root) Benutzerkonto. Es wird empfohlen, dass Sie für normale Tätigkeiten aus Sicherheitsgründen ein nicht privilegiertes Benutzerkonto anlegen. Geben Sie den Benutzernamen, Namen und Passwort ein.

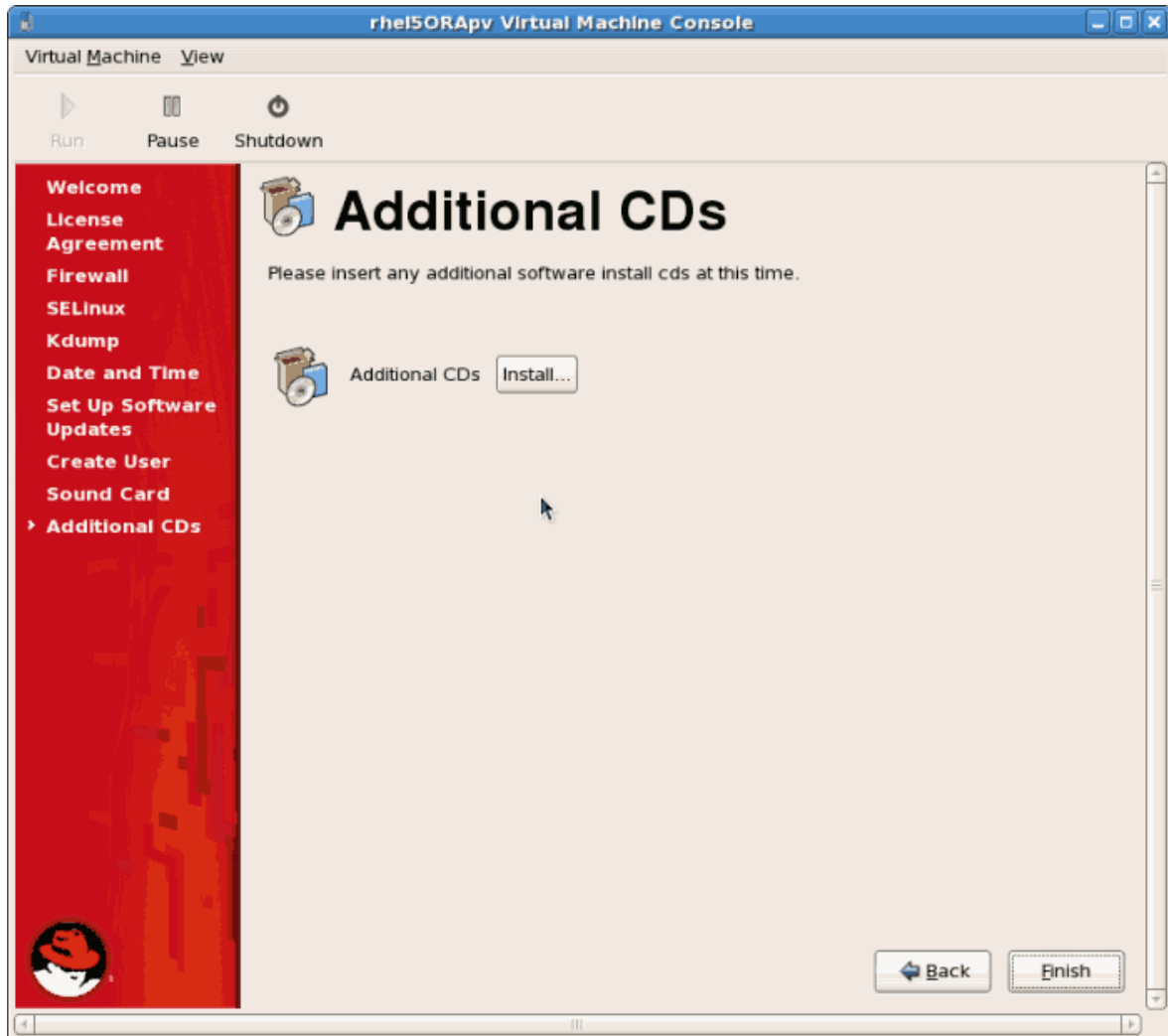


Klicken Sie auf **Weiter**.

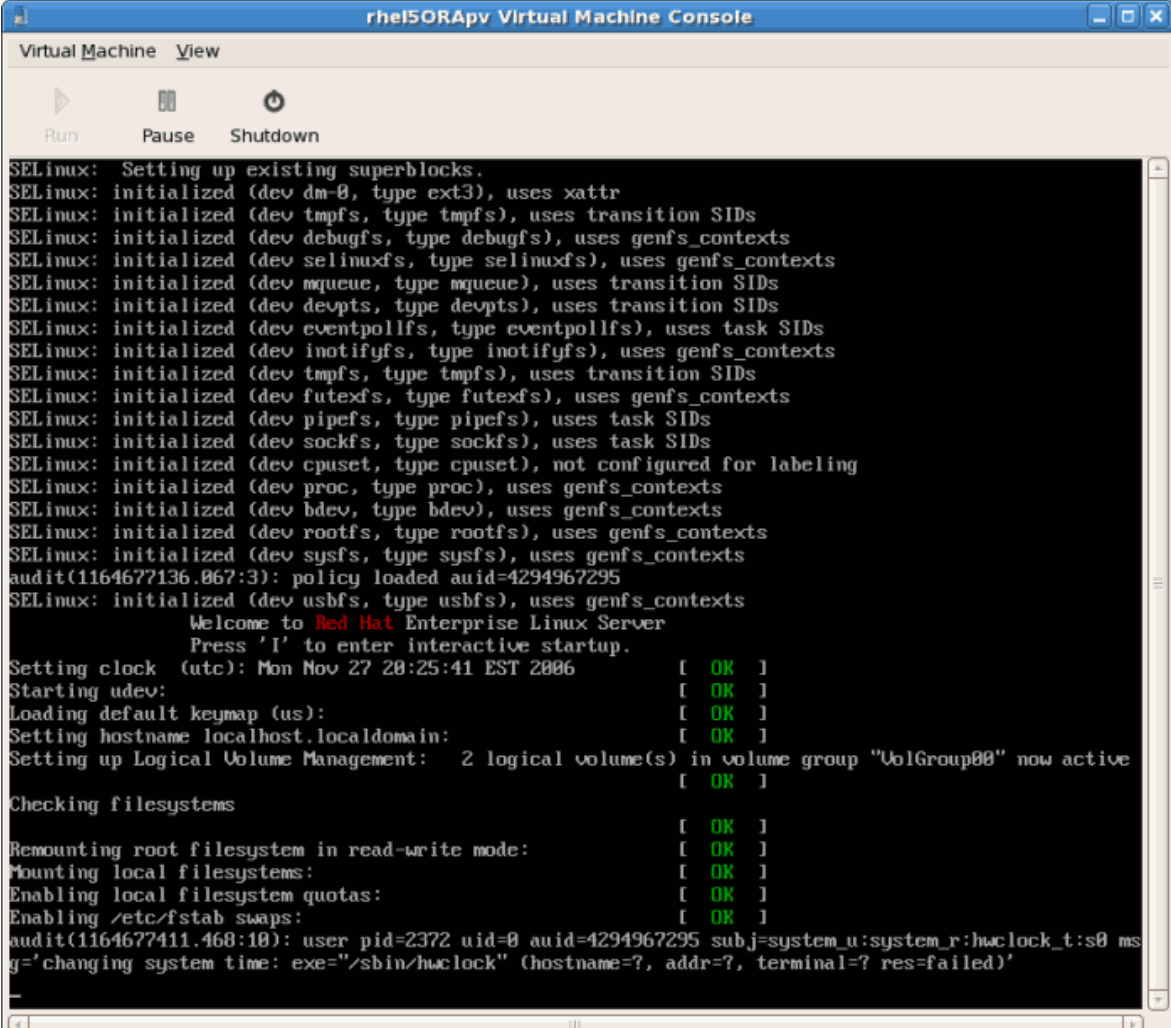
23. Wenn ein Audiogerät entdeckt wird und Sie Audioausgabe benötigen, kalibrieren Sie das Gerät. Schließen Sie den Vorgang ab und klicken auf **Weiter**.



24. Falls Sie zusätzliche Software-Pakete von CD installieren möchten, können Sie dies in diesem Bildschirm tun. Es ist oft effizienter, zu diesem Zeitpunkt noch keine zusätzliche Software zu installieren sondern diese später mit yum hinzuzufügen. Klicken Sie auf **Fertigstellen**.

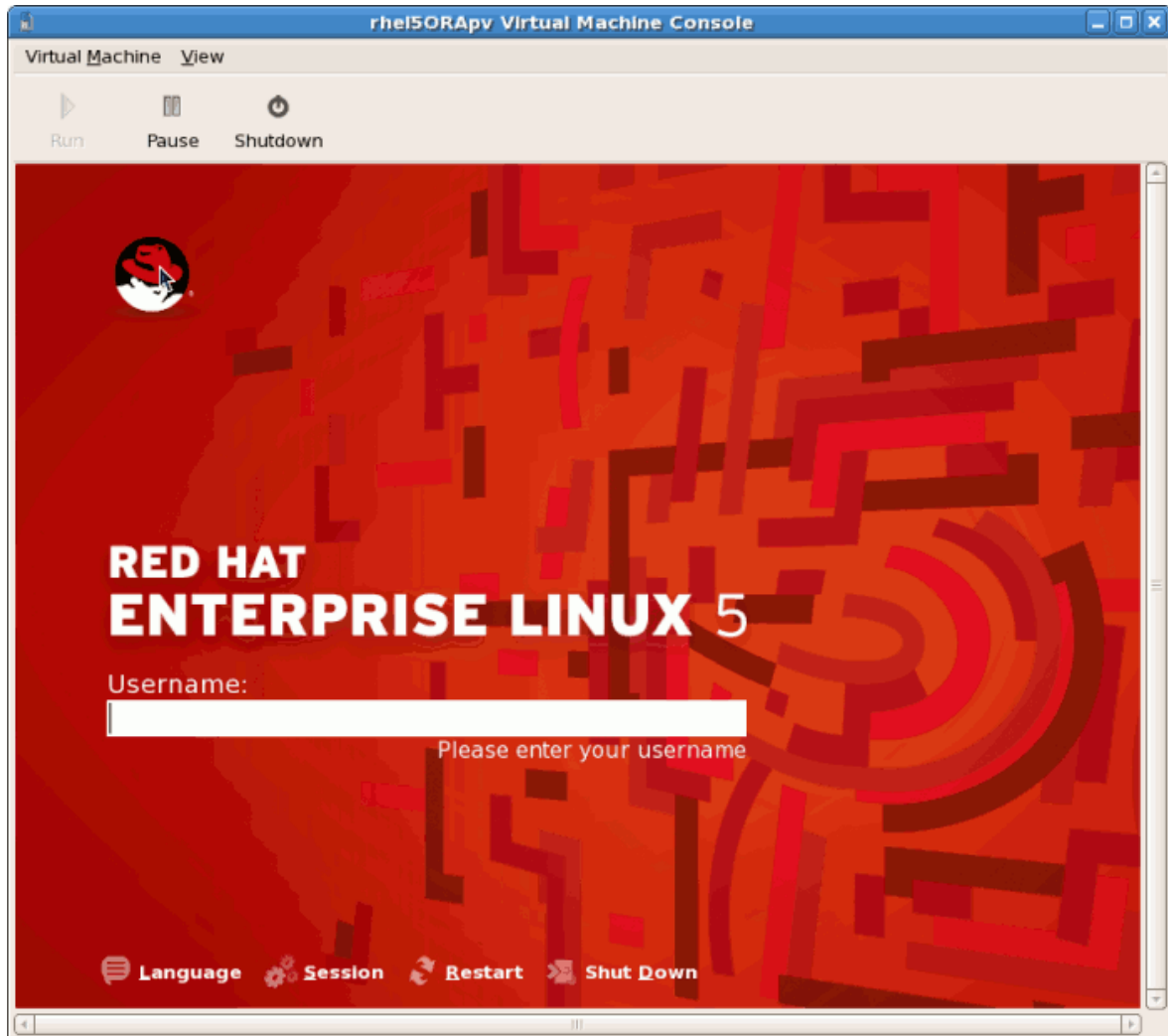


25. Der Gast konfiguriert nun alle von Ihnen vorgenommenen Einstellungen und fährt mit dem Bootvorgang fort.

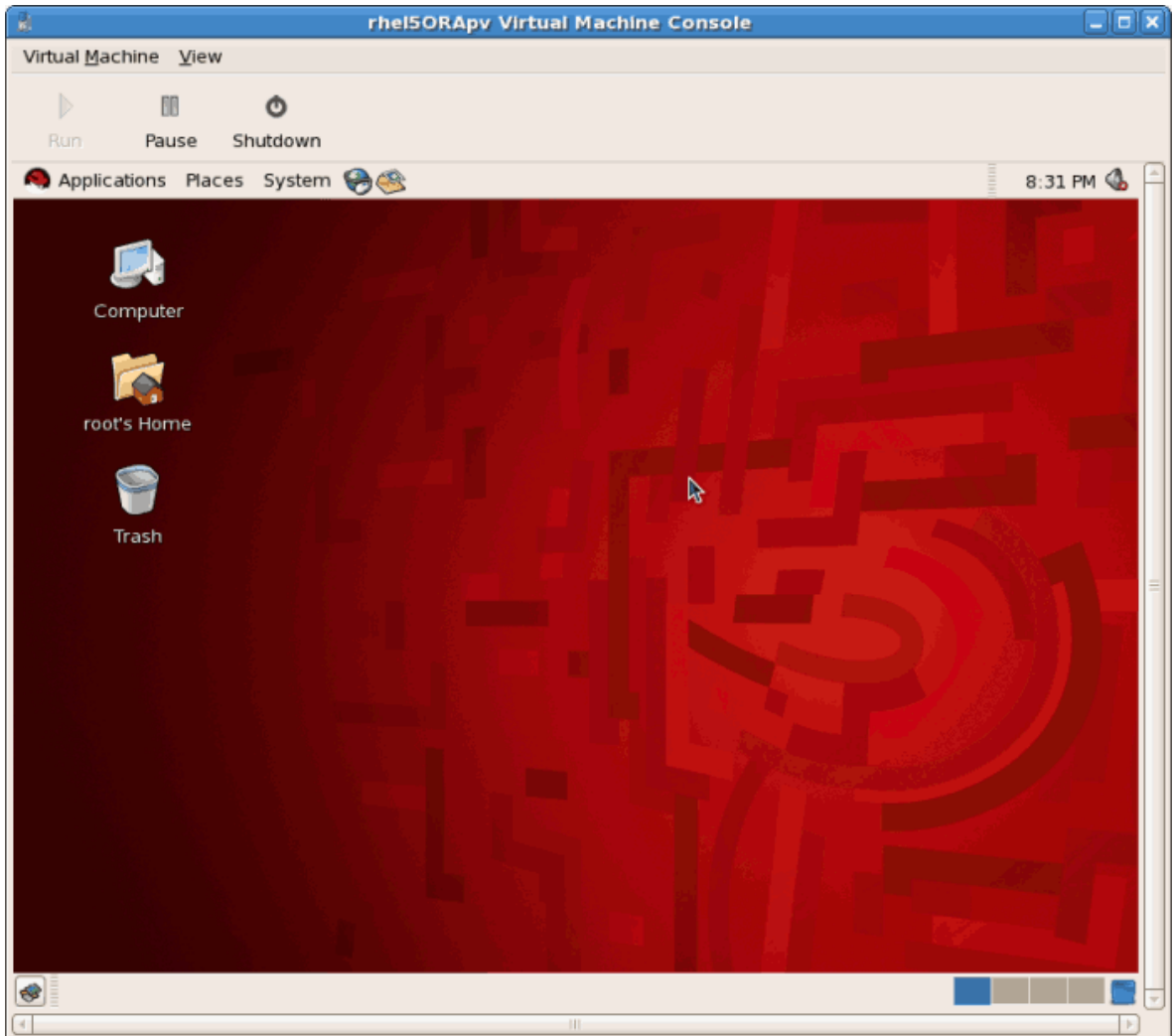


```
rhe5ORApv Virtual Machine Console
Virtual Machine View
Run Pause Shutdown
SELinux: Setting up existing superblocks.
SELinux: initialized (dev dm-0, type ext3), uses xattr
SELinux: initialized (dev tmpfs, type tmpfs), uses transition SIDs
SELinux: initialized (dev debugfs, type debugfs), uses genfs_contexts
SELinux: initialized (dev selinuxfs, type selinuxfs), uses genfs_contexts
SELinux: initialized (dev mqueue, type mqueue), uses transition SIDs
SELinux: initialized (dev devpts, type devpts), uses transition SIDs
SELinux: initialized (dev eventpollfs, type eventpollfs), uses task SIDs
SELinux: initialized (dev inotifyfs, type inotifyfs), uses genfs_contexts
SELinux: initialized (dev tmpfs, type tmpfs), uses transition SIDs
SELinux: initialized (dev futexfs, type futexfs), uses genfs_contexts
SELinux: initialized (dev pipefs, type pipefs), uses task SIDs
SELinux: initialized (dev sockfs, type sockfs), uses task SIDs
SELinux: initialized (dev cpuset, type cpuset), not configured for labeling
SELinux: initialized (dev proc, type proc), uses genfs_contexts
SELinux: initialized (dev bdev, type bdev), uses genfs_contexts
SELinux: initialized (dev rootfs, type rootfs), uses genfs_contexts
SELinux: initialized (dev sysfs, type sysfs), uses genfs_contexts
audit(1164677136.067:3): policy loaded auid=4294967295
SELinux: initialized (dev usbfs, type usbfs), uses genfs_contexts
Welcome to Red Hat Enterprise Linux Server
Press 'I' to enter interactive startup.
Setting clock (utc): Mon Nov 27 20:25:41 EST 2006 [ OK ]
Starting udev: [ OK ]
Loading default keymap (us): [ OK ]
Setting hostname localhost.localdomain: [ OK ]
Setting up Logical Volume Management: 2 logical volume(s) in volume group "VolGroup00" now active
[ OK ]
Checking filesystems [ OK ]
Remounting root filesystem in read-write mode: [ OK ]
Mounting local filesystems: [ OK ]
Enabling local filesystem quotas: [ OK ]
Enabling /etc/fstab swaps: [ OK ]
audit(1164677411.468:10): user pid=2372 uid=0 auid=4294967295 subj=system_u:system_r:hwclock_t:s0 msg='changing system time: exe="/sbin/hwclock" (hostname=?, addr=?, terminal=? res=failed)'
-
```

26. Der Red Hat Enterprise Linux 5 Anmeldebildschirm erscheint. Melden Sie sich mit dem Benutzerkonto an, das Sie in den vorangegangenen Schritten erstellt haben.



27. Sie haben nun erfolgreich einen paravirtualisierten Red Hat Enterprise Linux 5 Gast installiert.



3.2. Installation von Red Hat Enterprise Linux als voll virtualisierter Gast

Dieser Abschnitt beschreibt die Installation eines voll virtualisierten Red Hat Enterprise Linux 5 Gast.

Prozedur 3.3. Erzeugen eines voll virtualisierten Red Hat Enterprise Linux 5 Gasts mittels virt-manager

1. **Open virt-manager**
Start **virt-manager**. Launch the **Virtual Machine Manager** application from the **Applications** menu and **System Tools** submenu. Alternatively, run the **virt-manager** command as root.
2. **Select the hypervisor**
Select the hypervisor. If installed, select Xen or KVM. For this example, select KVM. Note that presently KVM is named qemu.

Verbinden Sie mit einem Hypervisor, falls noch nicht geschehen. Öffnen Sie das **Datei**-Menü und wählen die Option **Verbindung hinzufügen...** Siehe [Abschnitt 16.1, „Das Fenster "Verbindung öffnen"“](#).

Nachdem eine Hypervisor-Verbindung ausgewählt wurde, erscheint die Schaltfläche **Neu**. Klicken Sie auf **Neu**.

3. Start the new virtual machine wizard

Pressing the **New** button starts the virtual machine creation wizard.



Press **Forward** to continue.

4. Name the virtual machine

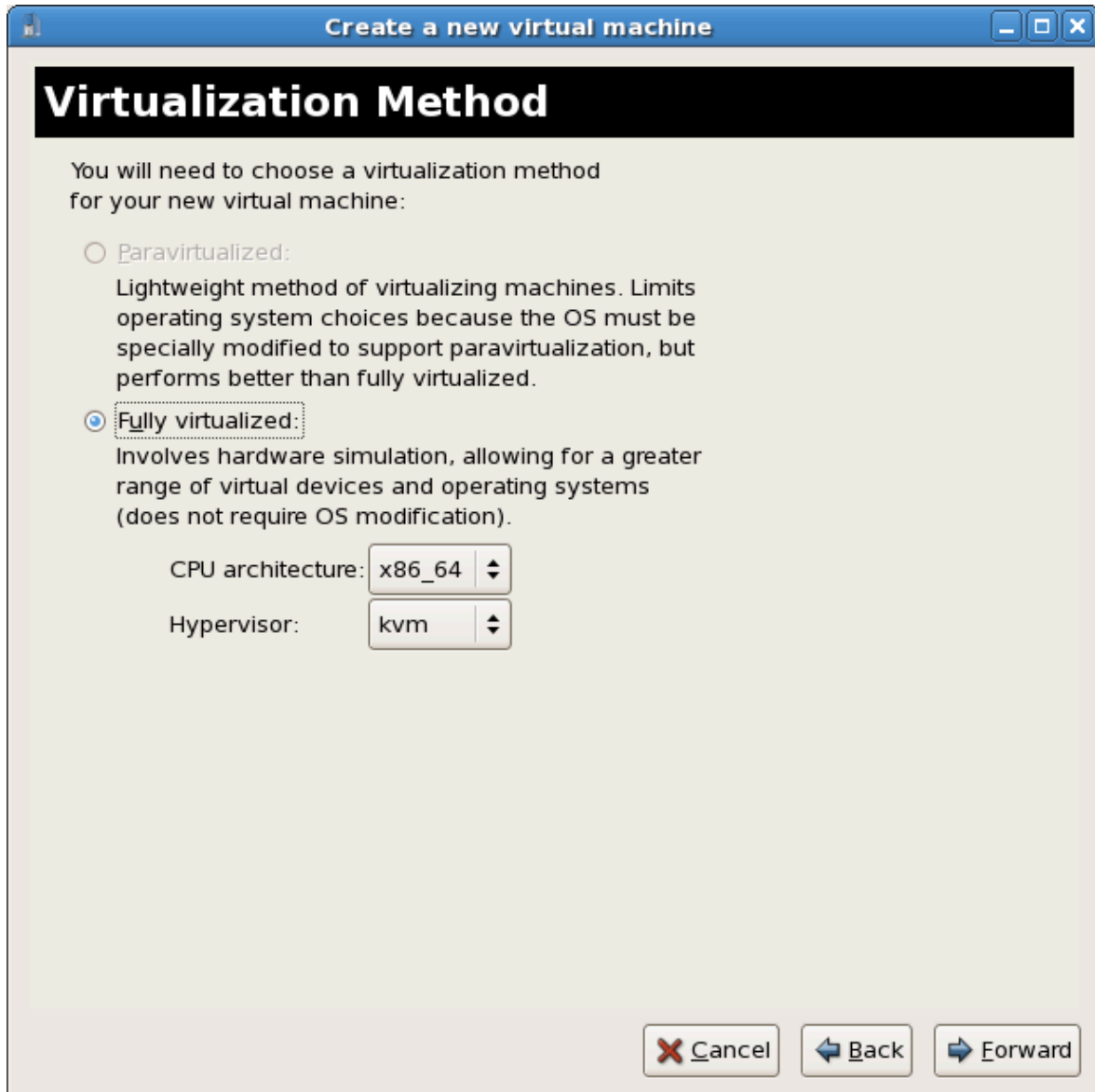
Geben Sie Ihrem virtualisierten Gast einen Namen. Satzzeichen und Leerstellen sind dabei nicht zulässig.



Klicken Sie auf **Weiter**, um fortzufahren.

5. **Choose a virtualization method**

Wählen Sie die Virtualisierungsmethode für den virtualisierten Gast. Beachten Sie, dass Sie nur eine installierte Virtualisierungsmethode auswählen können. Wenn Sie vorher KVM oder Xen gewählt haben ([Schritt 4](#)), müssen Sie den Hypervisor verwenden, den Sie ausgewählt haben. In diesem Beispiel wird der KVM-Hypervisor verwendet.



Klicken Sie auf **Weiter**, um fortzufahren.

6. **Select the installation method**

Wählen Sie **Lokales Installationsmedium** für eine Installation von einem optischen Speichermedium oder einem ISO-Abbild; **Netzwerkinstallationsbaum** für eine Installation von einem HTTP-, FTP-, oder NFS-Server; oder **Netzwerk-Boot** für eine Installation von einem PXE-Server.

Setzen Sie den **Betriebssystemtyp** auf **Linux** und die **Betriebssystemvariante** auf **Red Hat Enterprise Linux 5**, wie im Screenshot dargestellt.

The screenshot shows a window titled "Create a new virtual machine" with a sub-header "Installation Method". The main text asks the user to indicate where installation media is available. There are three radio button options: "Local install media (ISO image or CDROM)" (selected), "Network install tree (HTTP, FTP, or NFS)", and "Network boot (PXE)". Below this, the user is asked to choose the operating system. Two dropdown menus are shown: "OS Type" set to "Linux" and "OS Variant" set to "Red Hat Enterprise Linux 5". A lightbulb icon and text note that not all OS choices are supported by Red Hat, with a link to "Red Hat Enterprise Linux 5 virtualization support". At the bottom right, there are three buttons: "Cancel", "Back", and "Forward".

Klicken Sie auf **Weiter**, um fortzufahren.

7. **Locate installation media**

Wählen Sie "Speicherort des ISO-Abbilds" oder "CD-ROM oder DVD-Laufwerk". In diesem Beispiel wird ein ISO-Dateiabbild der Red Hat Enterprise Linux 5 Installations-DVD verwendet.

- a. Press the **Browse** button.
- b. Suchen Sie den Speicherort der ISO-Datei und wählen das ISO-Abbild aus. Klicken Sie auf **Öffnen**, um Ihre Auswahl zu bestätigen.
- c. Die Datei ist ausgewählt und bereit zur Installation.



Klicken Sie auf **Weiter**, um fortzufahren.



Image files and SELinux

Für ISO-Abbilddateien und Gast Speicherabbilder sollte das `/var/lib/libvirt/images/`-Verzeichnis verwendet werden. Abweichende Speicherorte erfordern unter Umständen zusätzliche Konfiguration von SELinux. Siehe *Abschnitt 7.1, „SELinux und Virtualisierung“* für weitere Einzelheiten.

8. Storage setup

Weisen Sie ein physisches Speichergerät (**Blockgerät**) oder ein dateibasiertes Abbild (**Datei**) zu. Dateibasierte Abbilder müssen im `/var/lib/libvirt/images/`-Verzeichnis abgelegt sein. Weisen Sie ausreichend Speicherplatz für Ihren virtualisierten Gast und dessen benötigte Anwendungen zu.

Create a new virtual machine

Storage

Please indicate how you'd like to assign space from the host for your new virtual machine. This space will be used to install the virtual machine's operating system.

Block device (partition):

Location:

i Example: /dev/hdc2

File (disk image):

Location:

Size: MB

Allocate entire virtual disk now

⚠ Warning: If you do not allocate the entire disk now, space will be allocated as needed while the virtual machine is running. If sufficient free space is not available on the host, this may result in data corruption on the virtual machine.

i Tip: You may add additional storage, including network-mounted storage, to your virtual machine after it has been created using the same tools you would on a physical system.

Klicken Sie auf **Weiter**, um fortzufahren.

Um diesen Gast zu migrieren

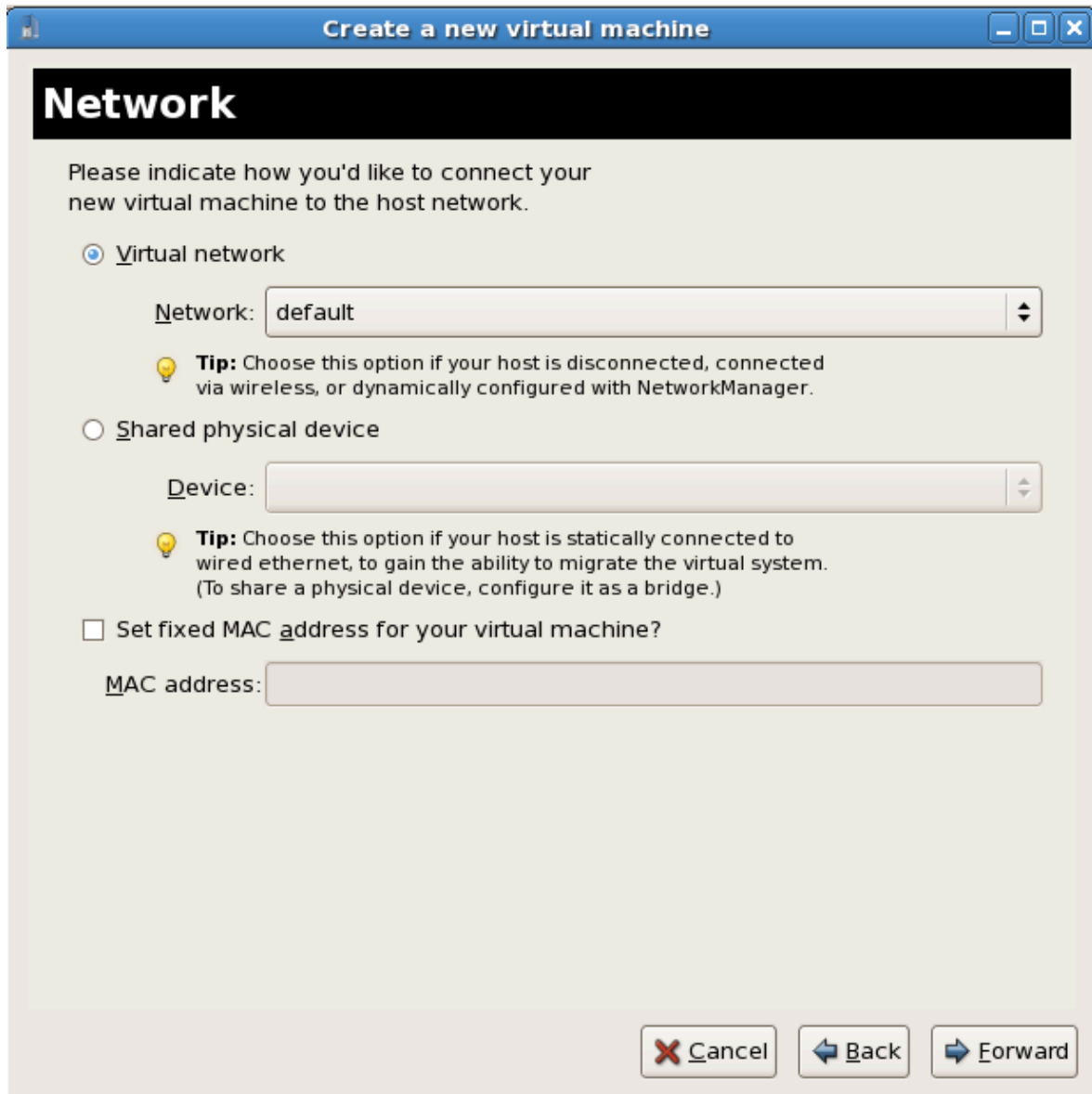
Für Live- und Offline-Migrationen ist es erforderlich, dass Gäste auf gemeinsam verwendeten Netzwerkspeicher installiert sind. Informationen zum Einrichten von gemeinsam genutzten Speicher für Gäste finden Sie unter *Kapitel 5, Gemeinsam verwendeter Speicher und Virtualisierung*.

9. Network setup

Select either **Virtual network** or **Shared physical device**.

The virtual network option uses Network Address Translation (NAT) to share the default network device with the virtualized guest. Use the virtual network option for wireless networks.

The shared physical device option uses a network bond to give the virtualized guest full access to a network device.



Press **Forward** to continue.

10. **Memory and CPU allocation**

The Allocate memory and CPU window displays. Choose appropriate values for the virtualized CPUs and RAM allocation. These values affect the host's and guest's performance.

Virtualized guests require sufficient physical memory (RAM) to run efficiently and effectively. Choose a memory value which suits your guest operating system and application requirements. Windows Server 2008. Remember, guests use physical RAM. Running too many guests or leaving insufficient memory for the host system results in significant usage of virtual memory and swapping. Virtual memory is significantly slower causing degraded system performance and responsiveness. Ensure to allocate sufficient memory for all guests and the host to operate effectively.

Assign sufficient virtual CPUs for the virtualized guest. If the guest runs a multithreaded application assign the number of virtualized CPUs it requires to run most efficiently. Do not assign more virtual CPUs than there are physical processors (or hyper-threads) available on

the host system. It is possible to over allocate virtual processors, however, over allocating has a significant, negative affect on guest and host performance due to processor context switching overheads.

Memory:
Please enter the memory configuration for this virtual machine. You can specify the maximum amount of memory the virtual machine should be able to use, and optionally a lower amount to grab on startup. Warning: setting virtual machine memory too high will cause out-of-memory errors in your host domain!

Total memory on host machine: 2.89 GB

Max memory (MB): 1024

Startup memory (MB): 1024

CPUs:
Please enter the number of virtual CPUs this virtual machine should start up with.

Logical host CPUs: 4

Maximum virtual CPUs: 16

Virtual CPUs: 2

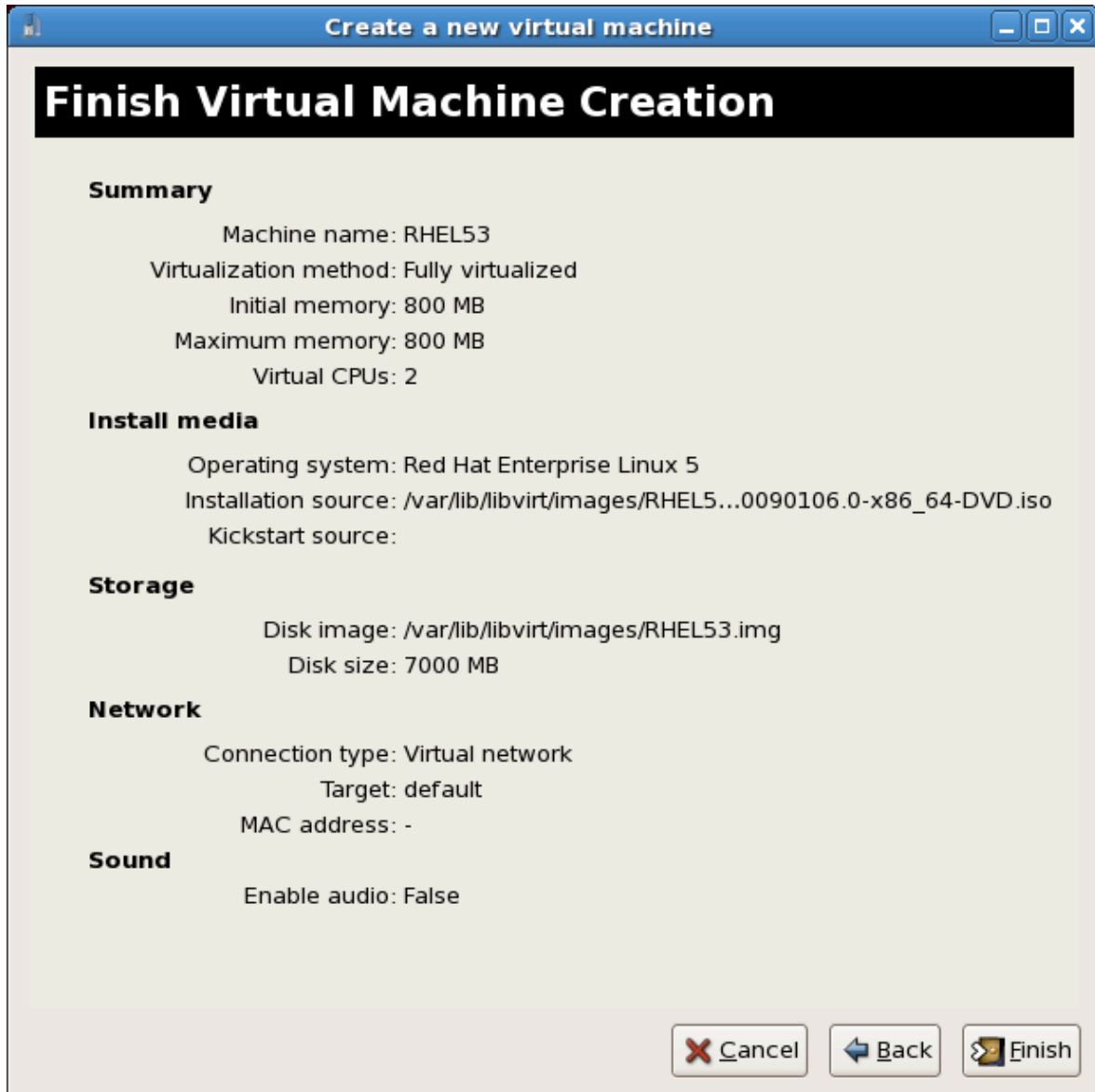
Tip: For best performance, the number of virtual CPUs should be less than (or equal to) the number of physical CPUs on the host system.

Cancel Back Forward

Press **Forward** to continue.

11. Verify and start guest installation

Überprüfen Sie die Konfiguration.



Klicken Sie auf **Fertigstellen**, um den Installationsvorgang zu starten.

12. Installation von Linux

Stellen Sie die Red Hat Enterprise Linux 5 Installationssequenz fertig. Die Installationssequenz wird im *Red Hat Enterprise Linux Installationshandbuch* behandelt, erhältlich unter <http://redhat.com/docs>.

Ein voll virtualisierter Red Hat Enterprise Linux 5 Gast ist nun fertig installiert.

3.3. Installation von Windows XP als voll virtualisierter Gast

Windows XP kann als voll virtualisierter Gast installiert werden. Dieser Abschnitt beschreibt, wie Windows XP als voll virtualisierter Gast auf Linux installiert werden kann.

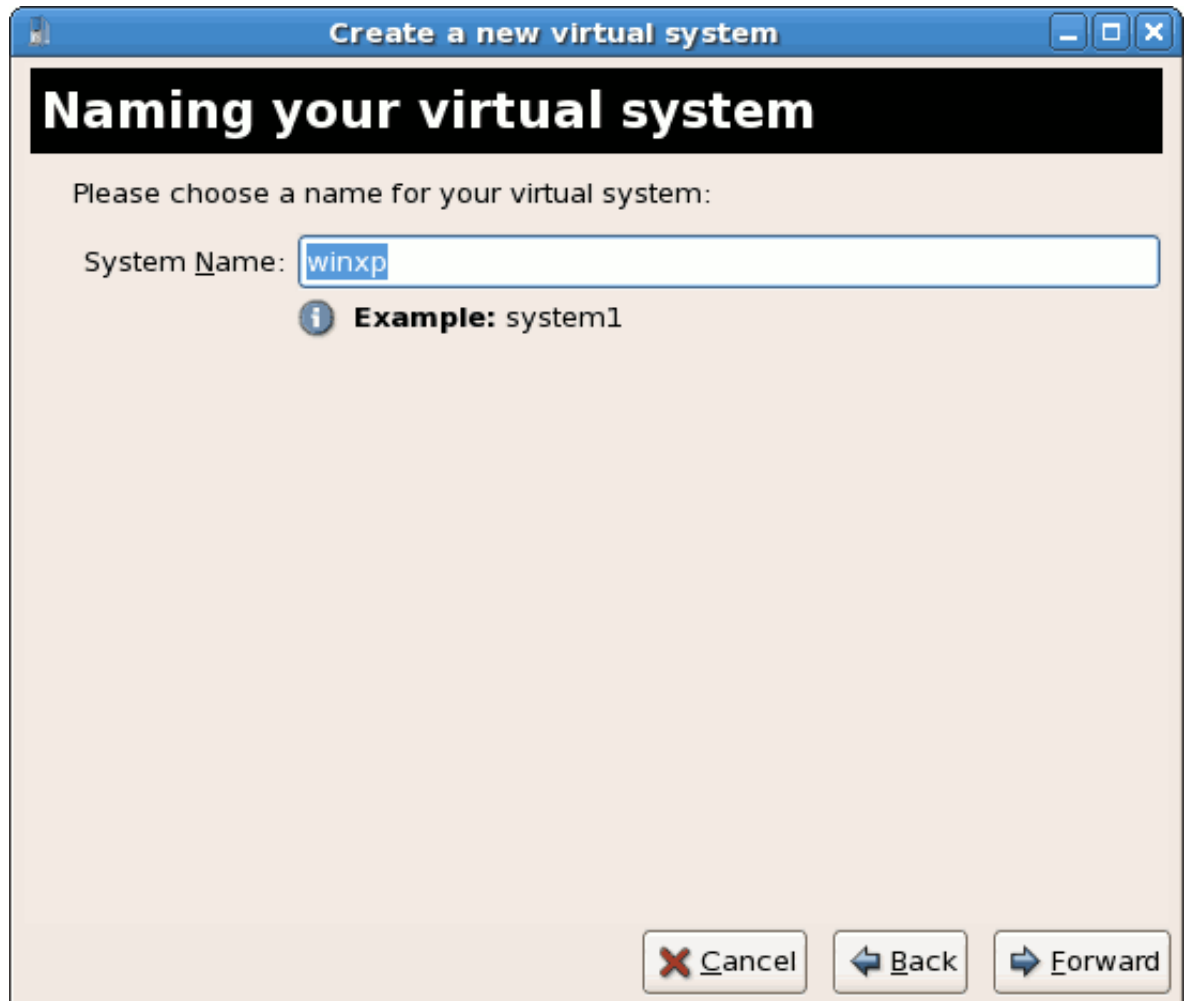
Bevor Sie mit der Prozedur beginnen, vergewissern Sie sich, dass Sie Root-Rechte besitzen.

1. **Starting virt-manager**

Öffnen Sie **Anwendungen > Systemwerkzeuge > Virtual Machine Manager**. Öffnen Sie eine Verbindung zum Host (klicken Sie dazu **Datei > Verbindung öffnen**). Klicken Sie auf die Schaltfläche **Neu**, um eine neue virtuelle Maschine anzulegen.

2. **Benennen Sie das virtuelle System**

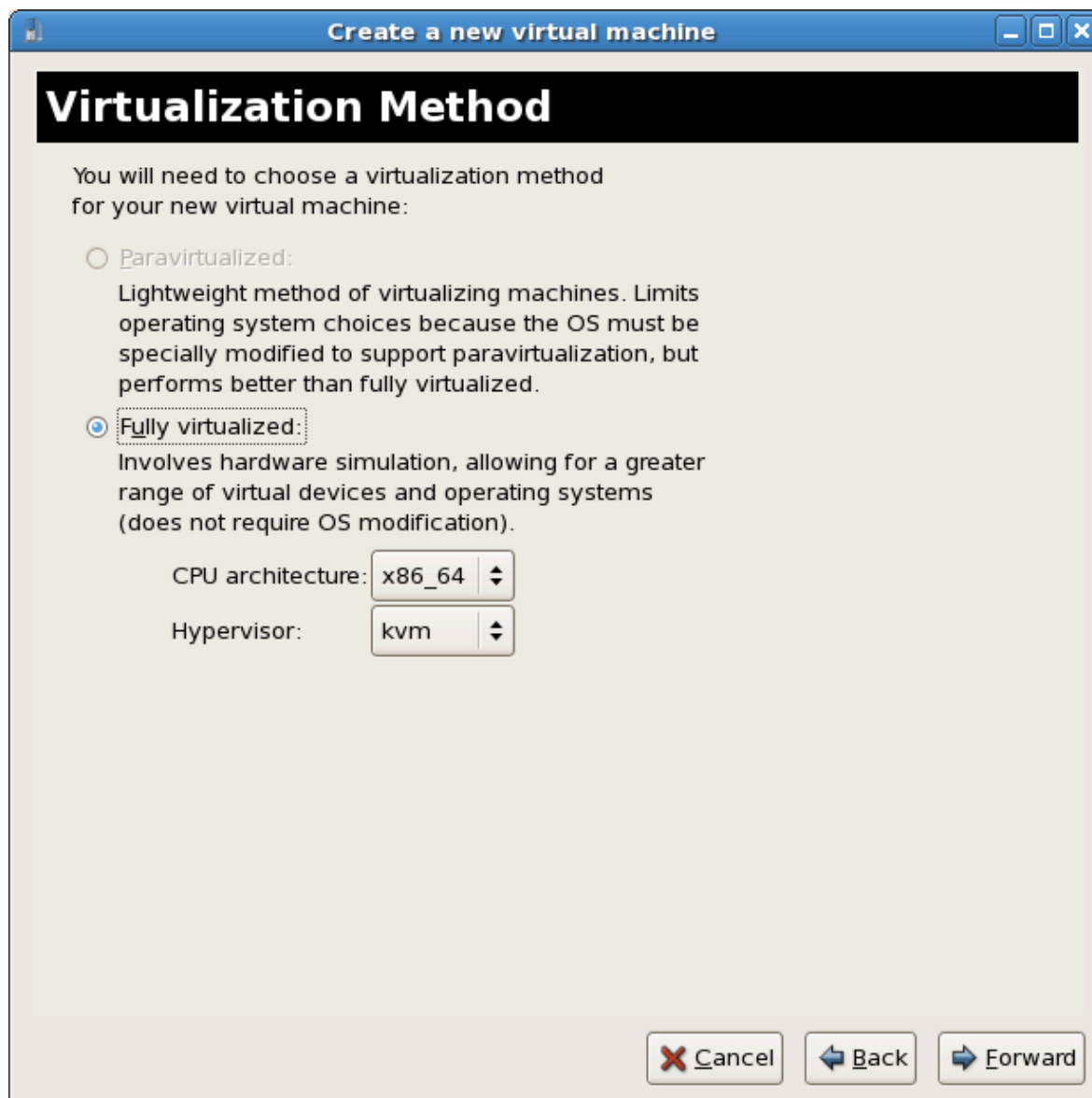
Geben Sie den **Systemnamen** ein und klicken anschließend auf **Weiter**.



3. **Wählen Sie eine Virtualisierungsmethode**

Wenn Sie vorher KVM oder Xen gewählt haben (*Schritt 1*), müssen Sie den Hypervisor verwenden, den Sie ausgewählt haben. In diesem Beispiel wird der KVM-Hypervisor verwendet.

Windows kann nur unter Verwendung der vollen Virtualisierung installiert werden.



4. **Wählen Sie die Installationsart**

In diesem Bildschirm können Sie die Installationsart und den Betriebssystemtyp wählen.

Wählen Sie für CD-ROM oder DVD-Installation das Gerät aus, das die Windows-Installations-CD bzw. DVD enthält. Falls Sie **Speicherort des ISO-Abbilds** wählen, geben Sie den Pfad zum .iso-Abbild der Windows-Installation an.

Wählen Sie **Windows** aus der Liste der **Betriebssystemtypen** und Microsoft Windows XP aus der Liste der **Betriebssystemversionen**.

PXE-Installation wird in diesem Kapitel nicht behandelt.

Create a new virtual system

Locating installation media

Please indicate where installation media is available for the operating system you would like to install on this **fully virtualized** virtual system:

ISO Image Location:

ISO Location:

CD-ROM or DVD:

Path to install media:

Network PXE boot

Please choose the type of guest operating system you will be installing:

OS Type:

OS Variant:

Press **Forward** to continue.



Image files and SELinux

Für ISO-Abbilddateien und Gastsspeicherabbilder sollte das `/var/lib/libvirt/images/`-Verzeichnis verwendet werden. Abweichende Speicherorte erfordern unter Umständen zusätzliche Konfiguration von SELinux. Siehe [Abschnitt 7.1, „SELinux und Virtualisierung“](#) für weitere Einzelheiten.

5. The **Assigning storage space** window displays. Choose a disk partition, LUN or create a file based image for the guest storage.

Alle Abbilddateien sollten im `/var/lib/libvirt/images/`-Verzeichnis abgelegt werden. Andere Speicherorte für dateibasierte Abbilder werden von SELinux verweigert. Falls Sie SELinux im Enforcing-Modus ausführen, werfen Sie einen Blick auf [Abschnitt 7.1, „SELinux und Virtualisierung“](#) für weitere Informationen über die Installation von Gästen.

Your guest storage image should be larger than the size of the installation, any additional packages and applications, and the size of the guests swap file. The installation process will choose the size of the guest's swap file based on size of the RAM allocated to the guest.

Allocate extra space if the guest needs additional space for applications or other data. For example, web servers require additional space for log files.

The screenshot shows a window titled "Create a new virtual system" with a sub-header "Assigning storage space". The main text reads: "Please indicate how you'd like to assign space on this physical host system for your new virtual system. This space will be used to install the virtual system's operating system." There are two radio button options: "Normal Disk Partition:" and "Simple File:". The "Simple File:" option is selected. Under "Simple File:", there is a "File Location:" field containing "/var/lib/libvirt/images/windows-" and a "Browse..." button. Below that is a "File Size:" field with a spinner set to "6000" and "MB". A checked checkbox "Allocate entire virtual disk now?" is present. A warning icon and text state: "Warning: If you do not allocate the entire disk at VM creation, space will be allocated as needed while the guest is running. If sufficient free space is not available on the host, this may result in data corruption on the guest." A tip icon and text state: "Tip: You may add additional storage, including network-mounted storage, to your virtual system after it has been created using the same tools you would on a physical system." At the bottom, there are buttons for "Help", "Cancel", "Back", and "Forward".

Choose the appropriate size for the guest on your selected storage type and click the **Forward** button.



Anmerkung

Es wird empfohlen, das Standardverzeichnis für die virtuellen Maschinenabbilder zu verwenden, also `/var/lib/libvirt/images/`. Falls Sie einen anderen Speicherort verwenden (wie z. B. `/images/` in diesem Beispiel), stellen Sie sicher, dass Sie ihn in der SELinux-Richtlinie hinzugefügt und neu gekennzeichnet haben,

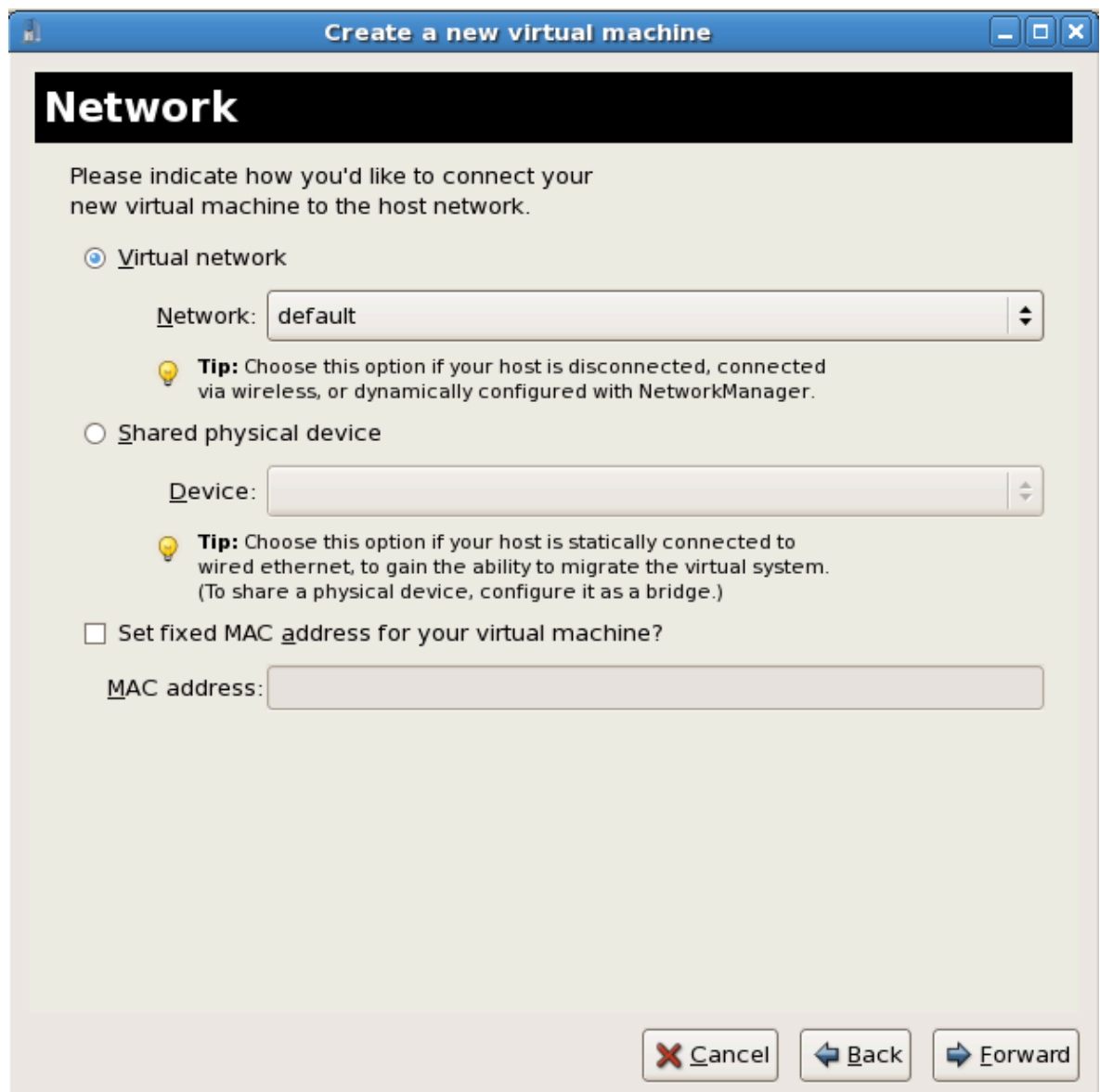
bevor Sie mit der Installation fortfahren (an späterer Stelle im Dokument finden Sie Informationen darüber, wie Sie Ihre SELinux-Richtlinie anpassen).

6. Network setup

Select either **Virtual network** or **Shared physical device**.

The virtual network option uses Network Address Translation (NAT) to share the default network device with the virtualized guest. Use the virtual network option for wireless networks.

The shared physical device option uses a network bond to give the virtualized guest full access to a network device.



The screenshot shows a window titled "Create a new virtual machine" with a "Network" sub-header. The main text asks the user to indicate how to connect the new virtual machine to the host network. There are three main options:

- Virtual network**: This option is selected. It includes a "Network:" dropdown menu currently set to "default". A tip below it reads: "Tip: Choose this option if your host is disconnected, connected via wireless, or dynamically configured with NetworkManager."
- Shared physical device**: This option is unselected. It includes a "Device:" dropdown menu. A tip below it reads: "Tip: Choose this option if your host is statically connected to wired ethernet, to gain the ability to migrate the virtual system. (To share a physical device, configure it as a bridge.)"
- Set fixed MAC address for your virtual machine?**: This checkbox is unselected. Below it is a "MAC address:" text input field.

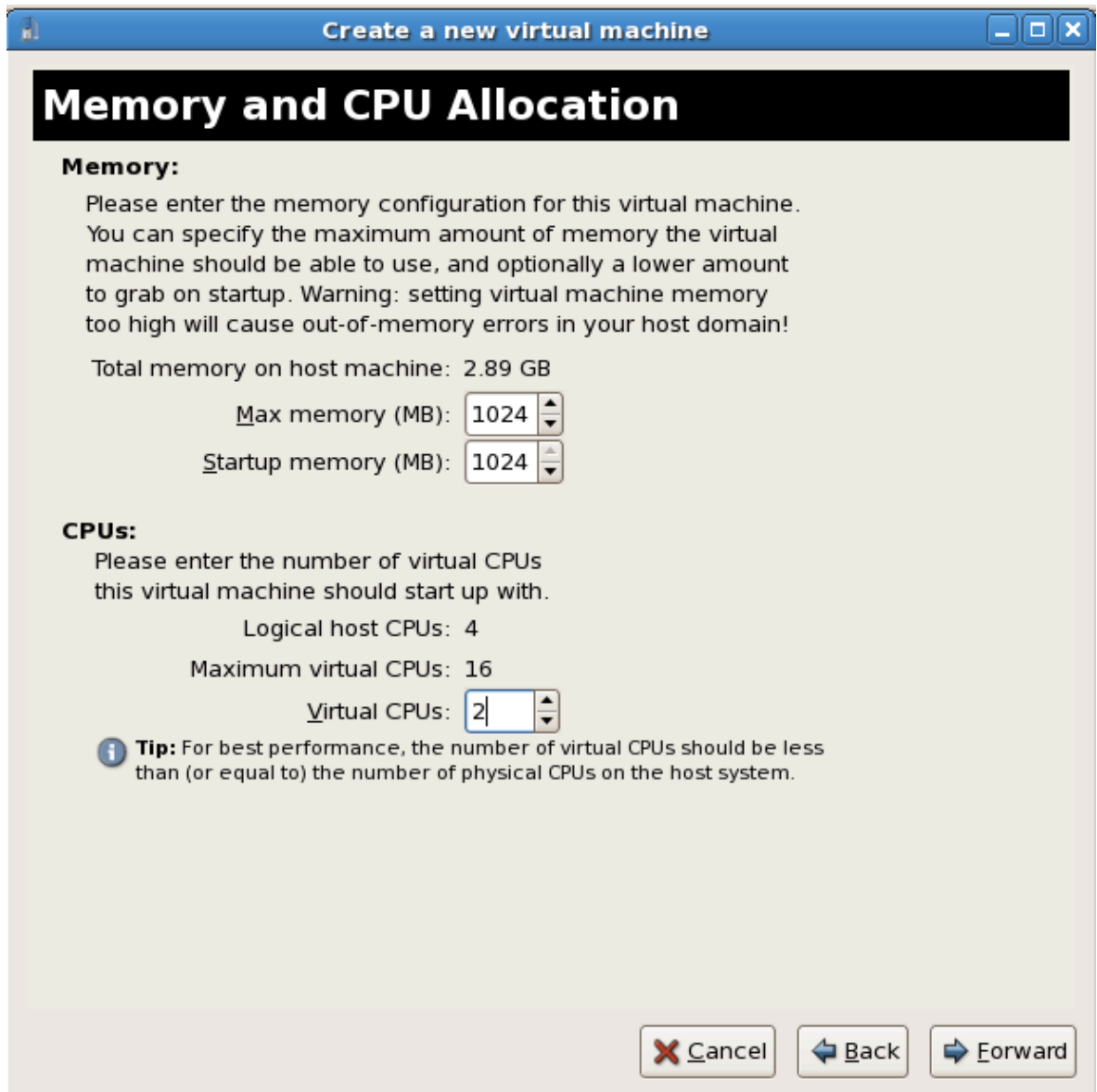
At the bottom right of the dialog, there are three buttons: "Cancel" (with a red X icon), "Back" (with a left arrow icon), and "Forward" (with a right arrow icon).

Press **Forward** to continue.

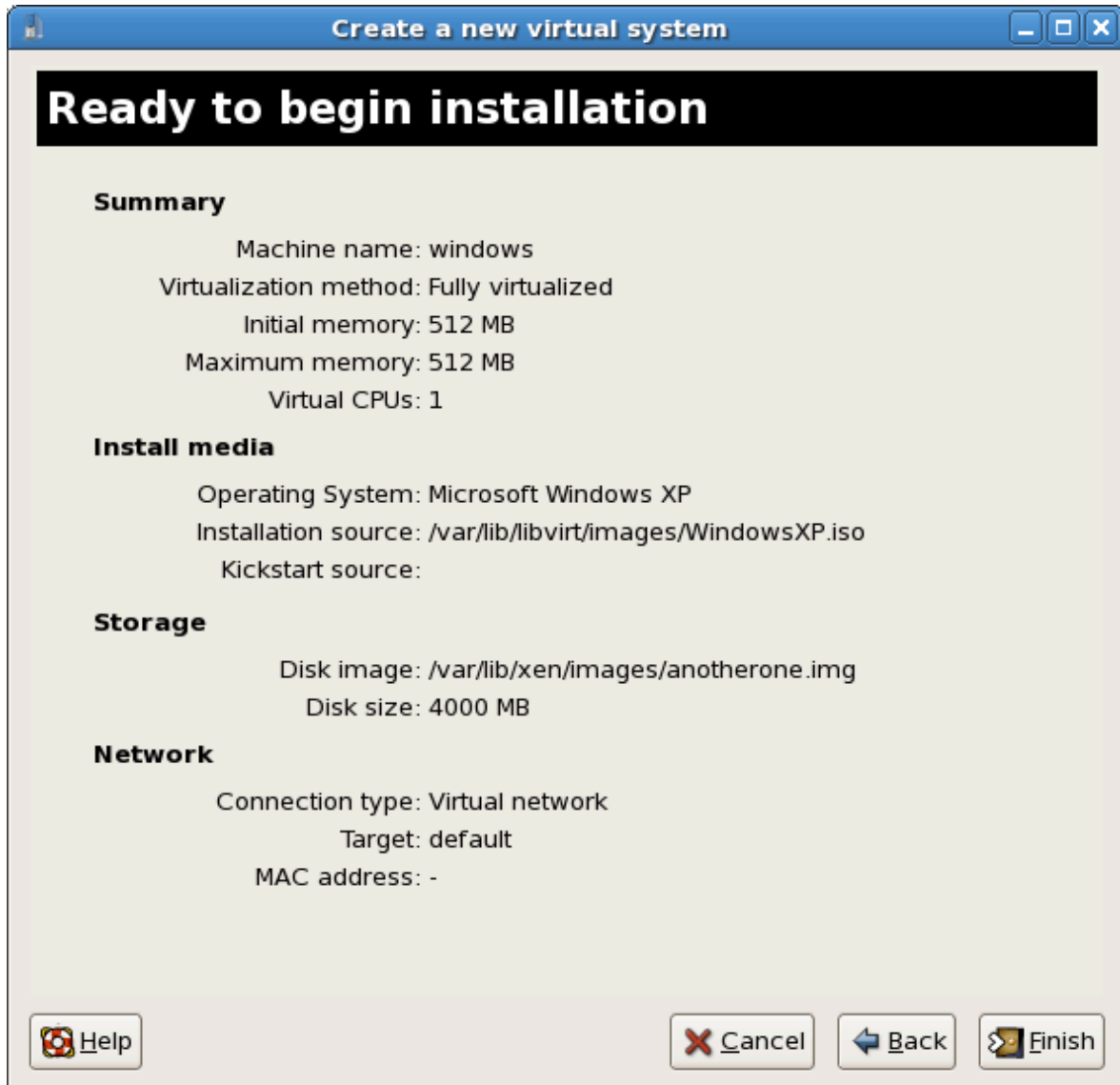
7. The Allocate memory and CPU window displays. Choose appropriate values for the virtualized CPUs and RAM allocation. These values affect the host's and guest's performance.

Gäste benötigen ausreichend physischen Arbeitsspeicher (RAM), um effektiv und effizient zu arbeiten. Wählen Sie einen Wert für den Speicher, der am Besten Ihrem Gastbetriebssystem und den Anforderungen der Anwendungen gerecht wird. Die meisten Betriebssysteme benötigen mindestens 512 MB RAM, um effizient zu arbeiten. Bedenken Sie, dass Gäste physischen RAM verbrauchen. Falls zu viele Gäste ausgeführt werden oder nicht genügend Arbeitsspeicher für das Host-System verbleibt, wird verstärkt virtueller Speicher genutzt. Virtueller Speicher hat jedoch deutlich langsamere Zugriffszeiten, infolgedessen sinkt die Leistung und Reaktionszeit des Systems. Stellen Sie daher sicher, dass Sie ausreichend Speicher zuweisen, damit sowohl alle Gäste als auch der Host effektiv arbeiten können.

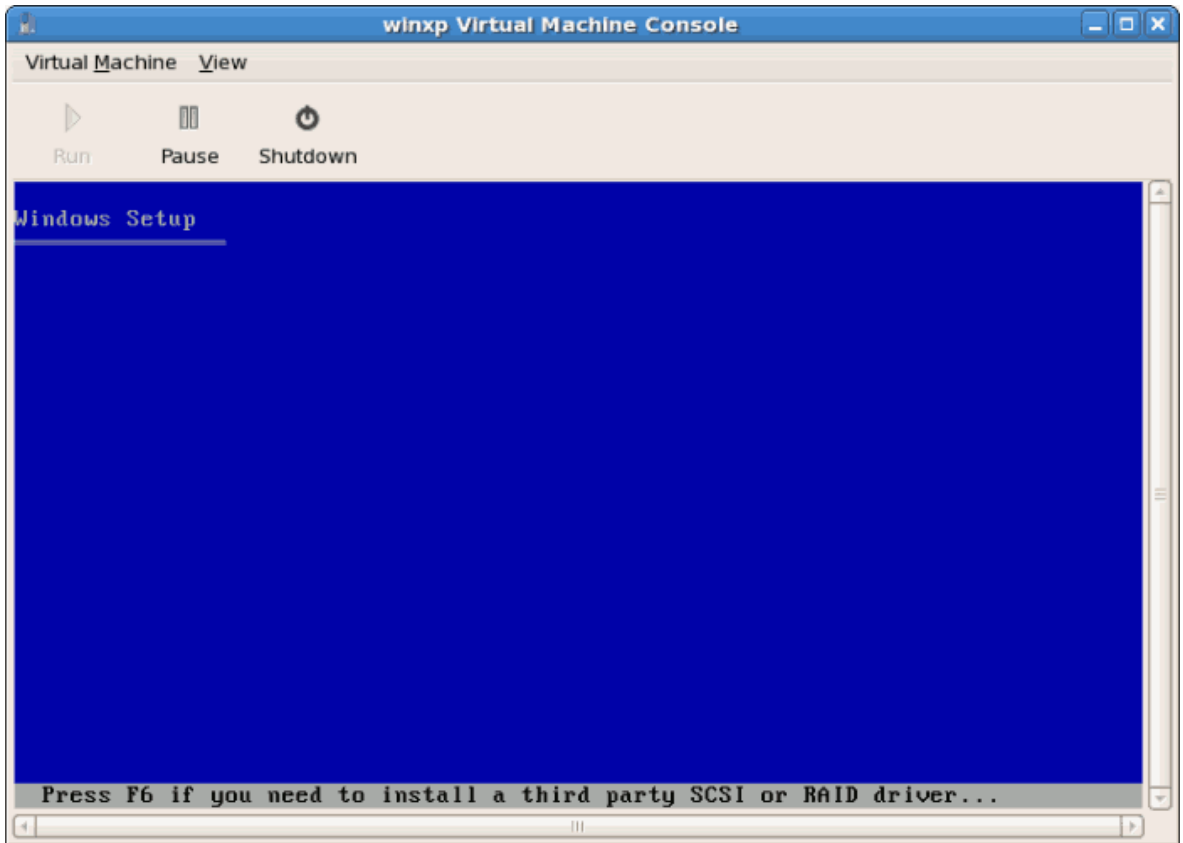
Assign sufficient virtual CPUs for the virtualized guest. If the guest runs a multithreaded application assign the number of virtualized CPUs it requires to run most efficiently. Do not assign more virtual CPUs than there are physical processors (or hyper-threads) available on the host system. It is possible to over allocate virtual processors, however, over allocating has a significant, negative affect on guest and host performance due to processor context switching overheads.



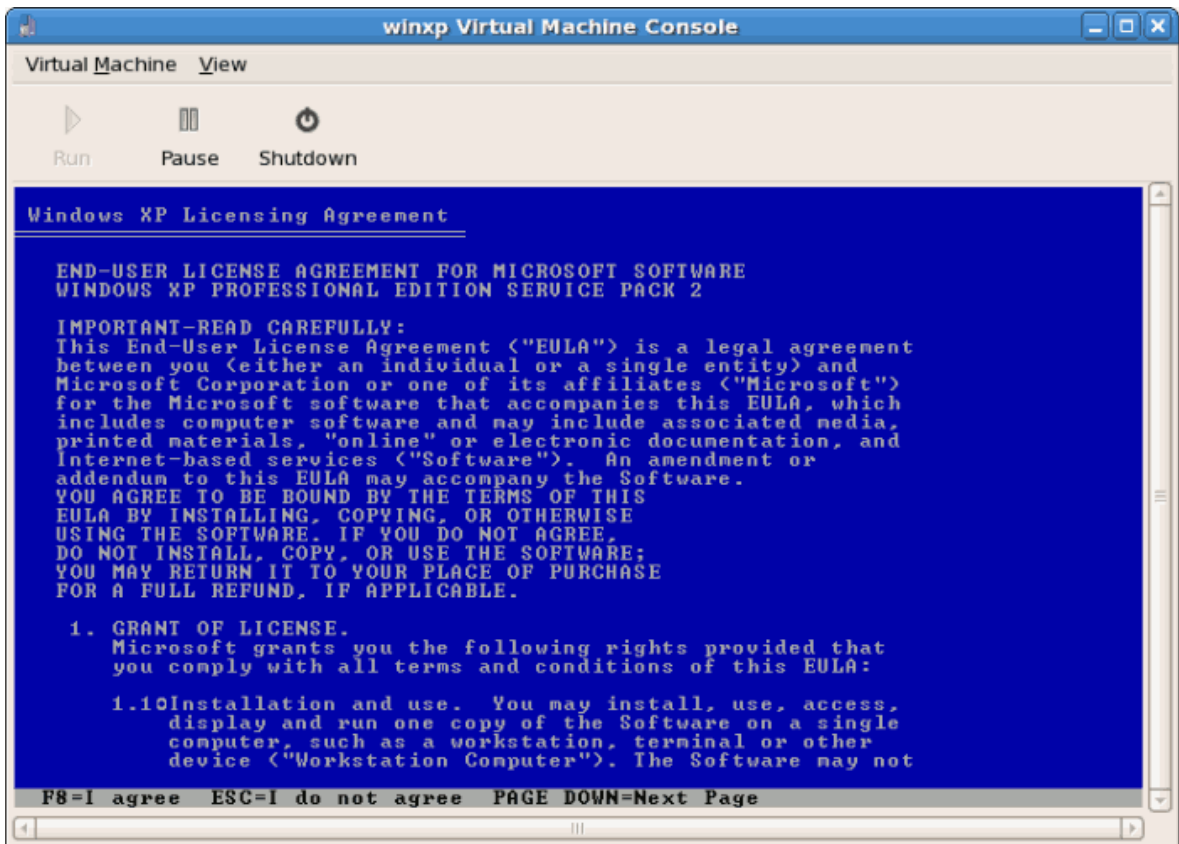
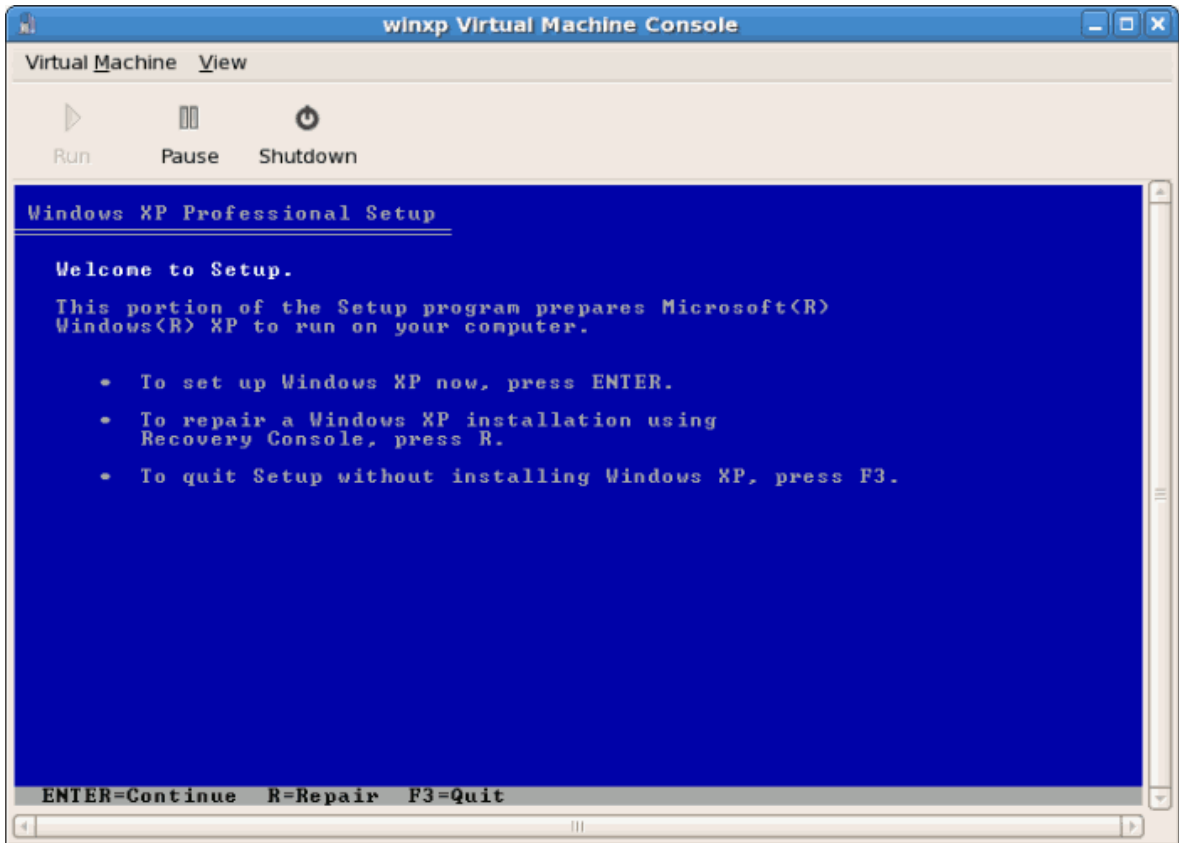
8. Bevor mit der Installation fortgefahren wird, sehen Sie eine Zusammenfassung auf dem Bildschirm. Klicken Sie auf **Fertigstellen**, um mit der Gastinstallation fortzufahren:



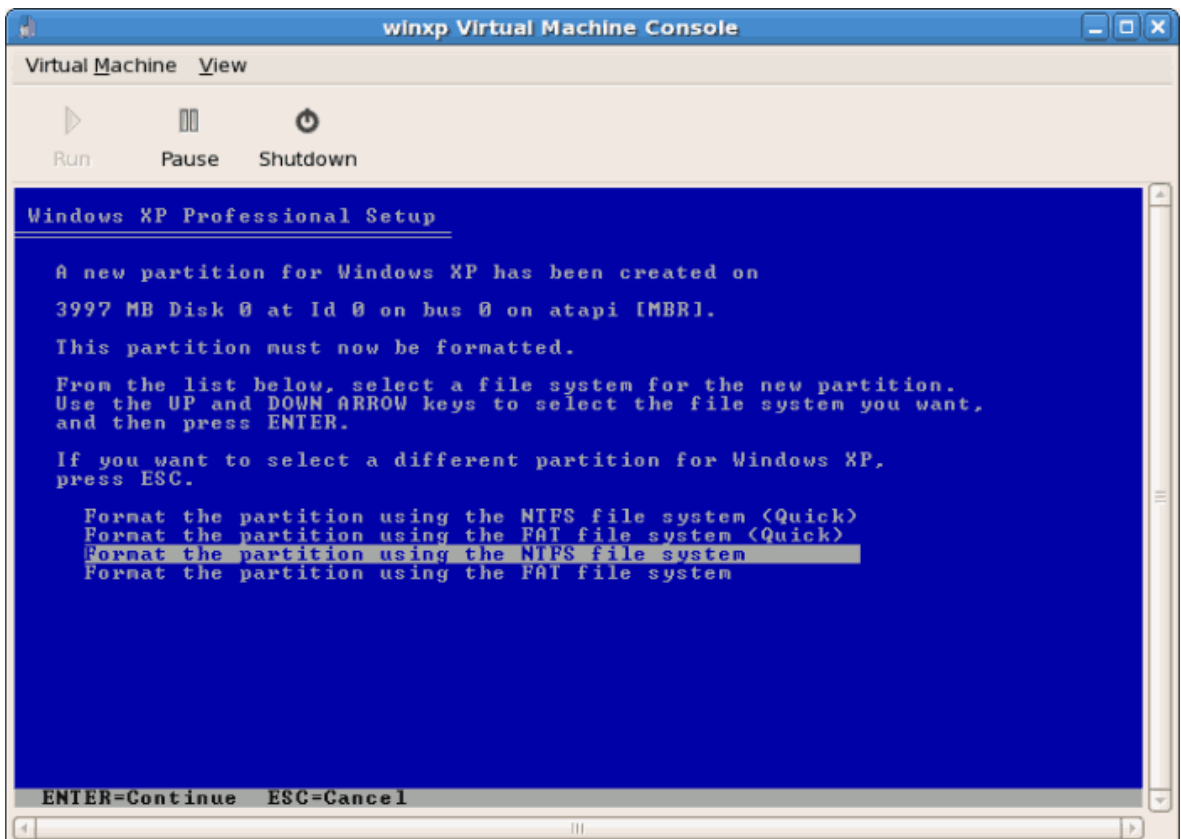
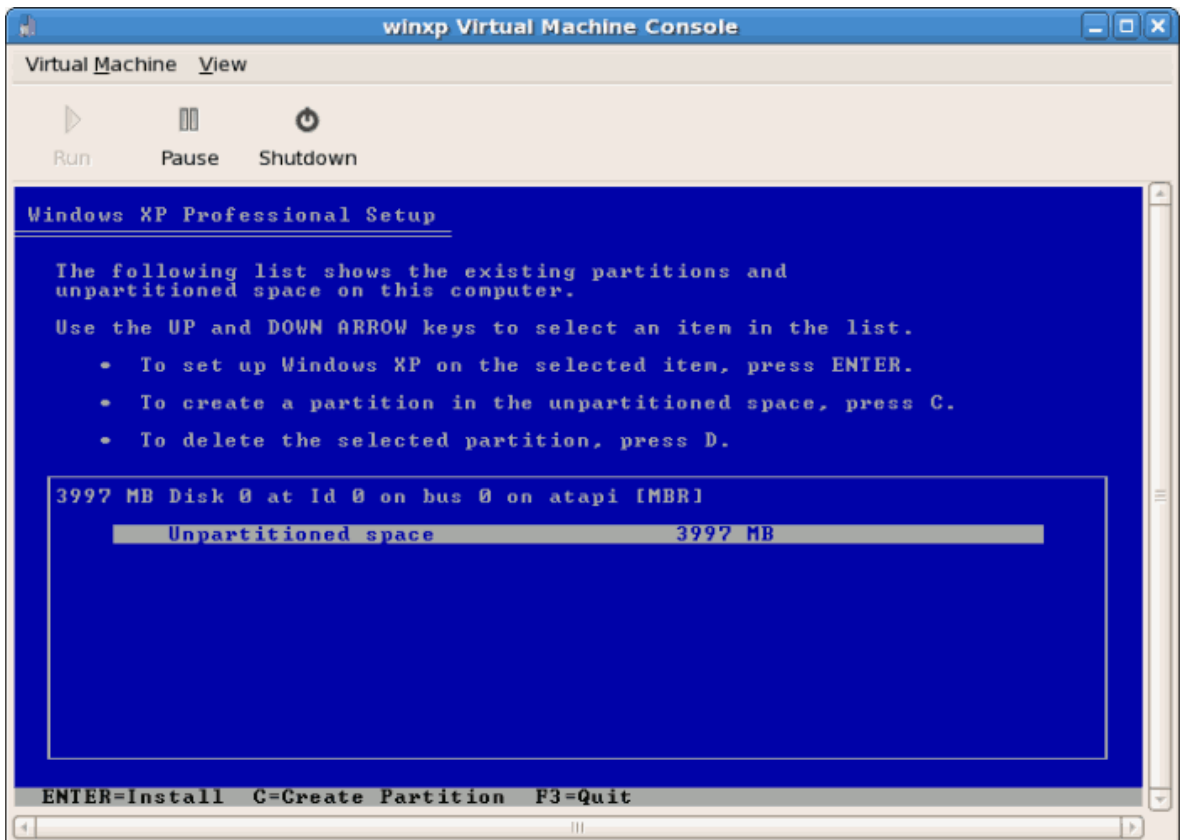
9. Da Sie eine Hardware-Auswahl treffen müssen, öffnen Sie sofort, nachdem die Installation gestartet wurde, ein Konsolenfenster. Klicken Sie auf **Fertigstellen**, wechseln dann zum Bildschirm mit der Zusammenfassung des **virt-manager** und wählen dort Ihren neu gestarteten Windows-Gast. Doppelklicken Sie auf den Systemnamen und das Konsolenfenster wird sich öffnen. Drücken Sie schnell mehrmals **F5**, um ein neues HAL auszuwählen, sobald Sie das Dialogfenster in der Windows-Installation bekommen, wählen Sie den 'Generische i486-Plattform'-Reiter aus (wechseln Sie die Auswahl mit den **Hoch** und **Runter**-Pfeilen).



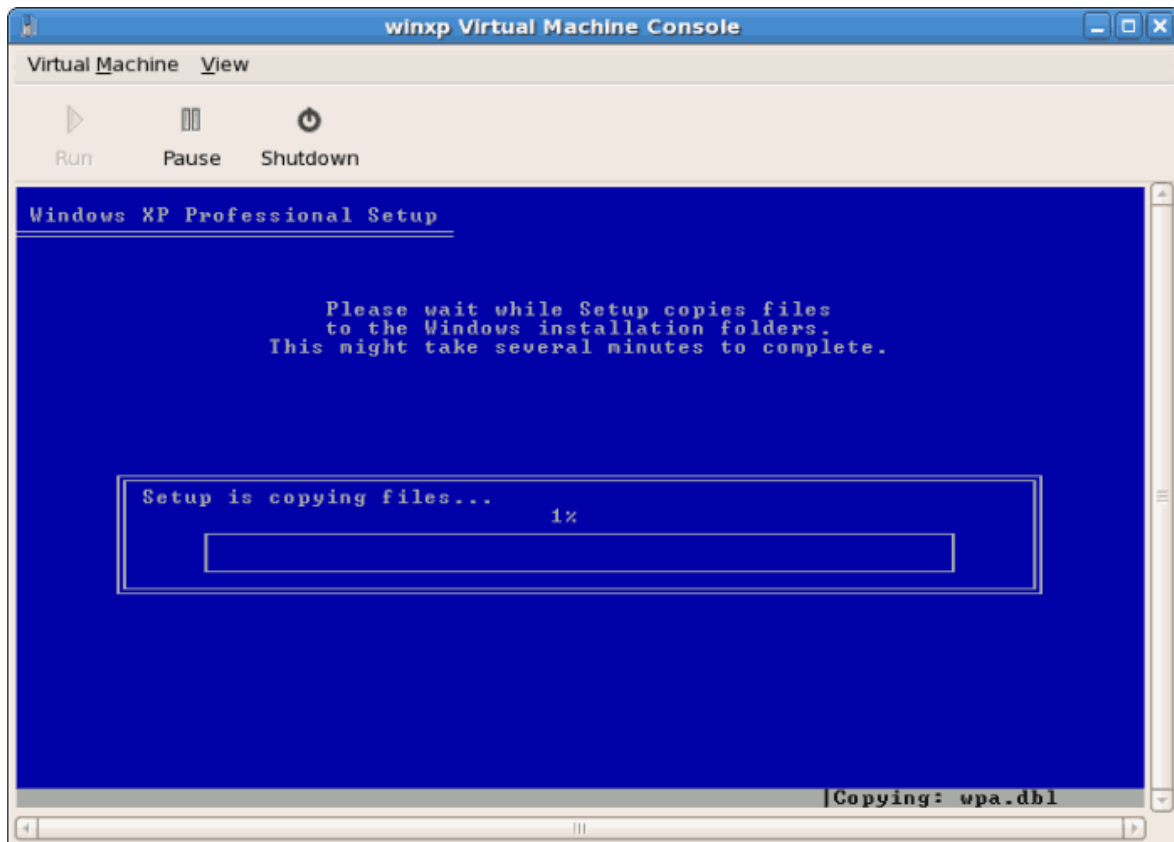
10. Die Installation fährt fort mit der standardmäßigen Windows Installation.

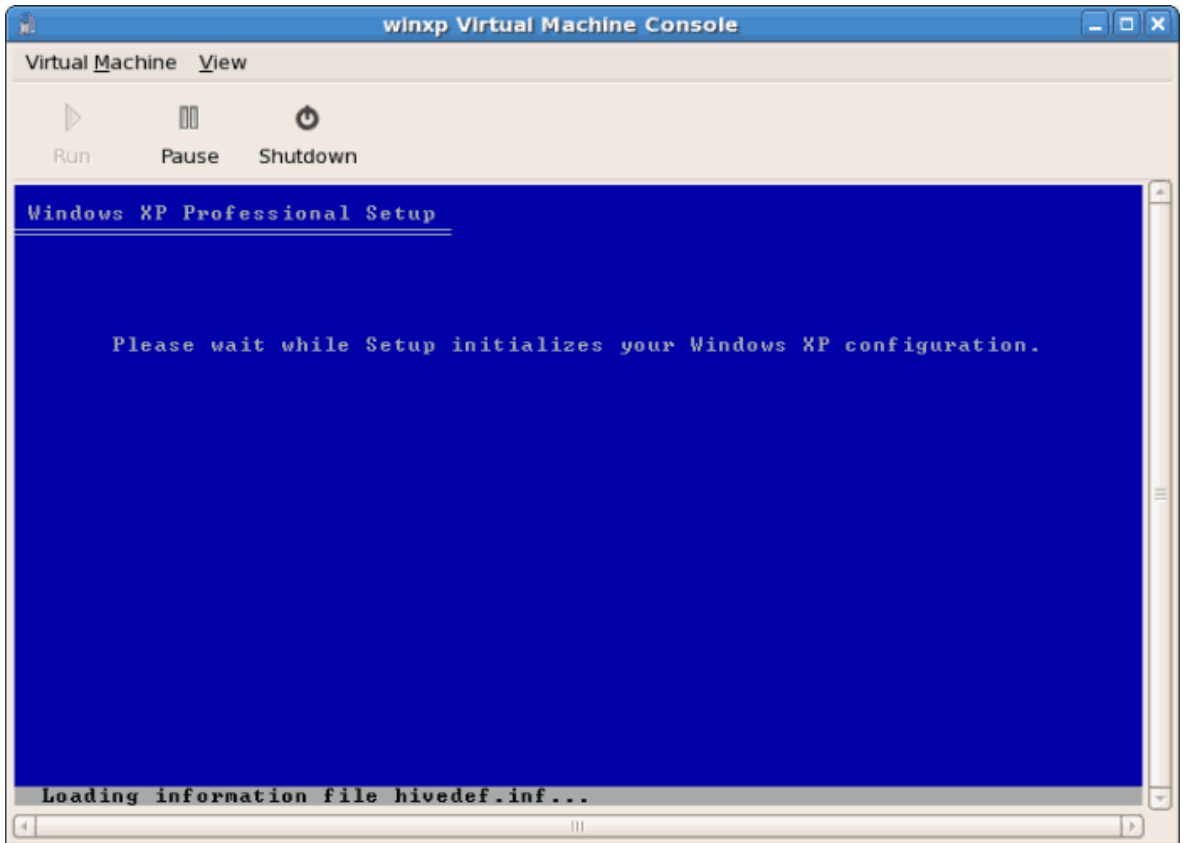


11. Partitionieren Sie die Festplatte, wenn Sie dazu aufgefordert werden.

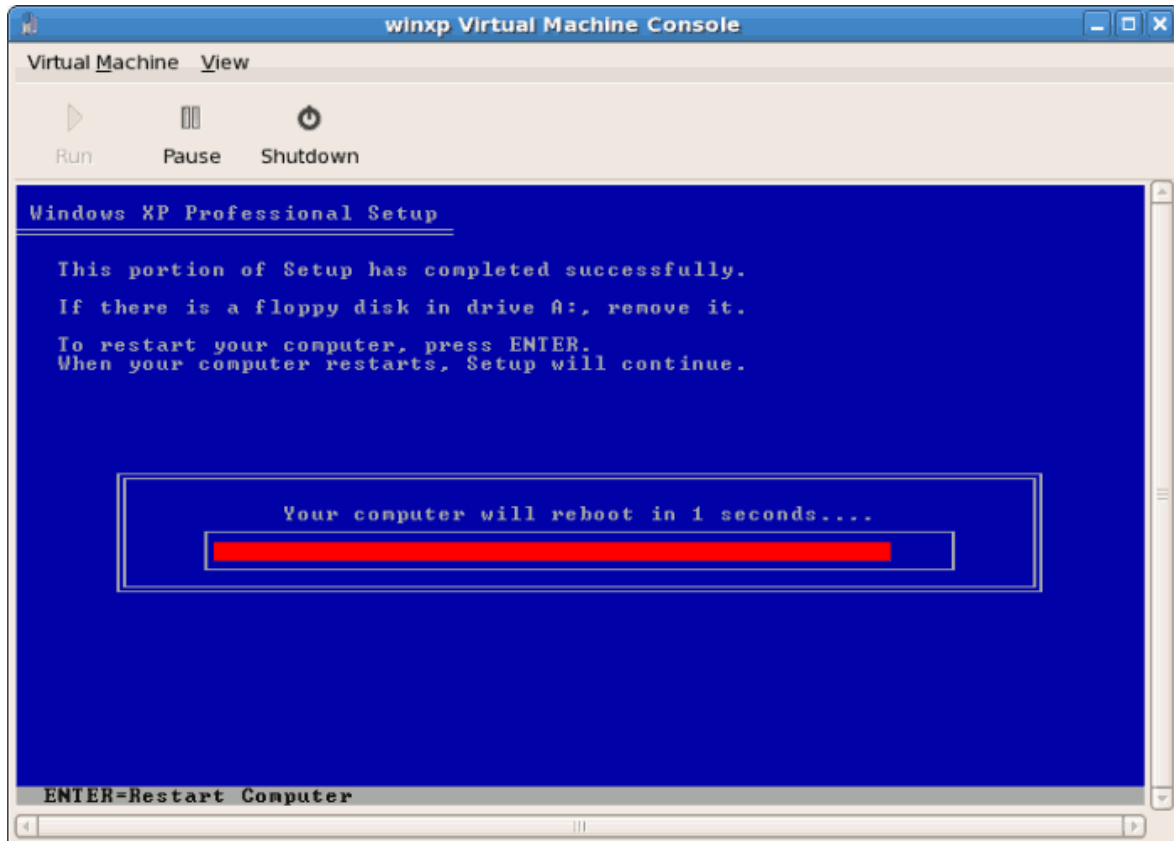


12. Nachdem die Festplatte formatiert wurde, beginnt Windows damit, die Dateien auf die Festplatte zu kopieren.





13. Die Dateien sind auf das Speichergerät kopiert, Windows startet nun neu.

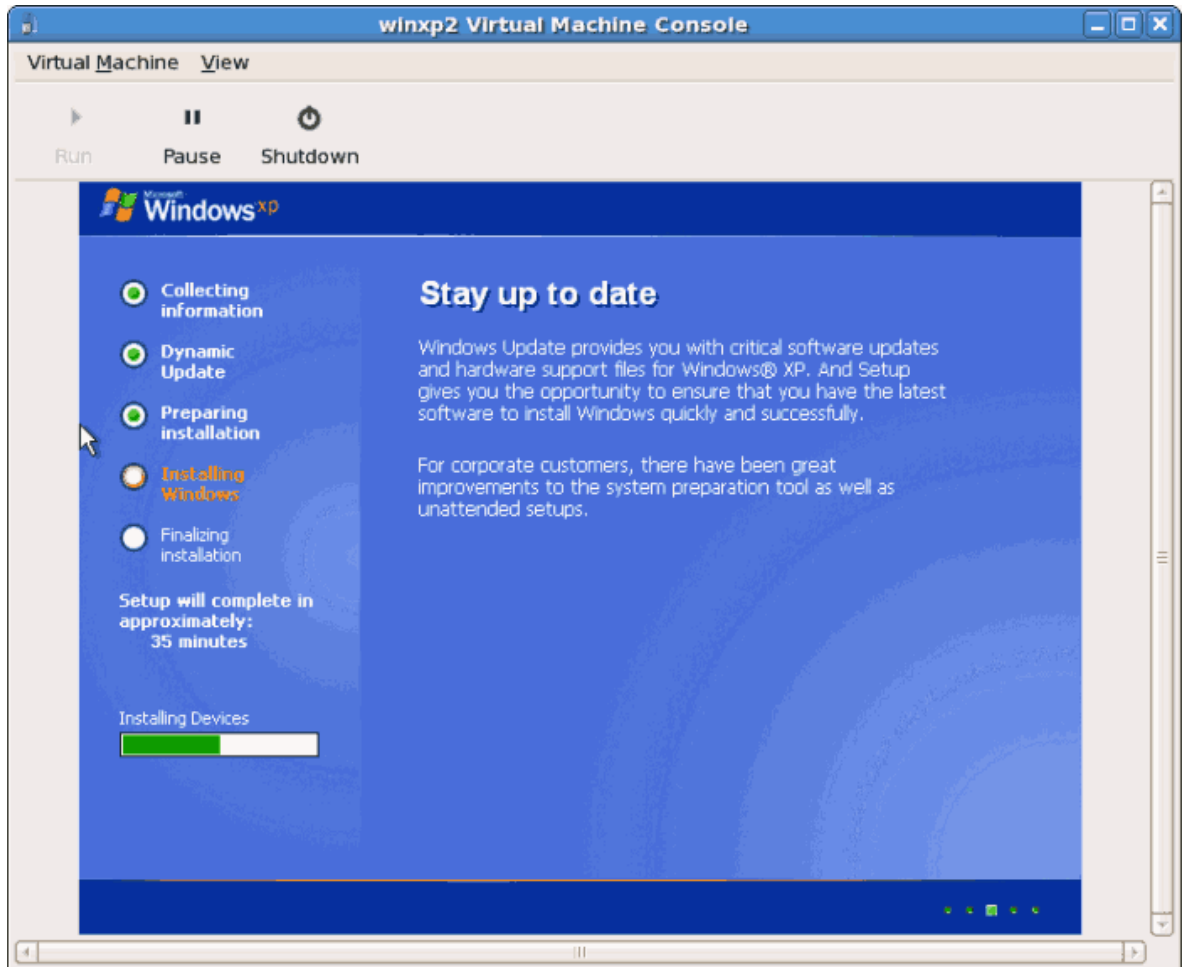


14. Starten Sie Ihren Windows-Gast neu:

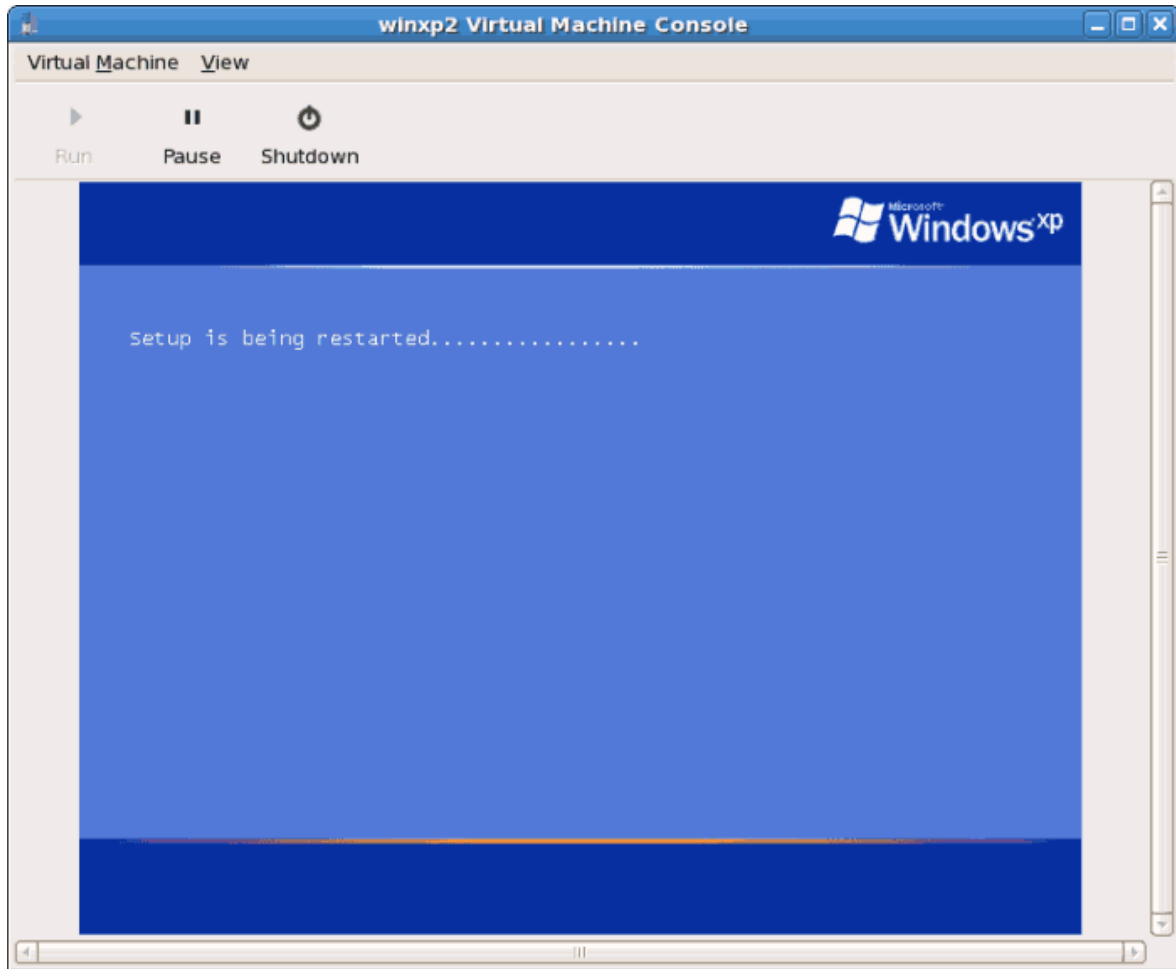
```
# virsh start WindowsGuest
```

Wobei *WindowsGuest* der Name Ihrer virtuellen Maschine ist.

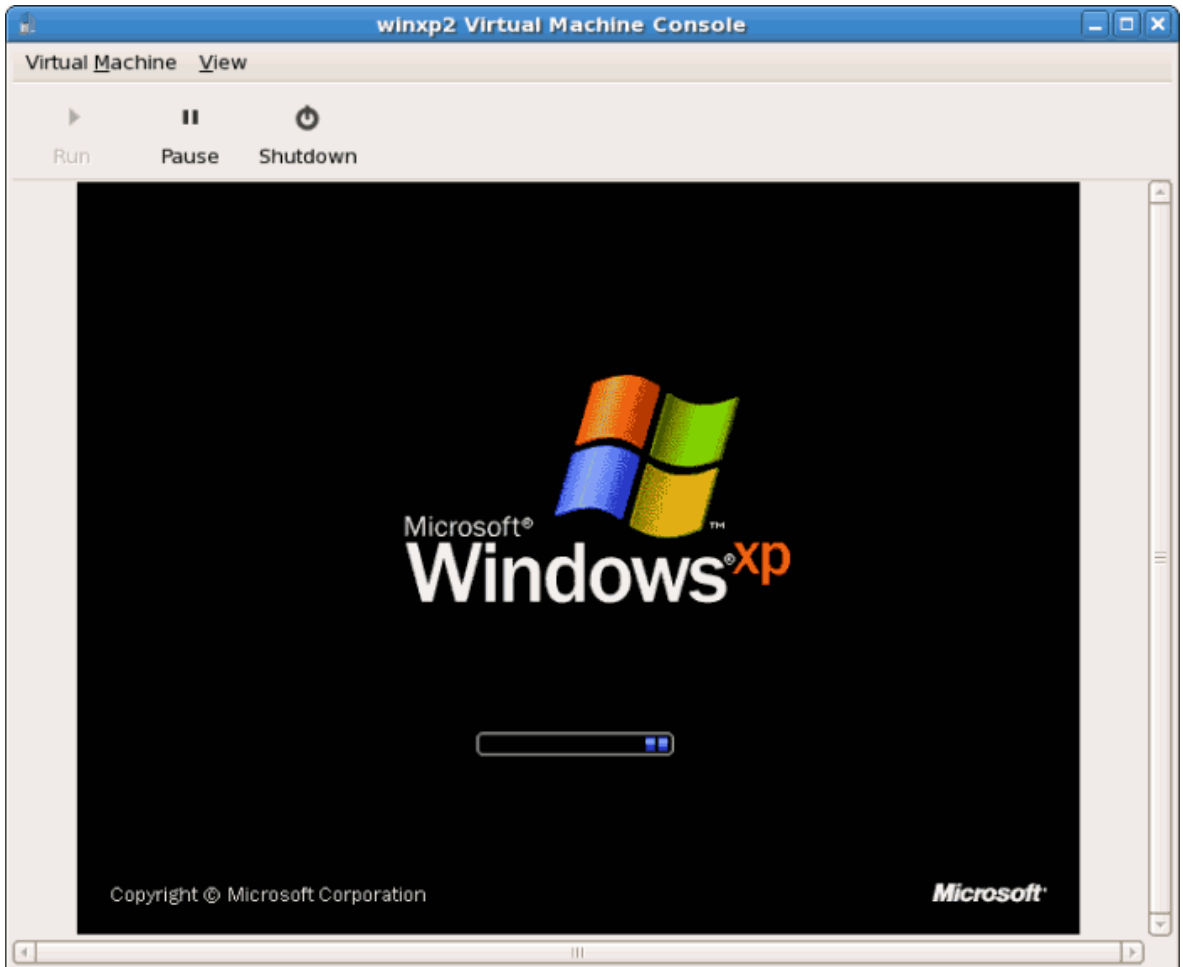
15. Wenn sich das Konsolenfenster öffnet, sehen Sie die Setup-Phase der Windows-Installation.



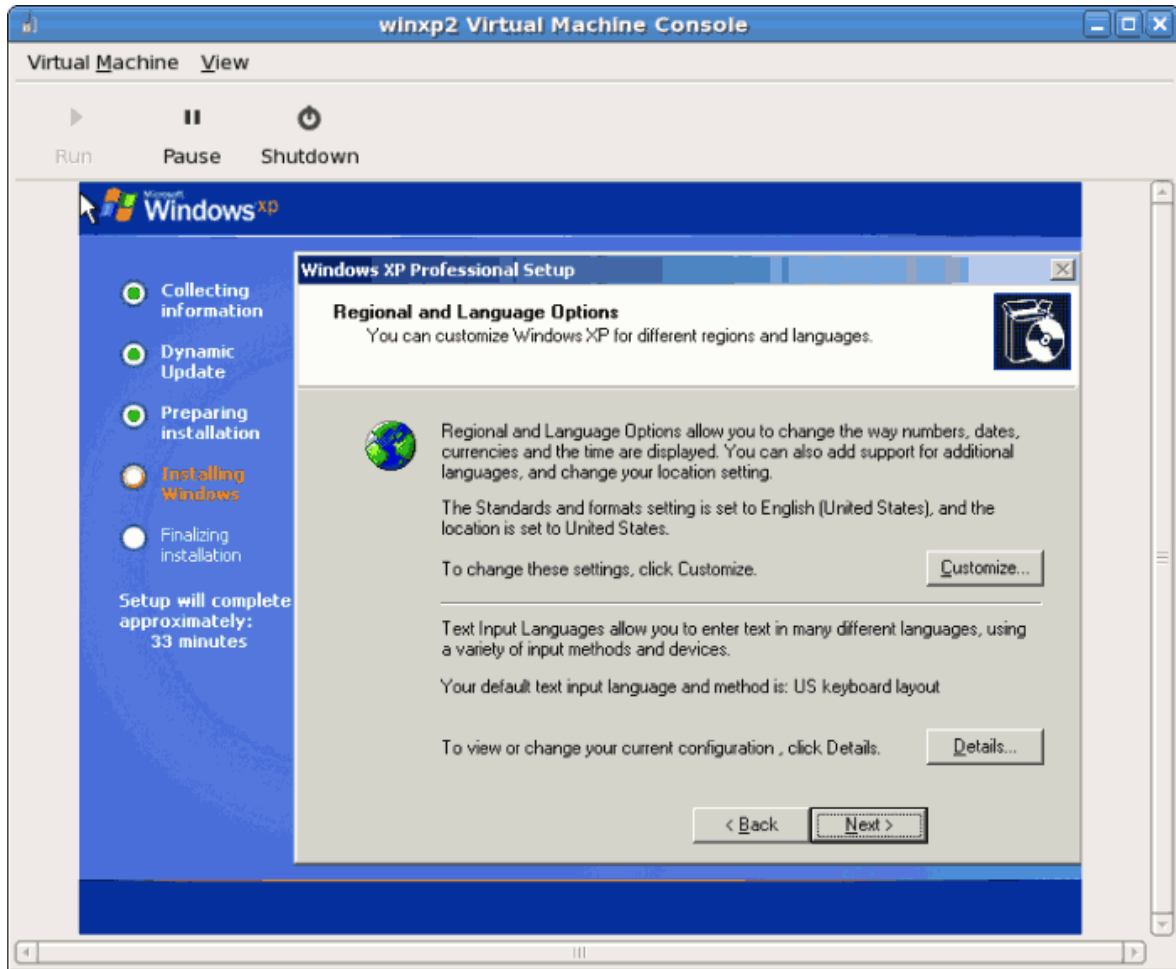
16. Falls Ihre Installation während der Setup-Phase zu hängen scheint, starten Sie den Gast mittels **virsh reboot *WindowsGuestName*** neu. Dies wird die Installation in der Regel wieder zum Laufen bringen. Wenn Sie Ihre virtuelle Maschine neu starten, werden Sie die Nachricht **Set up is being restarted** sehen:



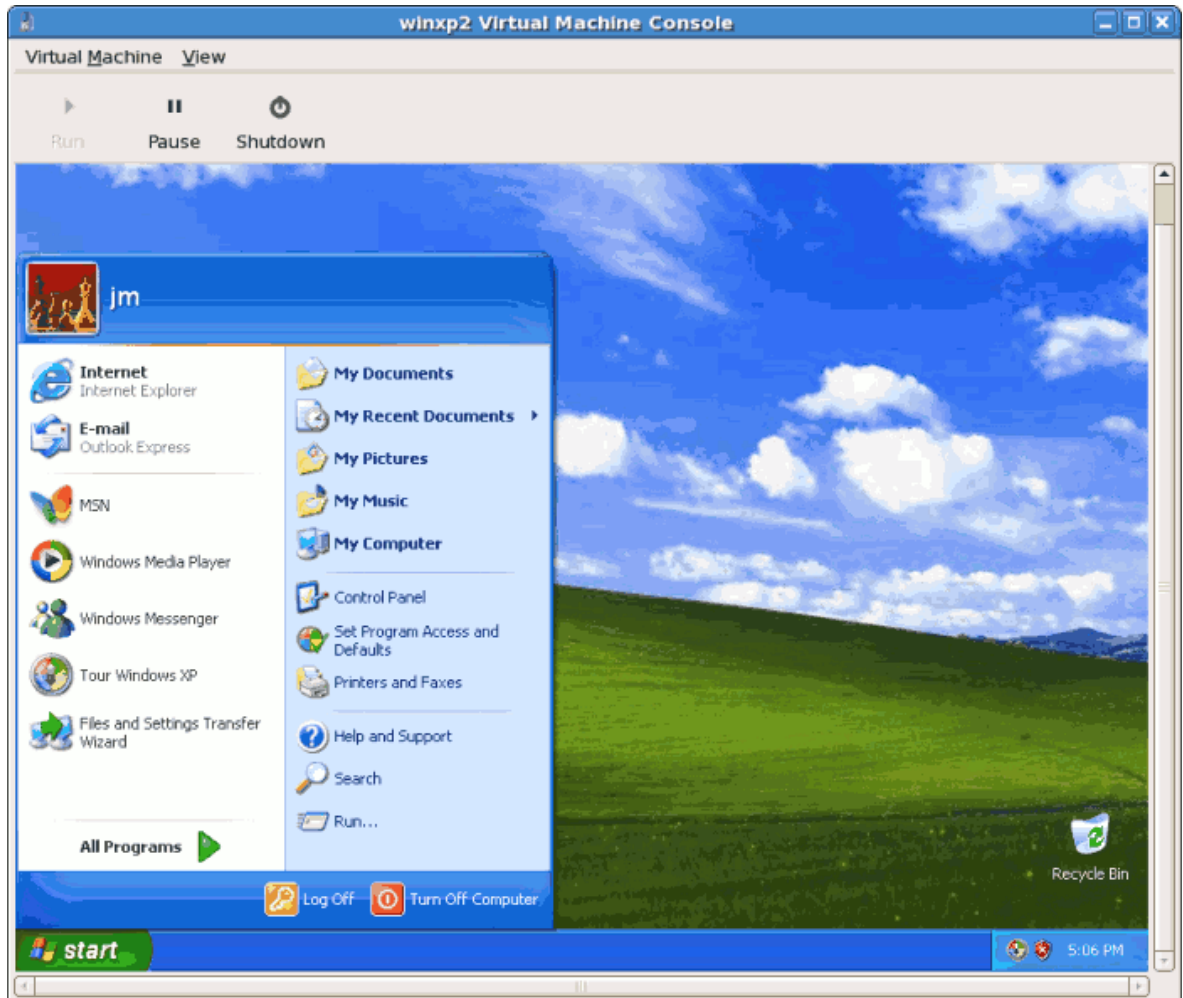
17. Nachdem das Setup abgeschlossen ist, sehen Sie den Windows-Boot-Bildschirm:



18. Sie können jetzt mit dem Standard-Setup für Ihre Windows-Installation fortfahren:



19. Der Setup-Prozess ist nun abgeschlossen, ein Windows-Bildschirm wird angezeigt.



3.4. Installation von Windows Server 2003 als voll virtualisierter Gast

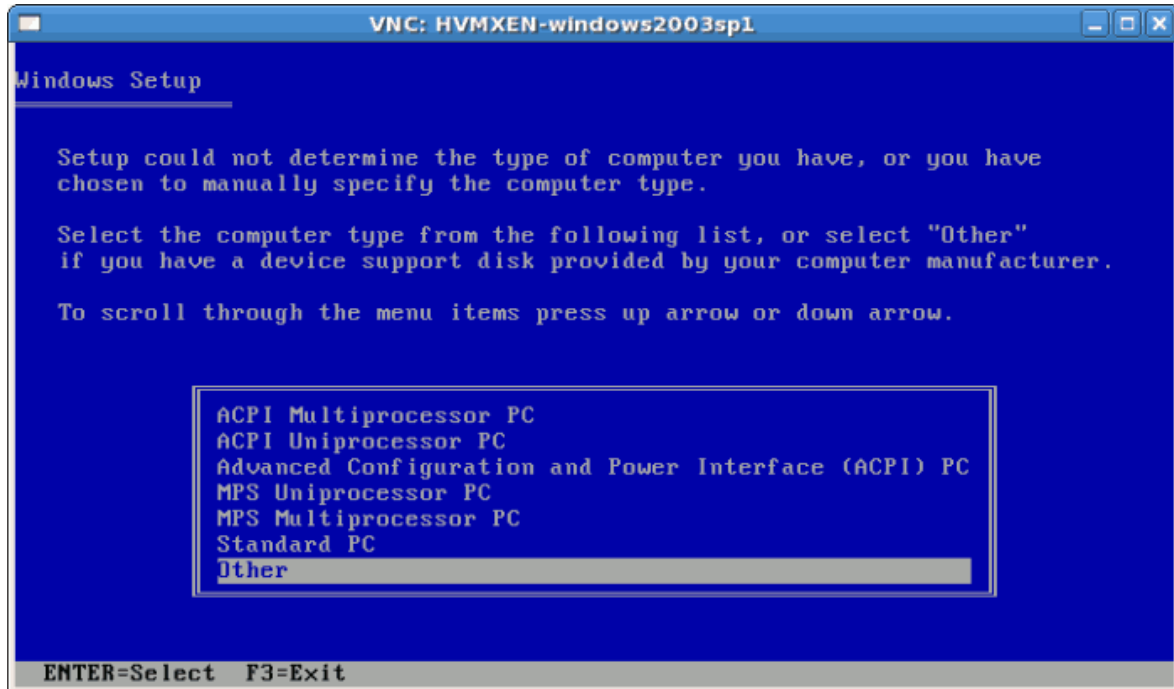
Dieses Kapitel behandelt die Installation eines Windows Server 2003 Gasts mit Hilfe des **virt-install**-Befehls. **virt-install** kann anstelle von **virt-manager** verwendet werden. Dieses Verfahren ähnelt der Windows XP Installation, die in [Abschnitt 3.3, „Installation von Windows XP als voll virtualisierter Gast“](#) beschrieben wird.

1. Durch Verwenden von **virt-install** zur Installation von Windows Server 2003 als Konsole für den Windows-Gast wird umgehend das virt-viewer-Fenster geöffnet. Sehen Sie hier ein Beispiel zur Verwendung von **virt-install** zur Installation eines Windows Server 2003 Gasts:

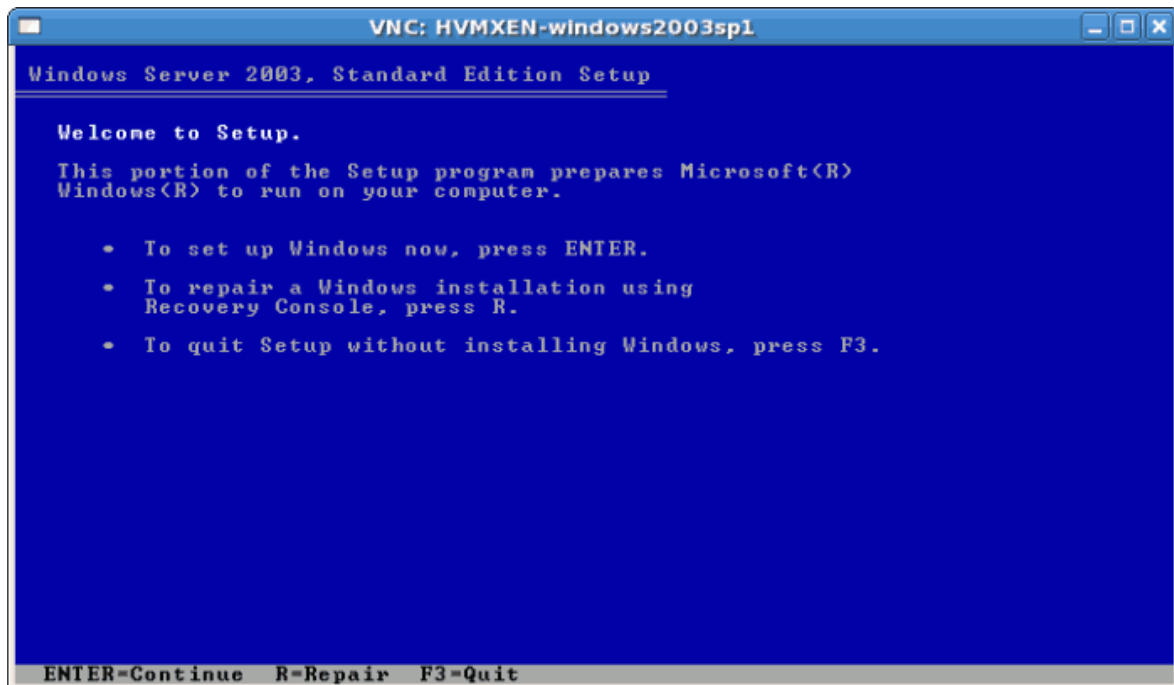
Starten Sie die Installation mit dem **virt-install**-Befehl.

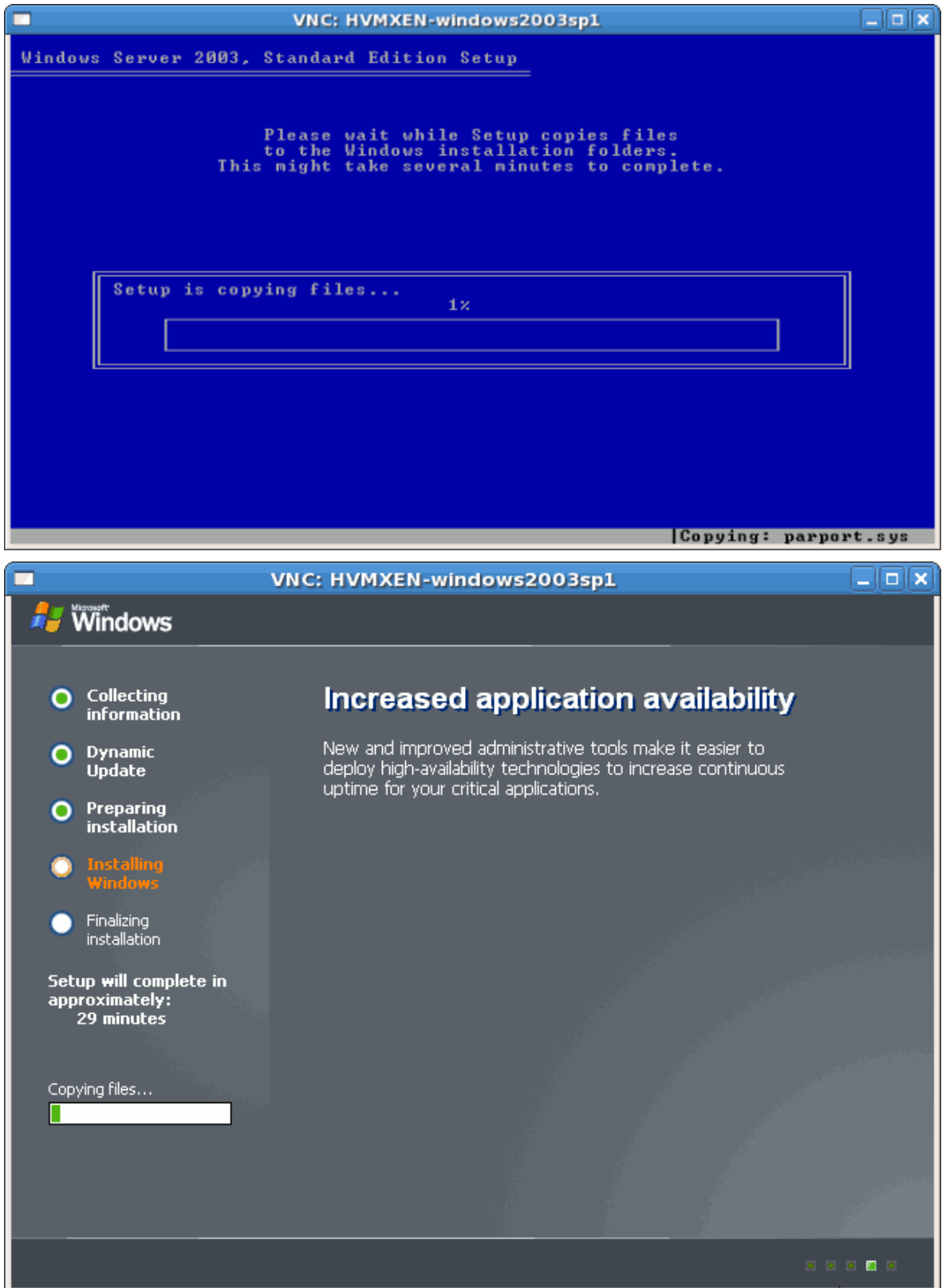
```
# virt-install -hvm -s 5 -f /var/lib/libvirt/images/windows2003sp1.dsk \
-n windows2003sp1 -cdrom=/ISOs/WIN/en_windows_server_2003_sp1.iso \
-vnc -r 1024
```

2. Sobald die Gastinstallation startet, müssen Sie schnell **F5** drücken. Wenn Sie nicht rechtzeitig **F5** drücken, müssen Sie die Installation neu beginnen. Durch Drücken von **F5** können Sie verschiedene **HAL** oder **Computer-Typen** auswählen. Wählen Sie Standard PC als Computer-Typ aus. Dies ist der einzige nötige Schritt, der nicht standardmäßig ist.



3. Führen Sie den Rest der Installation zu Ende.





4. Windows Server 2003 ist nun als voll virtualisierter Gast installiert.

3.5. Installation von Windows Server 2008 als voll virtualisierter Gast

Dieser Abschnitt beschreibt die Installation eines voll virtualisierten Windows Server 2008 Gasts.

Prozedur 3.4. Installation von Windows Server 2008 mit virt-manager

1. **Open virt-manager**

Start **virt-manager**. Launch the **Virtual Machine Manager** application from the **Applications** menu and **System Tools** submenu. Alternatively, run the **virt-manager** command as root.

2. **Select the hypervisor**

Select the hypervisor. If installed, select Xen or KVM. For this example, select KVM. Note that presently KVM is named qemu.

Nachdem die Option gewählt wurde, erscheint die Schaltfläche **Neu**. Klicken Sie auf **Neu**.

3. **Start the new virtual machine wizard**

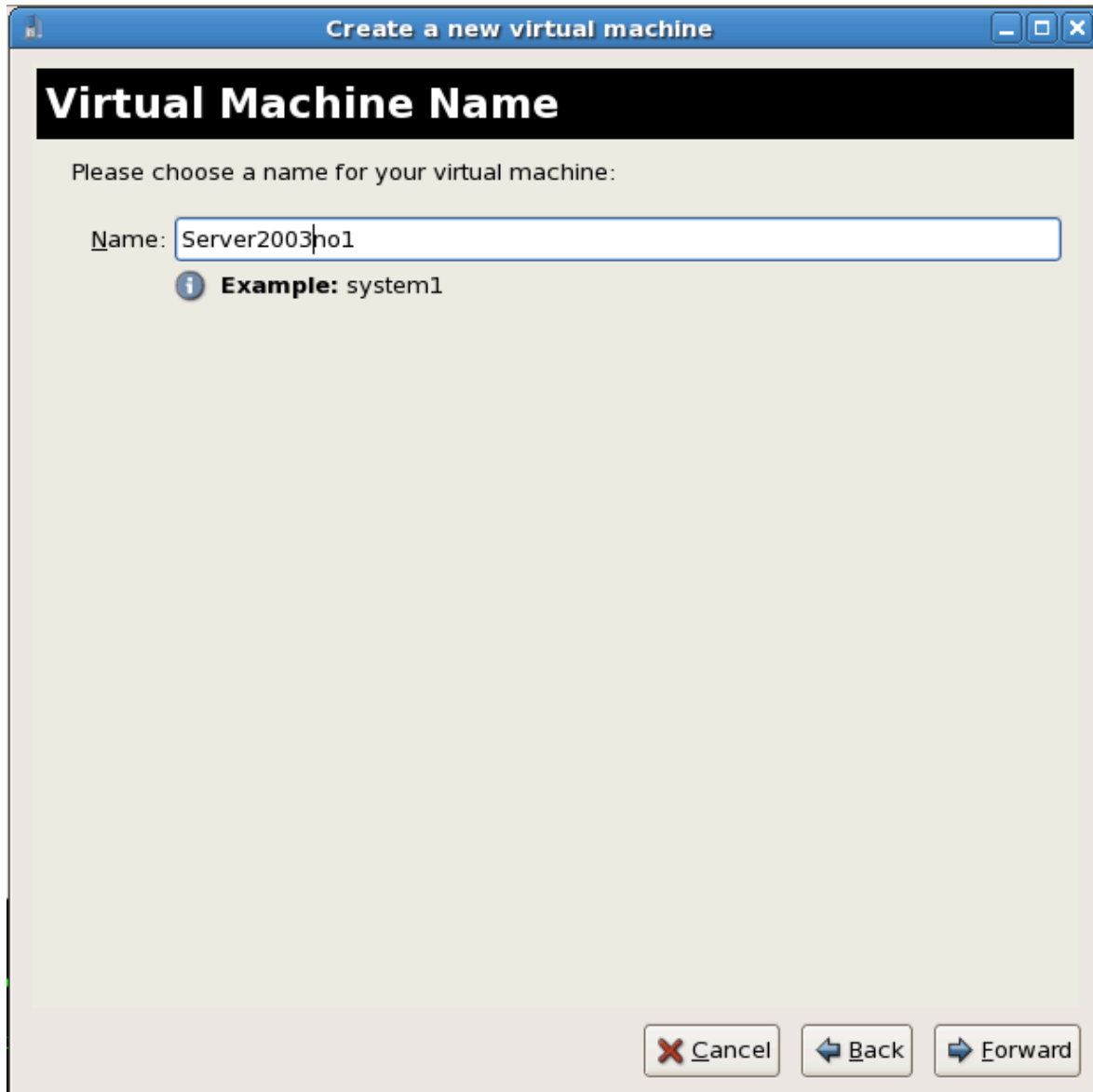
Pressing the **New** button starts the virtual machine creation wizard.



Press **Forward** to continue.

4. **Name the virtual machine**

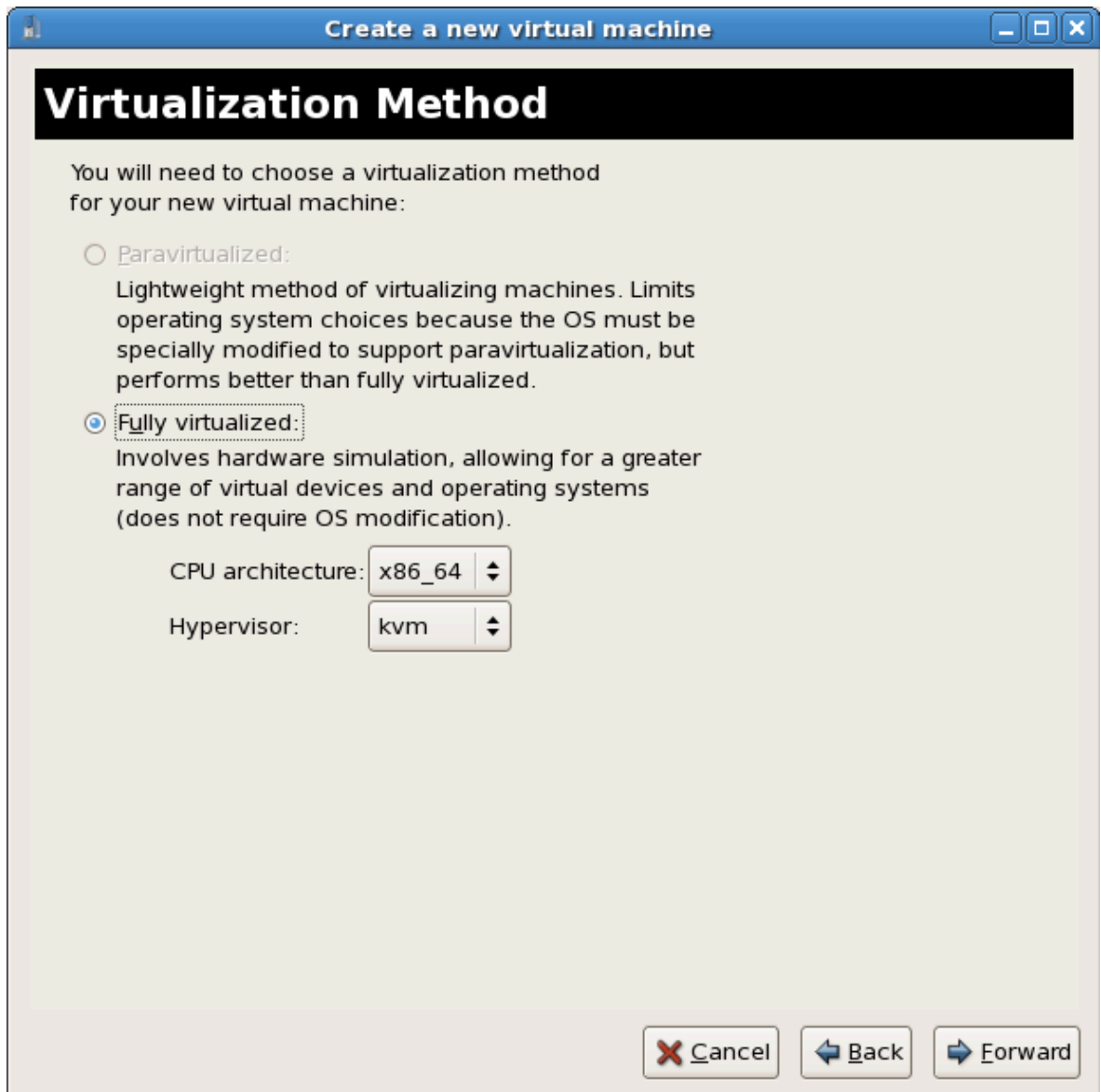
Geben Sie Ihrem virtualisierten Gast einen Namen. Satzzeichen und Leerstellen sind dabei nicht zulässig.



Klicken Sie auf **Weiter**, um fortzufahren.

5. **Choose a virtualization method**

Wählen Sie die Virtualisierungsmethode für den virtualisierten Gast. Beachten Sie, dass Sie nur eine installierte Virtualisierungsmethode auswählen können. Wenn Sie vorher KVM oder Xen gewählt haben (Schritt 2), müssen Sie den Hypervisor verwenden, den Sie ausgewählt haben. In diesem Beispiel wird der KVM-Hypervisor verwendet.



Klicken Sie auf **Weiter**, um fortzufahren.

6. **Select the installation method**

Für alle Windows-Versionen müssen Sie ein **Lokales Installationsmedium** verwenden, also entweder ein ISO-Abbild oder ein physisches, optisches Medium.

Falls Sie einen PXE-Server für Windows-Netzwerkinstallationen konfiguriert haben, kann auch PXE verwendet werden. PXE-Windows-Installation wird in diesem Handbuch jedoch nicht behandelt.

Setzen Sie den **Betriebssystemtyp** auf **Windows** und die **Betriebssystemvariante** auf **Microsoft Windows 2008**, wie im Screenshot dargestellt.

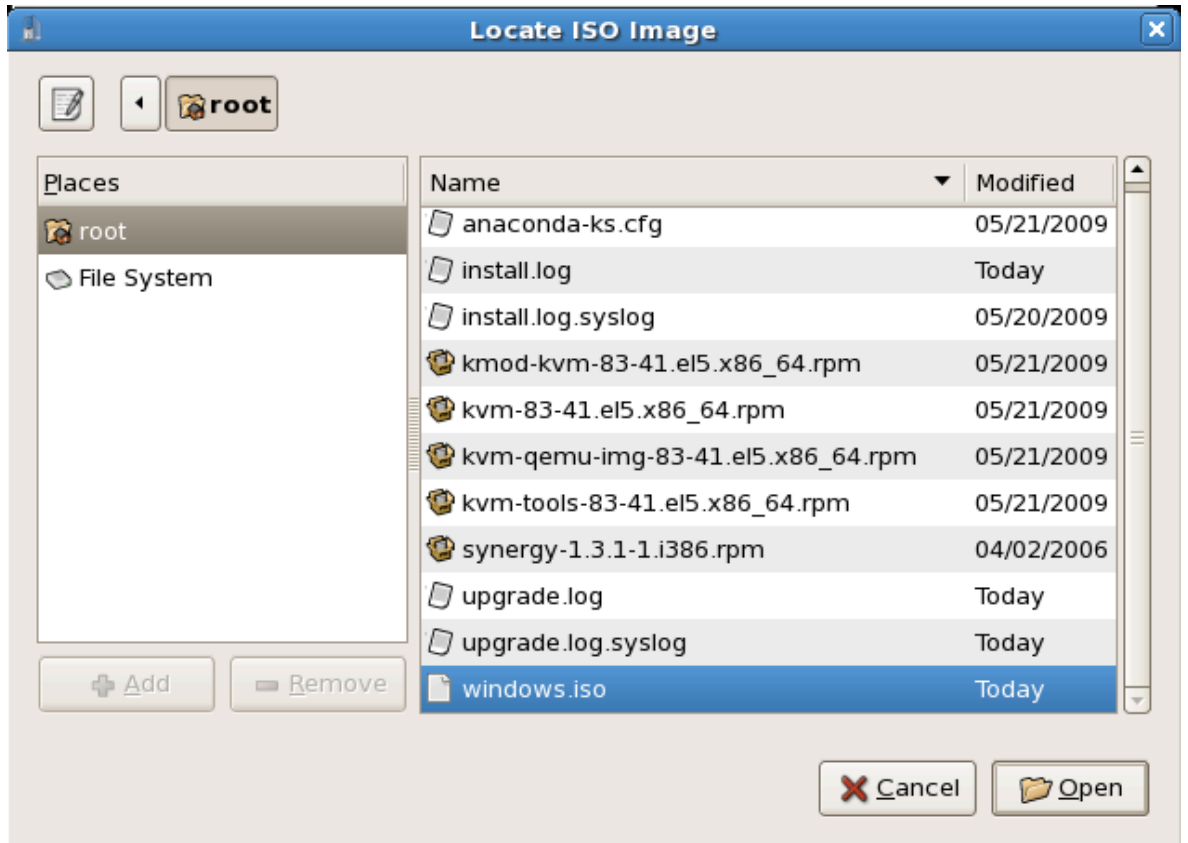


Klicken Sie auf **Weiter**, um fortzufahren.

7. **Locate installation media**

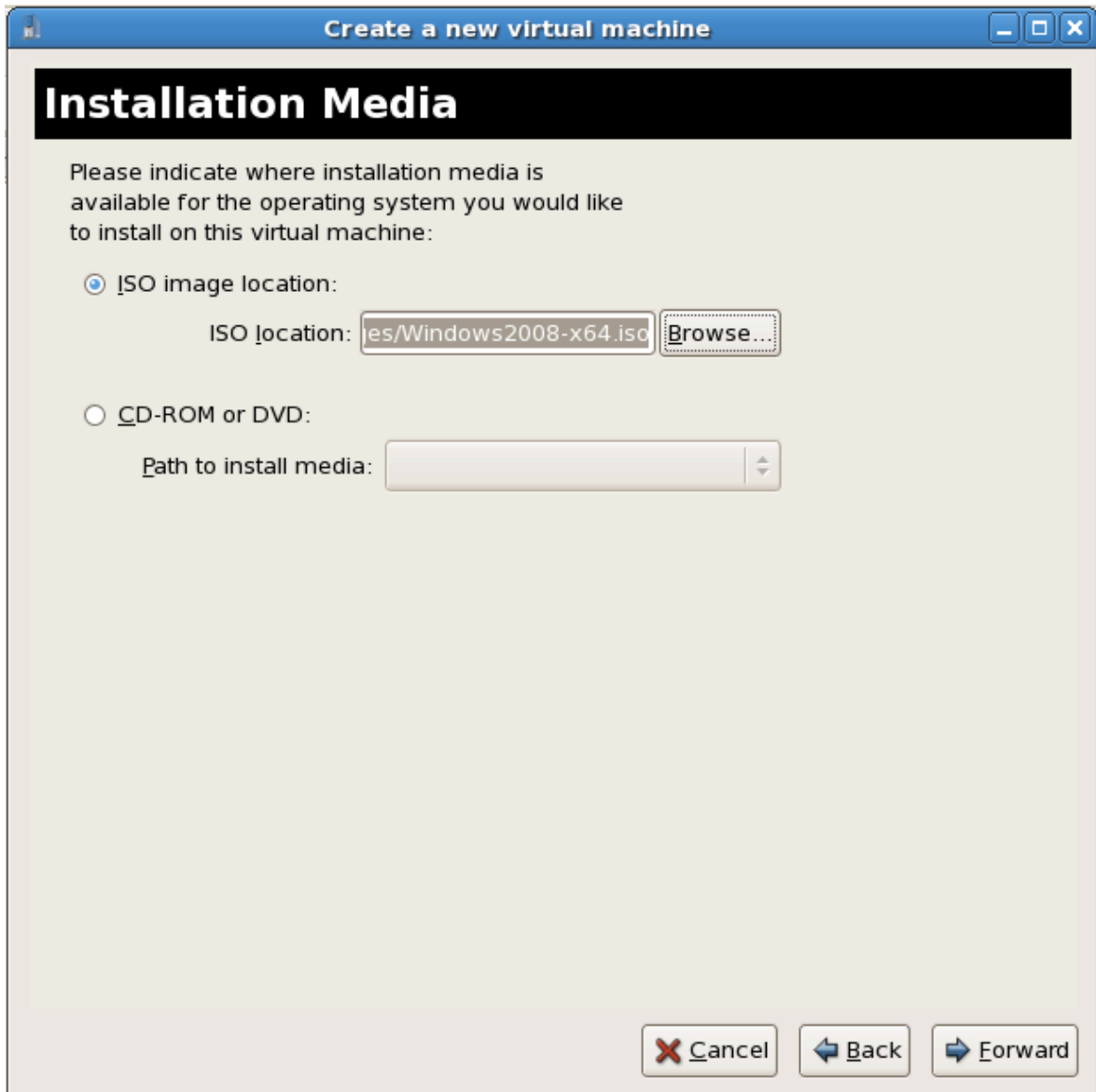
Wählen Sie "Speicherort des ISO-Abbilds" oder "CD-ROM oder DVD-Laufwerk". In diesem Beispiel wird ein ISO-Dateiabbild der Windows Server 2008 Installations-CD verwendet.

- a. Press the **Browse** button.
- b. Suchen Sie den Speicherort der ISO-Datei und wählen es aus.



Klicken Sie auf **Öffnen**, um Ihre Auswahl zu bestätigen.

- c. Die Datei ist ausgewählt und bereit zur Installation.



Klicken Sie auf **Weiter**, um fortzufahren.



Image files and SELinux

Für ISO-Abbilddateien und Gast Speicherabbilder sollte das `/var/lib/libvirt/images/`-Verzeichnis verwendet werden. Abweichende Speicherorte erfordern unter Umständen zusätzliche Konfiguration von SELinux. Siehe *Abschnitt 7.1, „SELinux und Virtualisierung“* für weitere Einzelheiten.

8. Storage setup

Weisen Sie ein physisches Speichergerät (**Blockgerät**) oder ein dateibasiertes Abbild (**Datei**) zu. Dateibasierte Abbilder müssen im `/var/lib/libvirt/images/`-Verzeichnis abgelegt sein. Weisen Sie ausreichend Speicherplatz für Ihren virtualisierten Gast und dessen benötigte Anwendungen zu.



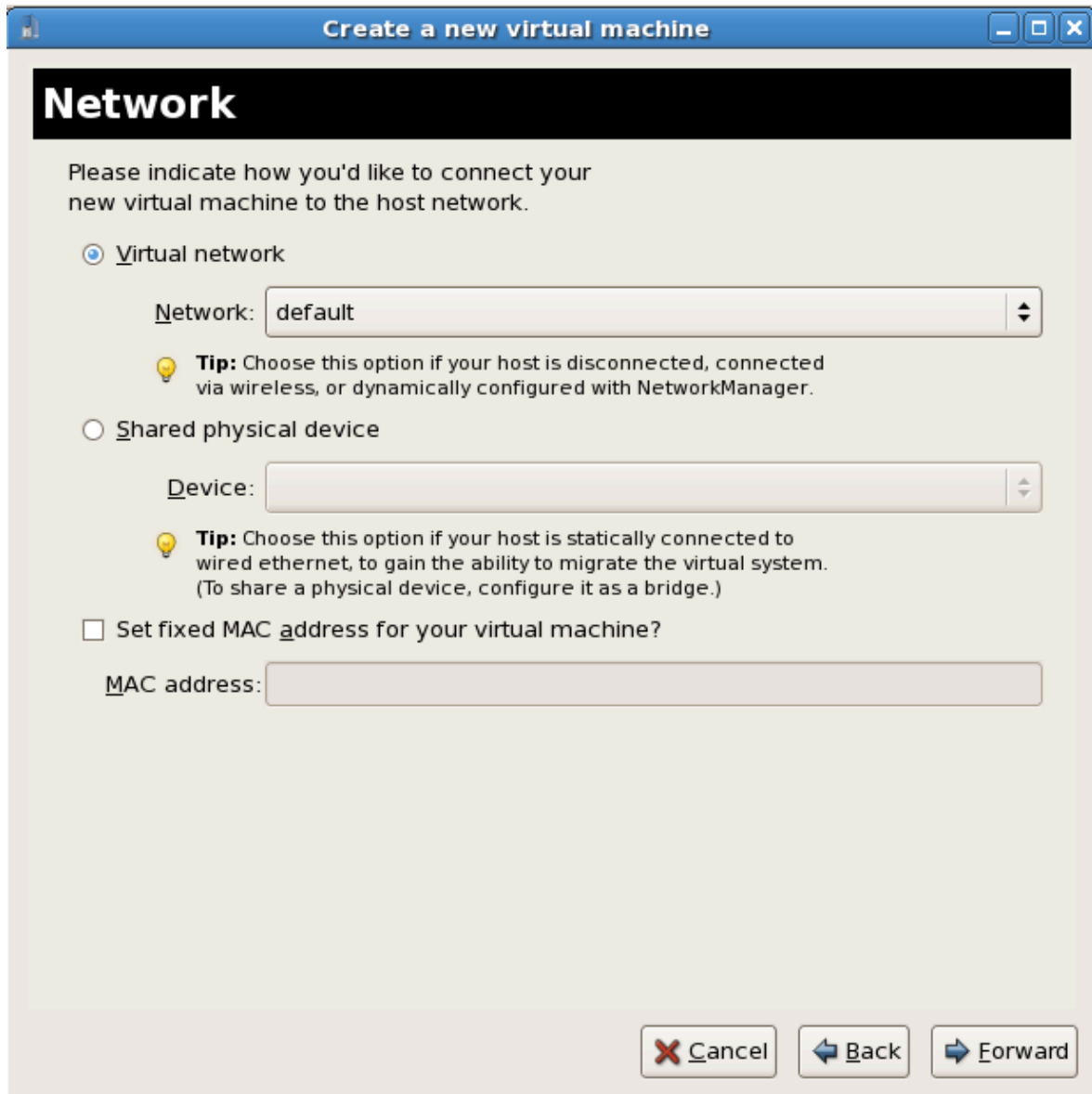
Klicken Sie auf **Weiter**, um fortzufahren.

9. Network setup

Select either **Virtual network** or **Shared physical device**.

The virtual network option uses Network Address Translation (NAT) to share the default network device with the virtualized guest. Use the virtual network option for wireless networks.

The shared physical device option uses a network bond to give the virtualized guest full access to a network device.



Press **Forward** to continue.

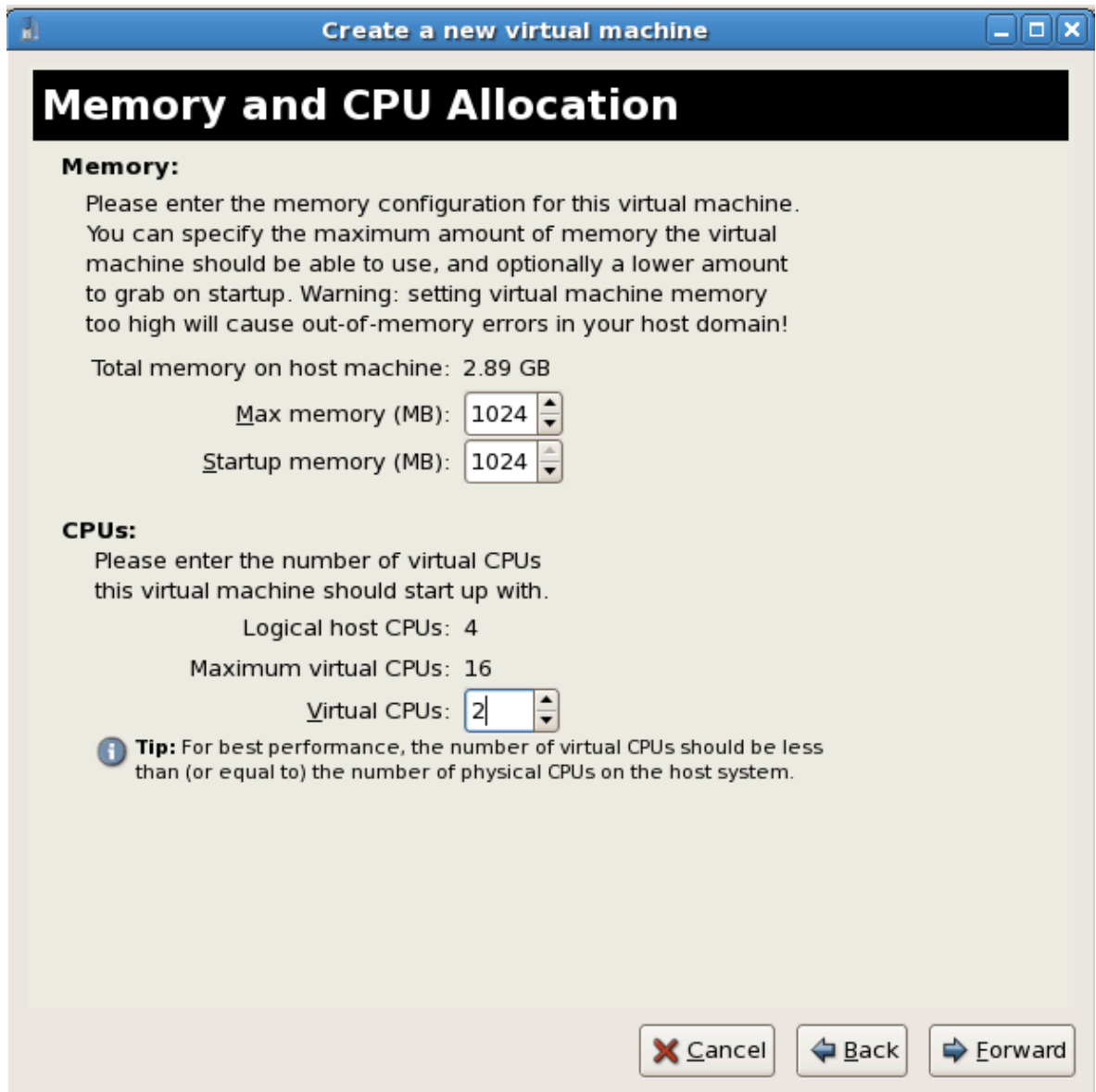
10. **Memory and CPU allocation**

The Allocate memory and CPU window displays. Choose appropriate values for the virtualized CPUs and RAM allocation. These values affect the host's and guest's performance.

Virtualized guests require sufficient physical memory (RAM) to run efficiently and effectively. Choose a memory value which suits your guest operating system and application requirements. Windows Server 2008. Remember, guests use physical RAM. Running too many guests or leaving insufficient memory for the host system results in significant usage of virtual memory and swapping. Virtual memory is significantly slower causing degraded system performance and responsiveness. Ensure to allocate sufficient memory for all guests and the host to operate effectively.

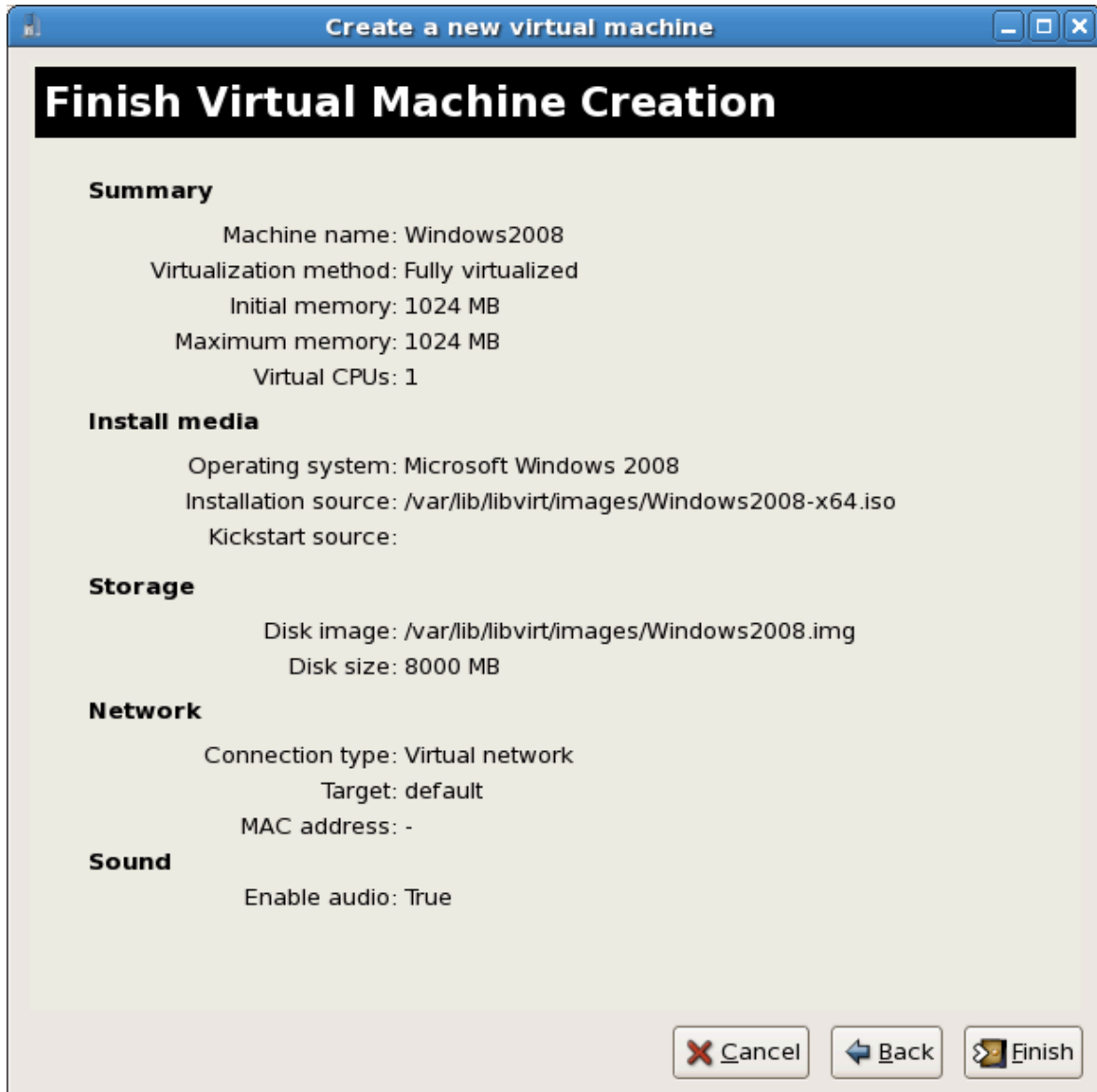
Assign sufficient virtual CPUs for the virtualized guest. If the guest runs a multithreaded application assign the number of virtualized CPUs it requires to run most efficiently. Do not assign more virtual CPUs than there are physical processors (or hyper-threads) available on

the host system. It is possible to over allocate virtual processors, however, over allocating has a significant, negative affect on guest and host performance due to processor context switching overheads.



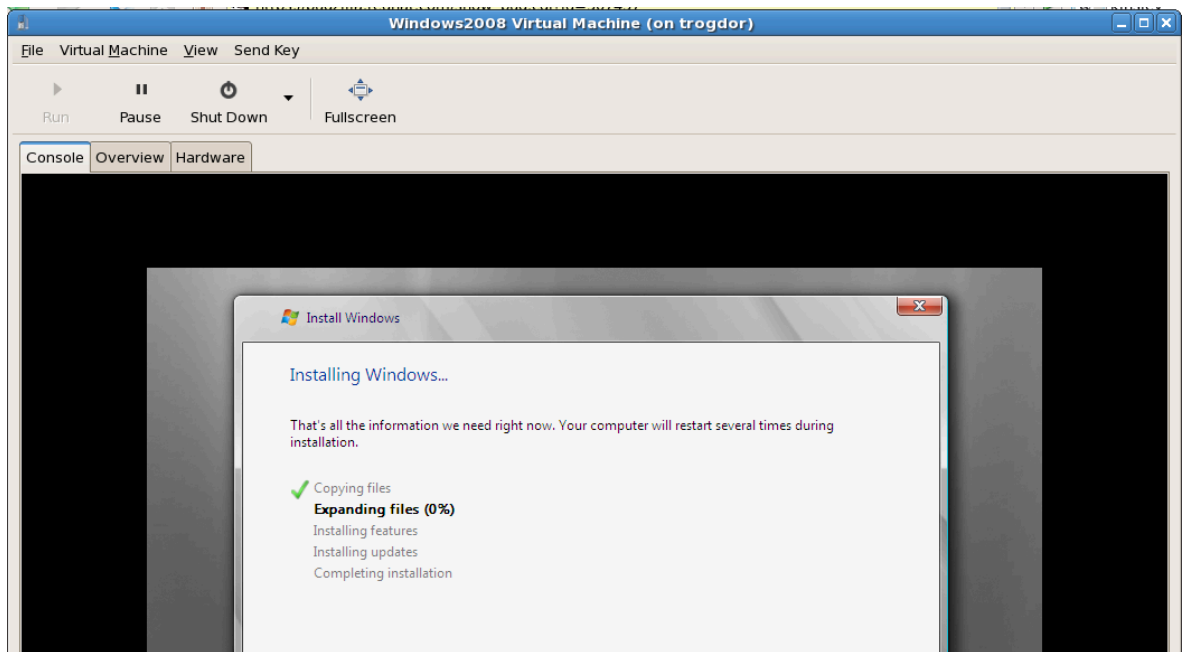
Press **Forward** to continue.

11. **Verify and start guest installation**
Überprüfen Sie die Konfiguration.



Klicken Sie auf **Fertigstellen**, um den Installationsvorgang zu starten.

12. Installation von Windows



Stellen Sie die Windows Server 2008 Installationssequenz fertig. Die Installationssequenz wird nicht in diesem Handbuch behandelt, Sie finden jedoch Informationen zur Installation von Windows in der [Dokumentation](#)¹ von Microsoft.

Teil II. Configuration

Konfigurieren der Virtualisierung in Fedora

Dieses Kapitel beschreibt die Konfigurationsverfahren für verschiedene fortgeschrittene Virtualisierungsaufgaben. Zu diesen Aufgaben gehören das Hinzufügen von Netzwerk- und Speichergeräten, das Verbessern der Sicherheit, das Steigern der Leistung sowie das Verwenden der paravirtualisierten Treibern auf voll virtualisierten Gästen.

Virtualisierte Blockgeräte

Dieses Kapitel erläutert die Installation und Konfiguration von Blockgeräten in virtualisierten Gästen. Der Begriff Blockgerät kann sich dabei auf unterschiedliche Arten von Speichergeräten beziehen.

4.1. Anlegen eines virtualisierten Floppy-Disk-Controllers

Floppy-Disk-Controller sind nötig für eine Reihe von älteren Betriebssystemen, insbesondere zur Installation von Treibern. Derzeit können virtualisierte Gäste nicht auf physische Floppy-Laufwerke zugreifen. Allerdings wird die Erstellung und der Zugriff auf Floppy-Diskettenabbilder durch virtualisierte Floppy-Laufwerke unterstützt. Dieser Abschnitt befasst sich mit dem Anlegen eines virtualisierten Floppy-Laufwerks.

Zunächst ist eine Abbilddatei einer Floppy-Diskette erforderlich. Floppy-Disketten-Abbilddateien können mit dem Befehl **dd** erstellt werden. Ersetzen Sie dabei `/dev/fd0` durch den Namen eines Floppy-Laufwerks und benennen die Diskette entsprechend.

```
# dd if=/dev/fd0 of=~/.legacydrivers.img
```



Anmerkung für paravirtualisierte Treiber

Die paravirtualisierten Treiber können voll virtualisierten Gästen physische Floppy-Geräte zuweisen.

Dieses Beispiel verwendet ein Gastsystem, das mittels **virt-manager** angelegt wurde und auf dem eine vollvirtualisierte Linux-Installation läuft, mit einem Abbild in `/var/lib/libvirt/images/rhel5FV.img`. In diesem Beispiel wird der Xen-Hypervisor verwendet.

1. Erstellen Sie die XML-Konfigurationsdatei für Ihr Gastabbild, indem Sie den Befehl **virsh** auf einem laufenden Gast ausführen.

```
# virsh dumpxml rhel5FV > rhel5FV.xml
```

Dies speichert die Konfigurationseinstellungen als XML-Datei, welche anschließend auf die vom Gast verwendeten Operationen und Geräte angepasst werden kann. Weitere Informationen zur Verwendung der virsh-XML-Konfigurationsdateien finden Sie unter [Kapitel 18, Erstellung angepasster libvirt-Skripte](#).

2. Erstellen Sie ein Floppy-Diskettenabbild für den Gast.

```
# dd if=/dev/zero of=/var/lib/libvirt/images/rhel5FV-floppy.img bs=512 count=2880
```

3. Fügen Sie den unteren Inhalt (mitsamt evtl. nötiger Änderungen) in die XML-Konfigurationsdatei Ihres Gasts ein. Dieses Beispiel erzeugt einen Gast mit einem Floppy-Gerät als dateibasiertes virtuelles Gerät.

```
<disk type='file' device='floppy'>
  <source file='/var/lib/libvirt/images/rhel5FV-floppy.img' />
```

```
<target dev='fda' />
</disk>
```

4. Stoppen Sie den Gast.

```
# virsh stop rhe15FV
```

5. Starten Sie den Gast neu unter Verwendung der XML-Konfigurationsdatei.

```
# virsh create rhe15FV.xml
```

Das Floppy-Laufwerk ist nun für den Gast erreichbar und als Abbilddatei auf dem Host gespeichert.

4.2. Hinzufügen von Speichergeräten zum Gast

Dieser Abschnitt erläutert das Hinzufügen von Speichergeräten zu virtualisierten Gästen. Zusätzlicher Speicher kann erst hinzugefügt werden, nachdem die Gäste angelegt wurden. Unterstützte Speichergeräte und -protokolle sind u. a.:

- Lokale Festplattenpartitionen
- Logische Datenträger
- Direkt mit dem Host verbundene Fibre Channel oder iSCSI
- Datei-Container in einem Dateisystem auf dem Host
- Direkt in der virtuellen Maschine eingehängte **NFS**-Dateisysteme
- Für den Gast direkt zugänglicher iSCSI-Speicher
- Cluster-Dateisysteme (**GFS**)

Hinzufügen von dateibasiertem Speicher zum Gast.

Dateibasierter Speicher oder dateibasierte Container sind Dateien auf dem Host-Dateisystem, die als virtualisierte Festplatten für virtualisierte Gäste fungieren. Um einen dateibasierten Container hinzuzufügen, sind die folgenden Schritte nötig:

1. Erzeugen Sie eine leere Container-Datei oder verwenden Sie eine bereits existierende Container-Datei (wie z. B. eine ISO-Datei).
 - a. Erzeugen Sie mit Hilfe des **dd**-Befehls eine Sparse-Datei. Beachten Sie bitte, dass die Verwendung von Sparse-Dateien aufgrund von Integritäts- und Performance-Problemen nicht empfohlen wird. Sparse-Dateien sind schnell erzeugt und können zum Testen verwendet werden, sollten jedoch nicht im Produktionsumfeld eingesetzt werden.

```
# dd if=/dev/zero of=/var/lib/libvirt/images/FileName.img bs=1M
seek=4096 count=0
```

- b. Vorab zugewiesene Nicht-Sparse-Dateien werden empfohlen für dateibasierte Speicher-Container. Führen Sie folgenden Befehl aus, um eine Nicht-Sparse-Datei zu erzeugen:

```
# dd if=/dev/zero of=/var/lib/libvirt/images/FileName.img bs=1M
count=4096
```

Beide Befehle erzeugen eine 400 MB große Datei, die als zusätzlicher Speicher für einen virtualisierten Gast genutzt werden kann.

- Erstellen Sie einen Speicherauszug der Gastkonfiguration. In diesem Beispiel heißt der Gast *Guest1* und die Datei wird im Benutzerverzeichnis gespeichert.

```
# virsh dumpxml Guest1 > ~/Guest1.xml
```

- Öffnen Sie die Konfigurationsdatei (in diesem Beispiel *Guest1.xml*) in einem Texteditor und suchen nach Einträgen, die mit "disk=" beginnen. Ein Eintrag könnte wie folgt aussehen:

```
>disk type='file' device='disk'<
  >driver name='tap' type='aio'</>
  >source file='/var/lib/libvirt/images/Guest1.img'</>
  >target dev='xvda'</>
</disk>
```

- Fügen Sie den zusätzlichen Speicher hinzu, indem Sie das Ende des disk=Eintrags ändern. Versichern Sie sich, dass Sie einen Gerätenamen für das virtuelle Blockgerät angeben, der noch nicht in der Konfigurationsdatei verwendet wird. Der folgende Beispieleintrag fügt eine Datei namens **FileName.img** als dateibasierten Speicher-Container hinzu:

```
>disk type='file' device='disk'<
  >driver name='tap' type='aio'</>
  >source file='/var/lib/libvirt/images/Guest1.img'</>
  >target dev='xvda'</>
</disk>
>disk type='file' device='disk'<
  >driver name='tap' type='aio'</>
  >source file='/var/lib/libvirt/images/FileName.img'</>
  >target dev='hda'</>
</disk>
```

- Starten Sie den Gast neu unter Verwendung der aktualisierten XML-Konfigurationsdatei.

```
# virsh create Guest1.xml
```

- Die folgenden Schritte sind spezifisch für einen Linux-Gast. Andere Betriebssysteme handhaben neue Speichergeräte auf andere Weise. Für Nicht-Linux-Systeme informieren Sie sich bitte in der Dokumentation Ihres Gast-Betriebssystems.

Der Gast verwendet nun die Datei **FileName.img** als ein Gerät namens **/dev/hdb**. Dieses Gerät muss vom Gast formatiert werden. Partitionieren Sie das Gerät auf dem Gast in eine einzige Primärpartition für das gesamte Gerät, und formatieren Sie anschließend das Gerät.

- Wählen Sie *n* für eine neue Partition.

```
# fdisk /dev/hdb  
Command (m for help):
```

- b. Wählen Sie *p* für eine Primärpartition.

```
Command action  
  e   extended  
  p   primary partition (1-4)
```

- c. Wählen Sie eine verfügbare Partitonsnummer. In diesem Beispiel wird die erste Partition ausgewählt durch Eingabe von *1*.

```
Partition number (1-4): 1
```

- d. Übernehmen Sie den Standardwert für den ersten Zylinder durch Drücken der *Eingabe*-Taste.

```
First cylinder (1-400, default 1):
```

- e. Wählen Sie die Größe der Partition. In diesem Beispiel wird die gesamte Festplatte zugewiesen durch Drücken der *Eingabe*-Taste.

```
Last cylinder or +size or +sizeM or +sizeK (2-400, default 400):
```

- f. Stellen Sie den Partitionstyp ein durch die Eingabe *t*.

```
Command (m for help): t
```

- g. Wählen Sie die Partition aus, die Sie im vorigen Schritt angelegt haben. In diesem Beispiel ist das die Partition *1*.

```
Partition number (1-4): 1
```

- h. Geben Sie *83* für eine Linux-Partition ein.

```
Hex code (type L to list codes): 83
```

- i. Speichern Sie Ihre Änderungen und beenden.

```
Command (m for help): w  
Command (m for help): q
```

- j. Formatieren Sie die neue Partition mit dem ext3-Dateisystem.

```
# mke2fs -j /dev/hdb
```

7. Hängen Sie die Festplatte auf dem Gast ein.

```
# mount /dev/hdb1 /myfiles
```

Der Gast besitzt nun ein zusätzliches, virtualisiertes, dateibasiertes Speichergerät.

Hinzufügen von Festplatten und anderen Blockgeräten zu einem Gast

Systemadministratoren nutzen zusätzliche Festplatten, um weiteren Speicherplatz zur Verfügung zu stellen oder auch um Systemdaten von Benutzerdaten trennen zu können. Dieses Verfahren, [Prozedur 4.1, „Hinzufügen physischer Blockgeräte zu virtualisierten Gästen“](#), beschreibt die Vorgehensweise zum Hinzufügen einer Festplatte auf dem Host zu einem virtualisierten Gast.

Dieses Verfahren funktioniert für alle physischen Blockgeräte einschließlich CD-ROM-, DVD- und Floppy-Laufwerke.

Prozedur 4.1. Hinzufügen physischer Blockgeräte zu virtualisierten Gästen

1. Schließen Sie die Festplatte physisch an den Host an. Konfigurieren Sie den Host entsprechend, falls auf das Laufwerk nicht standardmäßig zugegriffen werden kann.
2. Konfigurieren Sie das Gerät mit **multipath** und Persistenz auf dem Host, falls nötig.
3. Führen Sie den **virsh attach**-Befehl aus. Ersetzen Sie dabei *myguest* durch den Namen Ihres Gasts, */dev/hdb1* durch das hinzuzufügende Gerät und *hdc* durch den Ort des Geräts auf dem Gast. *hdc* muss ein noch nicht verwendeter Gerätenamen sein. Nutzen Sie die *hd**-Notation auch für Windows-Gäste, der Gast wird daraufhin das Gerät korrekt erkennen.

Hängen Sie für CD-ROM- oder DVD-Geräte dem Befehl den Parameter `--type hdd` an.

Hängen Sie für Floppy-Geräte dem Befehl den Parameter `--type floppy` an.

```
# virsh attach-disk myguest /dev/hdb1 hdc --driver tap --mode readonly
```

4. Der Gast besitzt nun ein neues Festplattengerät namens **/dev/hdb** unter Linux oder **D: drive** (oder ähnlich) unter Windows. Gegebenenfalls muss das Gerät noch formatiert werden.

4.3. Konfiguration von persistentem Speicher

Dieser Abschnitt bezieht sich auf Systeme mit Netzwerkspeicher oder externem Speicher, d. h. auf Fibre Channel oder iSCSI basierende Speichergeräte. Für derartige Systeme wird empfohlen, persistente Gerätenamen für Ihren Host zu konfigurieren. Dies ist nicht nur hilfreich bei der Live-Migration, sondern gewährleistet bei mehreren virtualisierten Geräten konsistente Gerätenamen und Speicher.

Universally Unique Identifiers (UUIDs) sind Teil eines standardisierten Verfahrens zur Identifizierung von Systemen und Geräten in verteilten Rechnernetzen. Dieser Abschnitt verwendet UUIDs dazu, iSCSI oder Fibre Channel LUNs zu identifizieren. UUIDs bleiben auch nach Neustarts, Abbruch der Verbindung oder Geräte austausch erhalten. Die UUID ist vergleichbar mit einer Kennung auf dem Gerät.

Systeme, auf denen **multipath** nicht läuft, müssen die *Konfiguration eines einzigen Pfads (Single Path)* durchführen. Systeme, auf denen **multipath** läuft, können die *Konfiguration multipler Pfade (Multiple Path)* durchführen.

Konfiguration eines einzigen Pfads (Single Path)

Dieses Verfahren implementiert *LUN*-Gerätersistenz mittels **udev**. Wenden Sie dieses Verfahren ausschließlich für Hosts an, die **multipath** nicht verwenden.

1. Bearbeiten Sie die `/etc/scsi_id.config`-Datei.
 - a. Vergewissern Sie sich, dass die Zeile **options=-b** auskommentiert ist.

```
# options=-b
```

- b. Fügen Sie folgende Zeile hinzu:

```
options=-g
```

Diese Option konfiguriert **udev** um anzunehmen, dass alle angeschlossenen SCSI-Geräte einen UUID (Unique Device Identifier) wiedergeben.

2. Um die UUID für ein bestimmtes Gerät anzuzeigen, führen Sie den Befehl `scsi_id -g -s /block/sd*` aus. Zum Beispiel:

```
# scsi_id -g -s /block/sd*
3600a0b800013275100000015427b625e
```

Die Ausgabe kann sich von dem obigen Beispiel unterscheiden. In der Ausgabe wird die UUID des Geräts `/dev/sdc` angezeigt.

3. Vergewissern Sie sich, dass die UUID-Ausgabe durch den `scsi_id -g -s /block/sd*`-Befehl identisch ist von Computern, die auf das Gerät zugreifen.
4. Erstellen Sie nun eine Regel für den Namen des Geräts. Legen Sie in `/etc/udev/rules.d` die Datei **20-names.rules** an. In dieser Datei fügen Sie neue Regeln hinzu. Alle Regeln werden in dieselbe Datei und in demselben Format eingefügt. Das Format für Regeln sieht folgendermaßen aus:

```
KERNEL="sd*", BUS="scsi", PROGRAM="/sbin/scsi_id -g -s", RESULT=UUID,
NAME=devicename
```

Ersetzen Sie *UUID* und *devicename* durch die vorher abgefragte UUID und den gewünschten Namen für das Gerät. Für das obige Beispiel könnte eine Regel wie folgt aussehen:

```
KERNEL="sd*", BUS="scsi", PROGRAM="/sbin/scsi_id -g -s",
RESULT="3600a0b800013275100000015427b625e", NAME="rack4row16"
```

Der **udev**-Daemon sucht daraufhin alle Geräte namens `/dev/sd*` für die UUID in der Regel. Sobald ein passendes Gerät mit dem System verbunden wird, wird dem Gerät der Name aus

der Regel zugewiesen. Das Gerät mit der UUID 3600a0b800013275100000015427b625e würde demnach als **/dev/rack4row16** erscheinen.

5. Fügen Sie folgende Zeile am Ende von **/etc/rc.local** an:

```
/sbin/start_udev
```

6. Kopieren Sie die Änderungen in den Dateien **/etc/scsi_id.config**, **/etc/udev/rules.d/20-names.rules** und **/etc/rc.local** für alle relevanten Hosts.

```
/sbin/start_udev
```

Netzwerksspeichergeräte mit konfigurierten Regeln haben jetzt persistente Namen auf all jenen Hosts, auf denen die Dateien aktualisiert wurden. Das bedeutet, dass Sie Gäste zwischen Hosts migrieren können mit Hilfe des gemeinsam genutzten Speichers, und die Gäste können auf die Speichergeräte in ihren Konfigurationsdateien zugreifen.

Konfiguration multipler Pfade (Multiple Path)

Das **multipath**-Paket wird für Systeme mit mehr als einem physischen Pfad vom Computer zu Speichergeräten verwendet. **multipath** bietet Fehlertoleranz, die Möglichkeit zum Failover, sowie verbesserte Leistung für Netzwerksspeichergeräte unter Linux-Systemen.

Um LUN-Persistenz in einer multipath-Umgebung zu implementieren, müssen Sie den Aliasnamen für die multipath-Geräte definieren. Jedes Speichergerät besitzt eine UUID, die als Schlüssel für die Aliasnamen fungiert. Sie können die UUID eines Geräts mit Hilfe des **scsi_id**-Befehls identifizieren.

```
# scsi_id -g -s /block/sdc
```

Die Multipath-Geräte werden im **/dev/mpath**-Verzeichnis angelegt. In dem nachfolgenden Beispiel sind vier Geräte in **/etc/multipath.conf** definiert:

```

multipaths {
    multipath {
        wwid          3600805f300159870000000000768a0019
        alias         oramp1
    }
    multipath {
        wwid          3600805f300159870000000000d643001a
        alias         oramp2
    }
    mulitpath {
        wwid          3600805f30015987000000000086fc001b
        alias         oramp3
    }
    mulitpath {
        wwid          3600805f300159870000000000984001c
        alias         oramp4
    }
}

```

Diese Konfiguration wird vier LUNs erzeugen namens `/dev/mpath/oramp1`, `/dev/mpath/oramp2`, `/dev/mpath/oramp3` und `/dev/mpath/oramp4`. Einmal eingegeben, ist die Zuordnung der Geräte-WWID zu ihren jeweiligen Namen auch nach einem Neustart persistent.

4.4. Hinzufügen eines virtualisierten CD-ROM- oder DVD-Laufwerks zu einem Gast

Um eine ISO-Datei in einen Gast einzufügen, während der Gast online ist, verwenden Sie **virsh** mit dem Parameter `attach-disk`.

```
# virsh attach-disk [domain-id] [source] [target] --driver file --type
  cdrom --mode readonly
```

Die Parameter `source` und `target` sind Pfade für die Dateien und Geräte im Host bzw. Gast. Der `source`-Parameter kann ein Pfad zu einer ISO-Datei oder das Gerät vom `/dev`-Verzeichnis sein.

Gemeinsam verwendeter Speicher und Virtualisierung

Dieses Kapitel behandelt den Einsatz von gemeinsam verwendetem Netzwerkspeicher mit Virtualisierung unter Fedora.

Die folgenden Methoden werden für Virtualisierung unterstützt:

- Fibre Channel
- iSCSI
- NFS
- GFS2

Netzwerkspeicher ist für Live- und Offline-Gastmigrationen unverzichtbar. Ohne gemeinsam verwendeten Speicher können Sie keine Migration von Gästen durchführen.

5.1. Verwenden von iSCSI zur Speicherung von Gästen

Dieser Abschnitt behandelt die Verwendung von iSCSI-basierten Geräten zur Speicherung von virtualisierten Gästen.

5.2. Verwenden von NFS zur Speicherung von Gästen

Dieser Abschnitt behandelt die Verwendung von NFS zur Speicherung von virtualisierten Gästen.

5.3. Verwenden von GFS2 zur Speicherung von Gästen

Dieser Abschnitt behandelt die Verwendung des Fedora Global File System 2 (GFS2) zur Speicherung von virtualisierten Gästen.

Beste Verfahren für Server

Die folgenden Tipps und Tricks können Ihnen dabei helfen, die Zuverlässigkeit Ihres Fedora Server Hosts (dom0) zu gewährleisten.

- Betreiben Sie SELinux im Enforcing-Modus. Sie können dies einstellen mit Hilfe des folgenden Befehls.

```
# setenforce 1
```

- Löschen oder deaktivieren Sie jeden unnötigen Dienst, wie z. B. **AutoFS**, **NFS**, **FTP**, **HTTP**, **NIS**, **telnetd**, **sendmail** und so weiter.
- Fügen Sie nur die minimale Anzahl von Benutzerkonten, die zur Verwaltung der Plattform auf dem Server benötigt werden, hinzu und löschen Sie unnötige Benutzerkonten.
- Vermeiden Sie, dass unnötige Applikationen auf Ihrem Host laufen. Das Ausführen von Applikationen auf dem Host kann Auswirkungen auf die Leistung Ihrer virtuellen Maschine haben und die Stabilität ihres Servers gefährden. Jede Applikation, die evtl. den Server zum Absturz bringen kann, würde infolgedessen auch alle virtuellen Maschinen auf dem Server zum Absturz bringen.
- Verwenden Sie einen zentralen Speicherort für Installationen und Abbilder virtueller Maschinen. Abbilder virtueller Maschinen sollten unter **/var/lib/libvirt/images/** gespeichert werden. Falls Sie ein anderes Verzeichnis für Ihre Abbilder virtueller Maschinen verwenden, stellen Sie sicher, dass Sie dieses Verzeichnis zu Ihrer SELinux-Richtlinie hinzufügen und vor Start der Installation neu kennzeichnen.
- Installationsquellen, -bäume und -abbilder sollten an einem zentralen Ort gespeichert werden, normalerweise der Ort Ihres vsftpd-Servers.

Sicherheit für Virtualisierung

Beim Einsatz der Virtualisierungstechnologie innerhalb der Infrastruktur Ihres Unternehmens müssen Sie sicherstellen, dass der Host nicht kompromittiert werden kann. Der Host im Xen-Hypervisor ist die privilegierte Domain, die die Systemverwaltung übernimmt und sämtliche virtuellen Maschinen verwaltet. Falls der Host unsicher ist, sind alle anderen Domains im System angreifbar. Es gibt verschiedene Möglichkeiten zur Verbesserung der Sicherheit auf Systemen mit Virtualisierung. Sie bzw. Ihre Organisation sollte einen *Einsatzplan* entwickeln, welcher nicht nur Funktionsspezifikationen enthält, sondern auch Dienste spezifiziert, die auf Ihren virtualisierten Gästen und Host-Servern laufen sollen, sowie die nötige Unterstützung für diese Dienste spezifiziert. Nachfolgend sind einige Themen im Zusammenhang mit der Sicherheit aufgeführt, die bei der Entwicklung eines Einsatzplans beachtet werden sollten:

- Führen Sie auf den Hosts nur absolut notwendige Dienste aus. Je weniger Dienste und Prozesse auf dem Host laufen, desto höher ist das Maß an Sicherheits und die Leistung.
- Aktivieren Sie [SELinux](#) auf dem Hypervisor. Lesen Sie [Abschnitt 7.1, „SELinux und Virtualisierung“](#) für mehr Informationen zu SELinux und Virtualisierung.
- Verwenden Sie eine Firewall, um den Datenverkehr zu dom0 einzuschränken. Sie können eine Firewall mit "default-reject"-Regeln einrichten, die dom0 gegen Attacken absichern. Weiterhin ist es wichtig, Dienste mit Netzwerkverbindung zu begrenzen.
- Erlauben Sie normalen Benutzern den Zugriff auf dom0 nicht. Wenn Sie normalen Benutzern den Zugriff auf dom0 gestatten, laufen Sie Gefahr, dom0 angreifbar zu machen. Denken Sie daran, dass dom0 privilegiert ist, das Einrichten von unprivilegierten Benutzerkonten kann daher das hohe Maß an Sicherheit gefährden.

7.1. SELinux und Virtualisierung

Security Enhanced Linux wurde von der NSA in Zusammenarbeit mit der Linux-Gemeinschaft dazu entwickelt, ein höheres Maß an Sicherheit für Linux zu erreichen. SELinux begrenzt die Möglichkeiten von Angreifern und verhindert viele der häufigen Sicherheitslücken wie z. B. Pufferüberläufe und Privilegeskalation. Aufgrund dieser Vorteile empfiehlt Fedora, dass auf allen Linux-Systemen SELinux im Enforcing-Modus aktiviert sein sollte.

SELinux verhindert das Laden von Gastabbildern, wenn SELinux aktiviert ist und die Abbilder nicht im richtigen Verzeichnis liegen. SELinux erfordert, dass alle Gastabbilder in `/var/lib/libvirt/images` gespeichert sind.

Hinzufügen von LVM-basiertem Speicher mit SELinux im Enforcing-Modus

Der folgende Abschnitt liefert ein Beispiel für das Hinzufügen eines logischen Datenträgers zu einem virtualisierten Gast bei aktiviertem SELinux. Diese Anleitung lässt sich auch auf Festplattenpartitionen übertragen.

[Prozedur 7.1. Erstellen und Einhängen eines logischen Datenträgers auf einem virtualisierten Gast mit aktiviertem SELinux](#)

1. Legen Sie einen logischen Datenträger an. Dieses Beispiel erstellt einen 5 GB großen logischen Datenträger namens `NewVolumeName` auf der Datenträgergruppe namens `volumeGroup`.

```
# lvcreate -n NewVolumeName -L 5G volumegroup
```

2. Formatieren Sie den logischen Datenträger *NewVolumeName* mit einem Dateisystem, das erweiterte Attribute unterstützt, wie z. B. ext3.

```
# mke2fs -j /dev/volumegroup/NewVolumeName
```

3. Erzeugen Sie ein neues Verzeichnis, um den neuen logischen Datenträger einzuhängen. Dieses Verzeichnis kann sich überall auf dem Dateisystem befinden, es wird jedoch empfohlen, es weder in einem der wichtigen Systemverzeichnisse zu erstellen (**/etc**, **/var**, **/sys**) noch in Benutzerverzeichnissen (**/home** oder **/root**). Dieses Beispiel verwendet ein Verzeichnis namens **/virtstorage**.

```
# mkdir /virtstorage
```

4. Hängen Sie den logischen Datenträger ein.

```
# mount /dev/volumegroup/NewVolumeName /virtstorage
```

5. Stellen Sie den richtigen SELinux-Typ für das Xen-Verzeichnis ein.

```
semanage fcontext -a -t xen_image_t "/virtualization(/.*)?"
```

Oder stellen Sie den richtigen SELinux-Typ für ein KVM-Verzeichnis ein.

```
semanage fcontext -a -t virt_image_t "/virtualization(/.*)?"
```

Falls die Targeted-Richtlinie verwendet wird (das ist der Standard), fügt der Befehl eine Zeile zur **/etc/selinux/targeted/contexts/files/file_contexts.local**-Datei hinzu, wodurch diese Änderung persistent gemacht wird. Die angefügte Zeile sieht etwa wie folgt aus:

```
/virtstorage(/.*)?    system_u:object_r:xen_image_t:s0
```

6. Führen Sie den Befehl aus, um den Typ des Einhängpunkts (**/virtstorage**) und aller darunterliegenden Dateien auf **xen_image_t** zu ändern (**restorecon** und **setfiles** lesen die Dateien in **/etc/selinux/targeted/contexts/files/**).

```
# restorecon -R -v /virtualization
```

7.2. Hinweise in Zusammenhang mit SELinux

Dieser Abschnitt enthält Dinge, die Sie beachten müssen, wenn Sie SELinux in Ihre Virtualisierungsumgebung implementieren. Wenn Sie Veränderungen im System durchführen oder Geräte hinzufügen, müssen Sie Ihre SELinux-Richtlinie entsprechend anpassen. Um einen LVM-Datenträger für einen Gast zu konfigurieren, müssen Sie den SELinux-Kontext für das entsprechende zu Grunde liegende Blockgerät und die Datenträgergruppe anpassen.

```
# semanage fcontext -a -t xen_image _t -f -b /dev/sda2  
# restorecon /dev/sda2
```

Der Boolesche Parameter **xend_disable_t** versetzt **xend** nach einem Neustart des Daemons in einen unbeschränkten Modus. Es ist besser, den Schutz für einen einzelnen Daemon zu deaktivieren, als für das gesamte System. Es wird empfohlen, dass Sie Verzeichnisse, die Sie an anderer Stelle verwenden werden, nicht als **xen_image_t** umkennzeichnen.

Netzwerkkonfiguration

Diese Seite gibt einen Überblick über gebräuchliche Netzwerkkonfigurationen, die von Anwendungen genutzt werden, die auf libvirt basieren. Diese Informationen gelten für alle Hypervisoren, egal ob Xen, KVM oder andere. Weitere Informationen finden Sie in der Dokumentation zur libvirt-Netzwerkarchitektur.

Die zwei gebräuchlichsten Konfigurationen sind "Virtuelles Netzwerk" und "Gemeinsam verwendetes physisches Gerät". Ersteres ist identisch über alle Distributionen hinweg und im Lieferumfang enthalten. Letzteres dagegen erfordert eine distributionspezifische, manuelle Konfiguration.

8.1. Network Address Translation (NAT) mit libvirt

Eine der häufigsten Methoden zur gemeinsamen Verwendung von Netzwerkverbindungen ist der Einsatz von Network Address Translation (NAT) Weiterleitung (auch "virtuelle Netzwerke" genannt).

Host-Konfiguration

In jeder standardmäßigen libvirt-Installation ist NAT-basierte Konnektivität zu virtuellen Maschinen bereits integriert. Dies ist das sogenannte "Virtuelle Standardnetzwerk". Überprüfen Sie dessen Verfügbarkeit mit dem Befehl **virsh net-list --all**.

```
# virsh net-list --all
Name                State      Autostart
-----
default             active    yes
```

Falls es nicht vorhanden ist, kann die Beispiel-XML-Konfigurationsdatei neu geladen und aktiviert werden:

```
# virsh net-define /usr/share/libvirt/networks/default.xml
```

Das Standardnetzwerk ist durch **/usr/share/libvirt/networks/default.xml** definiert.

Kennzeichnen Sie das Standardnetzwerk zum automatischen Start:

```
# virsh net-autostart default
Network default marked as autostarted
```

Starten Sie das Standardnetzwerk:

```
# virsh net-start default
Network default started
```

Sobald das libvirt-Standardnetzwerk läuft, werden Sie ein isoliertes Bridge-Gerät sehen. Dieses Gerät besitzt *keine* physischen Schnittstellen, da es NAT- und IP-Weiterleitung verwendet, um sich mit der Außenwelt zu verbinden. Fügen Sie keine neuen Schnittstellen hinzu.

```
# brctl show
bridge name      bridge id          STP enabled      interfaces
```

```
virbr0          8000.000000000000    yes
```

libvirt fügt **iptables**-Regeln hinzu, die Datenverkehr von und zu Gästen erlauben, die mit dem **virbr0**-Gerät in den **INPUT**, **FORWARD**, **OUTPUT** und **POSTROUTING**-Ketten verknüpft sind. **libvirt** versucht daraufhin, den **ip_forward**-Parameter zu aktivieren. Einige andere Anwendungen deaktivieren möglicherweise **ip_forward**, deshalb sollten Sie am Besten Folgendes zur **/etc/sysctl.conf**-Datei hinzufügen.

```
net.ipv4.ip_forward = 1
```

Gastkonfiguration

Sobald die Host-Konfiguration abgeschlossen ist, kann ein Gast basierend auf seinem Namen mit dem virtuellen Netzwerk verbunden werden. Um einen Gast mit dem virtuellen Standardnetzwerk zu verbinden, kann folgendes XML im Gast verwendet werden:

```
<interface type='network'>
  <source network='default' />
</interface>
```

Note

Es steht Ihnen frei, ob Sie eine MAC-Adresse angeben möchten. Falls Sie keine angeben, wird automatisch eine MAC-Adresse generiert. In einigen Fällen kann das manuelle Einstellen der MAC-Adresse jedoch sinnvoll sein.

```
<interface type='network'>
  <source network='default' />
  <mac address='00:16:3e:1a:b3:4a' />
</interface>
```

8.2. Bridged-Netzwerk mit libvirt

Bridged-Netzwerke (auch Gemeinsame Verwendung physischer Geräte genannt) werden eingesetzt, um einer virtuellen Maschine physische Geräte bereitzustellen. Bridging wird meist in fortgeschrittenen Konfigurationen und auf Servern mit mehreren Netzwerkschnittstellen angewendet.

Deaktivieren der Xen-Netzwerkskripte

Falls Ihr System eine Xen-Bridge verwendete, wird empfohlen, die standardmäßige Xen-Netzwerk-Bridge zu deaktivieren, indem Sie **/etc/xen/xend-config.sxp** bearbeiten und die folgende Zeile ändern, von:

```
(network-script network-bridge)
```

auf:

```
(network-script /bin/true)
```

Deaktivieren des NetworkManagers

Der NetworkManager unterstützt kein Bridging. Der NetworkManager muss daher deaktiviert werden, um die älteren Netzwerkskripte zu verwenden.

```
# chkconfig NetworkManager off
# chkconfig network on
# service NetworkManager stop
# service network start
```



Note

Anstatt den NetworkManager zu deaktivieren, können Sie "NM_CONTROLLED=no" zu den `ifcfg-*`-Skripten hinzufügen, die in den Beispielen verwendet werden.

Erstellen von Netzwerkinitialisierungsskripten

Erstellen oder bearbeiten Sie die folgenden zwei Netzwerkkonfigurationsdateien. Dieser Schritt kann mit verschiedenen Namen wiederholt werden für zusätzliche Netzwerk-Bridges.

Wechseln Sie in das `/etc/sysconfig/network-scripts`-Verzeichnis:

```
# cd /etc/sysconfig/network-scripts
```

Öffnen Sie die Netzwerkskripte für das Gerät, das Sie zur Bridge hinzufügen wollen. In diesem Beispiel definiert `ifcfg-eth0` die physische Netzwerkschnittstelle, die als Teil einer Bridge eingestellt wird:

```
DEVICE=eth0
# change the hardware address to match the hardware address your NIC uses
HWADDR=00:16:76:D6:C9:45
ONBOOT=yes
BRIDGE=br0
```



Tipp

Sie können die Maximum Transfer Unit (MTU) des Geräts konfigurieren, indem Sie eine `MTU`-Variable an das Ende der Konfigurationsdatei anfügen.

```
MTU=9000
```

Erstellen Sie ein neues Netzwerkskript in dem `/etc/sysconfig/network-scripts`-Verzeichnis namens `ifcfg-br0` oder ähnlich. `br0` ist der Name der Bridge. Dieser Name kann beliebig lauten, solange der Name der Datei dem Namen des `DEVICE`-Parameters entspricht.

```
DEVICE=br0
TYPE=Bridge
BOOTPROTO=dhcp
```

```
ONBOOT=yes  
DELAY=0
```



Warning

The line, `TYPE=Bridge`, is case-sensitive. It must have uppercase 'B' and lower case 'ridge'.

Starten Sie nach der Konfiguration das Netzwerk oder den Rechner neu.

```
# service network restart
```

Configure **iptables** to allow all traffic to be forwarded across the bridge.

```
# iptables -I FORWARD -m physdev --physdev-is-bridged -j ACCEPT  
# service iptables save  
# service iptables restart
```



Disable iptables on bridges

Alternatively, prevent bridged traffic from being processed by **iptables** rules. In `/etc/sysctl.conf` append the following lines:

```
net.bridge.bridge-nf-call-ip6tables = 0  
net.bridge.bridge-nf-call-iptables = 0  
net.bridge.bridge-nf-call-arptables = 0
```

Reload the kernel parameters configured with **sysctl**

```
# sysctl -p /etc/sysctl.conf
```

Restart the **libvirt** daemon.

```
# service libvirtd reload
```

Sie sollten jetzt über ein "Gemeinsam verwendetes physisches Gerät" verfügen, mit dem Gäste verknüpft werden können und so vollen LAN-Zugriff erlangen können. Überprüfen Sie Ihre neue Bridge wie folgt:

```
# brctl show  
bridge name      bridge id                STP enabled    interfaces  
virbr0           8000.00000000000000     yes            eth0  
br0              8000.000e0cb30550      no             eth0
```

Beachten Sie, dass die Bridge vollständig unabhängig ist von der **virbr0**-Bridge. Versuchen Sie *nicht*, ein physisches Gerät mit **virbr0** zu verknüpfen. Die **virbr0**-Bridge dient ausschließlich der Network Address Translation (NAT) Konnektivität.

KVM paravirtualisierte Treiber

Paravirtualisierte Treiber stehen für virtualisierte Windows-Gäste zur Verfügung, die auf KVM-Hosts laufen. Diese paravirtualisierten Treiber sind im virtio-Paket enthalten. Das virtio-Paket unterstützt Blockspeichergeräte und Netzwerkschnittstellen-Controller.

Paravirtualisierte Treiber verbessern die Leistung von voll virtualisierten Gästen. Mit paravirtualisierten Treibern verringert sich die Latenz bei I/O-Vorgängen des Gasts, und der Datendurchsatz wird erhöht auf nahezu dasselbe Niveau wie bei Bare-Metal-Implementierungen. Die Verwendung von paravirtualisierten Treibern wird empfohlen für voll virtualisierte Gäste, auf denen I/O-intensive Aufgaben und Anwendungen ausgeführt werden.

Die KVM paravirtualisierten Treiber werden in neueren Fedora-Versionen automatisch geladen und installiert. Diese Fedora-Versionen erkennen und installieren die Treiber, so dass zusätzliche Installationsschritte Ihrerseits nicht notwendig sind.

Wie auch mit dem KVM-Modul, so sind die virtio-Treiber nur auf Hosts verfügbar, auf denen neuere Fedora-Versionen laufen.



Note

Es stehen nur 28 PCI-Slots pro Gast für zusätzliche Geräte zur Verfügung. Jedes paravirtualisierte Netzwerk oder Blockgerät benötigt einen Slot. Jeder Gast kann bis zu 28 zusätzliche Geräte in einer beliebigen Kombination aus paravirtualisierten Netzwerk, paravirtualisierte Festplatten oder anderen PCI-Geräten mit VTd einsetzen.

Die folgenden Microsoft Windows Versionen besitzen unterstützte KVM paravirtualisierte Treiber:

- Windows XP,
- Windows Server 2003,
- Windows Vista, und
- Windows Server 2008.

9.1. Installation der KVM Windows paravirtualisierten Treiber

Dieser Abschnitt erläutert den Installationsvorgang für die KVM Windows paravirtualisierten Treiber. Die KVM paravirtualisierten Treiber können während der Windows-Installation geladen werden oder nach abgeschlossener Installation des Gasts installiert werden.

Sie können die paravirtualisierten Treiber mittels einer der folgenden Verfahren auf Ihrem Gast installieren:

- durch Hosten der Installationsdateien auf einem Netzwerk, auf das der Gast Zugriff hat,
- mittels eines virtualisierten CD-ROM-Laufwerks der Treiberinstallations-CD .iso-Datei, oder
- mittels eines virtualisierten Floppy-Laufwerks, um die Treiber zur Boot-Zeit zu installieren (für Windows-Gäste).

Dieses Handbuch beschreibt die Installation von der paravirtualisierten Installations-CD als virtualisiertes CD-ROM-Laufwerk.

1. Herunterladen der Treiber

Die Treiber sind erhältlich bei Microsoft (windowsservercatalog.com¹).

Das *virtio-win*-Paket installiert ein CD-ROM-Abbild namens **virtio-win.iso** im **/usr/share/virtio-win/**-Verzeichnis.

2. Installation der paravirtualisierten Treiber

Es wird empfohlen, die Treiber auf dem Gast zu installieren, bevor Sie ein Gerät, welches die paravirtualisierten Treiber verwenden soll, verknüpfen bzw. modifizieren.

Für Blockgeräte, auf denen Root-Dateisysteme gespeichert sind sowie für andere zum Booten des Gasts nötige Blockgeräte müssen die Treiber installiert sein, ehe das Gerät modifiziert wird. Sollten die Treiber nicht auf dem Gast installiert sein, und der Treiber wird auf den virtio-Treiber eingestellt, kann der Gast nicht starten.

Einhängen des Abbilds mit virt-Manager

Folgen Sie den Anweisungen unter [Prozedur 9.1](#), „[Verwenden von virt-manager zum Einhängen eines CD-ROM-Abbilds für einen Windows-Gast](#)“, um ein CD-ROM-Abbild mit **virt-manager** hinzuzufügen.

Prozedur 9.1. Verwenden von **virt-manager** zum Einhängen eines CD-ROM-Abbilds für einen Windows-Gast

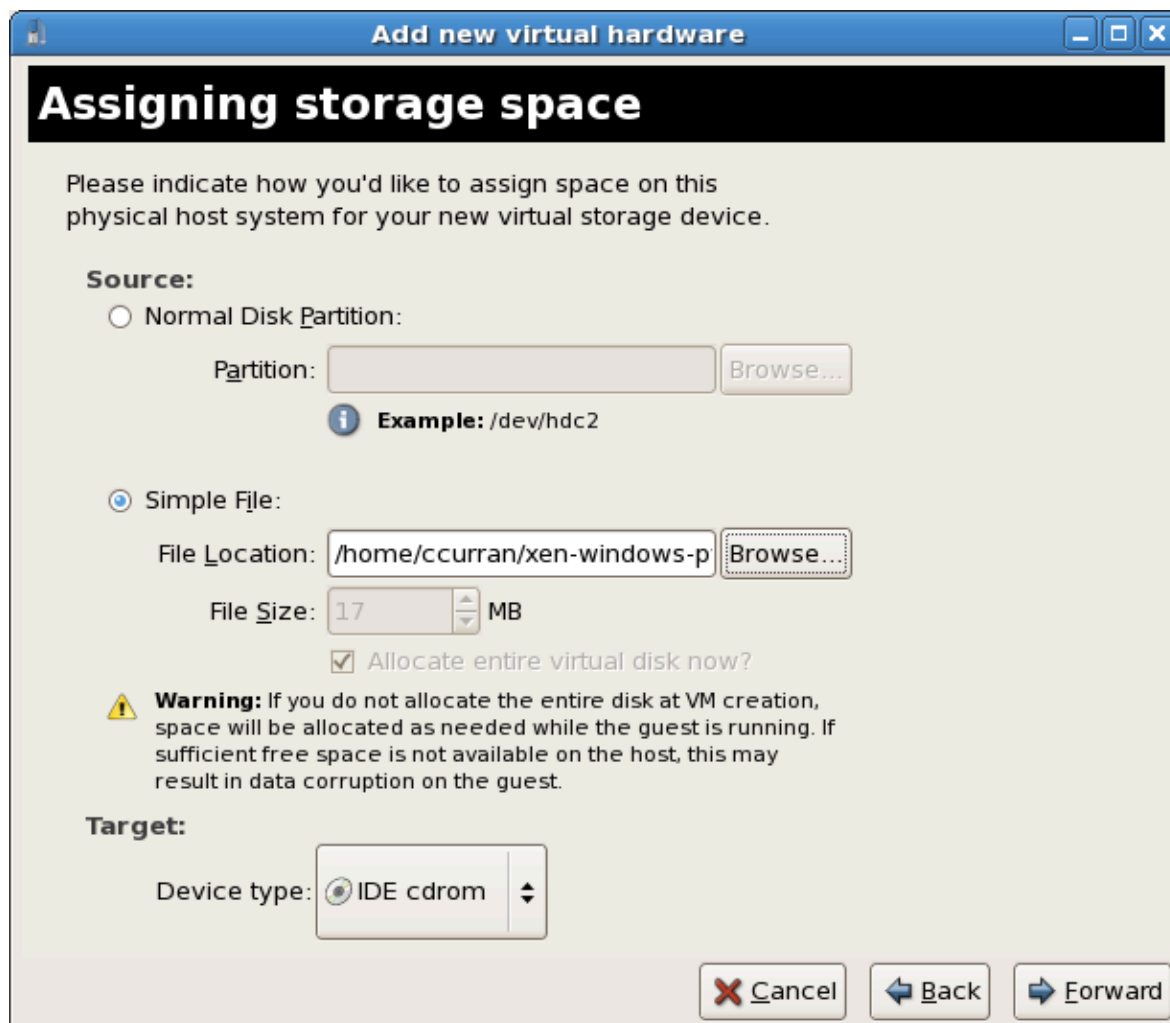
1. Öffnen Sie den **virt-manager**, wählen Sie Ihren virtualisierten Gast aus der Liste der virtuellen Maschinen aus und klicken die **Details**-Schaltfläche.
2. Klicken Sie die Schaltfläche **Hinzufügen** im **Details**-Fenster.
3. Dies startet einen Assistenten, der Sie beim Hinzufügen des Geräts unterstützt. Wählen Sie **Speichergerät** aus dem Drop-Down-Menü und klicken anschließend auf **Weiter**.



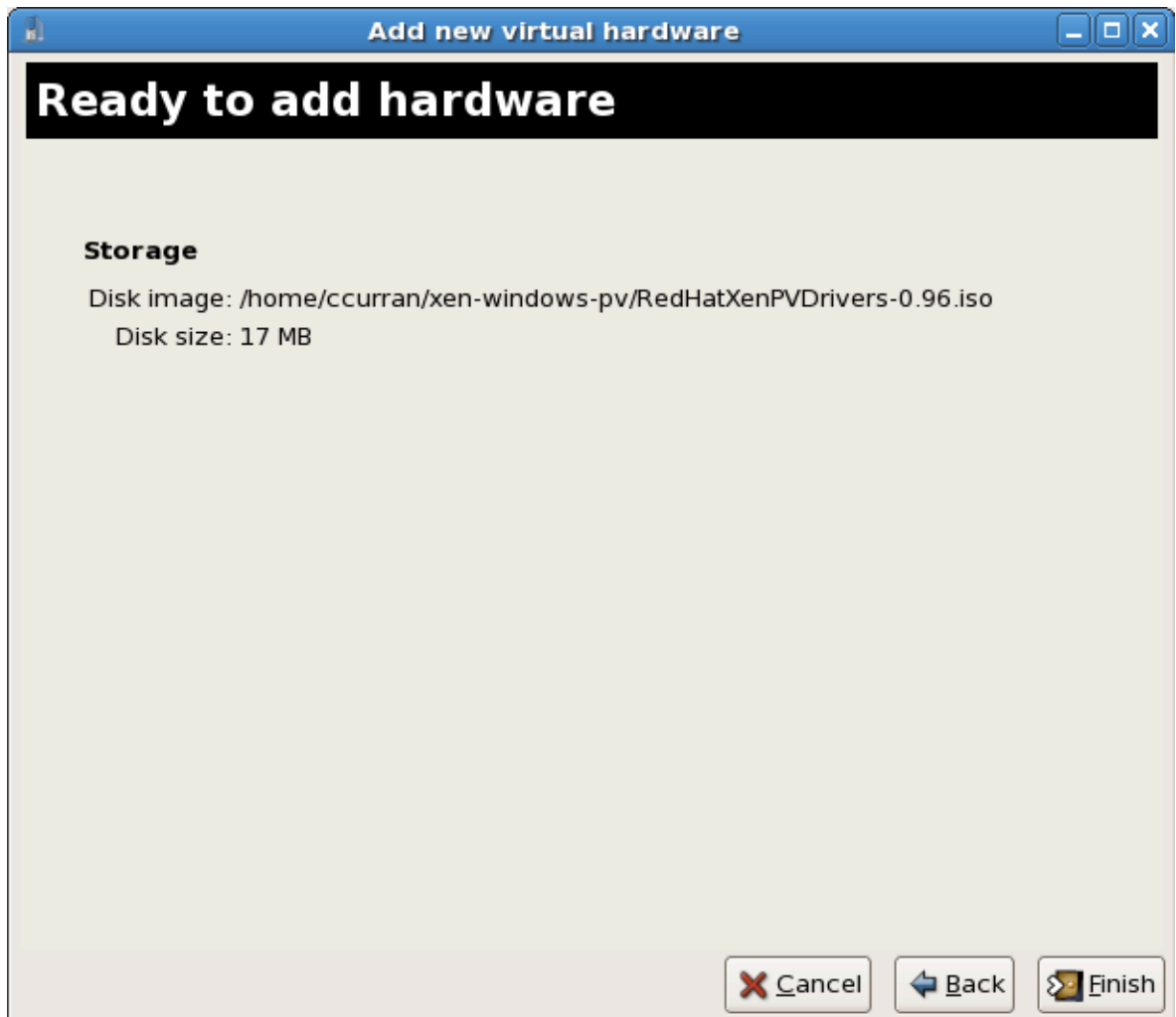
4. Wählen Sie die Option **Datei (Disk-Abbild)** und geben den Dateispeicherort der .iso-Datei des paravirtualisierten Treibers an. Der Speicherort der .iso-Dateien ist `/usr/share/xenpv-win`, sofern Sie **yum** zur Installation der paravirtualisierten Treiberpakete genutzt haben.

Falls die Treiber auf einer physischen CD vorliegen, wählen Sie die Option **Normale Plattenpartition**.

Setzen Sie den **Gerätetyp** auf **IDE CD-Rom** und klicken auf **Weiter**, um fortzufahren.



5. Die CD wurde zugewiesen und steht dem Gast nun zur Verfügung, sobald dieser gestartet wird. Klicken Sie auf **Abschließen**, um den Assistenten zu beenden, oder "Zurück", wenn Sie einen Fehler korrigieren möchten.



Installation mit einem virtualisierten Floppy-Laufwerk

Diese Anleitung zeigt die Installation der paravirtualisierten Treiber während einer Windows-Installation.

- Verbinden Sie bei der ersten Installation der Windows-VM mit Hilfe des "run-once"-Menüs **viostor.vfd** als Floppy.
 - a. **Windows Server 2003**

Wenn Windows Sie dazu auffordert, für Treiber von Drittanbietern F6 zu drücken, tun Sie dies und folgen den Anweisungen auf dem Bildschirm.
 - b. **Windows Server 2008**

Wenn das Installationsprogramm nach den Treibern verlangt, klicken Sie auf "Treiber laden", weisen auf Laufwerk A: und wählen den Treiber aus, der Ihrem Betriebssystem entspricht.

Verwenden der KVM paravirtualisierten Treiber für vorhandene Geräte

Modifizieren Sie eine bereits mit dem Gast verknüpfte Festplatte, so dass diese den **virtio**-Treiber anstelle des virtualisierten IDE-Treibers verwendet. In diesem Beispiel werden die libvirt-Konfigurationsdateien bearbeitet. Alternativ können auch **virt-manager**, **virsh attach-disk**

oder **virsh attach-interface** ein neues Gerät zur Verwendung der paravirtualisierten Treiber hinzufügen [Verwenden von KVM paravirtualisierten Treibern für neue Geräte](#).

1. Unten sehen Sie ein dateibasiertes Blockgerät, das den virtualisierten IDE-Treiber verwendet. Dies ist ein typischer Eintrag für einen virtualisierten Gast, der keine paravirtualisierten Treiber verwendet.

```
<disk type='file' device='disk'>
  <source file='/var/lib/libvirt/images/disk1.img' />
  <target dev='hda' bus='ide' />
</disk>
```

2. Um das paravirtualisierte Gerät zu verwenden, ändern Sie den Eintrag **bus=** auf **virtio**.

```
<disk type='file' device='disk'>
  <source file='/var/lib/libvirt/images/disk1.img' />
  <target dev='hda' bus='virtio' />
</disk>
```

Verwenden von KVM paravirtualisierten Treibern für neue Geräte

Diese Anleitung zeigt die Erstellung neuer Geräte zur Verwendung der KVM paravirtualisierten Treiber mittels **virt-manager**.

Alternativ können Sie auch die Befehle **virsh attach-disk** oder **virsh attach-interface** nutzen, um Geräte zu verknüpfen, die die paravirtualisierten Treiber verwenden.



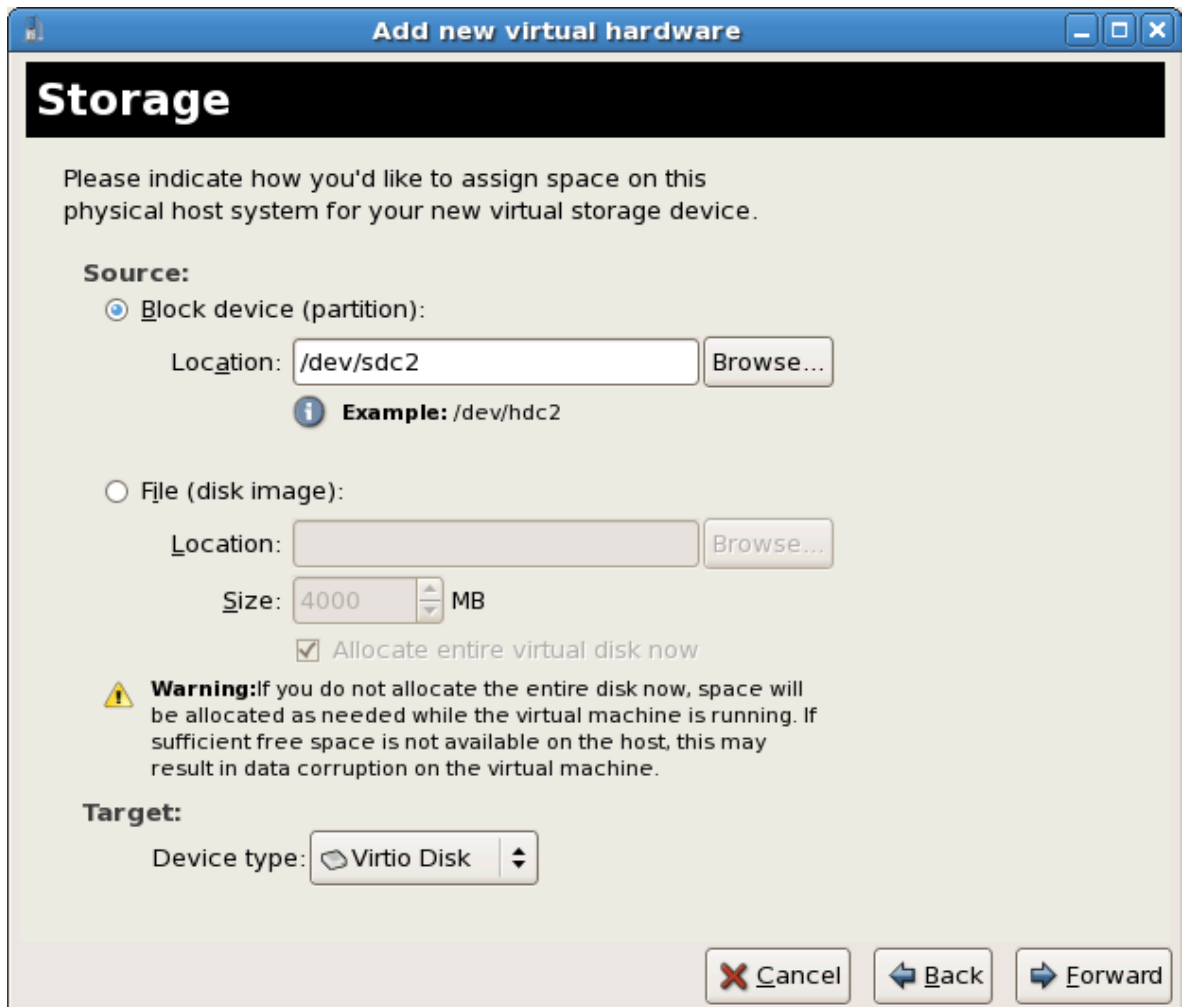
Installieren Sie die Treiber zuerst

Vergewissern Sie sich, dass die Treiber auf dem Windows-Gast installiert wurden, ehe Sie damit beginnen, neue Geräte zu installieren. Falls die Treiber nicht bereitstehen, kann das Gerät nicht erkannt werden und wird nicht funktionieren.

1. Öffnen Sie den virtualisierten Gast, indem Sie in **virt-manager** auf den Namen des Gasts doppelklicken.
2. Öffnen Sie den **Hardware**-Reiter.
3. Klicken Sie auf die Schaltfläche **Hardware hinzufügen**.
4. Wählen Sie unter dem Reiter "Hinzufügen virtueller Hardware" **Speicher** oder **Netzwerk** für den Gerätetyp.

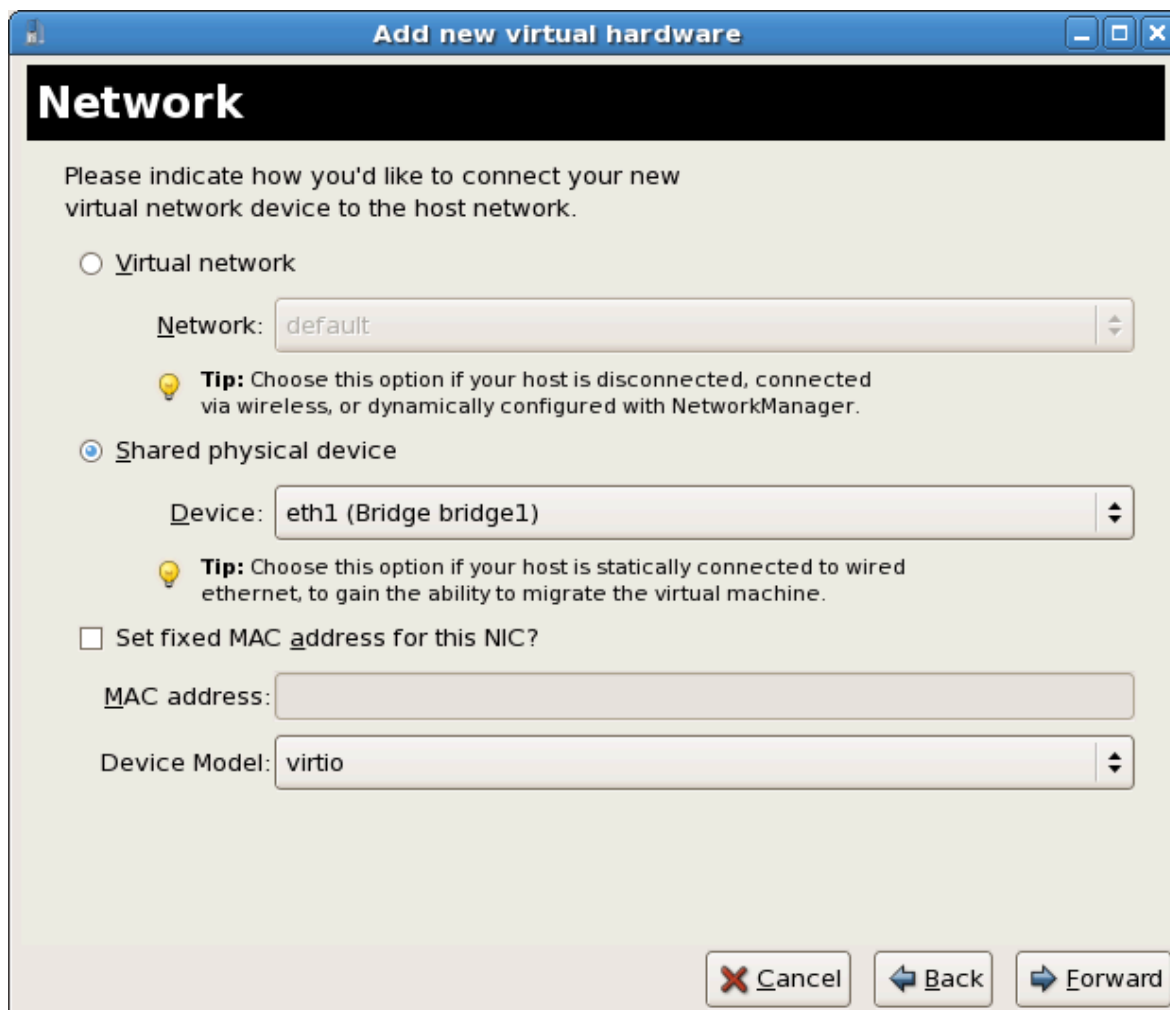
1. **Neue Festplatten**

Wählen Sie das Speichergerät oder dateibasierte Abbild. Wählen Sie **Virtio-Disk** als **Gerätetyp** und klicken auf **Weiter**.



2. Neue Netzwerkgeräte

Wählen Sie **Virtuelles Netzwerk** oder **Gemeinsam verwendetes physisches Gerät**.
Wählen Sie **Virtio** als **Gerätetyp** und klicken auf **Weiter**.



5. Klicken Sie auf **Abschließen**, um das Gerät zu speichern.



6. Starten Sie den Gast nun neu. Das Gerät wird unter Umständen vom Windows-Gast nicht erkannt, so lange er nicht neu gestartet wird.

Teil III. Administration

Verwalten virtualisierter Systeme

Dieses Kapitel beinhaltet Information zur Verwaltung des Hosts und der virtualisierten Gäste mit Hilfe von Tools, die in Fedora enthalten sind.

Gästeverwaltung mit Hilfe von xend

Der **xend**-Daemon führt gewisse Systemverwaltungsfunktionen durch, die im Zusammenhang mit virtuellen Maschinen stehen. Dieser Daemon kontrolliert die virtualisierten Ressourcen, demnach muss **xend** laufen, um mit den virtuellen Maschinen interagieren zu können. Bevor Sie **xend** starten, müssen Sie die Betriebsparameter angeben, indem Sie die **xend**-Konfigurationsdatei **xend-config.sxp**, die sich im Verzeichnis **etc/xen** befindet, bearbeiten. Nachfolgend finden Sie die Parameter, die Sie in der Konfigurationsdatei **xend-config.sxp** aktivieren oder deaktivieren können:

Element	Description
(console-limit)	Bestimmt das Arbeitsspeicher-Pufferlimit <code>xend_unix_server</code> des Konsolenservers und weist Werte auf einer pro-Domain-Basis zu.
(min-mem)	Bestimmt die minimale Anzahl an Megabytes, die für <code>domain0</code> reserviert werden (wenn Sie 0 eingeben, ändert sich der Wert nicht).
(dom0-cpus)	Bestimmt die Zahl der von <code>domain0</code> verwendeten CPUs (mindestens 1 CPU wird standardmäßig zugewiesen).
(enable-dump)	Bestimmt, dass im Falle eines Absturzes ein Dump aktiviert wird (Standard ist 0).
(external-migration-tool)	Bestimmt das Skript oder die Anwendung, die die externe Gerätemigration handhabt (die Skripte müssen sich in etc/xen/scripts/external-device-migrate befinden).
(logfile)	Bestimmt den Ort der Protokolldatei (standardmäßig /var/log/xend.log).
(loglevel)	Filtert die Werte des Protokollmodus aus: <code>DEBUG</code> , <code>INFO</code> , <code>WARNING</code> , <code>ERROR</code> oder <code>CRITICAL</code> (standardmäßig <code>DEBUG</code>).
(network-script)	Bestimmt das Skript, welches die Netzwerkumgebung aktiviert (Skripte müssen sich im Verzeichnis etc/xen/scripts befinden).
(xend-http-server)	Aktiviert den HTTP-Stream Paket-Management-Server (standardmäßig "no").
(xend-unix-server)	Aktiviert den UNIX Domain-Socket-Server, welcher ein Socket-Server ist ein Kommunikationsendpunkt, der Lowlevel-Netzwerkverbindungen handhabt und einkommende Verbindungen akzeptiert oder abweist. Der standardmäßige Wert ist Yes.
(xend-relocation-server)	Aktiviert den Migrations-Server für maschinenübergreifende Migrationen (standardmäßig "no").

Element	Description
(xend-unix-path)	Bestimmt den Ort, an dem der Befehl xend-unix-server Daten ausgibt (standardmäßig var/lib/xend/xend-socket)
(xend-port)	Bestimmt den Port, den der HTTP-Management-Server verwendet (standardmäßig 8000).
(xend-relocation-port)	Bestimmt den Port, den der Migrations-Server verwendet (standardmäßig 8002).
(xend-relocation-address)	Bestimmt die Host-Adressen, die zur Migration zugelassen sind. Der Standardwert ist der Wert der xend-address .
(xend-address)	Bestimmt die Adresse, mit der sich der Domain-Socket-Server verbindet. Der standardmäßige Wert erlaubt alle Verbindungen.

Tabelle 10.1. **xend**-Konfigurationsparameter

Nach dem Einrichten dieser Betriebsparameter sollten Sie überprüfen, dass **xend** läuft. Sollte dies nicht der Fall sein, initialisieren Sie den Daemon. Sie können den **xend**-Daemon am Kommandozeilenprompt starten, indem Sie Folgendes eingeben:

```
service xend start
```

Sie können **xend** auch dazu verwenden, den Daemon zu stoppen:

```
service xend stop
```

Dies beendet das Ausführen des Daemons.

Sie können mit **xend** den Daemon erneut starten:

```
service xend restart
```

Der Daemon startet erneut.

Und Sie können den Status des **xend**-Daemons überprüfen.

```
service xend status
```

Die Ausgabe zeigt den Status des Daemons.



xend zur Boot-Zeit aktivieren

Verwenden Sie den Befehl **chkconfig**, um **xend** in das **initscript** einzufügen.

```
chkconfig --level 345 xend
```

Der **xend** wird nicht auf den Runlevels 3,4 und 5 starten.

Zeitverwaltung bei KVM-Gästen

KVM verwendet die konstante Time Stamp Counter (TSC) Funktion in vielen modernen CPUs. Einige CPUs besitzen jedoch keinen konstanten Time Stamp Counter, was sich auf die Art und Weise auswirkt, wie Gäste unter KVM die Zeit messen. Gäste, die ohne genaue Zeitmessung laufen, können erhebliche Auswirkungen auf einige Netzwerkanwendungen haben, denn diese Gäste laufen schneller oder langsamer als die tatsächliche Zeit.

Mehrere Probleme können auftreten bei Gästen mit ungenauer Zeitmessung:

- Systemuhren sind ggf. nicht mehr synchron mit der tatsächlichen Zeit, was Sitzungen ungültig macht und Auswirkungen auf Netzwerke hat.
- Gäste mit langsameren Systemuhren haben ggf. Probleme mit der Migration.
- Gäste können stoppen oder abstürzen.

Diese Probleme existieren ebenso auf anderen Virtualisierungsplattformen; die Zeitmessung sollte daher immer überprüft werden.



NTP

Der Network Time Protocol (NTP) Daemon sollte sowohl auf dem Host als auch auf dem Gast laufen. Aktivieren Sie den ntpd-Dienst wie folgt:

```
# service ntpd start
```

Fügen Sie den ntpd-Dienst zur standardmäßigen Startup-Sequenz hinzu:

```
# chkconfig ntpd on
```

Die Verwendung des ntpd-Dienstes sollte die Folgen der Zeitabweichung in jedem Fall minimieren.

Feststellen, ob Ihre CPU über den konstanten Time Stamp Counter verfügt

Ihre CPU verfügt über den konstanten Time Stamp Counter, wenn das `constant_tsc`-Flag vorhanden ist. Um festzustellen, ob Ihre CPU das `constant_tsc`-Flag gesetzt hat, führen Sie den folgenden Befehl aus:

```
$ cat /proc/cpuinfo | grep constant_tsc
```

Wenn Sie eine Ausgabe erhalten, verfügt Ihre CPU über das `constant_tsc`-Bit. Falls keinerlei Ausgabe erfolgt, folgen Sie den unten stehenden Anweisungen.

Konfiguration von Hosts ohne konstanten Time Stamp Counter

Systeme ohne konstanten Time Stamp Counter erfordern zusätzliche Konfiguration. Funktionen der Energieverwaltung behindern die genaue Zeitmessung und müssen deaktiviert werden, damit Gäste mit KVM die genaue Zeit messen können.



Note

Diese Anweisungen gelten ausschließlich für AMD Revision F CPUs.

Falls die CPU nicht über das `constant_tsc`-Bit verfügt, deaktivieren Sie sämtliche Funktionen zur Energieverwaltung ([BZ#513138](https://bugzilla.redhat.com/show_bug.cgi?id=513138)¹). Jedes System hat mehrere Timer, die es zur Zeitmessung verwendet. Der TSC ist nicht stabil auf dem Host, was manchmal durch Änderungen an `cpufreq` verursacht werden kann, durch tiefen C-Status oder durch Migration auf einen Host mit einem schnelleren TSC. Um tiefe C-States, die den TSC anhalten können, zu vermeiden, fügen Sie auf dem Host "`processor.max_cstate=1`" zu den Boot-Optionen des Kernels in Grub hinzu:

```
term Fedora (vmlinuz-2.6.29.6-217.2.3.fc11)
    root (hd0,0)
    kernel /vmlinuz-vmlinuz-2.6.29.6-217.2.3.fc11 ro root=/dev/
VolGroup00/LogVol100 rhgb quiet processor.max_cstate=1
```

Deaktivieren Sie `cpufreq` (nur nötig auf Hosts ohne `constant_tsc`), indem Sie die Konfigurationsdatei `/etc/sysconfig/cpuspeed` bearbeiten und die Variablen `MIN_SPEED` und `MAX_SPEED` auf die höchstmögliche Frequenz ändern. Zulässige Höchstgrenzen finden Sie in den `/sys/devices/system/cpu/cpu*/cpufreq/scaling_available_frequencies`-Dateien.

Verwendung der paravirtualisierten Systemuhr mit Red Hat Enterprise Linux Gästen

Für bestimmte Red Hat Enterprise Linux Gäste sind zusätzliche Kernel-Parameter erforderlich. Diese Parameter können gesetzt werden, indem Sie an das Ende der `/kernel`-Zeile in der `/boot/grub/grub.conf`-Datei des Gasts angehängt werden.

Die Tabelle unten zeigt Red Hat Enterprise Linux Versionen und deren erforderliche Parameter für Gäste ohne konstanten Time Stamp Counter.

Red Hat Enterprise Linux	Zusätzliche Kernel-Parameter für den Gast
5.4 AMD64/Intel 64 mit der paravirtualisierten Uhr	Zusätzliche Parameter nicht notwendig
5.4 AMD64/Intel 64 ohne die paravirtualisierte Uhr	<code>divider=10 notsc lpj=n</code>
5.4 x86 mit der paravirtualisierten Uhr	Zusätzliche Parameter nicht notwendig
5.4 x86 ohne die paravirtualisierte Uhr	<code>divider=10 clocksource=acpi_pm lpj=n</code>
5.3 AMD64/Intel 64	<code>divider=10 notsc</code>
5.3 x86	<code>divider=10 clocksource=acpi_pm</code>
4.8 AMD64/Intel 64	<code>notsc divider=10</code>
4.8 x86	<code>clock=pmtmr divider=10</code>
3.9 AMD64/Intel 64	Zusätzliche Parameter nicht notwendig
3.9 x86	Zusätzliche Parameter nicht notwendig

¹ https://bugzilla.redhat.com/show_bug.cgi?id=513138

Verwenden der paravirtualisierten Uhr mit Windows-Gästen

Aktivieren Sie die paravirtualisierte Systemuhr auf Windows-Gästen durch Bearbeiten der Boot-Parameter. Die Windows-Boot-Einstellungen sind in der boot.ini-Datei gespeichert. Fügen Sie die folgende Zeile hinzu, um die paravirtualisierte Systemuhr zu aktivieren:

```
/use pmtimer
```

Weitere Information über die Windows-Boot-Einstellungen und die pmtimer-Option finden Sie unter [Verfügbare Befehlszeilenoptionen für die Boot.ini-Dateien von Windows XP und Windows Server 2003](#)².

² <http://support.microsoft.com/kb/833721>

KVM Live-Migration

Dieses Kapitel behandelt die Migration von Gästen auf einem KVM-Hypervisor zu einem anderen KVM-Host.

Migration bezeichnet den Vorgang, virtualisierte Gäste von einem Host auf einen anderen zu verschieben. Migration ist eine Schlüsseleigenschaft der Virtualisierung, da die Software vollständig von der Hardware getrennt ist. Migration ist hilfreich für:

- Load balancing - guests can be moved to hosts with lower usage when a host becomes overloaded.
- Hardware failover - when hardware devices on the host start to fail, guests can be safely relocated so the host can be powered down and repaired.
- Energy saving - guests can be redistributed to other hosts and host systems powered off to save energy and cut costs in low usage periods.
- Geographic migration - guests can be moved to another location for lower latency or in serious circumstances.

Eine Migration kann "live" oder "offline" erfolgen. Um Gäste zu migrieren, muss der Speicher gemeinsam verwendet werden. Migration funktioniert, indem der Speicher des Gasts zum Ziel-Host übertragen wird. Der gemeinsam verwendete Speicher speichert das Standarddateisystem des Gasts. Das Dateisystemabbild wird nicht über das Netzwerk vom Quell-Host zum Ziel-Host gesendet.

An offline migration suspends the guest then moves an image of the guests memory to the destination host. The guest is resumed on the destination host and the memory the guest used on the source host is freed.

Die Zeit, die eine Offline-Migration dauert, hängt von der Netzwerkbandbreite und Latenz ab. Ein Gast mit 2 GB Speicher sollte über eine 1 Gbit Ethernet-Verbindung etwa zehn Sekunden brauchen.

Bei einer Live-Migration läuft der Gast auf dem Quell-Host weiter ohne anzuhalten, während der Speicher verschoben wird. Jegliche Änderungen an den Speicherseiten werden nachverfolgt und zum Ziel gesendet, nachdem das Abbild übertragen wurde. Der Speicher wird dann mit den modifizierten Speicherseiten aktualisiert. Dieser Prozess läuft so lange, bis die Zeit, die dem Gast zum Pausieren zugestanden wird, der Zeit entspricht, die voraussichtlich für die Übertragung der letzten Seiten benötigt wird. KVM schätzt die verbleibende Zeit und versucht, die höchstmögliche Anzahl an Seiten von der Quelle zum Ziel zu übertragen, bis für KVM absehbar ist, dass die verbleibenden Seiten während einer kurzen Zeitspanne, die die virtuelle Maschine pausiert, übertragen werden können. Die Register werden nun auf dem neuen Host geladen und der Gast wird schließlich auf dem Ziel-Host wieder gestartet. Falls der Gast auf diese Weise nicht übertragen werden kann (was bei extrem hoher Auslastung des Gasts vorkommen kann), so wird er angehalten und stattdessen eine Offline-Migration eingeleitet.

Die Zeit, die eine Offline-Migration dauert, hängt von der Netzwerkbandbreite und Latenz ab. Bei hoher Auslastung des Netzwerks oder bei geringer Bandbreite dauert die Migration deutlich länger.

12.1. Voraussetzungen der Live-Migration

Für die Migration von Gästen müssen die folgenden Voraussetzungen erfüllt sein:

Migrationsvoraussetzungen

- Ein virtualisierter Gast, installiert auf gemeinsam verwendetem Netzwerkspeicher, unter Verwendung eines der folgenden Protokolle:
 - Fibre Channel
 - iSCSI
 - NFS
 - GFS2
- Zwei oder mehr Fedora-Systeme derselben Version mit denselben Aktualisierungen.
- Auf beiden Systemen müssen die entsprechenden Ports offen sein.
- Beide Systeme müssen eine identische Netzwerkkonfiguration besitzen. Sämtliches Bridging und sämtliche Netzwerkkonfiguration muss auf beiden Hosts genau übereinstimmen.
- Gemeinsam verwendeter Speicher muss auf dem Quell-Host und dem Ziel-Host an derselben Stelle eingehängt sein. Der Name des eingehängten Verzeichnisses muss identisch sein.

Konfiguration von Netzwerkspeicher

Konfigurieren Sie gemeinsam genutzten Speicher und installieren Sie Gäste auf diesem gemeinsamen Speicher. Anleitungen hierzu finden Sie unter [Kapitel 5, Gemeinsam verwendeter Speicher und Virtualisierung](#).

Alternativ dazu können Sie das NFS-Beispiel in [Abschnitt 12.2, „Beispiel für gemeinsam genutzten Speicher: NFS für eine einfache Migration“](#) verwenden.

12.2. Beispiel für gemeinsam genutzten Speicher: NFS für eine einfache Migration

Dieses Beispiel nutzt NFS, um Gastabbilder mit anderen KVM-Hosts gemeinsam zu verwenden. Dieses Beispiel ist für größere Installationen weniger praktisch, es soll lediglich dazu dienen, Migrationstechniken und kleine Einsätze zu veranschaulichen. Verwenden Sie dieses Beispiel nicht zur Migration oder Ausführung von mehr als einer Handvoll virtualisierter Gäste.

Weiterführende und genauere Anleitungen für gemeinsam genutzten Speicher finden Sie unter [Kapitel 5, Gemeinsam verwendeter Speicher und Virtualisierung](#)

1. Exportieren Sie Ihr libvirt-Abbildverzeichnis

Fügen Sie das standardmäßige Abbildverzeichnis zur `/etc/exports`-Datei hinzu:

```
/var/lib/libvirt/images *.bne.redhat.com(rw,no_root_squash,async)
```

Passen Sie den Host-Parameter auf Ihre Umgebung an.

2. Starten Sie NFS

a. Installieren Sie die NFS-Pakete, falls diese noch nicht installiert sind:

```
# yum install nfs
```


- b. Öffnen Sie in **iptables** die Ports für NFS und fügen NFS zur **/etc/hosts.allow**-Datei hinzu.
- c. Starten Sie den NFS-Dienst:

```
# service nfs start
```

3. Hängen Sie den gemeinsam verwendeten Speicher am Ziel ein

Hängen Sie das **/var/lib/libvirt/images**-Verzeichnis auf dem Zielsystem ein:

```
# mount sourceURL:/var/lib/libvirt/images /var/lib/libvirt/images
```



Einhängpunkte müssen auf dem Quell- und Zielsystem identisch sein.

Das für den Gast gewählte Verzeichnis muss auf dem Host und dem Gast exakt dasselbe sein. Dies gilt für alle Arten von gemeinsamem Speicher. Ist das Verzeichnis nicht identisch, schlägt die Migration fehl.

12.3. KVM Live-Migration mit virsh

Ein Gast kann mit Hilfe des **virsh**-Befehls auf einen anderen Host migriert werden. Der **migrate**-Befehl akzeptiert Parameter im folgenden Format:

```
# virsh migrate --live GuestName DestinationURL
```

The *GuestName* parameter represents the name of the guest which you want to migrate.

The *DestinationURL* parameter is the URL or hostname of the destination system. The destination system must run the same version of Fedora, be using the same hypervisor and have **libvirt** running.

Once the command is entered you will be prompted for the root password of the destination system.

Beispiel: Live-Migration mit virsh

Dieses Beispiel migriert von `test1.bne.redhat.com` nach `test2.bne.redhat.com`. Passen Sie die Host-Namen auf Ihre Umgebung an. Dieses Beispiel migriert eine virtuelle Maschine namens **CentOS4test**.

In diesem Beispiel wird davon ausgegangen, dass Sie gemeinsam verwendeten Speicher vollständig konfiguriert haben und alle Voraussetzungen erfüllt sind (siehe [Migrationsvoraussetzungen](#)).

1. Überprüfen Sie, dass der Gast läuft

Überprüfen Sie vom Quellsystem `test1.bne.redhat.com` aus, ob `CentOS4test` läuft:

```
[root@test1 ~]# virsh list
Id Name                               State
-----
```

```
10 CentOS4          running
```

2. Migrieren Sie den Gast

Führen Sie folgenden Befehl aus, um eine Live-Migration des Gasts zum Ziel `test2.bne.redhat.com` durchzuführen. Fügen Sie `/system` an das Ende der Ziel-URL an, um libvirt mitzuteilen, dass Sie umfassenden Zugriff benötigen.

```
# virsh migrate --live CentOS4test qemu+ssh://test2.bne.redhat.com/system
```

Once the command is entered you will be prompted for the root password of the destination system.

3. Warten Sie

Die Migration kann abhängig von der Größe und Auslastung des Gasts einige Zeit in Anspruch nehmen. `virsh` meldet nur Fehler. Der Gast wird solange weiterhin auf dem Quell-Host ausgeführt, bis er vollständig migriert ist.

4. Überprüfen Sie, ob der Gast auf dem Ziel-Host angekommen ist

Überprüfen Sie vom Zielsystem `test2.bne.redhat.com` aus, ob `CentOS4test` läuft:

```
[root@test2 ~]# virsh list
Id Name                State
-----
10 CentOS4             running
```

Die Live-Migration ist damit abgeschlossen.



Andere Netzwerkverfahren

libvirt unterstützt eine Vielzahl an Netzwerkverfahren einschließlich TLS/SSL, Unix-Sockets, SSH und unverschlüsseltes TCP. Siehe [Kapitel 13, Remote-Verwaltung virtualisierter Gäste](#) für weitere Informationen über den Gebrauch anderer Verfahren.

12.4. Migration mit virt-manager

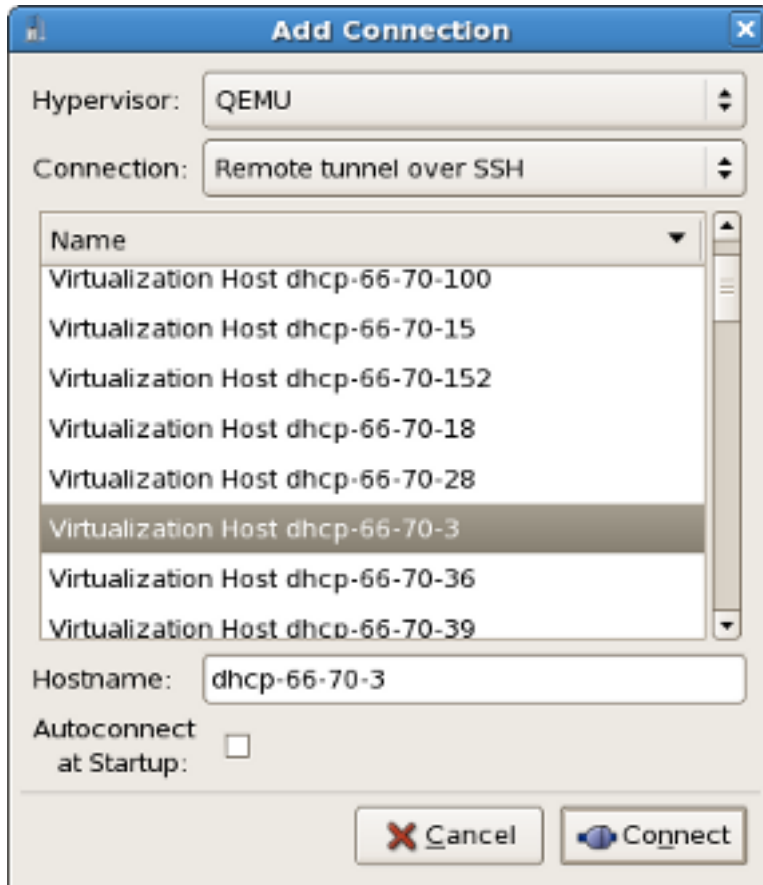
Dieser Abschnitt behandelt die Migration von KVM-basierten Gästen mit `virt-manager`.

1. Verbinden Sie die Quell- und Ziel-Hosts. Klicken Sie dazu im **Datei**-Menü auf **Verbindung hinzufügen**, woraufhin das Fenster **Verbindung hinzufügen** erscheint.

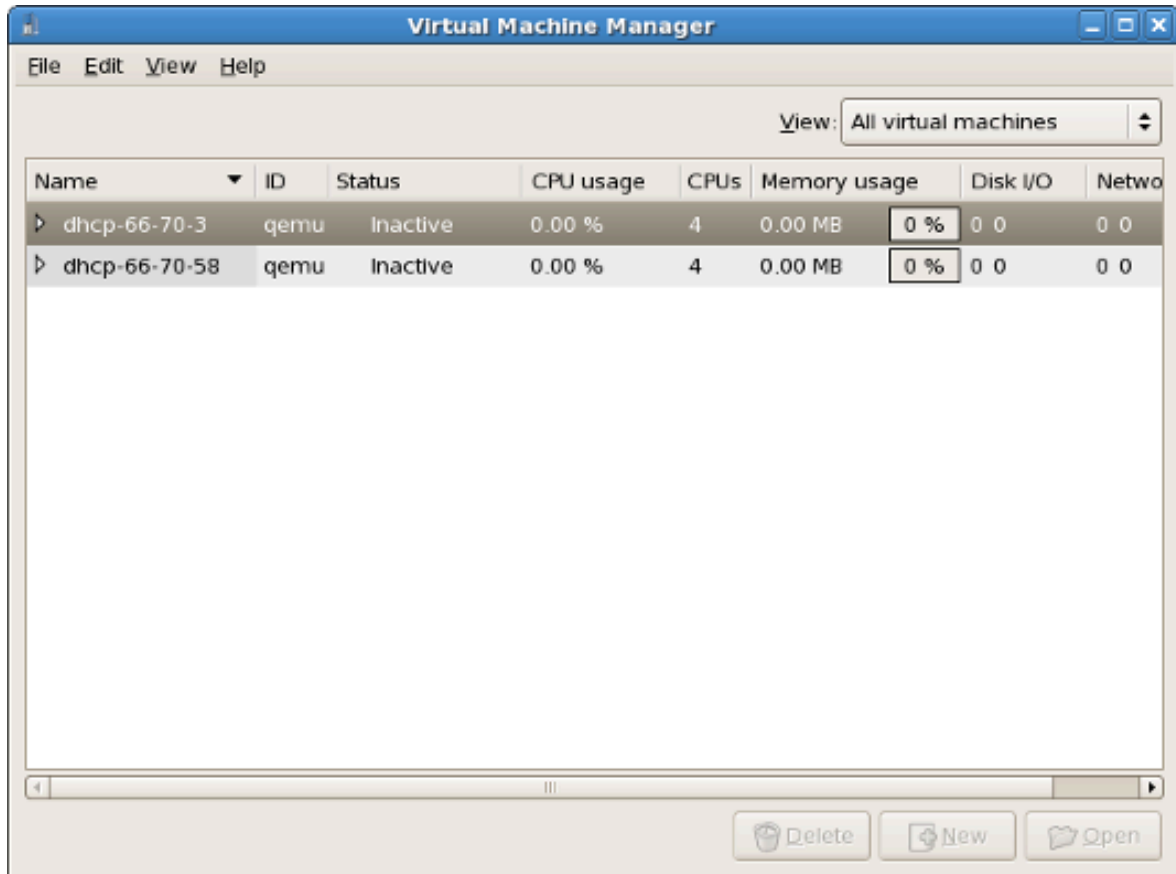
Geben Sie die folgenden Informationen ein:

- **Hypervisor:** Wählen Sie **QEMU**.
- **Verbindung:** Wählen Sie die Verbindungsart.
- **Host-Name:** Geben Sie den Host-Namen ein.

Klicken Sie auf **Verbinden**.



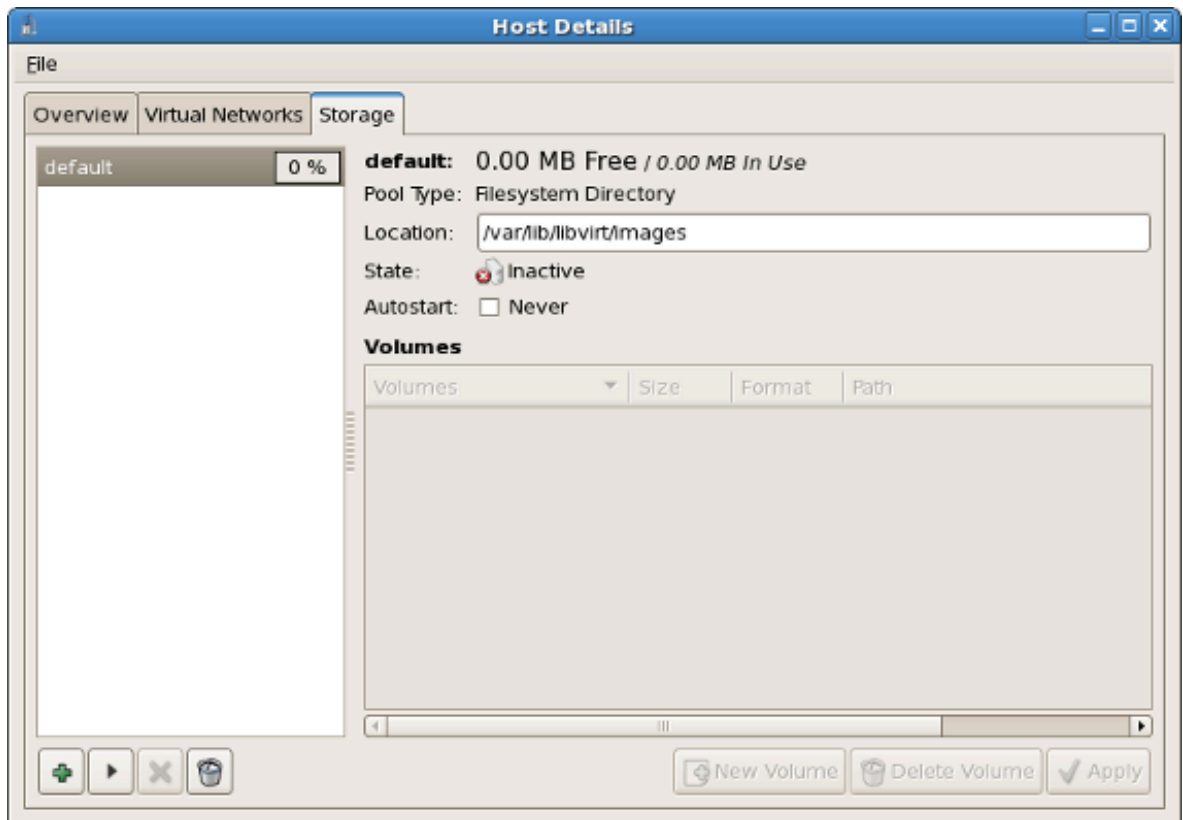
Der Virtual Machine Manager zeigt eine Liste verbundener Hosts an.



2. Fügen Sie einen Speicher-Pool mit demselben NFS zum Quell- und Ziel-Host hinzu.

Klicken Sie im **Bearbeiten**-Menü auf **Host-Details**, woraufhin das Fenster "Host-Details" erscheint.

Klicken Sie auf den Reiter **Speicher**.



3. Fügen Sie einen neuen Speicher-Pool hinzu. Klicken Sie oben links im Fenster auf die +- Schaltfläche. Es erscheint das Fenster "Neuen Speicher-Pool hinzufügen".

Geben Sie die folgenden Informationen ein:

- **Name:** Geben Sie den Namen des Speicher-Pools ein.
- **Typ:** Wählen Sie **netfs: Network Exported Directory**.



Klicken Sie auf **Weiter**.

4. Geben Sie die folgenden Informationen ein:

- **Format:** Wählen Sie den Speichertyp. Für Live-Migrationen muss dies entweder NFS oder iSCSI sein.
- **Host-Name:** Geben Sie die IP-Adresse oder den vollqualifizierten Domain-Namen des Speicher-Servers ein.

Add a New Storage Pool Step 2 of 2

Specify a storage location to be later split into virtual machine storage.

Target Path:

Format:

Host Name:

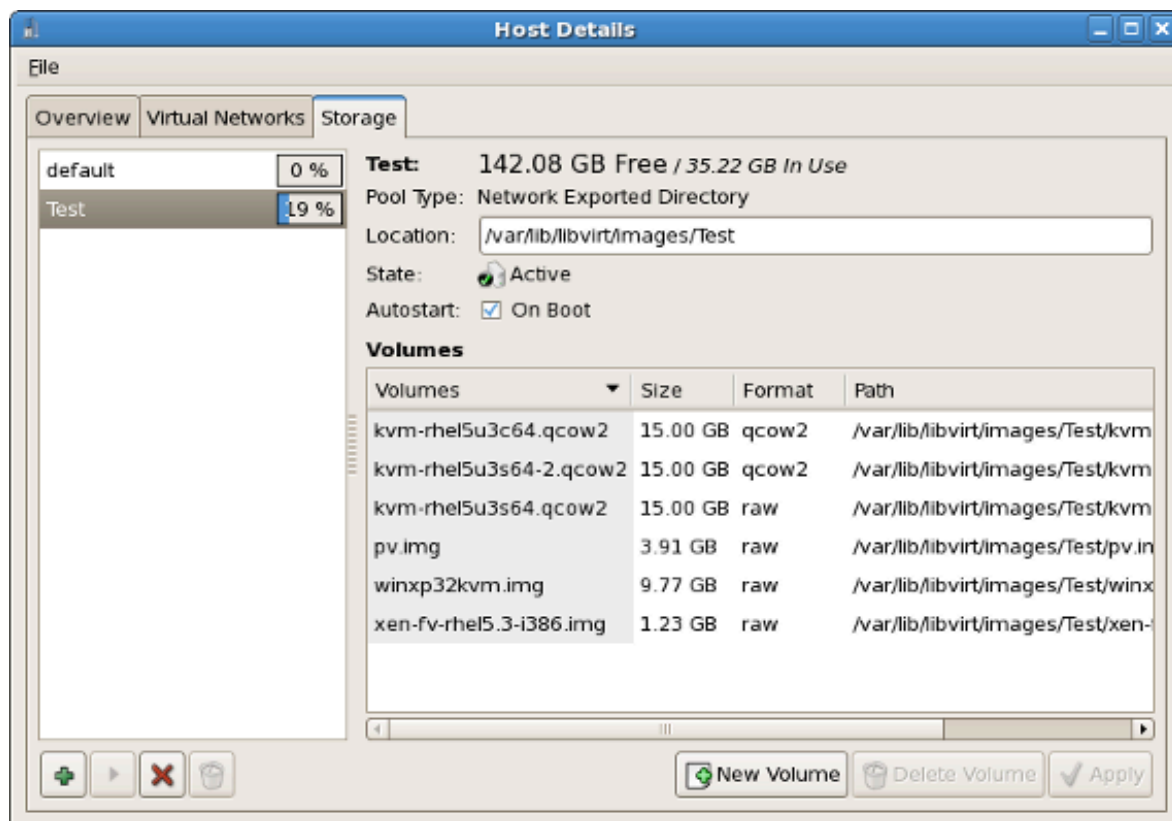
Source Path:

Build Pool:

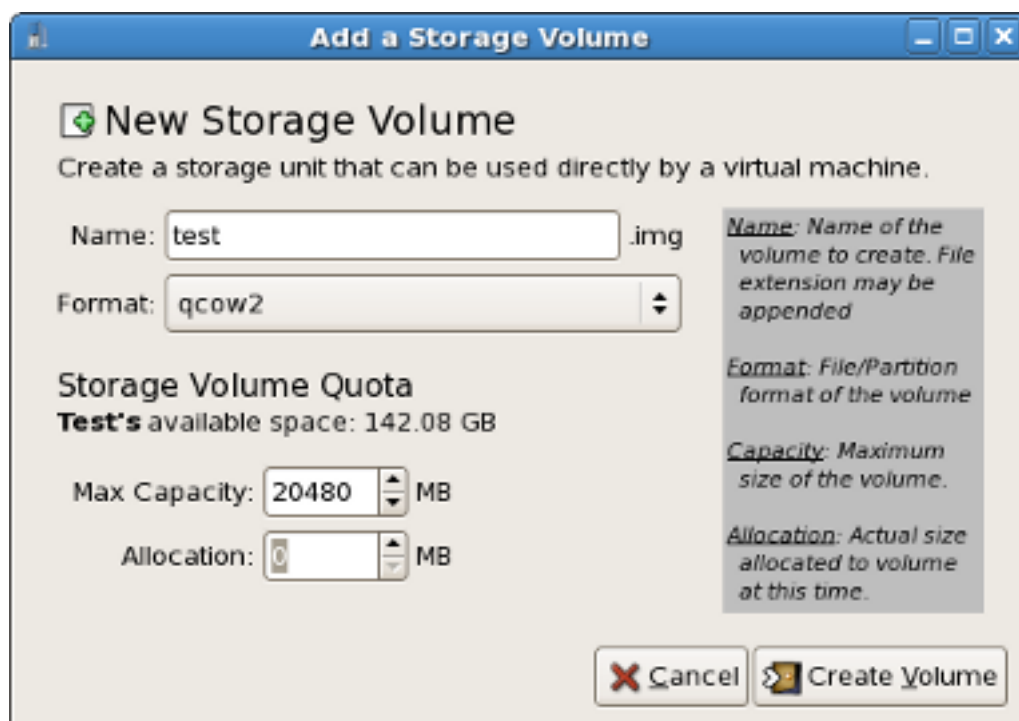
Source path: Path on the host that is being shared.

Klicken Sie auf **Abschließen**.

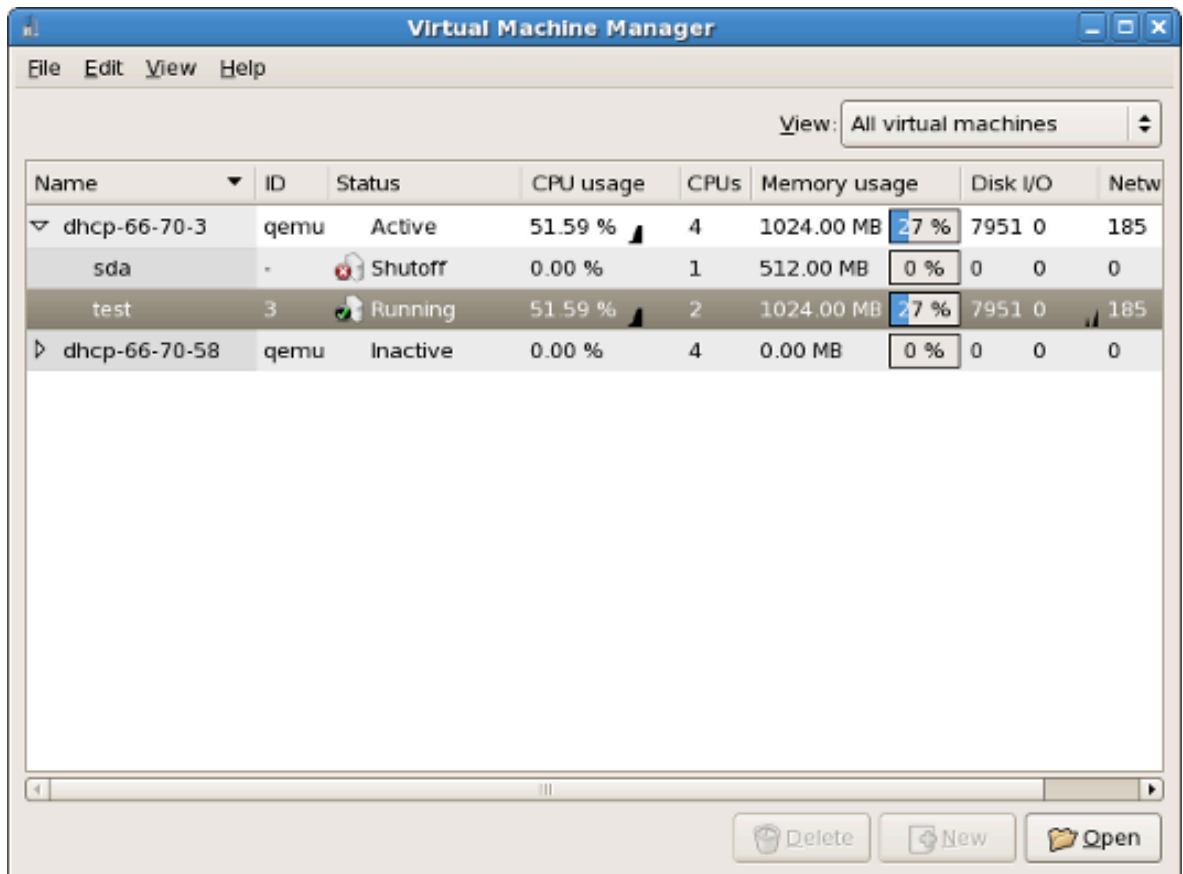
5. Erzeugen Sie einen neuen Datenträger in dem gemeinsam genutzten Speicher-Pool. Klicken Sie dazu auf **Neuer Datenträger**.



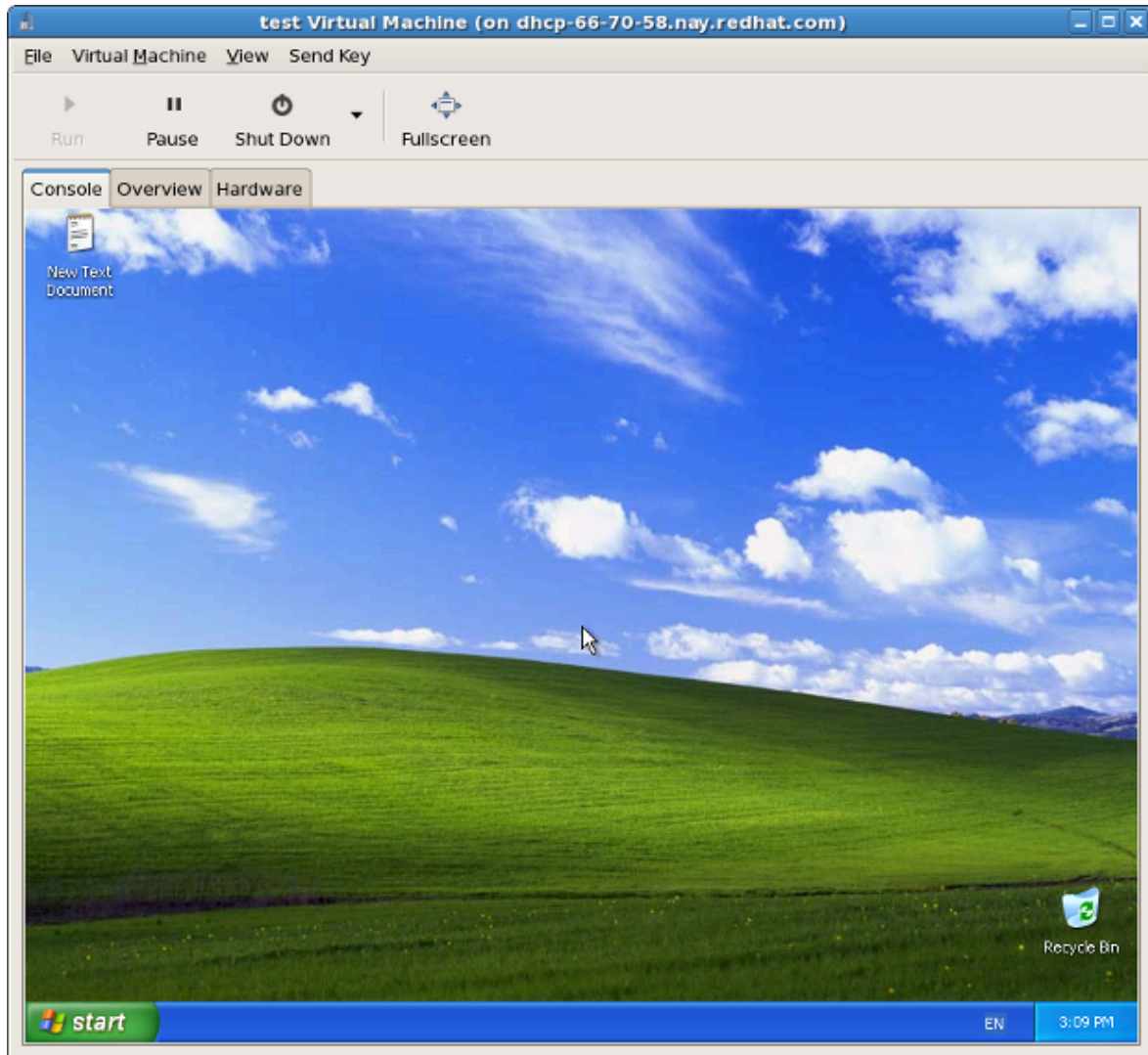
6. Geben Sie die Details ein und klicken anschließend auf **Datenträger erzeugen**.



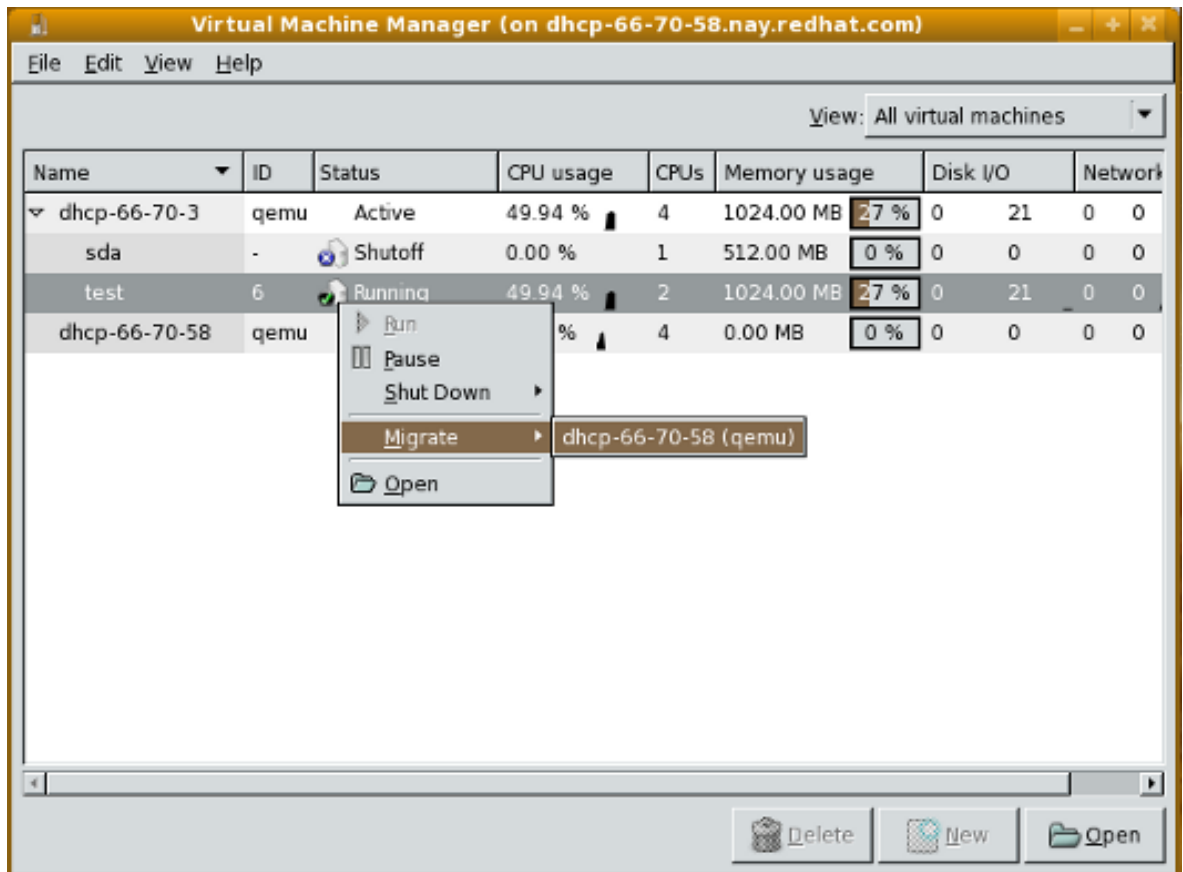
7. Erzeugen Sie eine virtuelle Maschine mit dem neuen Datenträger und starten anschließend die virtuelle Maschine.



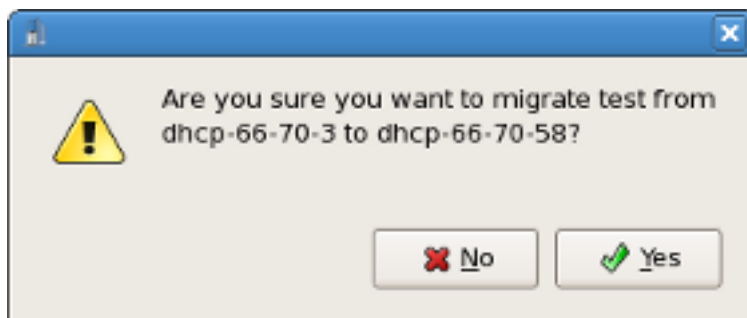
Das Fenster für die virtuelle Maschine erscheint.



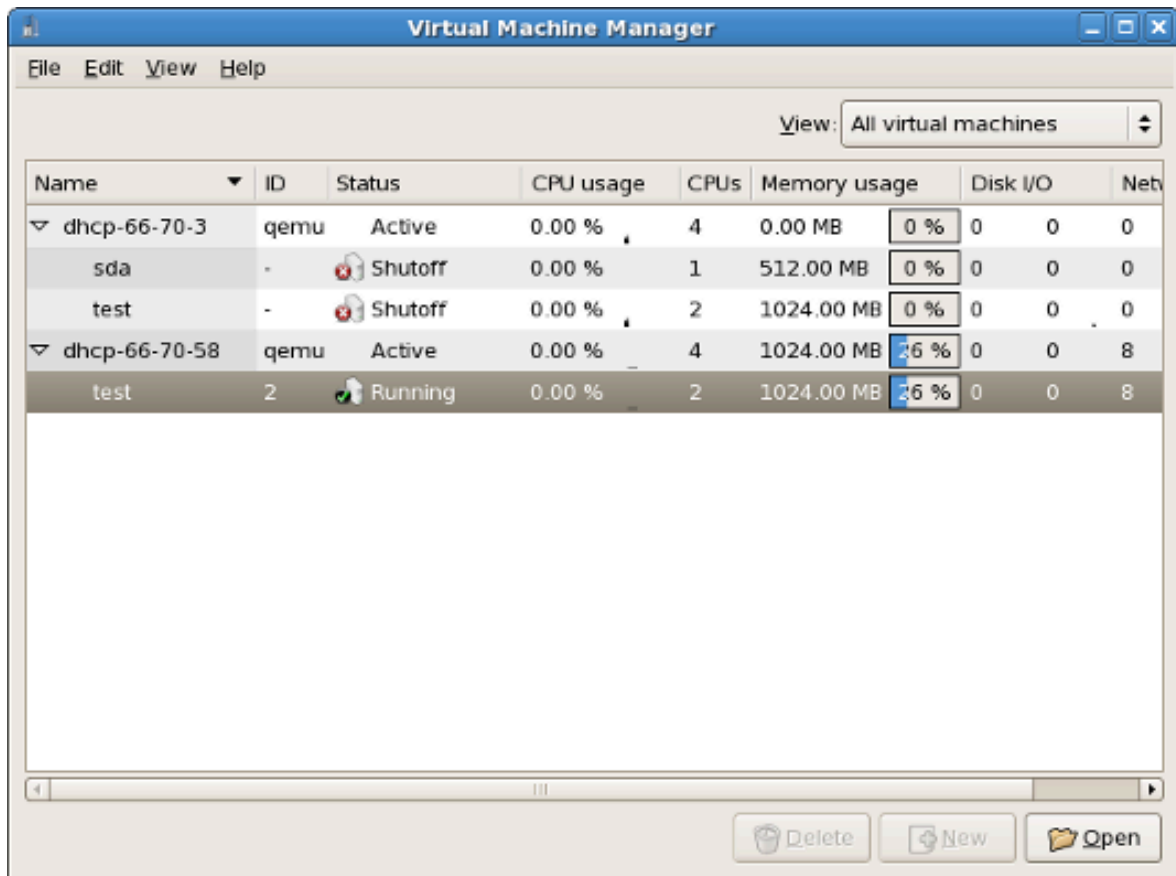
8. Klicken Sie im Fenster des Virtual Machine Managers mit der rechten Maustaste auf die virtuelle Maschine, wählen Sie dann **migrieren** aus und klicken den Migrationsort an.



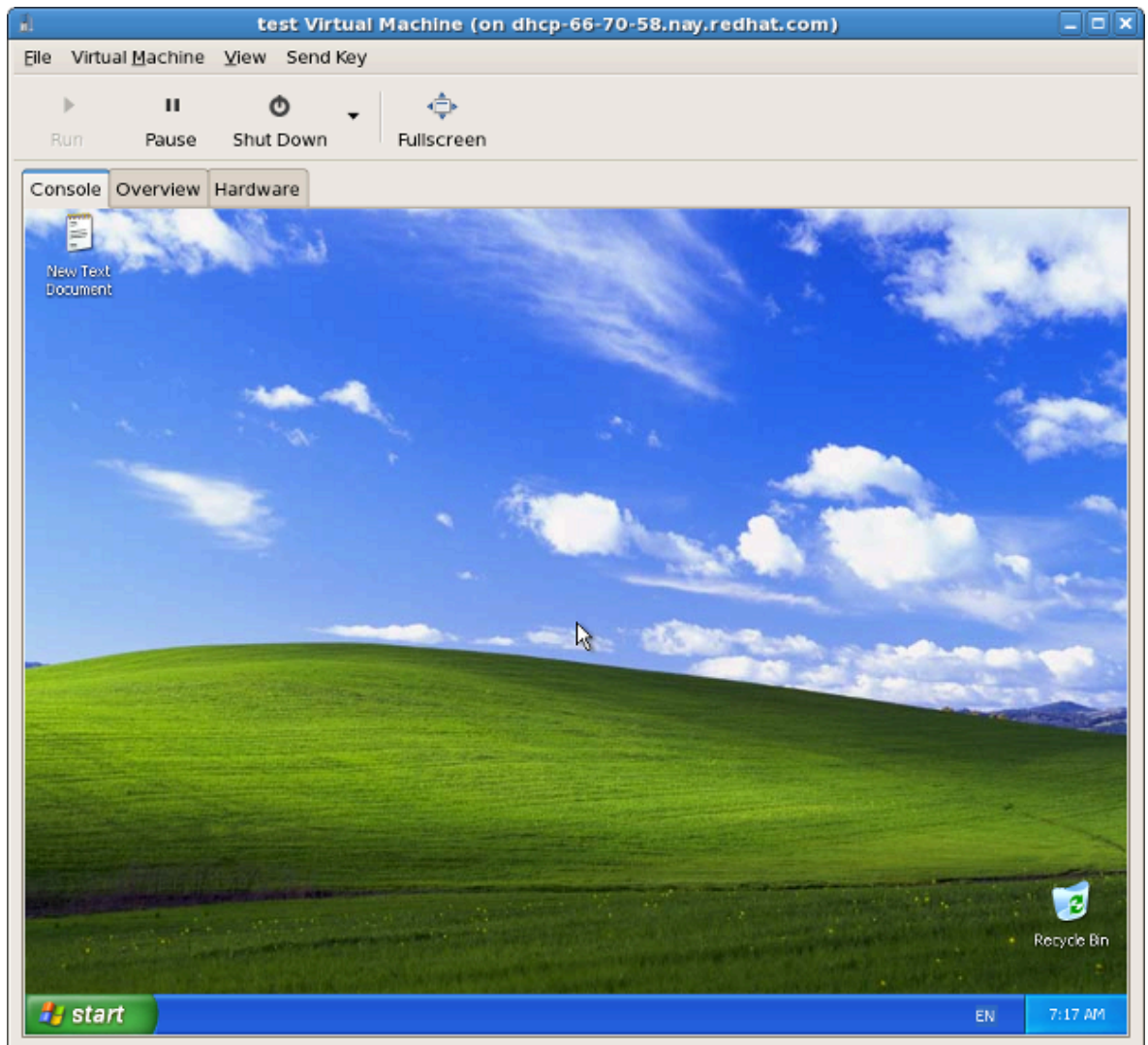
9. Klicken Sie auf **Ja**, um die Migration zu bestätigen.



Der Virtual Machine Manager zeigt die virtuelle Maschine an ihrem neuen Ort an.



Im Fenster der virtuellen Maschine wird nunmehr der neue Ort der virtuellen Maschine angezeigt.



Remote-Verwaltung virtualisierter Gäste

Dieser Abschnitt erläutert, wie Sie Ihren virtualisierten Gast mittels **ssh** oder TLS und SSL von Remote aus verwalten.

13.1. Remote-Verwaltung mit SSH

Das **ssh**-Paket stellt ein verschlüsseltes Netzwerkprotokoll zur Verfügung, mit Hilfe dessen Verwaltungsfunktionen sicher an entfernte Virtualisierungs-Server übertragen werden können. Das beschriebene Verfahren verwendet die **libvirt**-Verwaltungsverbindung, sicher getunnelt über eine **SSH**-Verbindung, um die Remote-Maschinen zu verwalten. Die Authentifizierung erfolgt mit **SSH**-Public-Key-Kryptografie und Passwörtern, die von Ihrem lokalen **SSH**-Agenten erfasst werden. Darüberhinaus wird die **VNC**-Konsole für jede virtuelle Gastmaschine über **SSH** getunnelt.

SSH wird normalerweise automatisch konfiguriert, deswegen verfügen Sie höchstwahrscheinlich bereits über ein SSH-Schlüssel-Setup und es müssen für den Zugriff auf den Verwaltungsdienst oder die **VNC**-Konsole keine zusätzlichen Firewall-Regeln hinzugefügt werden.

Sie sollten sich der Probleme bewusst sein, die der Gebrauch von **SSH** zum Fernsteuern Ihrer virtuellen Maschine mit sich bringt, so zum Beispiel:

- Sie benötigen Root-Login-Zugang zu der Remote-Maschine, um virtuelle Maschinen zu verwalten,
- der anfänglich Prozess zur Verbindungsherstellung kann ggf. langsam sein,
- es gibt keinen standardmäßigen oder einfachen Weg, um einen Benutzerschlüssel auf allen Host oder Gästen zu annullieren, und
- SSH ist bei einer großen Anzahl von Remote-Maschinen wenig praktikabel.

Konfiguration des SSH-Zugangs für **virt-manager**

In der folgenden Anleitung wird angenommen, dass Sie ganz von vorne anfangen und die **SSH**-Schlüssel noch nicht eingerichtet sind.

1. Sie benötigen ein öffentliches Schlüsselpaar auf der Maschine, auf der Sie **virt-manager** ausführen werden. Falls **ssh** bereits konfiguriert ist, können Sie diesen Befehl überspringen.

```
$ ssh-keygen -t rsa
```

2. Um eine Anmeldung von Remote aus zu erlauben, muss **virt-manager** eine Kopie des öffentlichen Schlüssels für jede Remote-Maschine haben, auf der **libvirt** läuft. Kopieren Sie die Datei **\$HOME/.ssh/id_rsa.pub** von der Maschine, die Sie für die Remote-Verwaltung verwenden wollen, mit Hilfe des **scp**-Befehls:

```
$ scp $HOME/.ssh/id_rsa.pub root@somehost:/root/key-dan.pub
```

3. Nachdem die Datei kopiert wurde, verwenden Sie den **ssh**-Befehl, um sich mit der Remote-Maschine als Root zu verbinden und fügen Sie die von Ihnen kopierte Datei in die Liste der

autorisierten Schlüssel ein. Falls der Root-Benutzer auf dem Remote-Host noch keine Liste mit autorisierten Schlüsseln hat, stellen Sie sicher, dass die Dateiberechtigungen korrekt gesetzt sind.

```
$ ssh root@somehost
# mkdir /root/.ssh
# chmod go-rwx /root/.ssh
# cat /root/key-dan.pub >> /root/.ssh/authorized_keys
# chmod go-rw /root/.ssh/authorized_keys
```

Der libvirt-Daemon (libvirtd)

Der libvirt-Daemon bietet eine Schnittstelle zur Verwaltung virtueller Maschinen. Der libvirtd-Daemon muss installiert und gestartet sein auf jedem Remote-Host, den Sie verwalten möchten.

```
$ ssh root@somehost
# chkconfig libvirtd on
# service libvirtd start
```

Nachdem libvirtd und **SSH** konfiguriert sind, sollten Sie in der Lage sein, auf Ihre virtuellen Maschinen von Remote aus zuzugreifen und diese zu verwalten. Sie sollten nunmehr auch über **VNC** auf Ihre Gäste zugreifen können.

13.2. Remote-Verwaltung über TLS und SSL

Sie können virtuelle Maschinen mit Hilfe von TLS und SSL verwalten. TLS und SSL bieten bessere Skalierbarkeit, sind aber komplizierter als SSH (siehe [Abschnitt 13.1](#), „Remote-Verwaltung mit SSH“). TLS und SSL basieren auf derselben Technologie, die auch von Webbrowsern für sichere Verbindungen verwendet wird. Die **libvirt**-Verwaltungsverbindung öffnet einen TCP-Port für eingehende Verbindungen, welcher sicher verschlüsselt und authentifiziert ist basierend auf x509-Zertifikaten. Zusätzlich wird die VNC-Konsole für jede virtuelle Gastmaschine für die Verwendung von TLS mit x509-Zertifikatsauthentifizierung eingerichtet.

Dieses Verfahren erfordert keinen Shell-Zugang auf den verwalteten Remote-Maschinen. Allerdings sind zusätzliche Firewall-Regeln erforderlich, um auf den Verwaltungsdienst oder die VNC-Konsole zuzugreifen. Eine Zertifikataufhebungsliste kann dazu benutzt werden, Benutzern den Zugang zu entziehen.

Schritte zum Einrichten von TLS/SSL-Zugang für virt-manager

In der folgenden Kurzanleitung wird angenommen, dass Sie ganz am Anfang beginnen und keinerlei Kenntnisse über TLS/SSL-Zertifikate besitzen. Falls Sie einen Zertifikats-Management-Server haben, können Sie die ersten Schritte wahrscheinlich überspringen.

libvirt-Server-Einrichtung

Mehr Informationen zur Erstellung von Zertifikaten finden Sie auf der **libvirt**-Website, <http://libvirt.org/remote.html>.

Xen-VNC-Server

Für den Xen-VNC-Server kann TLS aktiviert werden, indem die Konfigurationsdatei **/etc/xen/xend-config.sxp** bearbeitet wird. Entfernen Sie in der Konfigurationsdatei die Kommentierung des Konfigurationsparameters (**vnc-tls 1**).

Das `/etc/xen/vnc`-Verzeichnis benötigt die folgenden drei Dateien:

- `ca-cert.pem` – Das CA-Zertifikat
- `server-cert.pem` – Das von der CA signierte Server-Zertifikat
- `server-key.pem` – Den privaten Schlüssel des Servers

Dies stellt die Verschlüsselung des Datenkanals bereit. Es kann angebracht sein, von den Clients ihre eigenen x509-Zertifikate als Authentifizierung zu verlangen. Um dies zu aktivieren, entfernen Sie die Kommentierung des (`vnc-x509-verify 1`)-Parameters.

virt-manager und **virsh** Client-Einrichtung

Das Einrichten von Clients ist derzeit leicht inkonsistent. Um die **libvirt**-Verwaltungs-API über TLS zu aktivieren, müssen die CA- und Client-Zertifikate in `/etc/pki` platziert sein. Weitere Einzelheiten dazu finden Sie unter <http://libvirt.org/remote.html>.

Verwenden Sie in der **virt-manager**-Benutzeroberfläche die 'SSL/TLS'-Option als Transportmechanismus beim Verbinden mit einem Host.

Für **virsh** hat die URI das folgende Format:

- `qemu://hostname.guestname/system` für KVM.
- `xen://hostname.guestname/` für Xen.

Um SSL und TLS für VNC zu aktivieren, ist es notwendig, die CA- und Client-Zertifikate in `$HOME/.pki` zu platzieren. Es handelt sich um die folgenden drei Dateien:

- CA oder `ca-cert.pem` – Das CA-Zertifikat.
- `libvirt-vnc` oder `clientcert.pem` – Das von der CA signierte Client-Zertifikat.
- `libvirt-vnc` oder `clientkey.pem` – Der private Schlüssel des Clients.

13.3. Transportmodi

Für Remote-Verwaltung unterstützt **virsh list** die folgenden Transportmodi:

Transport Layer Security (TLS)

Transport Layer Security TLS 1.0 (SSL 3.1) authentifizierter und verschlüsselter TCP/IP-Socket, horcht in der Regel auf einen öffentlichen Port. Um dies nutzen zu können, müssen Sie Client- und Server-Zertifikate generieren. Der Standard-Port ist 16514.

UNIX-Sockets

Auf Unix-Domain-Sockets kann nur auf der lokalen Maschine zugegriffen werden. Sockets sind nicht verschlüsselt und verwenden UNIX-Berechtigungen oder SELinux zur Authentifizierung. Die Standard-Socket-Namen sind `/var/run/libvirt/libvirt-sock` und `/var/run/libvirt/libvirt-sock-ro` (für schreibgeschützte Verbindungen).

SSH

Transportiert über eine Secure Shell Protocol (SSH) Verbindung. Setzt voraus, dass Netcat (das `nc`-Paket) installiert ist. Der libvirt-Daemon (`libvirtd`) muss auf der Remote-Maschine laufen und Port

22 muss für SSH-Zugang offen sein. Sie sollten ein Verfahren zur SSH-Schlüsselverwaltung nutzen (z. B. das **ssh-agent**-Dienstprogramm), andernfalls werden Sie nach einem Passwort gefragt.

ext

Der `ext`-Parameter wird für alle externen Programme verwendet, die eine Verbindung zur Remote-Maschine herstellen können auf Wegen, die von `libvirt` nicht erfasst werden. Dies umfasst in der Regel nicht unterstützte Sicherheitsanwendungen von Drittanbietern.

tcp

Unverschlüsselter TCP/IP-Socket. Nicht empfohlen in Produktionsumgebungen. Dies ist normalerweise deaktiviert, kann jedoch von einem Administrator zu Testzwecken oder zum Gebrauch in einem vertrauenswürdigen Netzwerk aktiviert werden. Der Standard-Port ist 16509.

Sofern nichts anderes spezifiziert wurde, ist TLS der standardmäßige Transportmodus.

Remote-URIs

Ein Uniform Resource Identifier (URI) wird von `virsh` und `libvirt` verwendet, um mit einem Remote-Host zu verbinden. URIs können auch mit dem `--connect`-Parameter für den `virsh`-Befehl gebraucht werden, um einzelne Befehle oder Migrationen auf Remote-Hosts durchzuführen.

`libvirt`-URIs haben das folgende Format (Inhalte in eckigen Klammern, "[]", sind optionale Funktionen):

```
driver[+transport]://[username@][hostname][:port]/[path][?extraparameters]
```

Entweder die Transportmethode oder der Host-Name muss angegeben sein, um dies von einem lokalen URI unterscheidbar zu machen.

Beispiele für Parameter zur Remote-Verwaltung

- Verbinden mit einem entfernten Xen-Hypervisor auf einem Host namens `towada`, unter Verwendung des SSH-Transports und dem SSH-Benutzernamen `ccurran`.

```
xen+ssh://ccurran@towada/
```

- Verbinden mit einem entfernten Xen-Hypervisor auf einem Host namens `towada` unter Verwendung von TLS.

```
xen://towada/
```

- Verbinden mit einem entfernten Xen-Hypervisor auf einem Host namens `towada` unter Verwendung von TLS. Der Parameter `no_verify=1` weist `libvirt` an, das Zertifikat des Servers nicht zu überprüfen.

```
xen://towada/?no_verify=1
```

- Verbinden mit einem entfernten KVM-Hypervisor auf einem Host namens `towada` unter Verwendung von SSH.

```
qemu+ssh://towada/system
```

Beispiele zum Testen

- Verbinden mit dem lokalen KVM-Hypervisor unter Verwendung eines nicht standardmäßigen UNIX-Sockets. Der vollständige Pfad zum Unix-Socket ist in diesem Fall explizit angegeben.

```
qemu+unix:///system?socket=/opt/libvirt/run/libvirt/libvirt-sock
```

- Verbindet mit dem libvirt-Daemon unter Verwendung einer unverschlüsselten TCP/IP-Verbindung über den Server mit der IP-Adresse 10.1.1.10 auf Port 5000. Dies verwendet den Testtreiber mit Standardeinstellungen.

```
test+tcp://10.1.1.10:5000/default
```

Zusätzliche URI-Parameter

Es können zusätzliche Parameter an Remote-URIs angehängt werden. Die nachfolgende [Tabelle 13.1, „Zusätzliche URI-Parameter“](#) zeigt die zulässigen Parameter. Alle anderen Parameter werden ignoriert. Beachten Sie, dass Parameterwerte URI-spezifisch maskiert werden müssen (d. h. ein Fragezeichen (?) muss den Parametern vorangestellt werden und spezielle Zeichen werden in das URI-Format konvertiert).

Name	Transportmodus	Description	Beispielverwendung
name	alle Modi	Der Name, der an die Remote-virConnectOpen-Funktion übergeben wird. Der Name wird in der Regel gebildet, indem Transport, Host-Name, Port-Nummer, Benutzername und zusätzliche Parameter von der Remote-URI entfernt werden. In bestimmten, sehr komplexen Fällen ist es jedoch ratsam, den Namen explizit anzugeben.	name=qemu:///system
command	ssh und ext	Der externe Befehl. Für ext-Transport ist dies erforderlich. Für ssh ist der Standard ssh. Der PATH wird nach dem Befehl durchsucht.	command=/opt/openssh/bin/ssh

Name	Transportmodus	Description	Beispielverwendung
socket	unix und ssh	Der Pfad zum UNIX-Domain-Socket, was den Standard außer Kraft setzt. Für ssh-Transport wird dies an den Remote-Netcat-Befehl übergeben (siehe netcat).	socket=/opt/libvirt/run/libvirt/libvirt-sock
netcat	ssh	Der Name des Netcat-Befehls auf der entfernten Maschine. Der Standard ist nc. Für ssh-Transport konstruiert libvirt einen ssh-Befehl nach folgendem Schema: <code>command -p port [-l username] hostname netcat -U socket</code> . Dabei kann port, username, hostname als Teil der Remote-URI spezifiziert werden und command, netcat und socket stammen von zusätzlichen Parametern (oder vernünftigen Standards).	netcat=/opt/netcat/bin/nc
no_verify	tls	Falls auf einen anderen Wert als Null eingestellt, deaktiviert dies die Überprüfung des Server-Zertifikats durch den Client. Beachten Sie, dass Sie zum Deaktivieren der Überprüfung des Client-Zertifikats oder der IP-Adresse durch den Server die libvirtd-Konfiguration ändern müssen.	no_verify=1
no_tty	ssh	Falls auf einen anderen Wert als Null eingestellt, fragt ssh nicht nach einem Passwort, falls es sich nicht automatisch	no_tty=1

Name	Transportmodus	Description	Beispielverwendung
		bei der Remote-Maschine anmelden kann (zum Gebrauch von ssh-agent o. ä.). Verwenden Sie dies, wenn Sie keinen Zugriff auf ein Terminal haben, z. B. in grafischen Programmen, die libvirt verwenden.	

Tabelle 13.1. Zusätzliche URI-Parameter

Teil IV. Referenzhandbuch zur Virtualisierung

Referenz zu Befehlen, Systemwerkzeugen, Anwendungen und zusätzlichen Systemen für die Virtualisierung

Diese Kapitel bieten eine detaillierte Beschreibung der Virtualisierungsbefehle, Systemwerkzeuge und Anwendungen, die in Fedora enthalten sind. Benutzer, die Informationen zu fortgeschrittenen Funktionen und anderen Features benötigen, sollten diese Kapitel lesen.

Virtualisierungs-Tools

Nachfolgend sehen Sie eine Liste von Tools zur Verwaltung der Virtualisierung und zur Suche und Bereinigung von Programmfehlern (Debugging) sowie Netzwerk-Tools, die nützlich sind für Systeme, auf denen Xen läuft.

Systemadministrations-Tools

- **vmstat**
- **iostat**
- **lsof**

```
# lsof -i :5900
xen-vncfb 10635 root 5u IPv4 218738 TCP
grumble.boston.redhat.com:5900 (LISTEN)
```

- **qemu-img**

Fortgeschrittene Tools zur Suche und Bereinigung von Programmfehlern

- **systemTap**
- **crash**
- **xen-gdbserver**
- **sysrq**
- **sysrq t**
- **sysrq w**
- **sysrq c**

Netzwerk-Tools

brctl

- ```
brctl show
bridge name bridge id STP enabled interfaces
xenbr0 8000.fefffffffffff no vif13.0
 pdummy0
 vif0.0
```
- ```
# brctl showmacs xenbr0
port no  mac addr          is local?  aging timer
  1      fe:ff:ff:ff:ff:ff  yes        0.00
```
- ```
brctl showstp xenbr0
xenbr0
bridge id 8000.fefffffffffff
designated root 8000.fefffffffffff
```

|                       |                     |                      |
|-----------------------|---------------------|----------------------|
| root port             | 0                   | path cost            |
| 0                     |                     |                      |
| max age               | 20.00               | bridge max age       |
| 20.00                 |                     |                      |
| hello time            | 2.00                | bridge hello time    |
| 2.00                  |                     |                      |
| forward delay         | 0.00                | bridge forward delay |
| 0.00                  |                     |                      |
| aging time            | 300.01              |                      |
| hello timer           | 1.43                | tcn timer            |
| 0.00                  |                     |                      |
| topology change timer | 0.00                | gc timer             |
| 0.02                  |                     |                      |
| flags                 |                     |                      |
| vif13.0 (3)           |                     |                      |
| port id               | 8003                | state                |
| forwarding            |                     |                      |
| designated root       | 8000.feffffffffffff | path cost            |
| 100                   |                     |                      |
| designated bridge     | 8000.feffffffffffff | message age timer    |
| 0.00                  |                     |                      |
| designated port       | 8003                | forward delay timer  |
| 0.00                  |                     |                      |
| designated cost       | 0                   | hold timer           |
| 0.43                  |                     |                      |
| flags                 |                     |                      |
| pdummy0 (2)           |                     |                      |
| port id               | 8002                | state                |
| forwarding            |                     |                      |
| designated root       | 8000.feffffffffffff | path cost            |
| 100                   |                     |                      |
| designated bridge     | 8000.feffffffffffff | message age timer    |
| 0.00                  |                     |                      |
| designated port       | 8002                | forward delay timer  |
| 0.00                  |                     |                      |
| designated cost       | 0                   | hold timer           |
| 0.43                  |                     |                      |
| flags                 |                     |                      |
| vif0.0 (1)            |                     |                      |
| port id               | 8001                | state                |
| forwarding            |                     |                      |
| designated root       | 8000.feffffffffffff | path cost            |
| 100                   |                     |                      |
| designated bridge     | 8000.feffffffffffff | message age timer    |
| 0.00                  |                     |                      |
| designated port       | 8001                | forward delay timer  |
| 0.00                  |                     |                      |

---

```
designated cost 0 hold timer
0.43
flags
```

- **ifconfig**
- **tcpdump**

#### KVM-Tools

- **ps**
- **pstree**
- **top**
- **kvmtrace**
- **kvm\_stat**

#### Xen-Tools

- **xentop**
- **xm dmesg**
- **xm log**

---

---

# Das Verwalten von Gästen mit virsh

**virsh** ist ein Befehlszeilen-Tool zur Verwaltung der Gäste und des Hypervisors.

Das **virsh**-Tool setzt an der **libvirt**-Management-API an und fungiert als Alternative zum **xm**-Befehl und dem grafischen Gäste-Manager (**virt-manager**). Unprivilegierte Benutzer können **virsh** in schreibgeschütztem Modus nutzen. Sie können **virsh** dazu verwenden, Skripte für die Gastmaschinen auszuführen.

## Kurzanleitung zum virsh-Befehl

Die folgende Tabelle gibt eine Kurzanleitung für alle Befehlszeilenoptionen.

| Befehl          | Description                                                                         |
|-----------------|-------------------------------------------------------------------------------------|
| <b>help</b>     | Zeigt grundlegende Hilfe-Informationen.                                             |
| <b>list</b>     | Listet alle Gäste auf.                                                              |
| <b>dumpxml</b>  | Gibt die XML-Konfigurationsdatei für den Gast aus.                                  |
| <b>create</b>   | Erzeugt einen Gast anhand einer XML-Konfigurationsdatei und startet den neuen Gast. |
| <b>start</b>    | Startet einen inaktiven Gast.                                                       |
| <b>destroy</b>  | Zwingt einen Gast zum Beenden.                                                      |
| <b>define</b>   | Gibt eine XML-Konfigurationsdatei für einen Gast aus.                               |
| <b>domid</b>    | Zeigt die Gast-ID an.                                                               |
| <b>domuuid</b>  | Zeigt die Gast-UUID an.                                                             |
| <b>dominfo</b>  | Zeigt Gastinformationen.                                                            |
| <b>domname</b>  | Zeigt den Gastnamen.                                                                |
| <b>domstate</b> | Zeigt den Status eines Gasts an.                                                    |
| <b>quit</b>     | Beendet das interaktive Terminal.                                                   |
| <b>reboot</b>   | Startet einen Gast neu.                                                             |
| <b>restore</b>  | Stellt einen zuvor in einer Datei gespeicherten Gast wieder her.                    |
| <b>resume</b>   | Setzt einen angehaltenen Gast fort.                                                 |
| <b>save</b>     | Speichert den aktuellen Zustand eines Gasts in einer Datei.                         |
| <b>shutdown</b> | Führt einen Gast herunter.                                                          |
| <b>suspend</b>  | Hält einen Gast an.                                                                 |
| <b>undefine</b> | Löscht alle zu einem Gast gehörigen Dateien.                                        |
| <b>migrate</b>  | Migriert einen Gast auf einen anderen Host.                                         |

Tabelle 15.1. Befehle der Gästeverwaltung

Benutzen Sie die folgenden **virsh**-Befehle zur Verwaltung von Gast- und Hypervisor-Ressourcen:

| Befehl                  | Description                                                                                                     |
|-------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>setmem</b>           | Legt den zugewiesenen Speicher für einen Gast fest.                                                             |
| <b>setmaxmem</b>        | Legt die Höchstgrenze an Speicher für den Hypervisor fest.                                                      |
| <b>setvcpus</b>         | Ändert die Anzahl virtueller CPUs, die einem Gast zugewiesen sind.                                              |
| <b>vcpuinfo</b>         | Zeigt Informationen zur virtuellen CPU für einen Gast.                                                          |
| <b>vcpupin</b>          | Steuert die Affinität einer virtuellen CPU für einen Gast.                                                      |
| <b>domblkstat</b>       | Zeigt Blockgerätstatistiken für einen laufenden Gast.                                                           |
| <b>domifstat</b>        | Zeigt Netzwerkschnittstellenstatistiken für einen laufenden Gast.                                               |
| <b>attach-device</b>    | Verknüpft ein Gerät mit einem Gast mittels einer Gerätedefinition in einer XML-Datei.                           |
| <b>attach-disk</b>      | Verknüpft eine neue Festplatte mit einem Gast.                                                                  |
| <b>attach-interface</b> | Verknüpft eine neue Netzwerkschnittstelle mit einem Gast.                                                       |
| <b>detach-device</b>    | Löst verknüpftes Gerät von einem Gast, nimmt dieselben XML-Beschreibungen wie der Befehl <b>attach-device</b> . |
| <b>detach-disk</b>      | Löst verknüpfte Festplatte von einem Gast.                                                                      |
| <b>detach-interface</b> | Löst verknüpfte Netzwerkschnittstelle von einem Gast.                                                           |

Tabelle 15.2. Optionen zur Ressourcenverwaltung

Dies sind sonstige **virsh**-Optionen:

| Befehl          | Description                                 |
|-----------------|---------------------------------------------|
| <b>version</b>  | Zeigt die Version von <b>virsh</b> .        |
| <b>nodeinfo</b> | Gibt Informationen über den Hypervisor aus. |

Tabelle 15.3. Sonstige Optionen

### Verbinden mit dem Hypervisor

Verbinden Sie mit einer Hypervisor-Sitzung mittels **virsh**:

```
virsh connect {hostname OR URL}
```

Wobei **<name>** der Name der Maschine des Hypervisors ist. Um eine schreibgeschützte Verbindung herzustellen, hängen Sie an den oben aufgeführten Befehl **-readonly** an.

### Erstellen eines XML-Speicherauszugs einer virtuellen Maschine (Konfigurationsdatei)

Geben Sie die XML-Konfigurationsdatei eines Gasts mit **virsh** aus:

---

```
virsh dumpxml {domain-id, domain-name or domain-uuid}
```

Dieser Befehl gibt die XML-Konfigurationsdatei des Gasts über die Standardausgabe (**stdout**) aus. Sie können die Daten speichern, indem Sie die Ausgabe in eine Datei umleiten. Sehen Sie hier ein Beispiel für die Umleitung der Ausgabe in eine Datei namens *guest.xml*:

```
virsh dumpxml GuestID > guest.xml
```

Die Datei **guest.xml** kann dazu verwendet werden, einen Gast wiederherzustellen (siehe [Bearbeiten von Gastkonfigurationsdateien](#)). Sie können diese XML-Konfigurationsdatei bearbeiten, um zusätzliche Geräte zu konfigurieren oder Gäste hinzuzufügen. Werfen Sie einen Blick auf [Abschnitt 18.1, „Benutzung von XML-Konfigurationsdateien mit virsh“](#) für mehr Informationen über die Modifizierung von **virsh dumpxml**-Dateien.

Ein Beispiel einer **virsh dumpxml**-Ausgabe:

```
virsh dumpxml r5b2-mysQL01
<domain type='xen' id='13'>
 <name>r5b2-mysQL01</name>
 <uuid>4a4c59a7ee3fc78196e4288f2862f011</uuid>
 <bootloader>/usr/bin/pygrub</bootloader>
 <os>
 <type>linux</type>
 <kernel>/var/lib/libvirt/vmlinuz.2dgnU_</kernel>
 <initrd>/var/lib/libvirt/initrd.UQafMw</initrd>
 <cmdline>ro root=/dev/VolGroup00/LogVol00 rhgb quiet</cmdline>
 </os>
 <memory>512000</memory>
 <vcpu>1</vcpu>
 <on_poweroff>destroy</on_poweroff>
 <on_reboot>restart</on_reboot>
 <on_crash>restart</on_crash>
 <devices>
 <interface type='bridge'>
 <source bridge='xenbr0' />
 <mac address='00:16:3e:49:1d:11' />
 <script path='vif-bridge' />
 </interface>
 <graphics type='vnc' port='5900' />
 <console tty='/dev/pts/4' />
 </devices>
</domain>
```

## Erzeugen eines Gasts mit Hilfe einer Konfigurationsdatei

Gäste können aus XML-Konfigurationsdateien erzeugt werden. Sie können vorhandenes XML von einem bereits erstellten Gast kopieren oder die Option **dumpxml** verwenden (siehe **dumpxml** option (refer to [Erstellen eines XML-Speicherauszugs einer virtuellen Maschine \(Konfigurationsdatei\)](#))). Um einen Gast mit **virsh** aus einer XML-Datei zu erzeugen:

```
virsh create configuration_file.xml
```

### Bearbeiten von Gastkonfigurationsdateien

Statt die Option **dumpxml** zu verwenden (siehe [Erstellen eines XML-Speicherauszugs einer virtuellen Maschine \(Konfigurationsdatei\)](#)), können Gäste auch bearbeitet werden, während diese laufen oder offline sind. Diese Möglichkeit bietet Ihnen der **virsh edit**-Befehl. Um zum Beispiel einen Gast namens *softwaretesting* zu bearbeiten:

```
virsh edit softwaretesting
```

Dadurch wird ein Texteditor geöffnet. Der Standardtexteditor ist der **\$EDITOR**-Shell-Parameter (auf **vi** voreingestellt).

### Anhalten eines Gasts

Um einen Gast mit **virsh** anzuhalten:

```
virsh suspend {domain-id, domain-name or domain-uuid}
```

Wenn ein Gast angehalten ist, wird weiterhin Arbeitsspeicher des Systems verbraucht, jedoch keine Prozessorressourcen. Es finden keine Platten- oder Netzwerk-I/O-Vorgänge statt, solange der Gast angehalten ist. Diese Operation wird sofort umgesetzt, und der Gast kann mit Hilfe der Option **resume** ([Fortsetzen eines Gastes](#)) wieder fortgesetzt werden.

### Fortsetzen eines Gastes

Um einen angehaltenen Gast mit **virsh** wieder fortzusetzen, verwenden Sie die **resume**-Option:

```
virsh resume {domain-id, domain-name or domain-uuid}
```

Diese Operation wird sofort umgesetzt und die Gastparameter werden für **suspend**- und **resume**-Operationen beibehalten.

### Speichern eines Gasts

Speichern Sie den aktuellen Status eines Gasts mit Hilfe des **virsh**-Befehls in einer Datei:

```
virsh save {domain-name, domain-id or domain-uuid} filename
```

Dies hält den von Ihnen angegebenen Gast an und speichert die Daten in eine Datei. Das kann eine gewisse Zeit in Anspruch nehmen, je nachdem, wieviel Speicher von Ihrem Gast in Gebrauch ist. Sie können den Status Ihres Gasts mit der Option **restore** ([Wiederherstellen eines Gasts](#)) wiederherstellen. Die "save"-Option ist ähnlich der "pause"-Option, unterscheidet sich jedoch insofern, dass zusätzlich zum Anhalten der aktuelle Zustand des Gasts gespeichert wird.

### Wiederherstellen eines Gasts

Sie können einen Gast mit **virsh** wiederherstellen, den Sie zuvor mit der **virsh save**-Option gespeichert haben ([Speichern eines Gasts](#)):



---

```
virsh restore filename
```

Dies startet die gespeicherte virtuelle Maschine, was ggf. etwas Zeit benötigt. Der Name der virtuellen Maschine, sowie deren UUID bleiben erhalten, werden aber einer neuen ID zugewiesen.

### Herunterfahren eines Gasts

Herunterfahren eines Gasts mit dem Befehl **virsh**:

```
virsh shutdown {domain-id, domain-name or domain-uuid}
```

Sie können das Verhalten des neustartenden Gasts kontrollieren, indem Sie den Parameter **on\_shutdown** der Gastkonfigurationsdatei ändern.

### Neustarten eines Gasts

Neustarten eines Gasts mit dem Befehl **virsh**:

```
#virsh reboot {domain-id, domain-name or domain-uuid}
```

Sie können das Verhalten des neustartenden Gasts kontrollieren, indem Sie den Parameter **on\_reboot** der Gastkonfigurationsdatei ändern.

### Abbrechen eines Gasts

Abbrechen eines Gasts mit dem Befehl **virsh**:

```
virsh destroy {domain-id, domain-name or domain-uuid}
```

Dieser Befehl veranlasst ein abruptes Beenden und stoppt den angegebenen Gast. **virsh destroy** kann dabei möglicherweise das Gastdateisystem beschädigen. Verwenden Sie die **destroy**-Option nur, wenn der Gast nicht mehr reagiert. Für paravirtualisierte Gäste sollten Sie stattdessen die Option **shutdown** verwenden ([Herunterfahren eines Gasts](#)).

### Abrufen der Domain-ID eines Gasts

Um die Domain-ID eines Gasts zu erhalten:

```
virsh domid {domain-name or domain-uuid}
```

### Abrufen des Domain-Namens eines Gasts

Um den Domain-Namen eines Gasts zu erhalten:

```
virsh domname {domain-id or domain-uuid}
```

### Abrufen der UUID eines Gasts

Um die Universally Unique Identifier (UUID) eines Gasts zu erhalten:

```
virsh domuuid {domain-id or domain-name}
```

Ein Beispiel für eine **virsh domuuid**-Ausgabe:

```
virsh domuuid r5b2-mysql01
4a4c59a7-ee3f-c781-96e4-288f2862f011
```

### Anzeigen von Gastinformationen

Wenn Sie **virsh** zusammen mit der Domain-ID, dem Domain-Namen oder der UUID des Gasts verwenden, können Sie Informationen über den angegebenen Gast anzeigen:

```
virsh dominfo {domain-id, domain-name or domain-uuid}
```

Sehen Sie ein Beispiel für eine **virsh dominfo**-Ausgabe:

```
virsh dominfo r5b2-mysql01
id: 13
name: r5b2-mysql01
uuid: 4a4c59a7-ee3f-c781-96e4-288f2862f011
os type: linux
state: blocked
cpu(s): 1
cpu time: 11.0s
max memory: 512000 kb
used memory: 512000 kb
```

### Anzeigen von Host-Informationen

Um Informationen über den Host anzuzeigen:

```
virsh nodeinfo
```

Ein Beispiel für eine **virsh nodeinfo**-Ausgabe:

```
virsh nodeinfo
CPU model x86_64
CPU (s) 8
CPU frequency 2895 Mhz
CPU socket(s) 2
Core(s) per socket 2
Threads per core: 2
Numa cell(s) 1
Memory size: 1046528 kb
```

Dargestellt werden die Knoteninformationen und die Maschinen, die den Virtualisierungsprozess unterstützen.

---

## Anzeigen der Gäste

Um eine Gästeliste samt jeweiligem aktuellen Status mit **virsh** anzuzeigen:

```
virsh list
```

Andere verfügbare Optionen sind u. a.:

die Option **--inactive**, um inaktive Gäste aufzulisten (also Gäste, die zwar definiert wurden, zur Zeit jedoch nicht aktiv sind, und

die Option **--all**, um alle Gäste aufzulisten. Zum Beispiel:

```
virsh list --all
 Id Name State

 0 Domain-0 running
 1 Domain202 paused
 2 Domain010 inactive
 3 Domain9600 crashed
```

Die Ausgabe von **virsh list** kann kategorisiert werden als eine von sechs möglichen Stati (nachfolgend erläutert):

- Der Status **running** bezieht sich auf Gäste, die derzeit auf einer CPU aktiv sind.
- Gäste mit dem Status **blocked** sind blockiert und werden nicht ausgeführt bzw. können nicht ausgeführt werden. Dies können Gäste sein, die auf eine I/O warten (traditionell der "wait"-Status) oder die sich im Ruhezustand befinden.
- Der **paused**-Status bezieht sich auf Gäste, die angehalten wurden. Das ist der Fall, wenn ein Administrator die Schaltfläche **pause** im **virt-manager** klickt oder **xm pause** bzw. **virsh suspend** ausführt. Wenn ein Gast angehalten ist, verbraucht er weiterhin Arbeitsspeicher und andere Ressourcen, nimmt jedoch nicht am Scheduling teil und erhält keine CPU-Ressourcen vom Hypervisor.
- Der Status **shutdown** ist für Gäste, die gerade dabei sind herunterzufahren. Dem Gast wurde das Signal zum Herunterfahren gesendet und sollte im Begriff sein, seine Operationen zu beenden. Dies funktioniert ggf. nicht mit allen Betriebssystemen, denn einige Betriebssysteme reagieren nicht auf diese Signale.
- Gäste mit dem Status **dying** sind "am Sterben", d. h. der Gast wurde nicht vollständig heruntergefahren oder ist abgestürzt.
- Gäste mit dem Status **crashed** schlugen bei der Ausführung fehl und laufen nicht mehr. Dieser Status kann nur auftreten, wenn der Gast konfiguriert wurde, nach einem Absturz nicht neu zu starten.

## Anzeigen von Informationen zur virtuellen CPU

Um Information einer virtuellen CPU für einen Gast mittels **virsh** anzuzeigen:

```
virsh vcpuinfo {domain-id, domain-name or domain-uuid}
```

Hier ein Beispiel für eine `virsh vcpuinfo`-Ausgabe:

```
virsh vcpuinfo r5b2-mysql01
VCPU: 0
CPU: 0
State: blocked
CPU time: 0.0s
CPU Affinity: yy
```

### Konfigurieren der Affinität einer virtuellen CPU

Um die Affinität von virtuellen CPUs mit physischen CPUs zu konfigurieren:

```
virsh vcpupin {domain-id, domain-name or domain-uuid} vcpu, cpulist
```

Wobei **vcpu** die Nummer der virtuellen VCPU und **cpulist** die Anzahl der physischen CPUs angibt.

### Konfigurieren der Anzahl virtueller CPUs

Um mit `virsh` die Anzahl der CPUs zu ändern, die einem Gast zugewiesen sind:

```
virsh setvcpus {domain-name, domain-id or domain-uuid} count
```

Der neue *count*-Wert darf die Anzahl, die bei der Erstellung des Gasts festgelegt wurde, nicht überschreiten.

### Konfigurieren der Speicherzuweisung

Um die Speicherzuweisung für einen Gast mit `virsh` zu ändern:

```
virsh setmem {domain-id or domain-name} count
```

Sie müssen *count* in Kilobytes angeben. Der neue Wert darf die Menge, die Sie bei der Erstellung des Gasts festgelegt haben, nicht überschreiten. Werte kleiner als 64 MB funktionieren mit den meisten Betriebssystemen wahrscheinlich nicht. Ein höherer maximaler Speicherwert beeinflusst einen aktiven Gast nicht, es sei denn, der neue Wert ist niedriger, was zu einem Verkleinern der Speicherbelegung führen würde.

### Anzeigen von Blockgeräteinformationen für einen Gast

Verwenden Sie `virsh domblkstat`, um Blockgerätstatistiken für einen laufenden Gast anzuzeigen.

```
virsh domblkstat GuestName block-device
```

### Anzeigen von Netzwerkgeräteinformationen für einen Gast

Verwenden Sie `virsh domifstat`, um Netzwerkgerätstatistiken für einen laufenden Gast anzuzeigen.

```
virsh domifstat GuestName interface-device
```

---

## Gästeverwaltung mit virsh

Ein Gast kann mit Hilfe des **virsh**-Befehls auf einen anderen Host migriert werden. Migrieren Sie eine Domain auf einen anderen Host. Fügen Sie `--live` für eine Live-Migration hinzu. Der **migrate**-Befehl akzeptiert Parameter im folgenden Format:

```
virsh migrate --live GuestName DestinationURL
```

Der `--live`-Parameter ist optional. Fügen Sie den `--live`-Parameter für Live-Migrationen hinzu.

The *GuestName* parameter represents the name of the guest which you want to migrate.

The *DestinationURL* parameter is the URL or hostname of the destination system. The destination system must run the same version of Fedora, be using the same hypervisor and have **libvirt** running.

Once the command is entered you will be prompted for the root password of the destination system.

## Verwalten virtueller Netzwerke

Dieser Abschnitt behandelt die Verwaltung virtueller Netzwerke mit **virsh**. Um virtuelle Netzwerke aufzulisten:

```
virsh net-list
```

Dieser Befehl generiert eine Ausgabe ähnlich der folgenden:

```
virsh net-list
Name State Autostart

default active yes
vnet1 active yes
vnet2 active yes
```

Um Netzwerkinformationen für ein spezielles virtuelles Netzwerk anzusehen:

```
virsh net-dumpxml NetworkName
```

So werden Informationen über ein angegebenes virtuelles Netzwerk im XML-Format angezeigt:

```
virsh net-dumpxml vnet1
<network>
 <name>vnet1</name>
 <uuid>98361b46-1581-acb7-1643-85a412626e70</uuid>
 <forward dev='eth0' />
 <bridge name='vnet0' stp='on' forwardDelay='0' />
 <ip address='192.168.100.1' netmask='255.255.255.0'>
 <dhcp>
 <range start='192.168.100.128' end='192.168.100.254' />
 </dhcp>
 </ip>
```

```
</network>
```

Andere **virsh**-Befehle zur Verwaltung virtueller Netzwerke sind u. a.:

- **virsh net-autostart *network-name*** — Startet automatisch ein Netzwerk spezifiziert als *network-name*.
- **virsh net-create *XMLfile*** — Generiert und startet ein neues Netzwerk unter Verwendung einer vorhandenen XML-Datei.
- **virsh net-define *XMLfile*** — Generiert ein neues Netzwerkgerät von einer vorhandenen XML-Datei, ohne dieses zu starten.
- **virsh net-destroy *network-name*** — Zerstört ein Netzwerk spezifiziert als *network-name*.
- **virsh net-name *networkUUID*** — Konvertiert eine angegebene *networkUUID* in einen Netzwerknamen.
- **virsh net-uuid *network-name*** — Konvertiert einen spezifizierten *network-name* in eine Netzwerk-UUID.
- **virsh net-start *nameOfInactiveNetwork*** — Startet ein inaktives Netzwerk.
- **virsh net-undefine *nameOfInactiveNetwork*** — Löscht die Definition von einem inaktiven Netzwerk.

---

# Das Verwalten von Gästen mit dem Virtual Machine Manager (virt-manager)

Dieser Abschnitt beschreibt die Fenster, Dialogkästen und verschiedenen GUI-Kontrollmöglichkeiten des Virtual Machine Managers (**virt-manager**).

Der **virt-manager** bietet eine grafische Ansicht der Hypervisoren und Gäste auf Ihrem System und auf entfernten Maschinen. Sie können mit Hilfe von **virt-manager** sowohl paravirtualisierte als auch voll virtualisierte Gäste definieren. **virt-manager** kann zudem Aufgaben zur Verwaltung der Virtualisierung durchführen, u. a.:

- Speicher zuweisen,
- virtuelle CPUs zuweisen,
- Leistung im Betrieb überwachen,
- speichern und wiederherstellen, anhalten und fortsetzen, herunterfahren und starten von virtualisierten Gästen,
- verbindet mit den Text- und grafischen Konsolen, und
- Live- und Offline-Migrationen.

## 16.1. Das Fenster "Verbindung öffnen"

Zu Beginn erscheint dieses Fenster und fordert den Benutzer dazu auf, eine Hypervisor-Sitzung auszuwählen. Nicht privilegierte Benutzer können eine schreibgeschützte Sitzung initiieren. Root-Benutzer können eine Sitzung mit vollem Lese- und Schreibzugriff starten. Wählen Sie für normalen Gebrauch die **Lokaler Xen-Host** Option oder QEMU (für KVM).

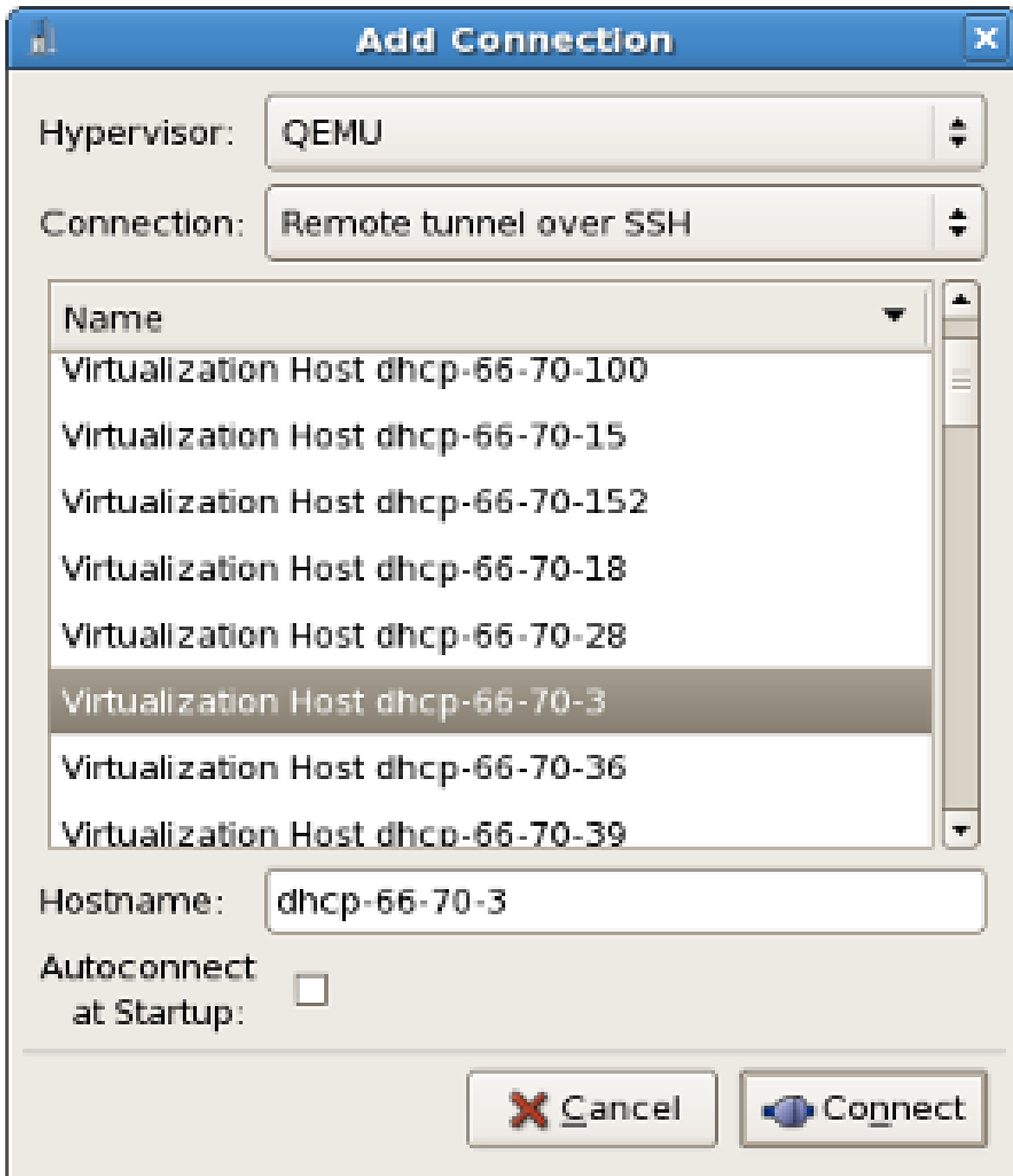


Abbildung 16.1. Das Fenster mit Verbindungen des Virtual Machine Manager

## 16.2. Das Hauptfenster des Virtual Machine Managers

Dieses Fenster zeigt alle laufenden virtuellen Maschinen, sowie deren zugeweilte Ressourcen (inklusive domain0). Sie können entscheiden, welche Felder angezeigt werden sollen. Ein Doppelklick auf die gewünschte virtuelle Maschine liefert die entsprechende Konsole für die bestimmte Maschine. Die Auswahl einer virtuellen Maschine und ein Doppelklick auf die Schaltfläche **Details** zeigt das Fenster "Details" für diese Maschine. Weiterhin können Sie auf das Menü **Datei** zugreifen, um eine neue virtuelle Maschine zu erstellen.



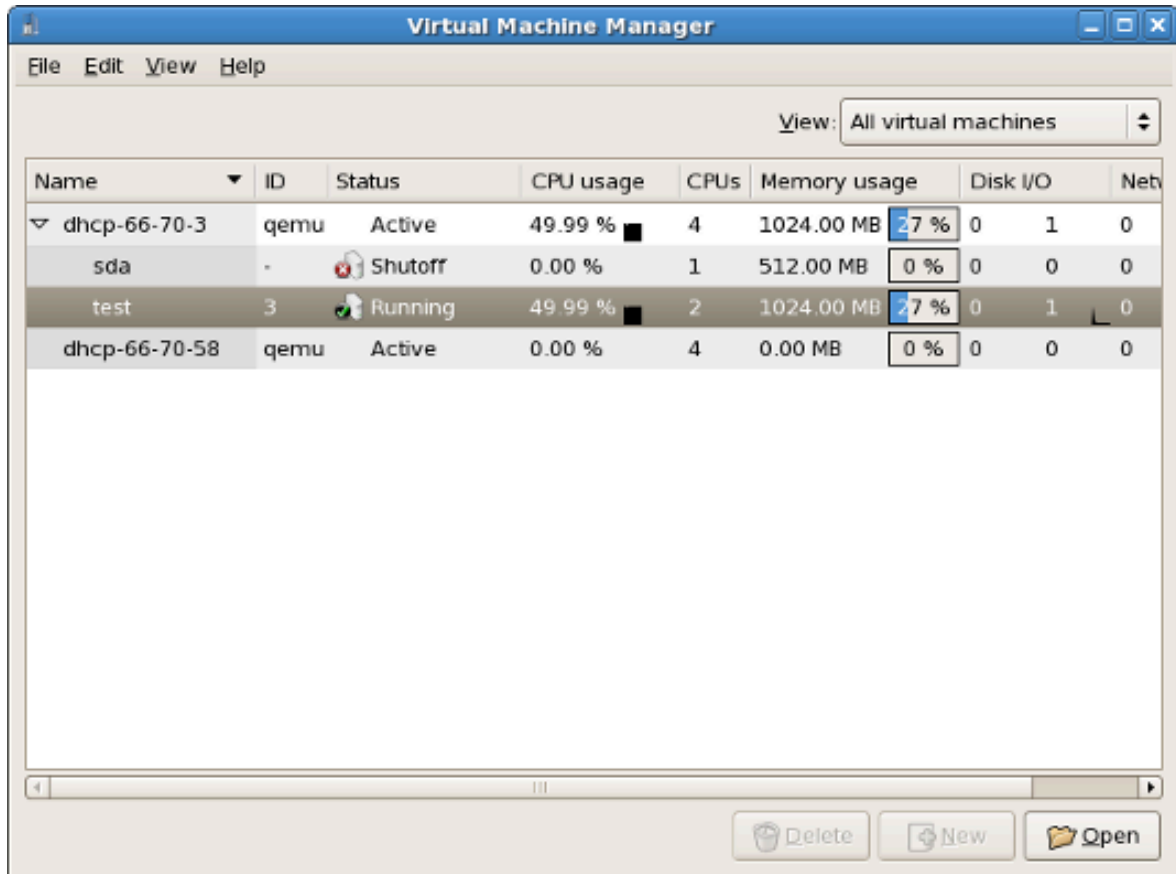


Abbildung 16.2. Hauptfenster des Virtual Machine Managers

### 16.3. Das Detail-Fenster des Virtual Machine Managers

Dieses Fenster zeigt Graphen und Statistiken der Nutzungsdaten von Ressourcen für einen Gast in Echtzeit an, die per **virt-manager** verfügbar sind. Das UUID-Feld zeigt den global eindeutigen Identifier für die virtuelle Maschine dar.

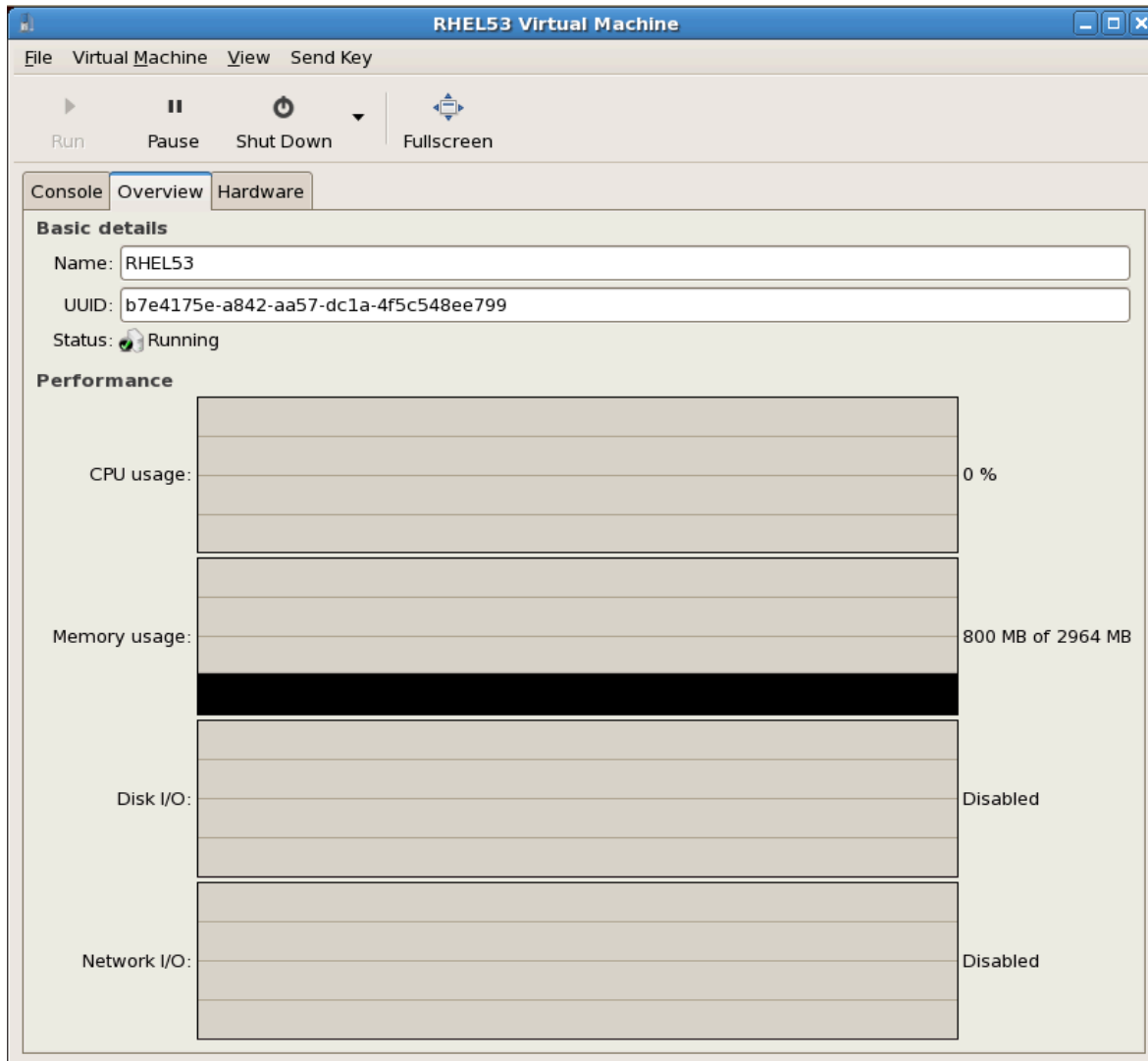


Abbildung 16.3. Das Detail-Fenster des virt-manager

## 16.4. Die grafische Konsole der virtuellen Maschine

Dieses Fenster stellt die grafische Konsole einer virtuellen Maschine dar. Paravirtualisierte und voll virtualisierte Maschinen verwenden unterschiedliche Techniken, um Ihre lokalen Framebuffer (Bildspeicher) zu exportieren. Beide Technologien verwenden jedoch **VNC**, um diese dem Konsolenfenster des Virtual Machine Manager zur Verfügung zu stellen. Falls Ihre virtuelle Maschine so konfiguriert ist, dass sie eine Authentifizierung erfordert, fordert Sie die grafische Konsole der virtuellen Maschine zur Eingabe eines Passworts auf, bevor die Anzeige erscheint.

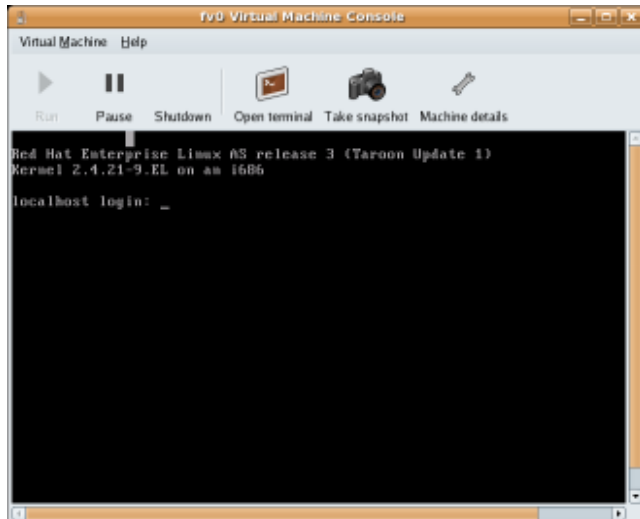


Abbildung 16.4. Das Fenster der grafischen Konsole



### Anmerkung zur Sicherheit und VNC

VNC wird von vielen Sicherheitsexperten als unsicher angesehen. Allerdings wurden einige Änderungen vorgenommen, um eine sichere Benutzung von VNC für die Virtualisierung unter Fedora zu gewährleisten. Die Gastmaschine horcht nur auf die Loopback-Adresse (127.0.0.1) des lokalen Hosts (dom0). Dadurch ist sichergestellt, dass nur diejenigen mit Shell-Privilegien auf dem Host auf den virt-manager und die virtuelle Maschine via VNC zugreifen können.

Administration von Remote aus ist möglich, wenn Sie den Anweisungen unter [Kapitel 13, Remote-Verwaltung virtualisierter Gäste](#) folgen. TLS kann bei der Verwaltung von Gast- und Host-Systemen Sicherheit auf Unternehmensebene bieten.

Ihr lokales Desktop-System kann Tastaturkombinationen unterbinden (z. B. Strg+Alt+F11), um zu verhindern, dass diese an die Maschine des Gasts gesendet werden. Sie können die **virt-manager** 'sticky key'-Fähigkeit des Virtual Machine Manager verwenden, um diese Tastaturfolge zu senden. Sie müssen eine Modifikatortaste (wie Strg oder Alt) dreimal drücken und dann wird die Taste, die Sie angeben, solange als aktiv behandelt, bis die nächste Taste, die keine Modifikatortaste ist, gedrückt wird. Anschließend können Sie Strg+Alt+F11 an den Gast senden, indem Sie die Tastaturfolge 'Strg Strg Strg Alt+F1' eingeben.

## 16.5. Starting virt-manager

Um eine **virt-manager**-Sitzung zu starten, öffnen Sie das **Anwendungen**-Menü, anschließend das **Systemwerkzeuge**-Menü und wählen dort den **Virtual Machine Manager (virt-manager)**.

Das Hauptfenster vom **virt-manager** erscheint.

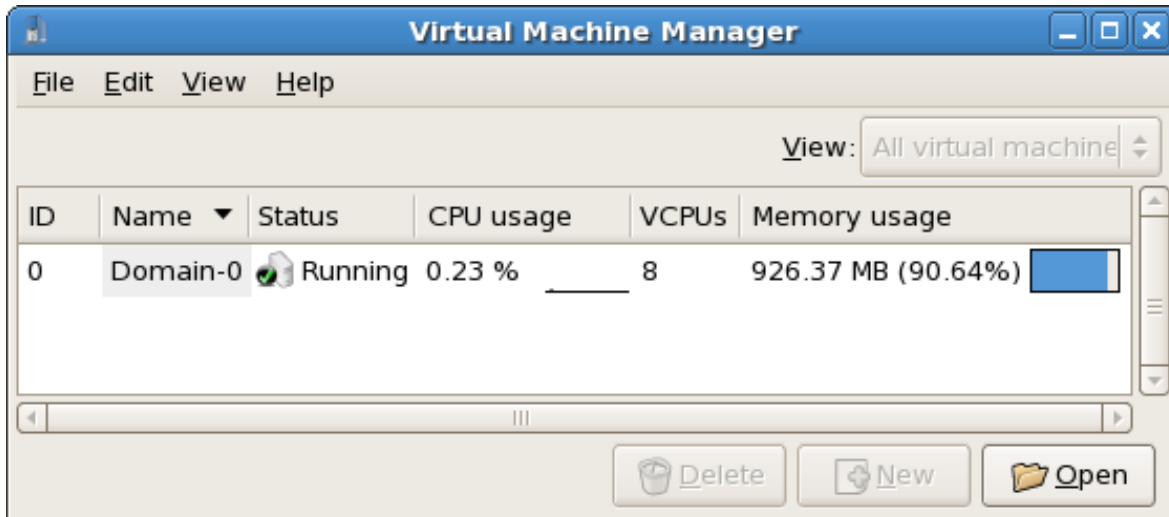


Abbildung 16.5. Das Starten von **virt-manager**

Alternativ kann **virt-manager** auch von Remote aus gestartet werden unter Verwendung von SSH, wie im folgenden Befehl veranschaulicht:

```
ssh -X host's address[remotehost]# virt-manager
```

Die Verwendung von **ssh** bei der Verwaltung virtueller Maschinen wird näher erläutert in [Abschnitt 13.1, „Remote-Verwaltung mit SSH“](#).

## 16.6. Wiederherstellen einer gespeicherten Maschine

Nachdem Sie den Virtual Machine Manager gestartet haben, werden alle virtuellen Maschinen auf Ihrem System im Hauptfenster angezeigt. Domain0 ist Ihr Host-System. Falls keine Maschinen existieren, bedeutet dies, dass derzeit keine Maschinen auf dem System laufen.

Um eine zuvor gespeicherte Sitzung wiederherzustellen:

1. Wählen Sie aus dem Menü **Datei** die Option **Gespeicherte Maschine wiederherstellen**.

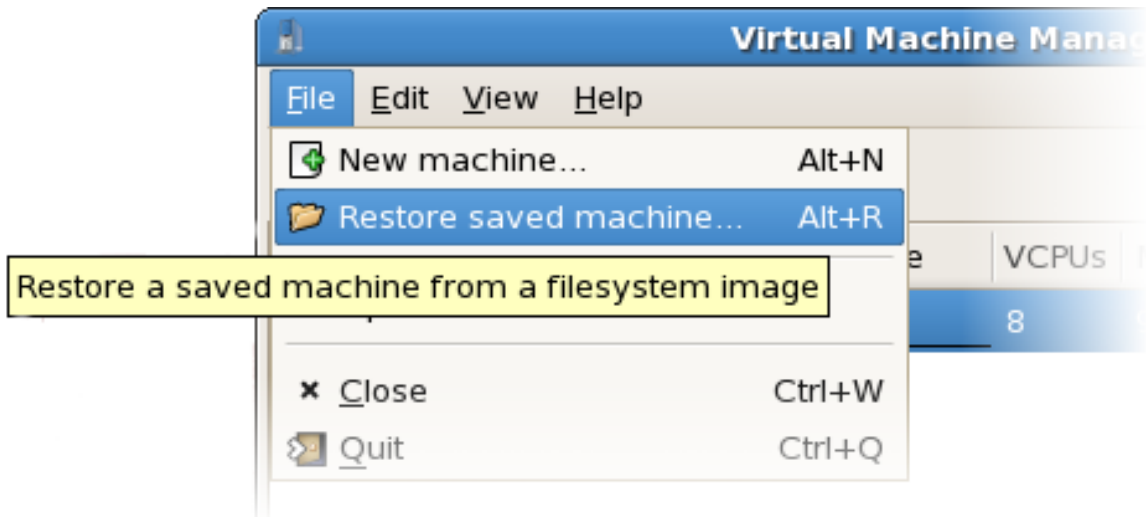


Abbildung 16.6. Wiederherstellen einer virtuellen Maschine

2. Das Hauptfenster **Virtuelle Maschine wiederherstellen** erscheint.
3. Begeben Sie sich in das korrekte Verzeichnis und wählen Sie die gespeicherte Sitzungsdatei.
4. Klicken Sie auf **Öffnen**.

Das gespeicherte virtuelle System erscheint im Fenster des Virtual Machine Manager.

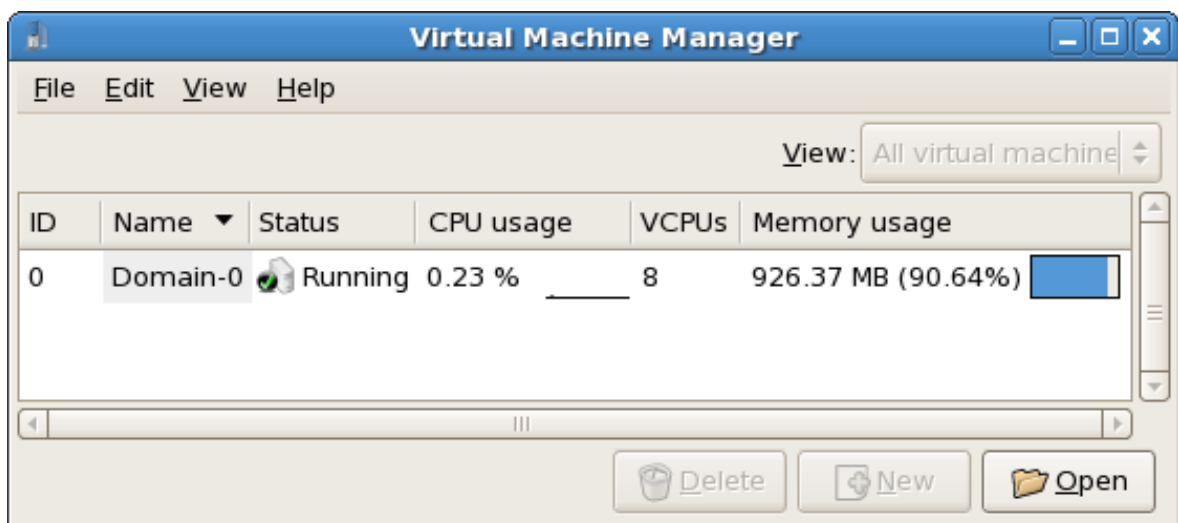


Abbildung 16.7. Die wiederhergestellte Sitzung des Virtual Machine Manager

## 16.7. Anzeigen von Gastdetails

Mit Hilfe des Virtual Machine Monitor können Sie sich die Daten zur Aktivität einer beliebigen virtuellen Maschine auf Ihrem System anschauen.

Um die Details eines virtuellen Systems anzusehen:

1. Markieren Sie im Hauptfenster des Virtual Machine Manager die virtuelle Maschine, die Sie ansehen möchten.

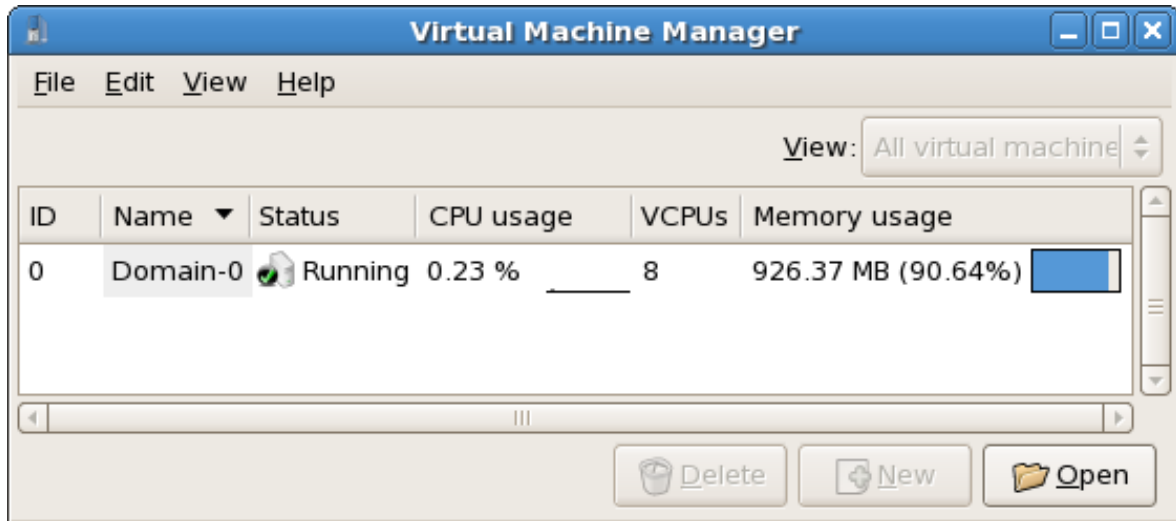


Abbildung 16.8. Auswahl einer anzuzeigenden virtuellen Maschine

2. Wählen Sie **Maschinendetails** aus dem Menü **Bearbeiten** des Virtual Machine Manager (oder klicken Sie auf die Schaltfläche **Details** am unteren Rand des Hauptfensters des Virtual Machine Manager).

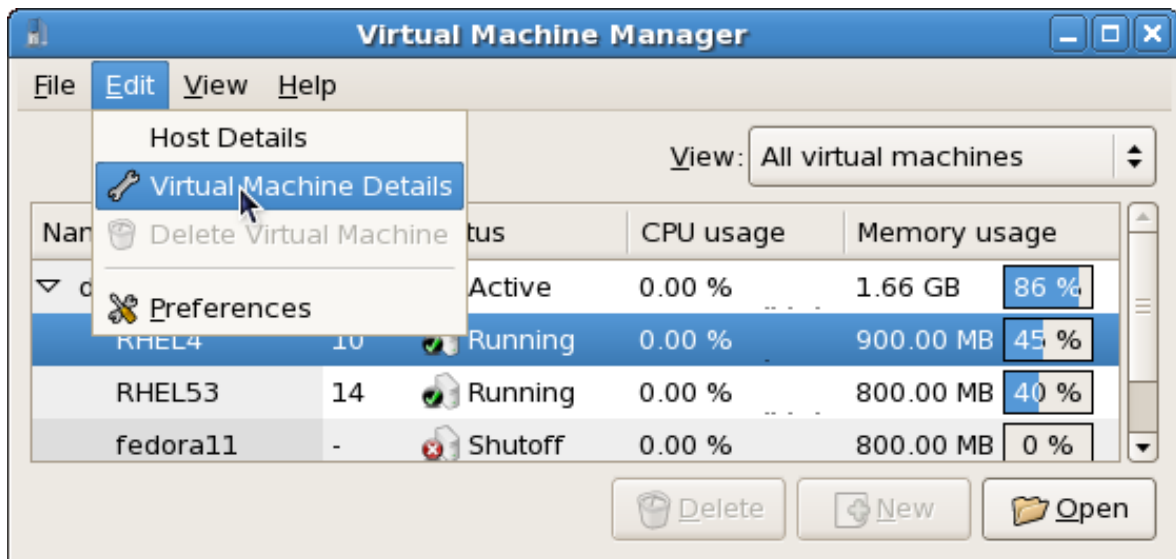


Abbildung 16.9. Anzeigen des Detail-Menüs der virtuellen Maschine

Das Fenster mit dem Überblick über die Details der virtuellen Maschine erscheint. Dieses Fenster fasst die Verwendung von CPU und Speicher für die von Ihnen angegebene(n) Domain(s) zusammen.

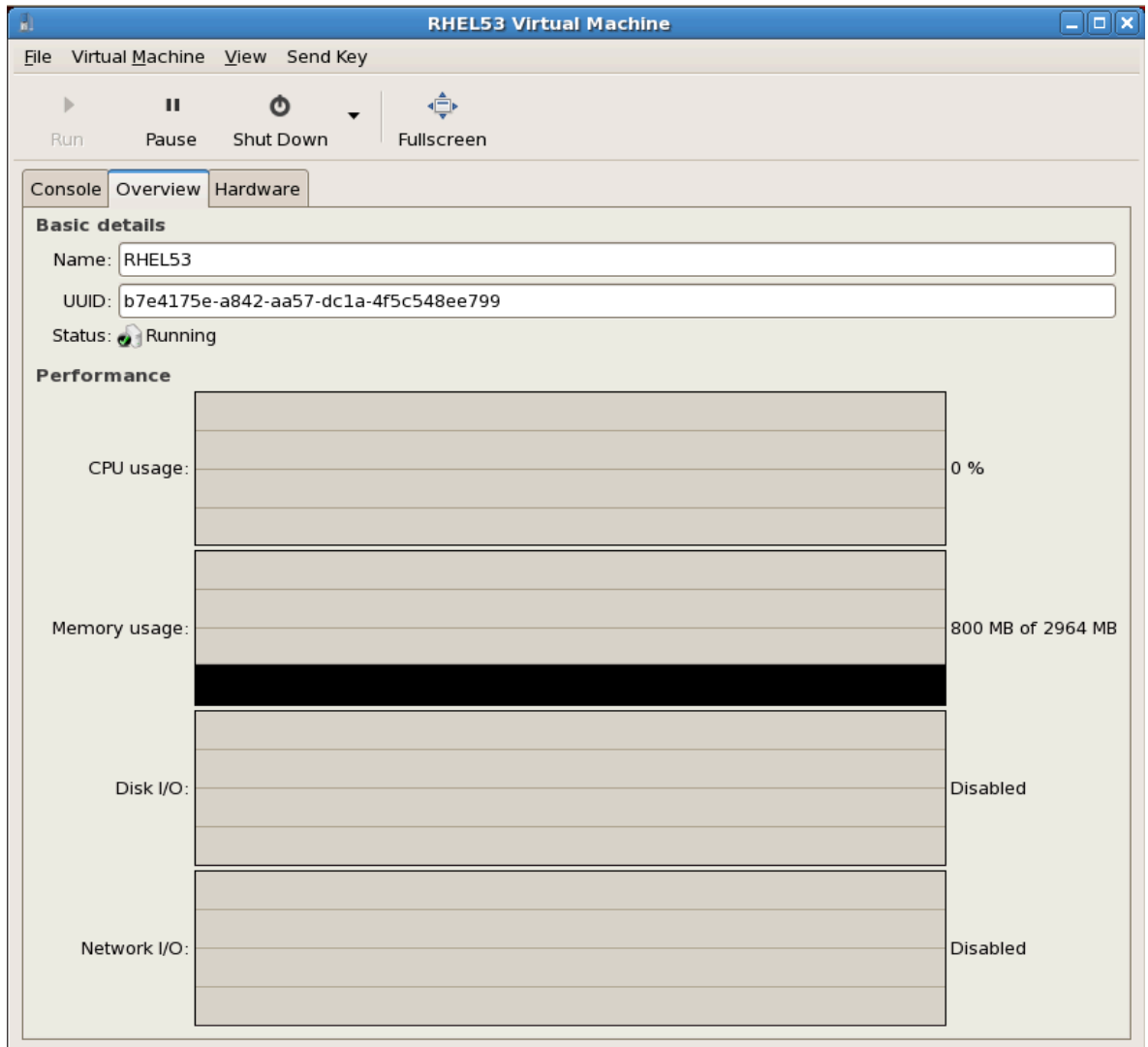


Abbildung 16.10. Anzeigen des Detail-Überblicks der virtuellen Maschine

3. Klicken Sie im Fenster **Details der virtuellen Maschine** auf den Reiter **Hardware**.  
Das Fenster **Hardware-Details der virtuellen Maschine** erscheint.

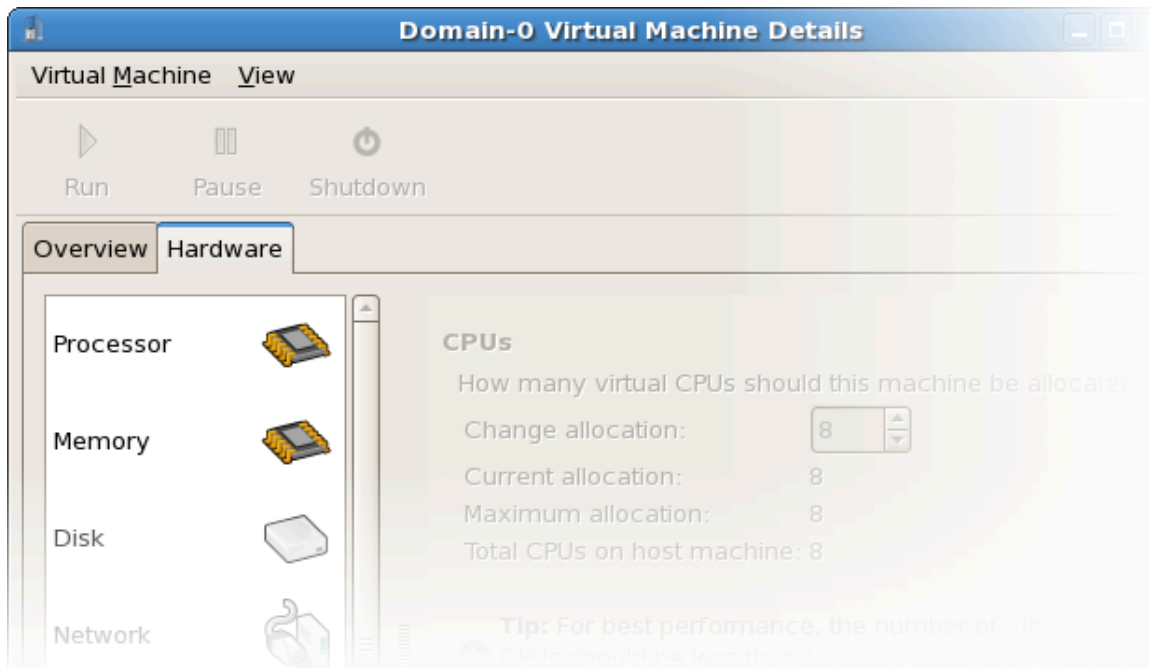


Abbildung 16.11. Anzeigen der Hardware-Details der virtuellen Maschine

4. Um die derzeitige Zuweisung von Prozessorspeicher zu betrachten oder zu verändern, klicken Sie auf **Prozessor** auf dem Reiter **Hardware**.



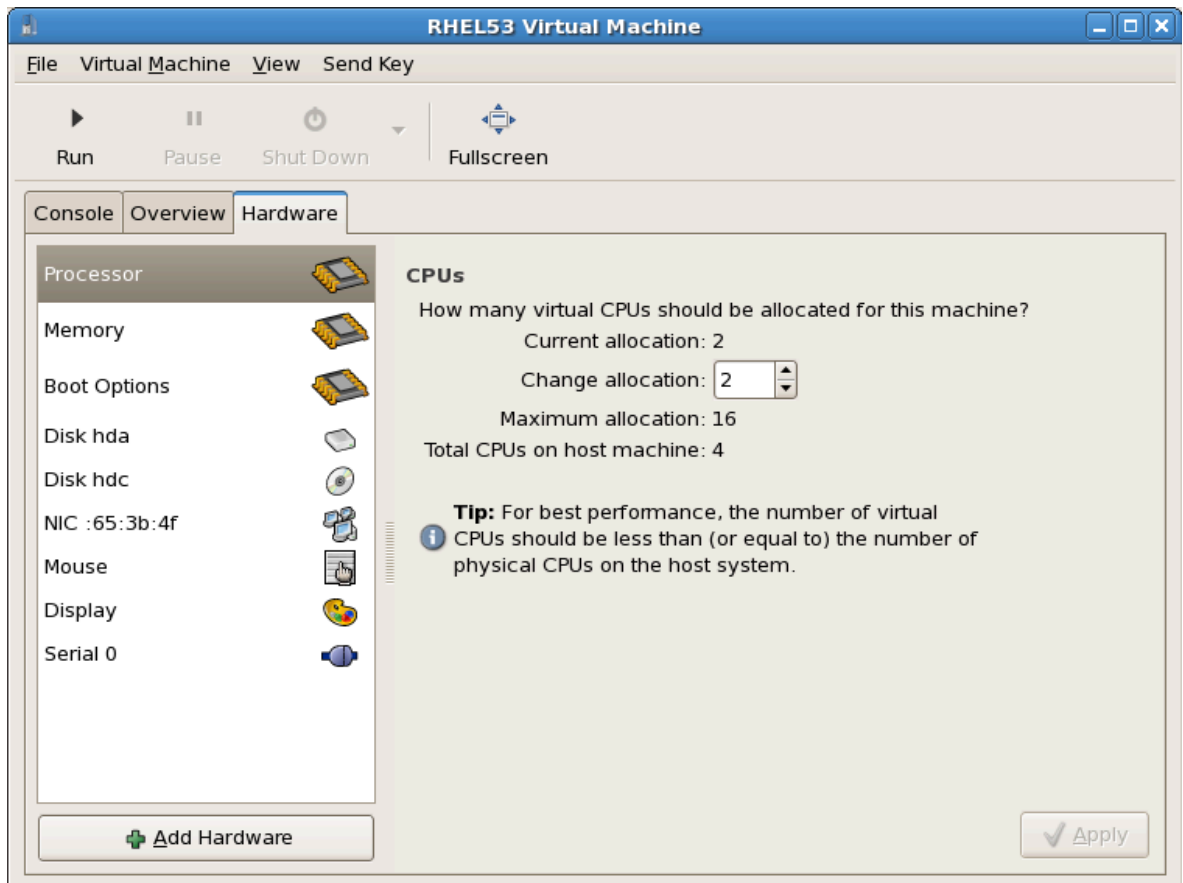


Abbildung 16.12. Anzeigen der Prozessorzuweisung

- Um die derzeitige Zuweisung von Arbeitsspeicher zu betrachten oder zu verändern, klicken Sie auf **Speicher** auf dem Reiter **Hardware**.

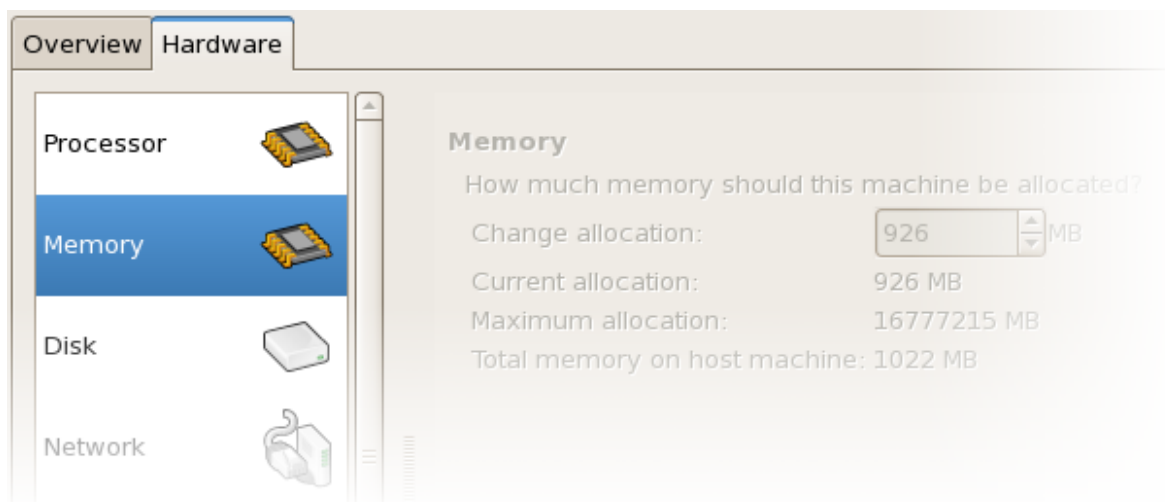


Abbildung 16.13. Anzeigen der Arbeitsspeicherzuweisung

- Um die derzeitige Festplattenkonfiguration zu betrachten oder zu verändern, klicken Sie auf **Festplatte** auf dem Reiter **Hardware**.

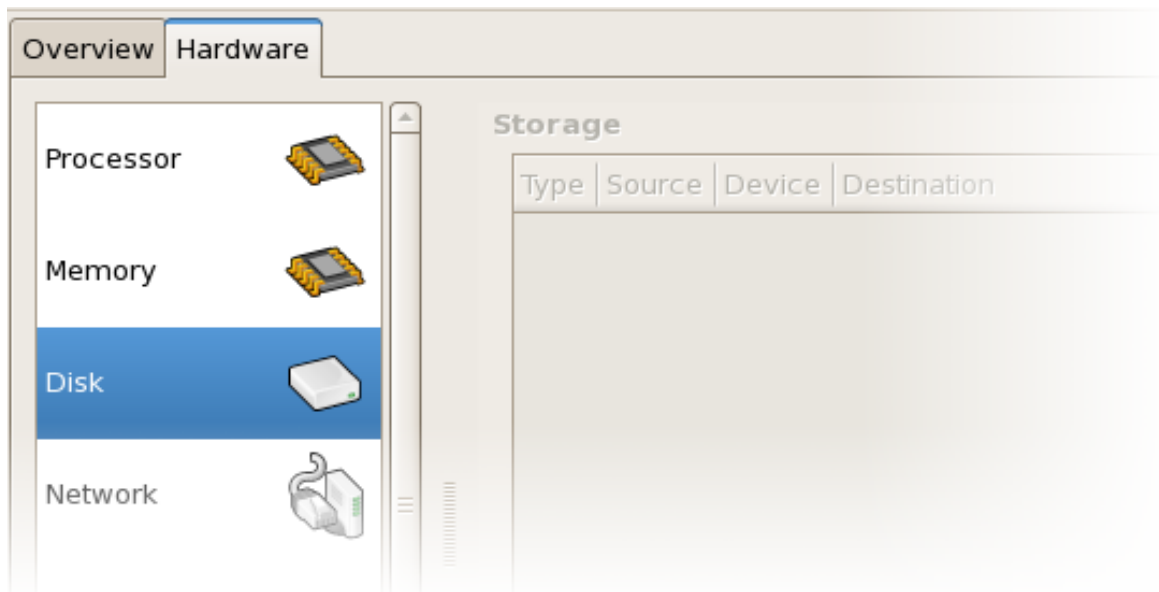


Abbildung 16.14. Anzeigen der Festplattenkonfiguration

- Um die derzeitige Netzwerkkonfiguration zu betrachten oder zu verändern, klicken Sie auf **Netzwerk** auf dem Reiter **Hardware**.

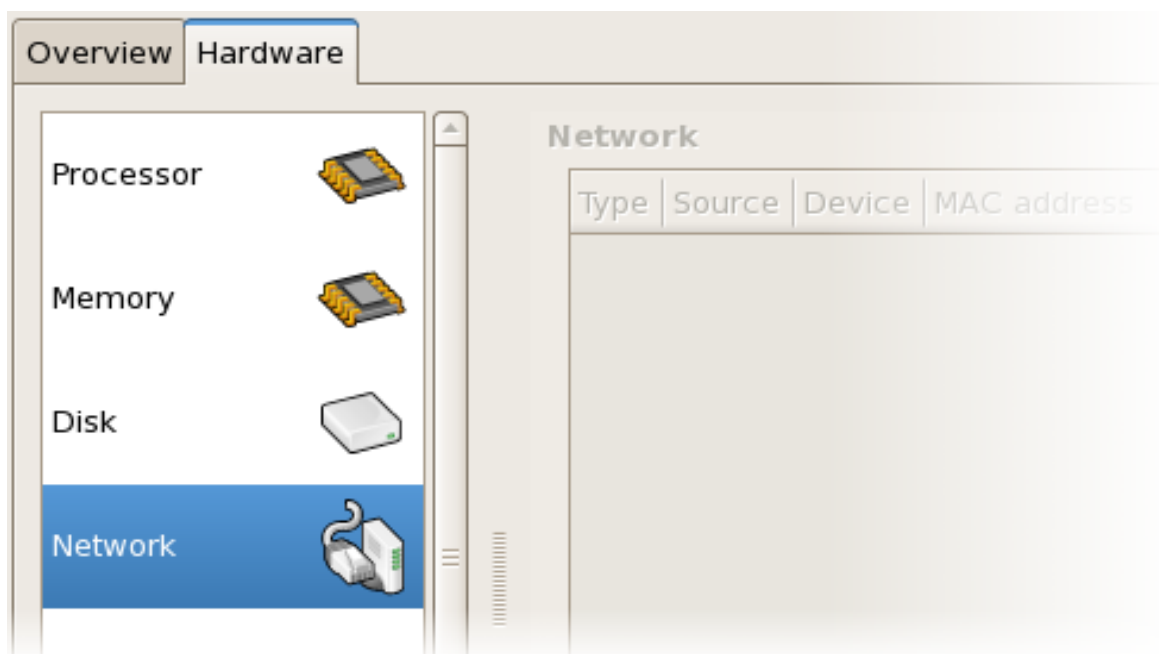


Abbildung 16.15. Anzeigen der Netzwerkkonfiguration

## 16.8. Überwachen des Status

Mit Hilfe des Virtual Machine Manager können Sie die Statusüberwachung des virtuellen Systems modifizieren.

Gehen Sie wie folgt vor, um die Statusüberwachung zu konfigurieren und Konsolen zu aktivieren:

- Wählen Sie **Präferenzen** aus dem Menü **Bearbeiten**.

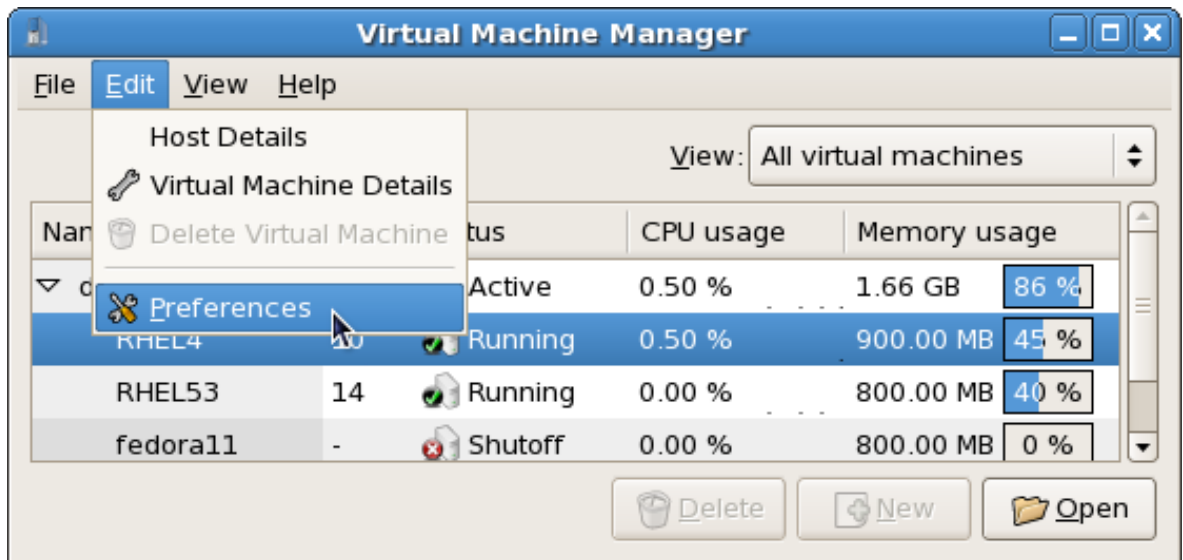


Abbildung 16.16. Modifizieren der Präferenzen des Gasts

Das Fenster "Präferenzen" des Virtual Machine Manager erscheint.

2. Geben Sie in der Auswahlbox der Statusüberwachung die Zeit (in Sekunden) an, die Sie für das System-Update festlegen möchten.

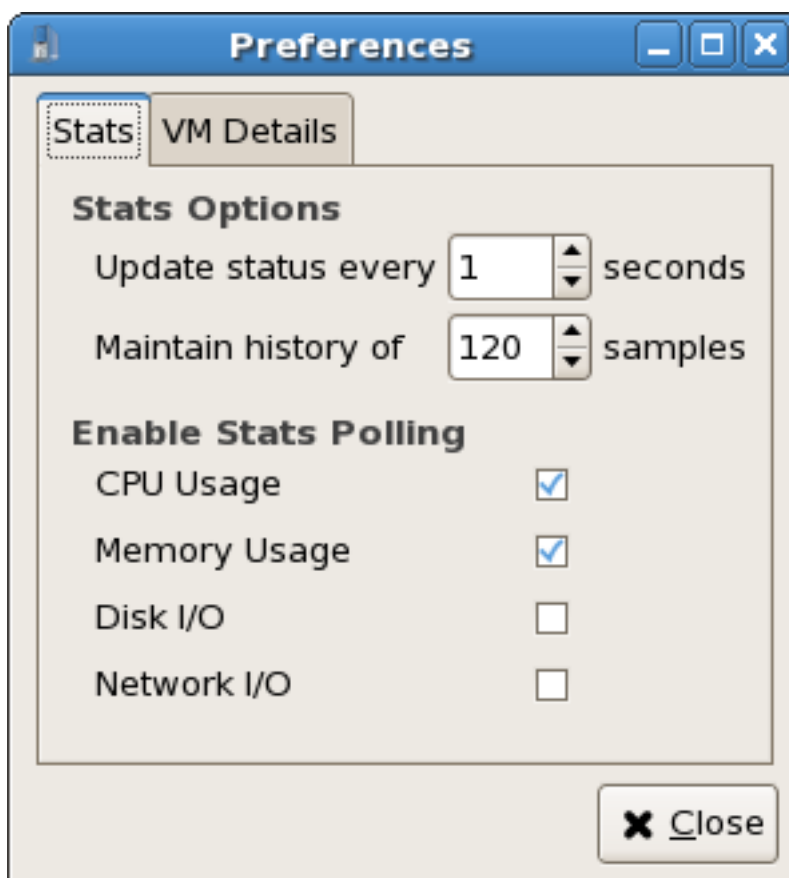


Abbildung 16.17. Konfiguration der Statusüberwachung

3. Geben Sie im Konsolenbereich an, wie eine Konsole geöffnet werden soll und legen Sie ein Eingabegerät fest.

### 16.9. Anzeigen der Gast-Identifizier

Gehen Sie wie folgt vor, um die Gast-IDs für alle virtuellen Maschinen auf Ihrem System anzusehen:

1. Wählen Sie das Kontrollkästchen **Domain-ID** aus dem Menü **Anzeigen**.

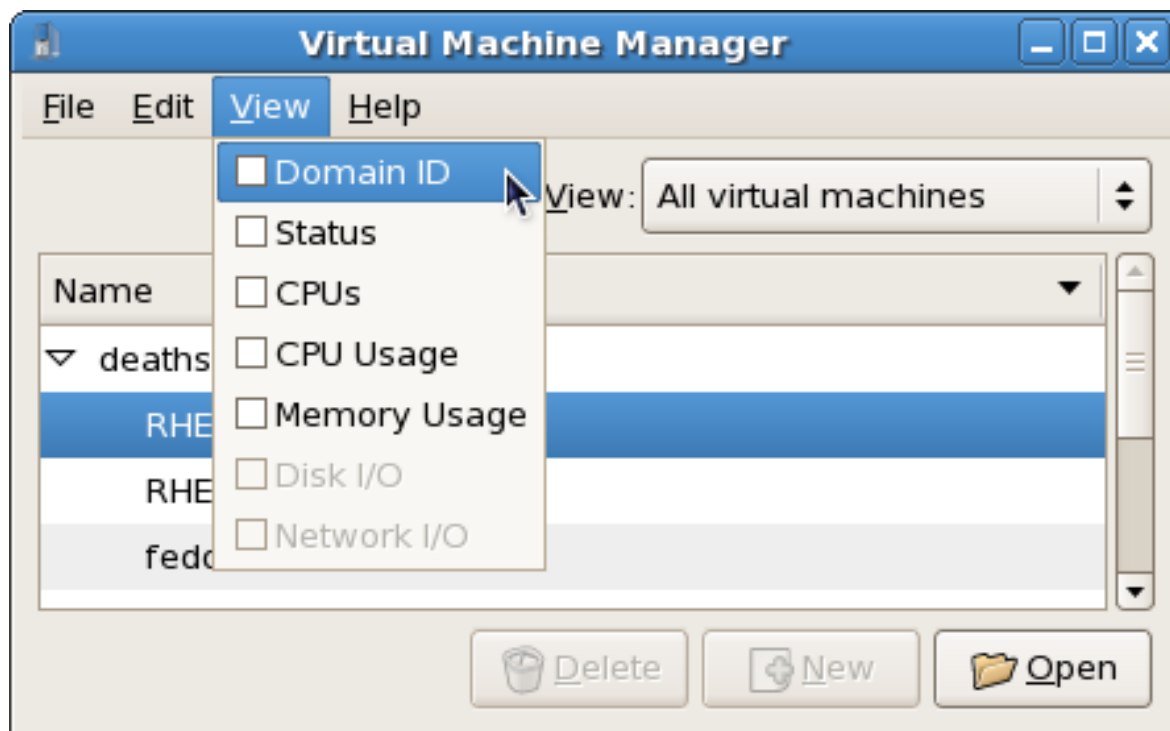


Abbildung 16.18. Anzeigen der Gast-IDs

2. Der Virtual Machine Manager listet die Domain-IDs für alle Domains auf Ihrem System auf.

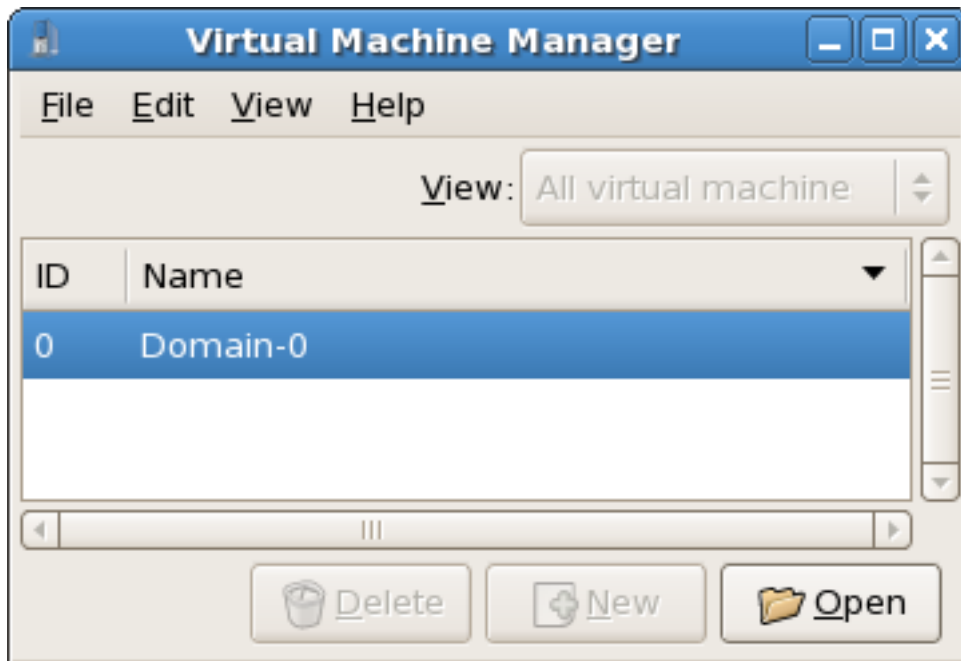


Abbildung 16.19. Anzeigen der Domain-ID

## 16.10. Anzeigen des Gaststatus

Gehen Sie wie folgt vor, um den Status aller virtuellen Maschinen auf Ihrem System zu betrachten:

1. Wählen Sie das Kontrollkästchen **Status** aus dem Menü **Ansicht**.

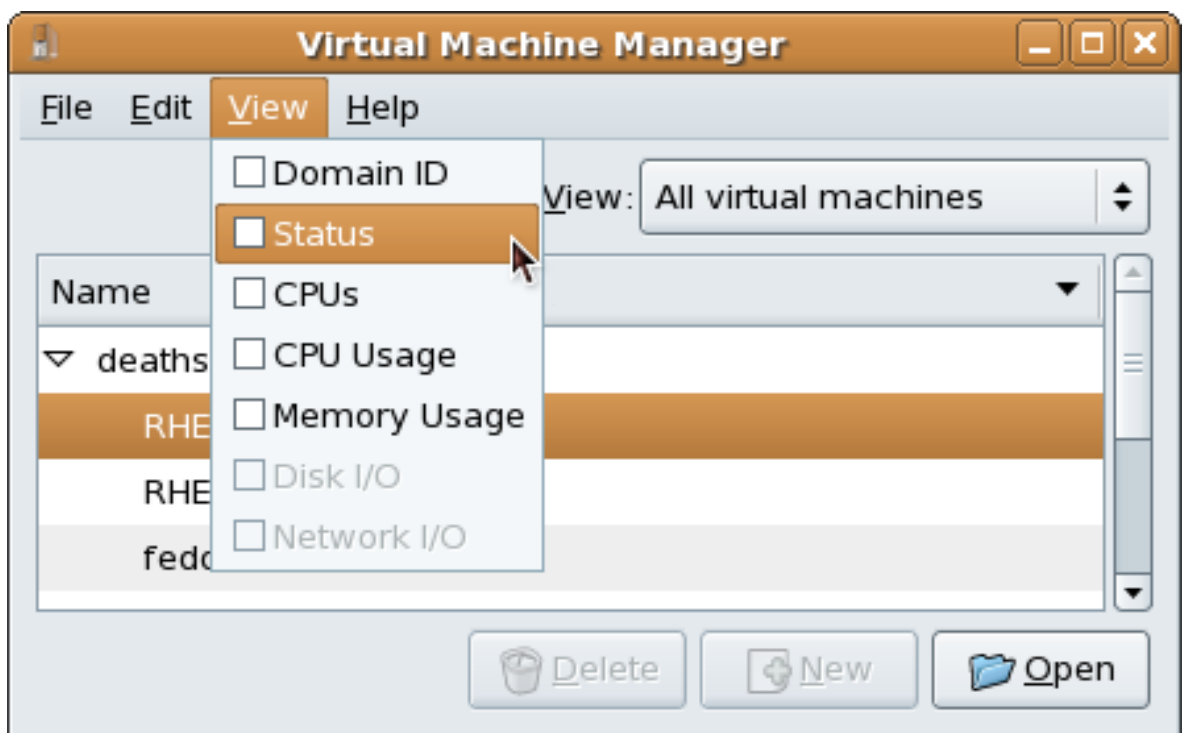


Abbildung 16.20. Auswahl des Status der virtuellen Maschine

- Der Virtual Machine Manager listet den Status aller virtuellen Maschinen auf Ihrem System auf.

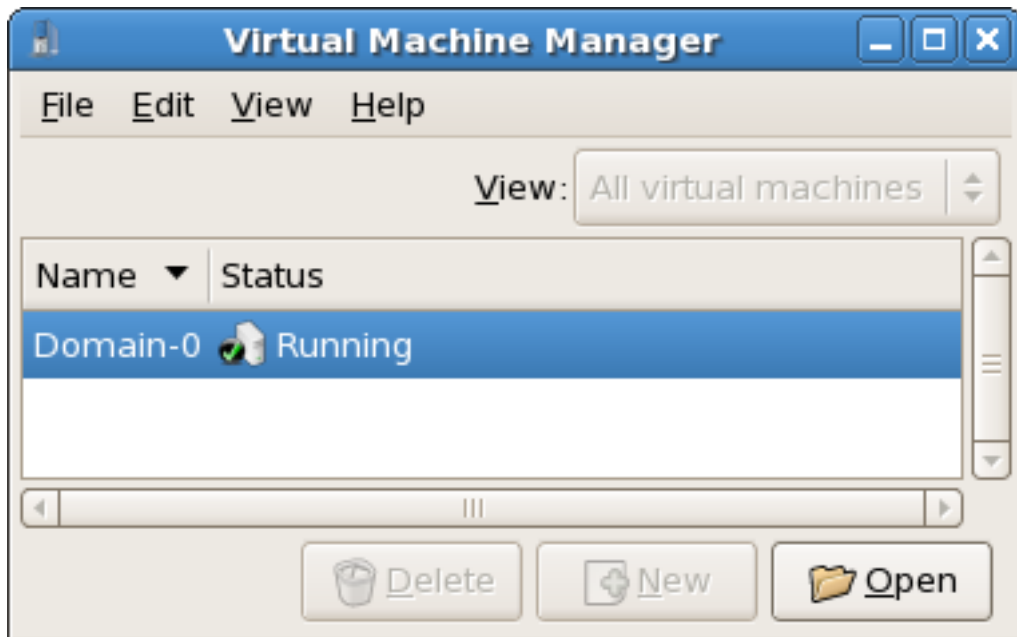


Abbildung 16.21. Anzeigen des Status der virtuellen Maschine

## 16.11. Anzeigen virtueller CPUs

Gehen Sie wie folgt vor, um die Anzahl der virtuellen CPUs für alle virtuellen Maschinen auf Ihrem System anzusehen:

- Wählen Sie aus dem Menü **Anzeigen** das Kontrollkästchen **Virtuelle CPUs**.

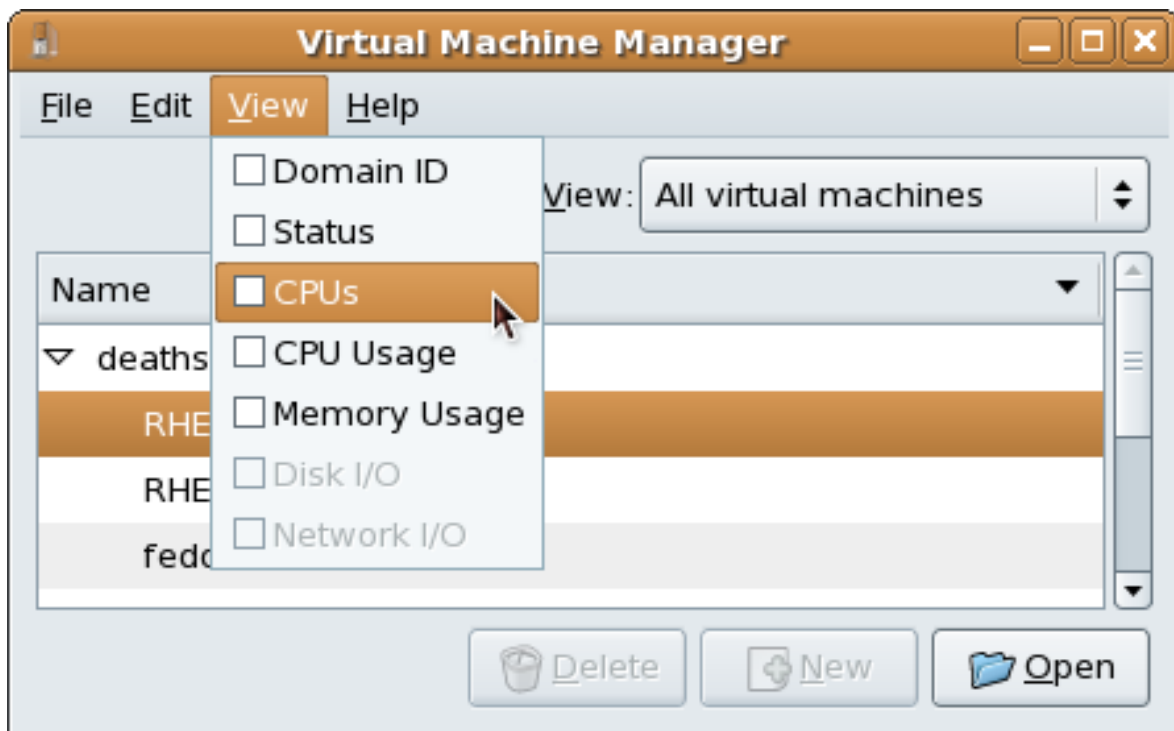


Abbildung 16.22. Auswahl der virtuellen CPU-Optionen

2. Der Virtual Machine Manager listet die virtuellen CPUs für alle virtuellen Maschinen auf Ihrem System auf.

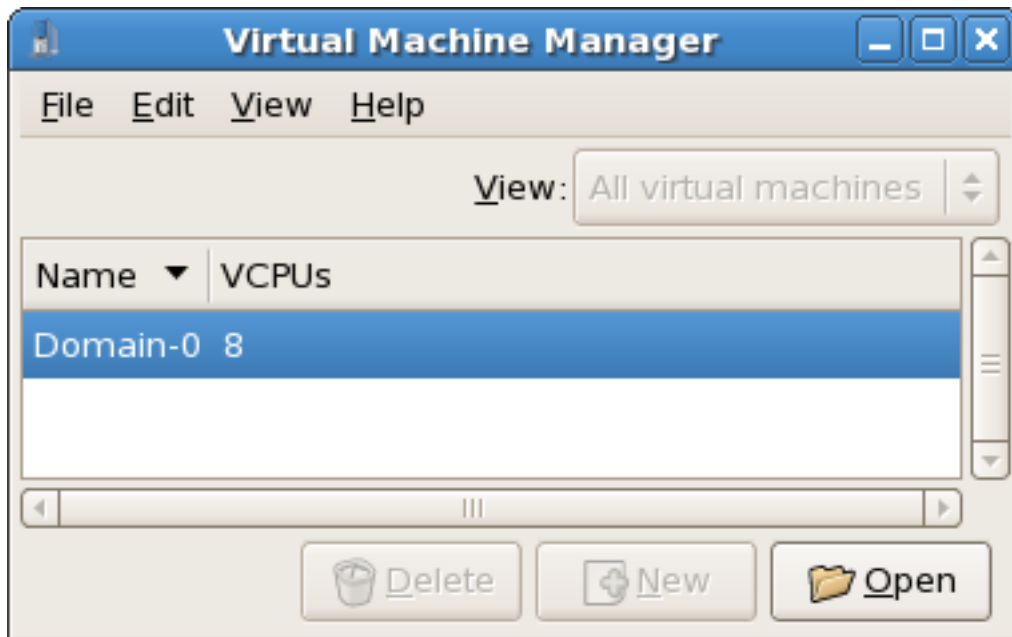


Abbildung 16.23. Anzeigen virtueller CPUs

## 16.12. Anzeigen der CPU-Auslastung

Gehen Sie wie folgt vor, um die CPU-Auslastung für alle virtuellen Maschinen auf Ihrem System anzusehen:

1. Wählen Sie das Kontrollkästchen **CPU-Auslastung** aus dem Menü **Anzeige**.

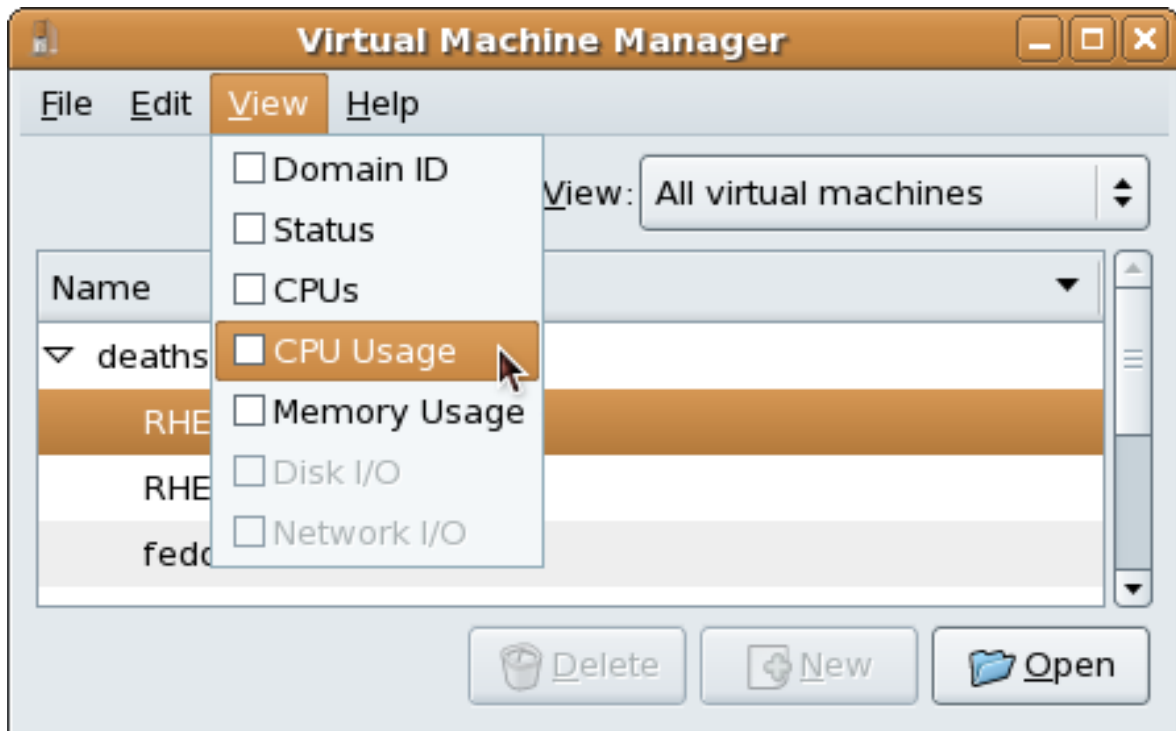


Abbildung 16.24. Auswahl der CPU-Auslastung

- 2. Der Virtual Maschine Manager listet die CPU-Auslastung in Prozent für alle virtuellen Maschinen auf Ihrem System auf.

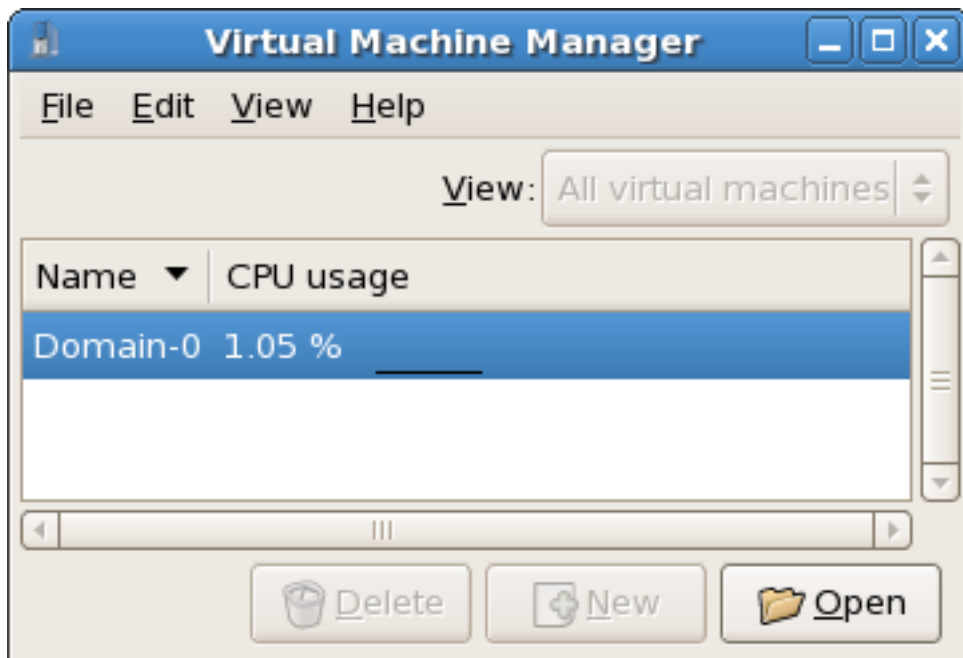


Abbildung 16.25. Anzeigen der CPU-Auslastung



## 16.13. Anzeigen des Speicherverbrauchs

Gehen Sie wie folgt vor, um den Speicherverbrauch für alle virtuellen Maschinen auf Ihrem System zu betrachten:

1. Wählen Sie aus dem Menü **Anzeigen** das Kontrollkästchen **Speicherbelegung**.

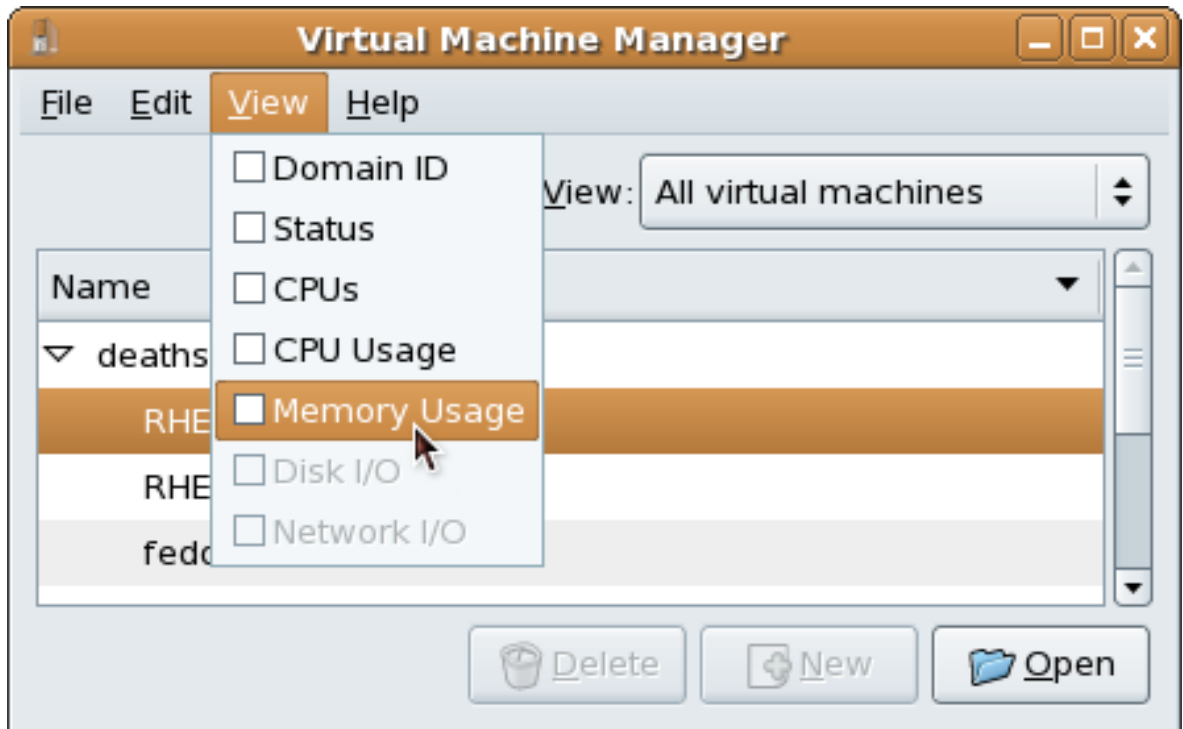


Abbildung 16.26. Auswahl des Speicherverbrauchs

2. Der Virtual Machine Manager listet den Anteil des Speicherverbrauchs (in Megabytes) für alle virtuellen Maschinen auf Ihrem System auf.

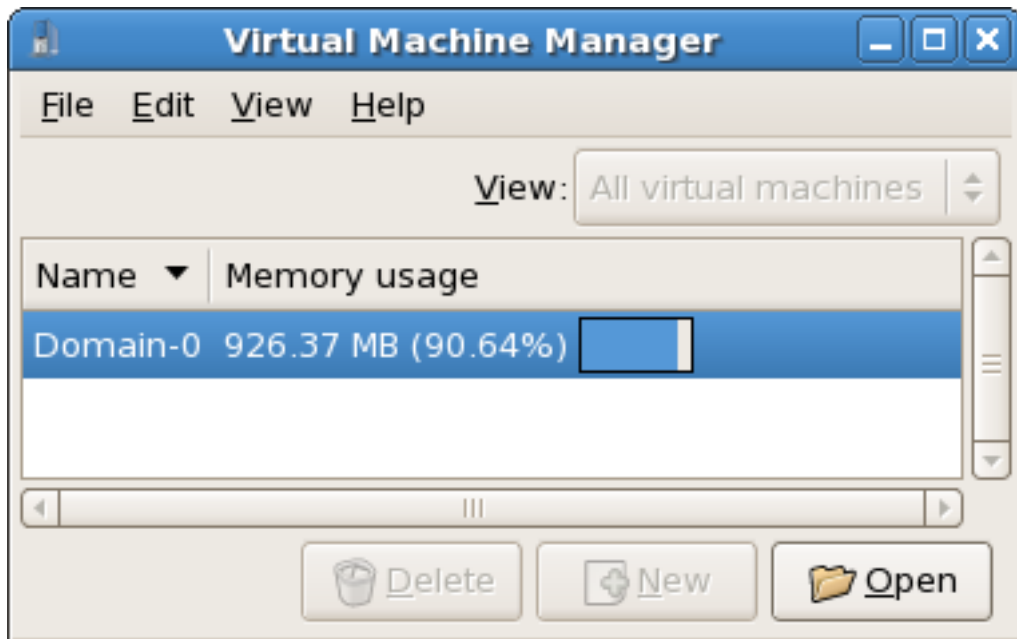


Abbildung 16.27. Anzeigen des Speicherverbrauchs

## 16.14. Verwalten eines virtuellen Netzwerks

Zur Konfiguration eines virtuellen Netzwerks auf Ihrem System:

1. Wählen Sie **Host-Details** aus dem Menü **Bearbeiten**.

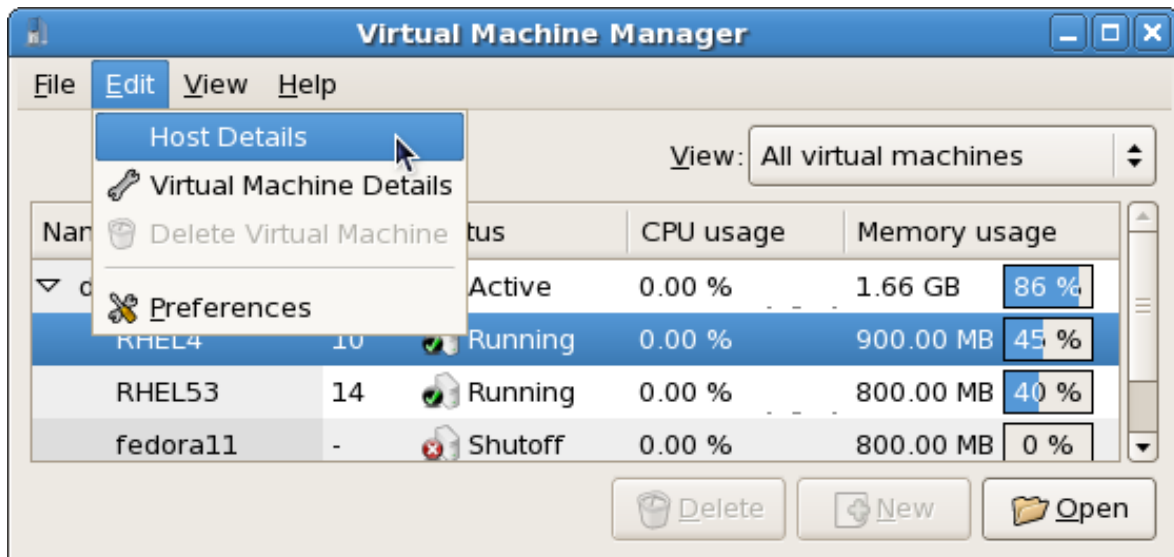


Abbildung 16.28. Auswählen der Host-Details

2. Dies öffnet das Menü **Host-Details**. Klicken Sie auf den Reiter **Virtuelle Netzwerke**.

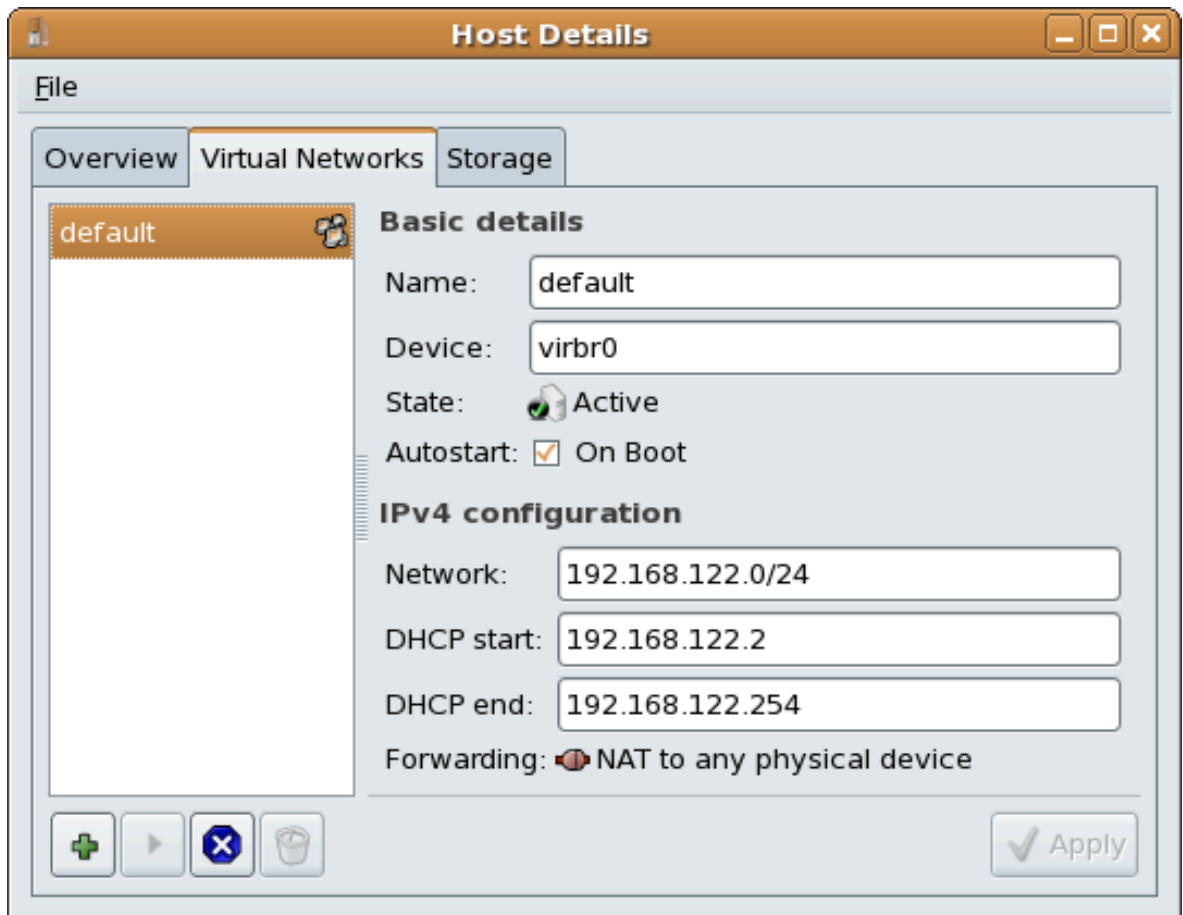


Abbildung 16.29. Virtuelle Netzwerkkonfiguration

3. Alle verfügbaren virtuellen Netzwerke sind im linken Kasten des Menüs aufgelistet. Sie können die Konfiguration eines virtuellen Netzwerks bearbeiten, indem Sie es aus diesem Kasten auswählen und nach Bedarf ändern.

## 16.15. Erstellen eines virtuellen Netzwerks

Um ein virtuelles Netzwerk auf Ihrem System zu erstellen:

1. Öffnen Sie das **Host-Details**-Menü (siehe [Abschnitt 16.14](#), „*Verwalten eines virtuellen Netzwerks*“) und klicken Sie auf **Hinzufügen**.

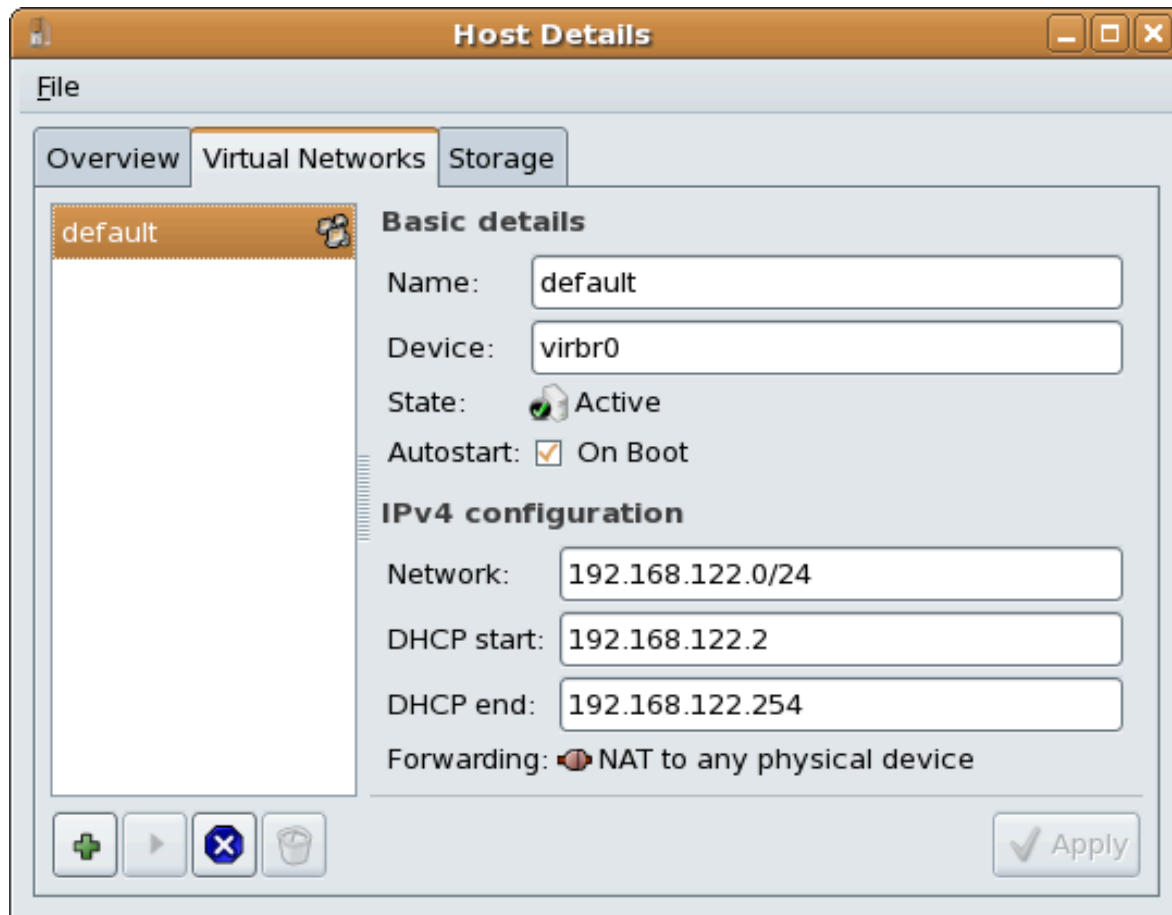


Abbildung 16.30. Virtuelle Netzwerkkonfiguration

Dies öffnet das Menü **Neues virtuelles Netzwerk erstellen**. Klicken Sie auf **Weiter**, um fortzufahren.

## Creating a new virtual network

This assistant will guide you through creating a new virtual network. You will be asked for some information about the virtual network you'd like to create, such as:

- A **name** for your new virtual network
- The IPv4 **address** and **netmask** to assign
- The **address range** from which the **DHCP** server will allocate addresses for virtual machines
- Whether to **forward** traffic to the physical network

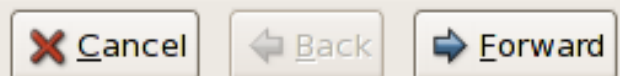


Abbildung 16.31. Erstellung eines neuen virtuellen Netzwerks

2. Geben Sie den Namen Ihres virtuellen Netzwerks ein und klicken Sie auf **Weiter**.



Abbildung 16.32. Benennung Ihres virtuellen Netzwerks

3. Geben Sie einen IPv4-Adressraum für Ihr virtuelles Netzwerk ein und klicken Sie auf **Weiter**.

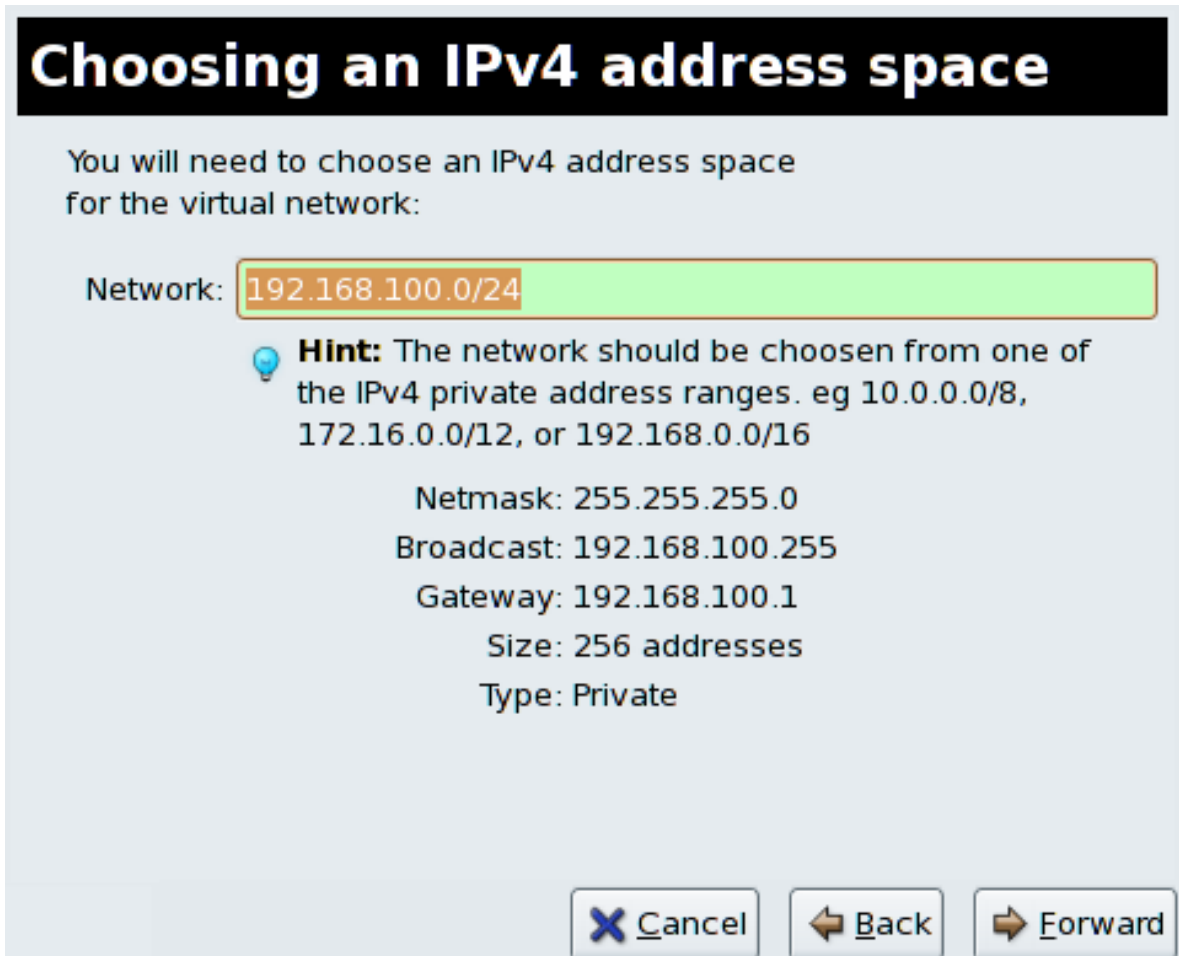


Abbildung 16.33. Auswahl eines IPv4-Adressraums

4. Definieren Sie den DHCP-Bereich für Ihr virtuelles Netzwerk durch die Angabe eines **Start-** und **End-** Bereichs von IP-Adressen. Klicken Sie auf **Weiter**, um fortzufahren.

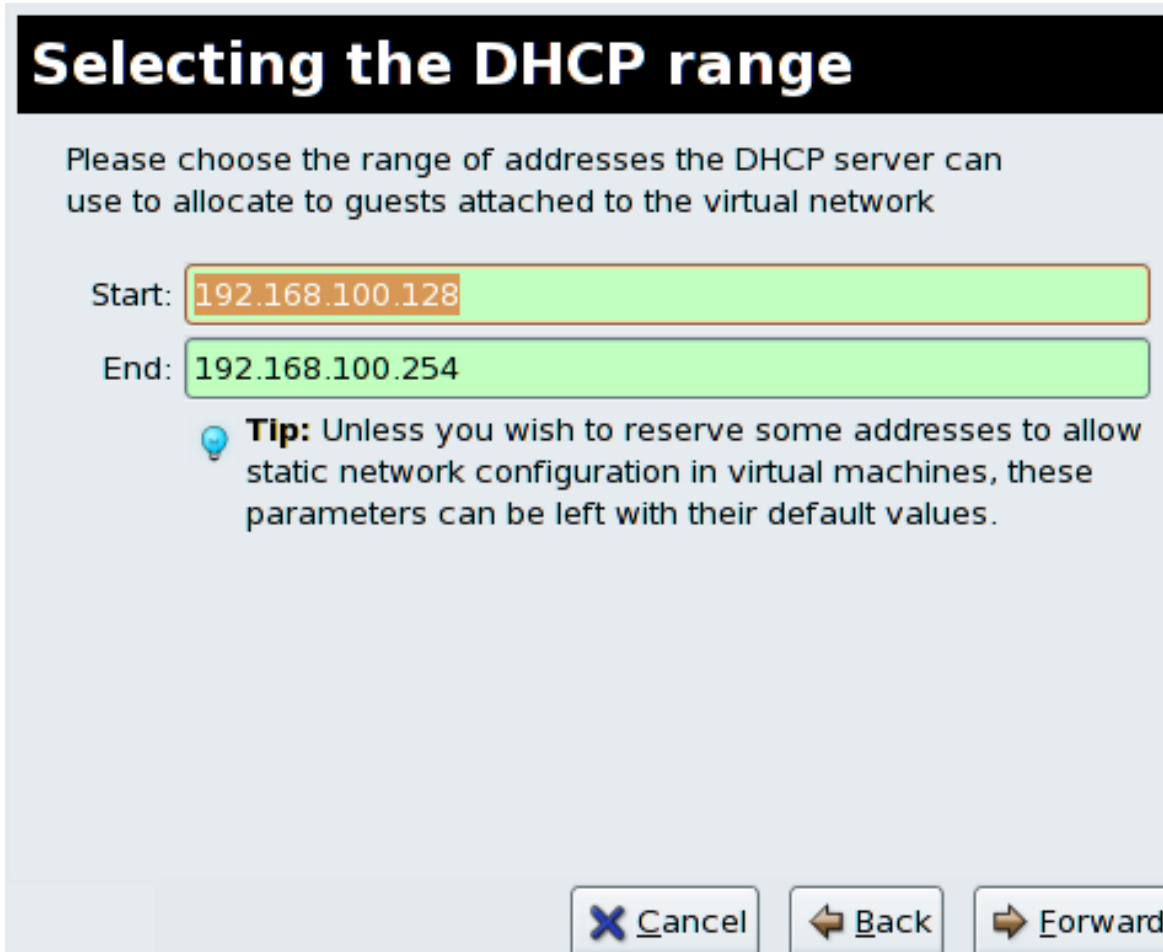


Abbildung 16.34. Auswahl des DHCP-Bereichs

5. Wählen Sie aus, wie sich das virtuelle Netzwerk mit dem physischen Netzwerk verbinden soll.



## Connecting to physical network

Please indicate whether this virtual network should be connected to the physical network.

Isolated virtual network

Forwarding to physical network

Desination:

Abbildung 16.35. Verbindung mit dem physischen Netzwerk

Falls Sie sich für **An physisches Netzwerk weiterleiten** entscheiden, wählen Sie, ob das **Ziel** entweder **NAT zu allen physischen Geräten** oder **NAT zum physischen Gerät eth0** sein soll.

Click **Forward** to continue.

6. Sie sind nun bereit, das Netzwerk einzurichten. Überprüfen Sie die Konfiguration Ihres Netzwerks und klicken Sie auf **Fertigstellen**.

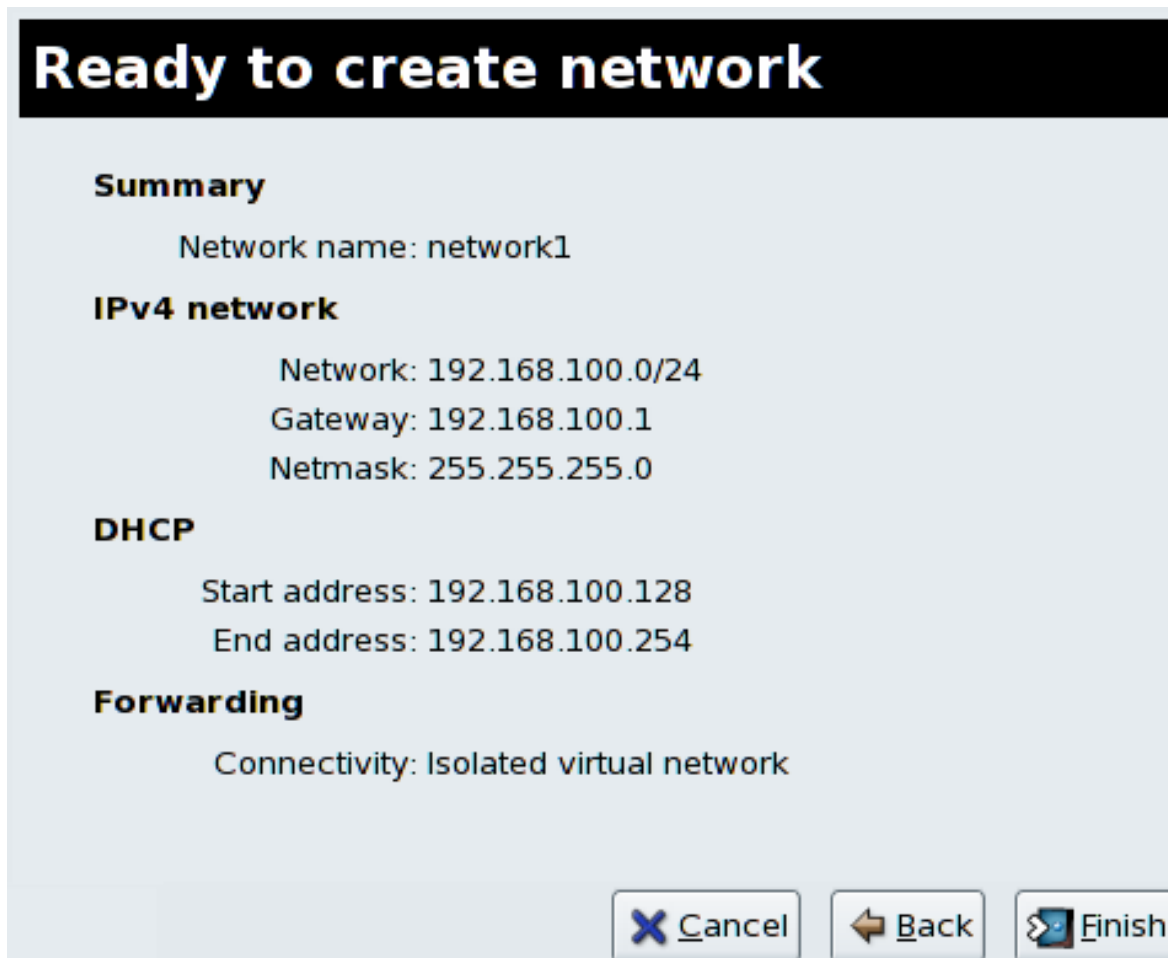


Abbildung 16.36. Bereit zur Erstellung des Netzwerks

7. Das neue virtuelle Netzwerk steht nun im Reiter **Virtuelles Netzwerk** des Menüs **Host-Details** zur Verfügung.

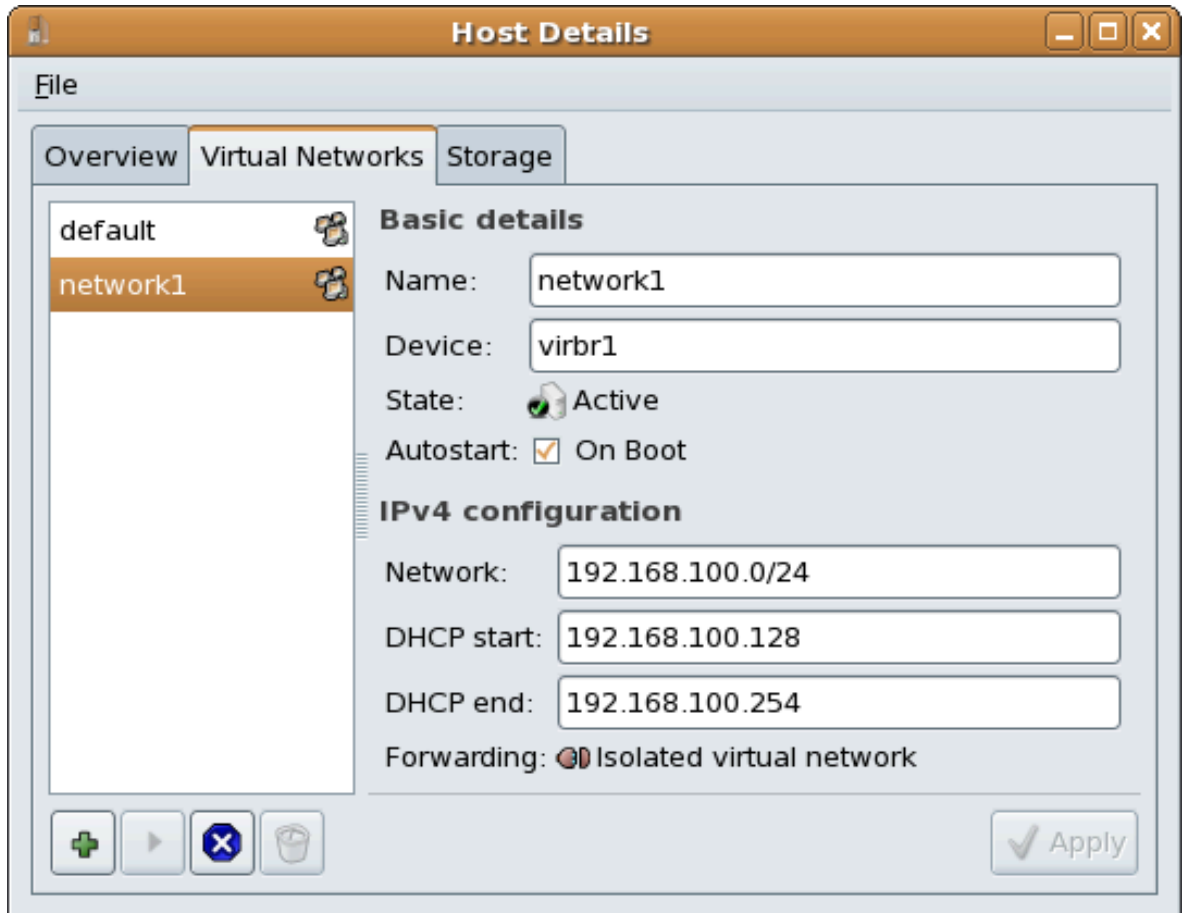


Abbildung 16.37. Neues virtuelles Netzwerk ist nun verfügbar



---

## Teil V. Tips and Tricks

# Tipps und Tricks für eine verbesserte Produktivität

Dieses Kapitel beinhaltet nützliche Hinweise und Tipps, um die Leistung, Skalierbarkeit und Stabilität der Virtualisierung zu verbessern.

---

---

---

# Tipps und Tricks

Dieses Kapitel beinhaltet nützliche Hinweise und Tipps, um die Leistung, Skalierbarkeit und Stabilität der Virtualisierung zu verbessern.

## 17.1. Gäste automatisch starten

Dieser Abschnitt beschreibt, wie ein virtualisierter Gast automatisch während des Hochfahrens des Host-Systems gestartet werden kann.

Dieses Beispiel verwendet **virsh**, um einen Gast namens *TestServer* so einzustellen, dass dieser automatisch startet, wenn der Host hochfährt.

```
virsh autostart TestServer
Domain TestServer marked as autostarted
```

Der Gast startet nun automatisch mit dem Host.

Um zu deaktivieren, dass der Gast automatisch startet, verwenden Sie den `--disable`-Parameter.

```
virsh autostart --disable TestServer
Domain TestServer unmarked as autostarted
```

Der Gast startet nun nicht mehr automatisch mit dem Host.

## 17.2. Wechseln zwischen dem KVM- und Xen-Hypervisor

Dieser Abschnitt behandelt das Wechseln zwischen dem KVM- und Xen-Hypervisor.

Fedora unterstützt nur einen aktiven Hypervisor zur selben Zeit.



### Migration von virtualisierten Gästen zwischen Hypervisoren

Derzeit existiert keine Applikationen zum Wechseln von Xen-basierten Gästen zu KVM-basierten Gästen bzw. umgekehrt. Gäste können ausschließlich auf dem Hypervisor-Typ verwendet werden, auf dem sie ursprünglich erzeugt wurden.

### 17.2.1. Xen zu KVM

Das folgende Verfahren beschreibt den Wechsel vom Xen-Hypervisor zum KVM-Hypervisor. Dieses Verfahren setzt voraus, dass das *kernel-xen*-Paket installiert und aktiviert ist.

#### 1. Installieren Sie das KVM-Paket

Installieren Sie das *kvm*-Paket, sofern Sie das nicht bereits getan haben.

```
yum install kvm
```

#### 2. Überprüfen Sie, welcher Kernel verwendet wird

Das *kernel-xen*-Paket kann installiert sein. Verwenden Sie den **uname**-Befehl, um festzustellen, welcher Kernel ausgeführt wird:

```
$ uname -r
2.6.23.14-107.fc8xen
```

Der aktuelle Kernel, "2.6.23.14-107.fc8xen", läuft auf dem System. Falls der Standard-Kernel, "2.6.23.14-107.fc8", ausgeführt wird, können Sie diesen Unterschritt überspringen.

- **Wechsel vom Xen-Kernel zum Standard-Kernel**

In der **grub.conf**-Datei wird festgelegt, welcher Kernel gebootet wird. Um den Standard-Kernel zu ändern, bearbeiten Sie die **/boot/grub/grub.conf**-Datei wie unten veranschaulicht.

```
default=1
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Fedora (2.6.23.14-107.fc8)
 root (hd0,0)
 kernel /vmlinuz-2.6.23.14-107.fc8 ro root=/dev/VolGroup00/
LogVol00 rhgb quiet
 initrd /initrd-2.6.23.14-107.fc8.img
title Fedora (2.6.23.14-107.fc8xen)
 root (hd0,0)
 kernel /xen.gz-2.6.23.14-107.fc8
 module /vmlinuz-2.6.23.14-107.fc8xen ro root=/dev/
VolGroup00/LogVol00 rhgb quiet
 module /initrd-2.6.23.14-107.fc8xen.img
```

Beachten Sie den **default=1**-Parameter. Dadurch wird der GRUB Bootloader angewiesen, den zweiten Eintrag – also den Xen-Kernel – zu booten. Ändern Sie den Standard auf 0 (oder die Nummer für den Standard-Kernel):

```
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Fedora (2.6.23.14-107.fc8)
 root (hd0,0)
 kernel /vmlinuz-2.6.23.14-107.fc8 ro root=/dev/VolGroup00/
LogVol00 rhgb quiet
 initrd /initrd-2.6.23.14-107.fc8.img
title Fedora (2.6.23.14-107.fc8xen)
 root (hd0,0)
 kernel /xen.gz-2.6.23.14-107.fc8
 module /vmlinuz-2.6.23.14-107.fc8xen ro root=/dev/
VolGroup00/LogVol00 rhgb quiet
 module /initrd-2.6.23.14-107.fc8xen.img
```



### 3. Starten Sie neu, um den neuen Kernel zu laden

Starten Sie das System neu. Der Computer wird mit dem neuen Standard-Kernel starten. Das KVM-Modul sollte automatisch mit dem Kernel geladen werden. Überprüfen Sie, ob KVM läuft:

```
$ lsmod | grep kvm
kvm_intel 85992 1
kvm 222368 2 ksm,kvm_intel
```

Sofern alles einwandfrei geklappt hat, sind nun das **kvm**-Module sowie entweder das **kvm\_intel**-Modul oder das **kvm\_amd**-Modul vorhanden.

## 17.2.2. KVM zu Xen

Das folgende Verfahren beschreibt den Wechsel vom KVM-Hypervisor zum Xen-Hypervisor. Dieses Verfahren setzt voraus, dass das *kvm*-Paket installiert und aktiviert ist.

### 1. Installieren Sie die Xen-Pakete

Installieren Sie das *kernel-xen*-Paket, sofern Sie das nicht bereits getan haben.

```
yum install kernel-xen xen
```

Das *kernel-xen*-Paket kann installiert, aber deaktiviert sein.

### 2. Überprüfen Sie, welcher Kernel verwendet wird

Verwenden Sie den **uname**-Befehl, um festzustellen, welcher Kernel derzeit läuft.

```
$ uname -r
2.6.23.14-107.fc8
```

Der aktuelle Kernel, "**2.6.23.14-107.fc8**", läuft auf dem System. Dies ist der Standard-Kernel. Wenn der Kernel auf **xen** endet (z. B. **2.6.23.14-107.fc8xen**), dann läuft der Xen-Kernel bereits und Sie können diesen Unterschritt überspringen.

- **Wechsel vom Standard-Kernel zum Xen-Kernel**

In der **grub.conf**-Datei wird festgelegt, welcher Kernel gebootet wird. Um den Standard-Kernel zu ändern, bearbeiten Sie die **/boot/grub/grub.conf**-Datei wie unten veranschaulicht.

```
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Fedora (2.6.23.14-107.fc8)
 root (hd0,0)
 kernel /vmlinuz-2.6.23.14-107.fc8 ro root=/dev/VolGroup00/
LogVol00 rhgb quiet
 initrd /initrd-2.6.23.14-107.fc8.img
title Fedora (2.6.23.14-107.fc8xen)
 root (hd0,0)
 kernel /xen.gz-2.6.23.14-107.fc8
```

```
module /vmlinuz-2.6.23.14-107.fc8xen ro root=/dev/
VolGroup00/LogVol100 rhgb quiet
module /initrd-2.6.23.14-107.fc8xen.img
```

Beachten Sie den **default=0**-Parameter. Dadurch wird der GRUB Bootloader angewiesen, den ersten Eintrag – also den Standard-Kernel – zu booten. Ändern Sie den Standard auf **1** (oder die Nummer für den Xen-Kernel):

```
default=1
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Fedora (2.6.23.14-107.fc8)
 root (hd0,0)
 kernel /vmlinuz-2.6.23.14-107.fc8 ro root=/dev/VolGroup00/
LogVol100 rhgb quiet
 initrd /initrd-2.6.23.14-107.fc82.6.23.14-107.fc8.img
title Fedora (2.6.23.14-107.fc8xen)
 root (hd0,0)
 kernel /xen.gz-2.6.23.14-107.fc8
 module /vmlinuz-2.6.23.14-107.fc8xen ro root=/dev/
VolGroup00/LogVol100 rhgb quiet
 module /initrd-2.6.23.14-107.fc8xen.img
```

### 3. Starten Sie neu, um den neuen Kernel zu laden

Starten Sie das System neu. Der Computer wird mit dem neuen Xen-Kernel starten. Überprüfen Sie dies mit dem **uname**-Befehl:

```
$ uname -r
2.6.23.14-107.fc8xen
```

Falls die Ausgabe auf xen endet, wird der Xen-Kernel ausgeführt.

## 17.3. Verwenden von qemu-img

Das **qemu-img**-Befehlszeilen-Tool wird zur Formatierung verschiedener von Xen und KVM genutzter Dateisysteme verwendet. **qemu-img** sollte verwendet werden, um virtualisierte Gastabbilder zu formatieren, zusätzliche Speichergeräte und Netzwerkspeicher hinzuzufügen. Die Verwendung und die verfügbaren Optionen von **qemu-img** werden im Folgenden erläutert.

### Formatierung und Erstellung neuer Abbilder oder Geräte

Erstellen Sie das neue Disk-Abbild "filename" mit der Größe "size" und dem Format "format".

```
qemu-img create [-6] [-e] [-b base_image] [-f format] filename [size]
```

Falls **base\_image** spezifiziert wird, so wird das Abbild nur die Unterschiede zu **base\_image** aufzeichnen. In diesem Fall ist keine Angabe der Größe nötig. **base\_image** wird nie modifiziert werden, es sei denn, Sie verwenden den "commit" Monitor-Befehl.

## Umwandeln eines existierenden Abbilds in ein anderes Format

Die "convert"-Option wird verwendet, um ein anerkanntes Format in ein anderes Abbildformat umzuwandeln.

Befehlsformat:

```
qemu-img convert [-c] [-e] [-f format] filename [-O output_format]
output_filename
```

Konvertieren Sie das Disk-Abbild "filename" zu Disk-Abbild "output\_filename" unter Verwendung des Formats "output\_format". Optional kann es verschlüsselt ("-e"-Option) oder komprimiert ("-c"-Option) werden.

Nur das Format "qcow" unterstützt Verschlüsselung oder Komprimierung. Die Komprimierung ist schreibgeschützt. Das heißt, wenn ein komprimierter Sektor neu geschrieben wird, wird dieser unkomprimiert beschrieben.

Die Verschlüsselung verwendet das AES-Format mit sehr sicherer 128 Bit Verschlüsselung. Wählen Sie ein langes Passwort (16 Zeichen) für höchstmögliche Sicherheit.

Abbildkonvertierung ist außerdem sehr hilfreich, um kleinere Abbilder zu erhalten, wenn Sie ein Abbildformat verwenden, das wachsen kann wie z. B. **qcow** oder **cow**. Die leeren Sektoren werden erkannt und für das Zielabbild ausgeblendet.

## Erhalten von Abbildinformationen

Der **info**-Parameter zeigt Informationen über ein Disk-Abbild. Das Format der **info**-Option ist wie folgt:

```
qemu-img info [-f format] filename
```

Gibt Informationen über das Disk-Abbild "filename". Verwenden Sie dies insbesondere dazu, die reservierte Größe auf der Festplatte festzustellen, denn diese kann abweichen von der angezeigten Größe. Falls vm-Schnappschüsse im Disk-Abbild gespeichert sind, werden auch diese angezeigt.

## Unterstützte Formate

Das Format eines Abbilds wird normalerweise automatisch erkannt. Die folgenden Formate werden unterstützt:

### raw

Raw Disk-Abbildformat (Standard). Dieses Format hat den Vorteil, dass es einfach exportierbar auf alle anderen Emulatoren ist. Falls Ihr Dateisystem LÖcher unterstützt (z. B. ext2 oder ext3 unter Linux bzw. NTFS unter Windows), dann werden nur die beschriebenen Sektoren Platz belegen. Verwenden Sie **qemu-img info**, um die Größe festzustellen, die wirklich vom Abbild gebraucht wird, oder **ls -ls** unter Unix/Linux.

### qcow2

QEMU Abbildformat, das vielseitigste Format. Verwenden Sie es, um kleinere Abbilder zu erhalten, (nützlich, wenn Ihr Dateisystem keine LÖcher unterstützt, z. B. unter Windows), optional AES-Verschlüsselung, zlib-basierte Komprimierung und Unterstützung verschiedener VM-Schnappschüsse.

### qcow

Altes QEMU Abbildformat. Nur zwecks Kompatibilität mit älteren Versionen enthalten.

### cow

Benutzermodus Linux "Copy On Write" Abbildformat. Das **cow**-Format ist nur zwecks Kompatibilität mit älteren Versionen enthalten. Es funktioniert nicht mit Windows.

### vmdk

VMware 3 und 4 kompatibles Abbildformat.

### cloop

Linux Compressed Loop Image, nur nützlich zum Wiederverwenden direkt komprimierter CD-ROM Abbilder, wie sie z. B. auf Knoppix CD-ROMs vorliegen.

## 17.4. Overcommitting mit KVM

Der KVM-Hypervisor unterstützt den sog. CPU-Overcommit und Speicher-Overcommit. Beim Overcommitting werden mehr virtualisierte CPUs oder mehr Speicher zugewiesen, als physische Ressourcen auf dem System vorhanden sind. Mit CPU-Overcommit können nicht voll ausgelastete virtualisierte Server oder Desktops auf weniger physischen Servern laufen, was sowohl Energie als auch Geld spart.



### Xen-Unterstützung

CPU-Overcommitting wird nicht unterstützt für den Xen-Hypervisor. Overcommitting von CPUs mit dem Xen-Hypervisor kann zur Systeminstabilität und zu Abstürzen des Hosts und der virtualisierten Gäste führen.

### Overcommitting von Speicher

Die meisten Betriebssysteme und Anwendungen nutzen nicht ständig 100% des zur Verfügung stehenden RAM. Dieses Verhalten kann mit KVM ausgenutzt werden, um mehr Speicher für virtualisierte Gäste zu verwenden, als physisch verfügbar ist.

Unter KVM werden virtuelle Maschinen wie Linux-Prozesse gehandhabt. Gäste auf dem KVM-Hypervisor haben keine zugewiesenen Blöcke physischen RAMs, sondern funktionieren stattdessen als Prozesse. Jedem Prozess wird Speicher zugewiesen, wenn dieser mehr Speicher anfragt. KVM nutzt dies, um Speicher für Gäste zuzuweisen, wenn das Gastbetriebssystem mehr oder weniger Speicher anfragt. Der Gast braucht nur wenig mehr physischen Speicher, als das virtualisierte Betriebssystem zu verwenden scheint.

Wenn der physische Speicher nahezu vollständig verbraucht ist oder der Prozess seit einer gewissen Zeit inaktiv ist, verlegt Linux den Speicher des Prozesses in den Swap-Bereich. Swap-Speicher ist in der Regel eine Partition auf einem Festplattenlaufwerk (HDD) oder Festkörperlaufwerk (SSD), das Linux zum Erweitern des virtuellen Speichers nutzt. Swap ist deutlich langsamer als RAM:

Da virtuelle Maschinen unter KVM Linux-Prozesse sind, kann der Speicher, der vom virtualisierten Gast verwendet wird, in den Swap-Bereich ausgelagert werden, wenn der Gast gerade wenig oder gar nicht ausgelastet ist. Speicher kann zugewiesen werden über die Gesamtgröße des Swap und physischen RAMs. Dies kann zu Problemen führen, wenn virtualisierte Gäste ihren RAM vollständig verbrauchen. Wenn nicht ausreichend Swap-Space für die Prozesse der virtuellen Maschinen zur Verfügung steht, beginnt der **pdflush**-Prozess. **pdflush** beendet Prozesse, um

Speicher freizugeben und das System so vor einem Absturz zu bewahren. **pdflush** kann Fehler im Dateisystem verursachen und kann dazu führen, dass die virtualisierten Gäste nicht mehr starten können.



### Warning

Falls nicht genügend Swap-Speicher verfügbar ist, werden Gastbetriebssysteme zwangsweise heruntergefahren. Dies kann dazu führen, dass Gäste funktionsunfähig werden. Sie sollten dies vermeiden und daher niemals mehr Speicher zuweisen, als Swap zur Verfügung steht.

Die Swap-Partition wird verwendet, um wenig verwendeten Speicher zur Festplatte auszulagern, so dass die Speicherleistung erhöht wird. Die Standardgröße der Swap-Partition wird auf Grundlage der RAM-Menge und Overcommit-Rate berechnet. Es wird empfohlen, Ihre Swap-Partition größer anzulegen, wenn Sie beabsichtigen, Speicher-Overcommit mit KVM durchzuführen. Die empfohlene Rate für das Overcommitting liegt bei 50% (0,5). Die verwendete Formel lautet:

$$(0,5 \times \text{RAM}) + (\text{Overcommit-Rate} \times \text{RAM}) = \text{Empfohlene Swap-Größe}$$

In der Red Hat Knowledgebase finden Sie einen Artikel darüber, wie die Größe der Swap-Partition sicher und effizient bestimmt werden kann — siehe [Knowledgebase](#)<sup>1</sup>.

Es ist möglich, eine Overcommit-Rate von mehr als dem Zehnfachen der Anzahl virtualisierter Gäste über dem physischen RAM des Systems zu haben. Dies funktioniert nur mit bestimmten Anwendungsauslastungen (z. B. Desktop-Virtualisierung mit weniger als 100% Auslastung). Die Formel zum Einstellen der Overcommit-Rate ist nicht sehr kompliziert, Sie müssen nur Ihre Rate testen und an Ihre Umgebung anpassen.

### Overcommitting von virtualisierten CPUs

Der KVM-Hypervisor unterstützt das Overcommitting von virtualisierten CPUs. Das Overcommitting von virtualisierten CPUs wird nur durch die Auslastungsgrenze virtualisierter Gäste beschränkt. Seien Sie beim Overcommitting von virtualisierten CPUs jedoch vorsichtig, denn Auslastungen von beinahe 100% können dazu führen, dass Anfragen verworfen werden oder die Antwortzeiten untragbar lang werden.

Das Overcommitting von virtualisierten CPUs ist am besten, wenn jeder virtualisierte Gast nur über eine einzige VCPU verfügt. Der Linux Scheduler arbeitet sehr effizient mit dieser Art Auslastung. KVM sollte Gäste mit Auslastungen unter 100% bei einer Rate von 5 VCPUs sicher unterstützen. Overcommitting von virtualisierten Gästen mit einfachem VCPU stellt kein Problem dar.

Sie können kein Overcommitting bei Gästen mit symmetrischem Multiprocessing für mehr als die Anzahl physischer Prozessorkerne durchführen. Zum Beispiel sollte ein Gast mit vier VCPUs nicht auf einem Host mit Dual Core Prozessor ausgeführt werden. Overcommitting bei Gästen mit symmetrischem Multiprocessing für mehr als die Anzahl physischer Prozessorkerne hat deutliche Leistungseinbußen zur Folge.

Sie können Gästen VCPUs bis zur Anzahl der physischen Prozessorkerne zuweisen, dies funktioniert einwandfrei. So können Sie beispielsweise virtualisierte Gäste mit vier VCPUs auf einem Quad Core Host ausführen. Gäste mit Auslastungen von unter 100% sollten mit dieser Konfiguration effektiv arbeiten können.

<sup>1</sup> <http://kbase.redhat.com/faq/docs/DOC-15252>



### Grundsätzlich vorher testen

Wenden Sie in einer Produktionsumgebung kein Overcommitting von Speicher oder CPUs an, ohne vorher umfangreiche Tests durchgeführt zu haben. Anwendungen, die 100% des Speichers oder der Prozessorressourcen brauchen, können in Umgebungen mit Overcommitment instabil werden. Testen Sie also gründlich vor dem Einsatz.

## 17.5. Modifizieren von `/etc/grub.conf`

Dieser Abschnitt beschreibt, wie Sie Ihre `/etc/grub.conf`-Datei sicher und richtig ändern, um den virtualisierten Kernel zu verwenden. Sie müssen den xen-Kernel verwenden, um den Xen-Hypervisor auszuführen. Kopieren Sie ihren existierenden xen-Kernel-Eintrag. Stellen Sie dabei sicher, dass Sie alle der wichtigen Zeilen kopieren, ansonsten wird Ihr System Probleme beim Booten haben (`initrd` wird die Länge `0` haben). Wenn Sie xen-Hypervisor-spezifische Werte benötigen, müssen Sie diese in der xen-Zeile Ihres Grub-Eintrages hinzufügen.

Die Ausgabe unten ist ein Beispiel für einen `grub.conf`-Eintrag eines Systems, auf dem das `kernel-xen`-Paket läuft. Die `grub.conf` auf Ihrem System kann davon abweichen. Der wichtige Teil in dem unteren Beispiel ist der Abschnitt ab der `title`-Zeile bis zum Beginn der nächsten neuen Zeile.

```
#boot=/dev/sda
default=0
timeout=15
#splashimage=(hd0,0)/grub/splash.xpm.gz hiddenmenu
serial --unit=0 --speed=115200 --word=8 --parity=no --stop=1
terminal --timeout=10 serial console

title Fedora (2.6.23.14-107.fc8xen)
 root (hd0,0)
 kernel /xen.gz-2.6.23.14-107.fc8 com1=115200,8n1
 module /vmlinuz-2.6.23.14-107.fc8xen ro root=/dev/VolGroup00/
LogVol100
 module /initrd-2.6.23.14-107.fc8xen.img
```



### Wichtiger Hinweis zum Bearbeiten von `grub.conf`

Ihre `grub.conf` kann sehr unterschiedlich aussehen, falls sie manuell bearbeitet wurde oder von einem Beispiel kopiert wurde.

Um die Menge an Arbeitsspeicher für Ihr Host-System zur Bootzeit auf 250 MB zu stellen, müssen Sie `dom0_mem=256M` in der xen-Zeile in Ihrer `grub.conf` eingeben. Sehen Sie eine modifizierte Version der Grub-Konfigurationsdatei im folgenden Beispiel:

```
#boot=/dev/sda
default=0
timeout=15
#splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
serial --unit=0 --speed=115200 --word=8 --parity=no --stop=1
```

```
terminal --timeout=10 serial console

title Fedora (2.6.23.14-107.fc8xen)
 root (hd0,0)
 kernel /xen.gz-2.6.23.14-107.fc8 com1=115200,8n1 dom0_mem=256MB
 module /vmlinuz-2.6.23.14-107.fc8xen ro
 root=/dev/VolGroup00/LogVol100
 module /initrd-2.6.23.14-107.fc8xen.img
```

## 17.6. Überprüfen der Virtualisierungserweiterungen

Dieser Abschnitt hilft Ihnen dabei festzustellen, ob Ihr System die nötigen Hardware-Virtualisierungserweiterungen besitzt. Virtualisierungserweiterungen (Intel VT oder AMD-V) sind für volle Virtualisierung erforderlich.



### Kann ich Virtualisierung ohne die Virtualisierungserweiterungen nutzen?

Falls Virtualisierungserweiterungen nicht vorhanden sind, können Sie die Xen-Paravirtualisierung mit dem Fedora *kernel-xen*-Paket nutzen.

Führen Sie den folgenden Befehl aus, um zu überprüfen, ob die CPU-Virtualisierungserweiterungen zur Verfügung stehen:

```
$ grep -E 'svm|vmx' /proc/cpuinfo
```

Die folgende Ausgabe enthält einen vmx-Eintrag, wodurch angezeigt wird, dass ein Intel-Prozessor mit den Intel VT Erweiterungen vorhanden ist:

```
flags : fpu tsc msr pae mce cx8 apic mtrr mca cmov pat pse36 clflush
 dts acpi mmx fxsr sse sse2 ss ht tm syscall lm constant_tsc pni
 monitor ds_cpl
 vmx est tm2 cx16 xtpr lahf_lm
```

Die folgende Ausgabe enthält einen svm-Eintrag, wodurch angezeigt wird, dass ein AMD-Prozessor mit den AMD-V Erweiterungen vorhanden ist:

```
flags : fpu tsc msr pae mce cx8 apic mtrr mca cmov pat pse36 clflush
 mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt lm 3dnowext 3dnow
 pni cx16
 lahf_lm cmp_legacy svm cr8legacy ts fid vid ttp tm stc
```

Der "flags:" Inhalt kann mehrmals erscheinen für jeden Hyperthread, Kern oder CPU auf dem System.

Die Virtualisierungserweiterungen können ggf. im BIOS deaktiviert sein. Falls die Erweiterungen nicht angezeigt werden oder volle Virtualisierung nicht funktioniert, werfen Sie einen Blick auf [Prozedur 19.1, „Aktivieren der Virtualisationserweiterungen im BIOS“](#).

### 17.7. Identifizieren des Gasttyps und der Implementierung

Das nachfolgende Skript kann identifizieren, ob die Umgebung, in der eine Anwendung oder ein Skript ausgeführt wird, eine paravirtualisierte Umgebung, ein voll virtualisierter Gast oder auf dem Hypervisor ist.

```
#!/bin/bash
declare -i IS_HVM=0
declare -i IS_PARA=0
check_hvm()
{
 IS_X86HVM="$(strings /proc/acpi/dsdt | grep int-xen)"
 if [x"${IS_X86HVM}" != x]; then
 echo "Guest type is full-virt x86hvm"
 IS_HVM=1
 fi
}
check_para()
{
 if $(grep -q control_d /proc/xen/capabilities); then
 echo "Host is dom0"
 IS_PARA=1
 else
 echo "Guest is para-virt domU"
 IS_PARA=1
 fi
}
if [-f /proc/acpi/dsdt]; then
 check_hvm
fi

if [${IS_HVM} -eq 0]; then
 if [-f /proc/xen/capabilities] ; then
 check_para
 fi
fi

if [${IS_HVM} -eq 0 -a ${IS_PARA} -eq 0]; then
 echo "Baremetal platform"
fi
```



#### Untersuchen von Hosts

Um Hosts zu untersuchen, verwenden Sie den Befehl `virsh capabilities`.

### 17.8. Generieren einer neuen, eindeutigen MAC-Adresse

In manchen Fällen müssen Sie eine neue, eindeutige *MAC address* für Ihren Gast generieren. Es gibt derzeit kein Befehlszeilen-Tool, um neue MAC-Adressen zum Zeitpunkt des Schreibens zu generieren. Das unten bereitgestellte Skript kann neue MAC-Adressen für Ihre Gäste generieren.



Speichern Sie das Skript auf Ihren Gast als **macgen.py**. Nun können Sie von diesem Verzeichnis aus das Skript mittels **./macgen.py** starten und es wird eine neue MAC-Adresse generieren. Die Ausgabe sollte etwa wie dieses Beispiel aussehen:

```
$./macgen.py
00:16:3e:20:b0:11

#!/usr/bin/python
macgen.py script to generate a MAC address for virtualized guests on Xen
#
import random
#
def randomMAC():
 mac = [0x00, 0x16, 0x3e,
 random.randint(0x00, 0x7f),
 random.randint(0x00, 0xff),
 random.randint(0x00, 0xff)]
 return ':'.join(map(lambda x: "%02x" % x, mac))
#
print randomMAC()
```

### Eine andere Methode zum Generieren einer neuen MAC-Adresse für Ihren Gast

Sie können auch die integrierten Module von **python-virtinst** verwenden, um eine neue MAC-Adresse und **UUID** zur Verwendung in einer Gastkonfigurationsdatei zu generieren:

```
echo 'import virtinst.util ; print\
virtinst.util.uuidToString(virtinst.util.randomUUID())' | python
echo 'import virtinst.util ; print virtinst.util.randomMAC()' | python
```

Das obige Skript kann ebenfalls als eine Skriptdatei implementiert werden, wie unten gezeigt.

```
#!/usr/bin/env python
-*- mode: python; -*-
print ""
print "New UUID:"
import virtinst.util ; print
virtinst.util.uuidToString(virtinst.util.randomUUID())
print "New MAC:"
import virtinst.util ; print virtinst.util.randomMAC()
print ""
```

## 17.9. Very Secure ftpd

vsftpd bietet Zugang zu Installationsbäumen für paravirtualisierte Gäste oder anderen Daten. Falls Sie während der Server-Installation vsftpd noch nicht installiert haben, können Sie das RPM-Paket im **Server**-Verzeichnis Ihres Installationsmediums finden und es mittels **rpm -ivh vsftpd\*.rpm** installieren (beachten Sie, dass sich das RPM-Paket in Ihrem aktuellen Verzeichnis befinden muss).

1. Um vsftpd zu konfigurieren, bearbeiten Sie **/etc/passwd** mit **vipw** und ändern Sie das FTP-Benutzerverzeichnis auf das Verzeichnis, in dem Sie die Installationsbäume für Ihre

paravirtualisierten Gäste ablegen möchten. Ein Beispieleintrag für Ihren FTP-Benutzer könnte wie folgt aussehen:

```
ftp:x:14:50:FTP User:/xen/pub:/sbin/nologin
```

- Um `vsftpd` beim Hochfahren automatisch zu starten, verwenden Sie das `chkconfig`-Dienstprogramm, um den automatischen Start von `vsftpd` zu aktivieren.
- Überprüfen Sie mittels `chkconfig --list vsftpd`, dass `vsftpd` nicht aktiviert ist:

```
$ chkconfig --list vsftpd
vsftpd 0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

- Führen Sie `chkconfig --levels 345 vsftpd on` aus, damit `vsftpd` automatisch für Runlevel 3, 4 und 5 gestartet wird.
- Verwenden Sie den Befehl `chkconfig --list vsftpd`, um sich zu vergewissern, dass `vsftpd` für den automatischen Start beim Hochfahren des Systems aktiviert wurde:

```
$ chkconfig --list vsftpd
vsftpd 0:off 1:off 2:off 3:on 4:on 5:on 6:off
```

- Verwenden Sie `service vsftpd start vsftpd`, um den `vsftpd`-Dienst zu starten:

```
$service vsftpd start vsftpd
Starting vsftpd for vsftpd: [OK]
```

## 17.10. Konfiguration von LUN-Persistenz

Dieser Abschnitt beschreibt, wie *LUN*-Persistenz in Gästen und auf der Host-Maschine mit oder ohne Multipath implementiert werden kann.

### Implementierung von LUN-Persistenz ohne Multipath

Falls Ihr System kein Multipath verwendet, können Sie `udev` verwenden, um LUN-Persistenz zu implementieren. Bevor Sie die LUN-Persistenz auf Ihrem System implementieren, stellen Sie sicher, dass Sie die passenden UUIDs erhalten. Sobald Sie diese erhalten, können Sie die LUN-Persistenz konfigurieren, indem Sie die Datei `scsi_id` bearbeiten, die sich im Verzeichnis `/etc` befindet. Sobald Sie diese Datei in einem Texteditor geöffnet haben, müssen Sie die folgende Zeile auskommentieren:

```
options=-b
```

Ersetzen Sie dies anschließend mit diesem Parameter:

```
options=-g
```

Dies veranlasst `udev`, alle SCSI-Geräte des Systems auf zurückkehrende UUIDs zu überwachen. Um die UUIDs des Systems zu ermitteln, verwenden Sie den Befehl `scsi_id`:

```
scsi_id -g -s /block/sdc
3600a0b80001327510000015427b625e
```

Diese lange Zeichenkette ist die UUID. Die UUID verändert sich nicht, wenn Sie ein neues Gerät zu Ihrem System hinzufügen. Ermitteln Sie die UUID für jedes Gerät, um Regeln für diese Geräte erstellen zu können. Um neue Regeln für Geräte zu erstellen, müssen Sie die Datei **20-names.rules** bearbeiten, die sich im Verzeichnis **/etc/udev/rules.d** befindet. Die Regeln zur Benennung der Geräte haben das folgende Format:

```
KERNEL="sd*", BUS="scsi", PROGRAM="sbin/scsi_id", RESULT="UUID",
NAME="devicename"
```

Ersetzen Sie Ihre bestehende *UUID* und den *devicename* mit dem oben erhaltenen UUID-Eintrag. Die Regel sollte daher wie folgt lauten:

```
KERNEL="sd*", BUS="scsi", PROGRAM="sbin/scsi_id",
RESULT="3600a0b80001327510000015427b625e", NAME="mydevicename"
```

Dies veranlasst das System, sämtliche Geräte, die mit **/dev/sd\*** übereinstimmen, zu aktivieren, um die festgelegten UUID zu untersuchen. Wird ein passendes Gerät gefunden, wird ein Geräteknoten mit der Bezeichnung **/dev/devicename** erstellt. In diesem Beispiel ist der Geräteknoten **/dev/mydevice**. Abschließend müssen Sie noch die Datei **/etc/rc.local** anhängen mit der Zeile:

```
/sbin/start_udev
```

### Implementierung von LUN-Persistenz mit Multipath

Um LUN-Persistenz in einer Multipath-Umgebung zu implementieren, müssen Sie den Alias-Namen für die Multipath-Geräte definieren. Bei diesem Beispiel müssen Sie vier Geräte-Aliasse definieren, indem Sie die Datei **multipath.conf**, die sich im Verzeichnis **/etc/** befindet, bearbeiten:

```
multipath {
 wwid 3600a0b80001327510000015427b625e
 alias oramp1
}
multipath {
 wwid 3600a0b80001327510000015427b6
 alias oramp2
}
multipath {
 wwid 3600a0b80001327510000015427b625e
 alias oramp3
}
multipath {
 wwid 3600a0b80001327510000015427b625e
 alias oramp4
}
```

Dies definiert vier LUNs: **/dev/mpath/oramp1**, **/dev/mpath/oramp2**, **/dev/mpath/oramp3** und **/dev/mpath/oramp4**. Diese Geräte befinden sich im Verzeichnis **/dev/mpath**. Die LUN-Namen

bleiben auch über Neustarts hinaus bestehen, da Alias-Namen auf den wwid (weltweiten ID) der LUNs erstellt werden.

### 17.11. Abschalten der SMART-Disk Überwachung von Gästen

Die SMART-Disk Überwachung kann deaktiviert werden, da wir nur mit virtuellen Platten arbeiten und der eigentliche physische Speicher vom Host verwaltet wird.

```
/sbin/service smartd stop
/sbin/chkconfig --del smartd
```

### 17.12. Klonen von Gastkonfigurationsdateien

Sie können eine existierende Konfigurationsdatei kopieren, um einen neuen Gast zu erstellen. Sie müssen dazu nur den Namensparameter Ihrer Gastkonfigurationsdatei modifizieren. Der neue, eindeutige Name erscheint dann im Hypervisor und ist durch die Verwaltungsdienstprogramme einsehbar. Sie müssen ebenfalls eine neue UUID generieren, indem Sie den Befehl **uuidgen** ausführen. Für die **vif**-Einträge müssen Sie eine eindeutige MAC-Adresse für jeden Gast definieren (falls Sie eine Gastkonfiguration von einem existierenden Gast kopieren, können Sie dazu ein Skript erstellen). Falls Sie eine existierende Gastkonfigurationsdatei auf einen neuen Host verschieben, müssen Sie für die Xen-Bridge-Information den Eintrag **xenbr** aktualisieren, damit er mit Ihren lokalen Netzwerkkonfigurationen übereinstimmt. Für die Geräteinträge müssen Sie die Einträge in dem '**disk=**'-Abschnitt modifizieren, um auf das korrekte Gastabbild zu verweisen.

Sie müssen diese Systemkonfigurationseinstellungen auch auf Ihrem Gast modifizieren. Sie müssen den HOSTNAME-Eintrag der Datei **/etc/sysconfig/network** modifizieren, um mit den neuen Host-Namen des Gasts übereinzustimmen.

Sie müssen die **HWADDR**-Adresse der **/etc/sysconfig/network-scripts/ifcfg-eth0**-Datei modifizieren, um mit den Ausgabe von **ifconfig eth0** übereinzustimmen und falls Sie eine statische IP-Adresse verwenden, müssen Sie den Eintrag **IPADDR** modifizieren.

### 17.13. Kopieren eines existierenden Gasts und seiner Konfigurationsdatei

Dieser Abschnitt beschreibt das Kopieren von existierenden Konfigurationsdateien, um einen neuen Gast zu erstellen. Es gibt Schlüsselparameter in Ihrer Gastkonfigurationsdatei, auf die Sie achten müssen und die Sie modifizieren müssen, um einen Gast erfolgreich zu duplizieren.

**name**

Der Name Ihres Gasts, wie er im Hypervisor bekannt ist und in Verwaltungsdiensten angezeigt wird. Dieser Eintrag sollte in Ihrem System eindeutig sein.

**uuid**

Eine eindeutige Kennung für den Gast, eine neue UUID kann durch den Befehl **uuidgen** erstellt werden. Eine Beispiel-Ausgabe der UUID:

```
$ uuidgen
```

```
a984a14f-4191-4d14-868e-329906b211e5
```

vif

- Die *MAC address* muss eine eindeutige MAC-Adresse für jeden Gast definieren. Dies geschieht automatisch, wenn die Standard-Tools verwendet werden. Falls Sie eine Gastkonfiguration von einem bereits existierenden Gast kopieren, können Sie das Skript unter [Abschnitt 17.8](#), „Generieren einer neuen, eindeutigen MAC-Adresse“ verwenden.
- Falls Sie eine existierende Gastkonfigurationsdatei auf einen neuen Host verschieben oder kopieren, müssen Sie sicherstellen, dass Sie den `xenbr`-Eintrag anpassen, um mit Ihrer lokalen Netzwerkkonfiguration übereinzustimmen (Sie können die Bridge-Informationen mit Hilfe des Befehls `brctl show` erhalten.)
- Stellen Sie sicher, dass Sie die Geräteeinträge in dem `disk=`-Abschnitt eingestellt haben, um auf das richtige Gastabbild zu verweisen.

Passen Sie nun die Systemkonfigurationseinstellungen auf Ihrem Gast an:

**/etc/sysconfig/network**

Ändern Sie den `HOSTNAME`-Eintrag auf den neuen `hostname` des Gasts.

**/etc/sysconfig/network-scripts/ifcfg-eth0**

- Ändern Sie die `HWADDR`-Adresse auf die Ausgabe von `ifconfig eth0`.
- Modifizieren Sie den `IPADDR`-Eintrag, falls statische IP-Adressen verwendet werden.



---

# Erstellung angepasster libvirt-Skripte

Die Informationen in diesem Abschnitt ist hilfreich für Programmierer und Systemadministratoren, die angepasste Skripte schreiben wollen und dies mit Hilfe von **libvirt** vereinfachen können.

[Kapitel 17, Tipps und Tricks](#) sollte von Programmierern gelesen werden, die beabsichtigen, neue Anwendungen mit Hilfe von **libvirt** zu schreiben.

## 18.1. Benutzung von XML-Konfigurationsdateien mit virsh

**virsh** kann mit XML-Konfigurationsdateien umgehen. Sie können dies zu ihrem Vorteil verwenden beim Skripting großer Deployments mit speziellen Optionen. Sie können Geräte, die in einer XML-Datei definiert sind, zu einem laufenden paravirtualisierten Gast hinzufügen. Um beispielsweise eine ISO-Datei als **hdc** zu einem laufenden Gast hinzuzufügen, erstellen Sie eine XML-Datei:

```
cat satelliteiso.xml
<disk type="file" device="disk">
 <driver name="file"/>
 <source file="/var/lib/libvirt/images/rhn-satellite-5.0.1-11-
redhat-linux-as-i386-4-embedded-oracle.iso"/>
 <target dev="hdc"/>
 <readonly/>
</disk>
```

Führen Sie **virsh attach-device** aus, um das ISO als **hdc** mit einem Gast namens "satellite" zu verknüpfen:

```
virsh attach-device Satellite satelliteiso.xml
```





---

## Teil VI. Troubleshooting

# Einführung in die Suche und Beseitigung von Fehlern (Troubleshooting)

Das folgende Kapitel bietet Informationen, die Ihnen bei Suche und Beseitigung von Fehlern, die beim Einsatz der Virtualisierung auftreten können, helfen sollen.



### Wichtiger Hinweis zu Virtualisierungsproblemen

Aufgrund der ständigen Weiterentwicklung, bei der sowohl Fehler behoben als auch neue verursacht werden, ist Ihr spezielles Problem eventuell noch nicht in diesem Handbuch enthalten. Werfen Sie für die aktuellste Liste aller bekannten Fehler, Probleme und Problemlösungen einen Blick auf die Fedora *Release Notes* für Ihre Version und Hardware-Architektur. Die *Release Notes* finden Sie im Dokumentationsbereich der Fedora-Website, <http://docs.fedoraproject.org>.

---

---

---

# Troubleshooting

Dieses Kapitel behandelt häufige Probleme mit Fedora Virtualisierung und deren Lösungen.

## 19.1. Fehler bei Loop-Gerät

Falls dateibasierte Gastabbilder verwendet werden, müssen Sie evtl. die Anzahl von konfigurierten Loop-Geräten erhöhen. Die standardmäßige Konfiguration erlaubt bis zu acht aktive Loop-Geräte. Falls mehr als acht dateibasierte Gäste oder Loop-Geräte gebraucht werden, kann dies unter `/etc/modprobe.conf` eingestellt werden. Bearbeiten Sie `/etc/modprobe.conf` und fügen Sie die nachfolgende Zeile ein:

```
options loop max_loop=64
```

Dieses Beispiel verwendet 64, aber Sie können auch eine andere Anzahl als maximalen Loop-Wert spezifizieren. Sie müssen evtl. auch Loop-Geräte unterstützte Gäste implementieren. Um dies für ein Loop-Gerät unterstützten Gast für einen paravirtualisierten Gast anzulegen, verwenden Sie `phy: block device` oder `tap:aio`. Um dasselbe für einen voll virtualisierten Gast zu machen, verwenden Sie den Befehl `phy: device` oder `file: file`.

## 19.2. Aktivieren der Intel VT und AMD-V Virtualisierungs-Hardware-Erweiterungen im BIOS

Dieser Abschnitt beschreibt, wie Hardware-Virtualisierungserweiterungen identifiziert und in Ihrem BIOS aktiviert werden können, falls sie deaktiviert sind.

Die Intel VT Erweiterungen können im BIOS deaktiviert werden. Bestimmte Laptop-Hersteller haben die Intel VT Erweiterungen standardmäßig in Ihren CPUs deaktiviert.

Die Virtualisierungserweiterungen können für AMD-V-fähige Prozessoren, die in einem Rev 2 Socket installiert sind, nicht im BIOS deaktiviert werden.

Die Virtualisierungserweiterungen wurden mitunter im BIOS deaktiviert, in der Regel durch den Laptop-Hersteller. Anweisungen zum Aktivieren von deaktivierten Virtualisierungserweiterungen finden Sie unter [Abschnitt 19.2, „Aktivieren der Intel VT und AMD-V Virtualisierungs-Hardware-Erweiterungen im BIOS“](#).

Vergewissern Sie sich zunächst, dass die Virtualisierungserweiterungen im BIOS aktiviert sind. Die BIOS-Einstellungen für Intel® oder AMD-V befinden sich normalerweise in den **Chipsatz** oder **Prozessor**-Menüs. Die Menünamen können jedoch von den Angaben in diesem Handbuch abweichen, und die Einstellungen für die Virtualisierungserweiterungen liegen möglicherweise unter **Sicherheitseinstellungen** oder anderen obskuren Menüs.

### Prozedur 19.1. Aktivieren der Virtualisationserweiterungen im BIOS

1. Starten Sie den Computer neu und öffnen Sie das BIOS-Menü des Systems. Dies funktioniert normalerweise mit **Entfernen** oder **Alt + F4**.
2. Wählen Sie **Restore Defaults**, und danach **Save & Exit** aus.
3. Schalten Sie die Maschine aus und nehmen Sie sie zusätzlich vom Netz.

4. Schalten Sie die Maschine ein und öffnen Sie **BIOS Setup Utility**. Öffnen Sie die den **Processor**-Abschnitt und aktivieren Sie **Intel®Virtualization Technology** oder **AMD-V**. Die Werte können auf manchen Maschinen auch den Namen **Virtualization Extensions** haben. Wählen Sie **Save & Exit** aus.
5. Schalten Sie die Maschine aus und nehmen Sie sie zusätzlich vom Netz.
6. Führen Sie `cat /proc/cpuinfo | grep vmx svm` aus. Falls der Befehl eine Ausgabe liefert, sind die Virtualisierungserweiterungen nunmehr aktiviert. Falls keine Ausgabe erscheint, verfügt Ihr System eventuell nicht über die Virtualisierungserweiterungen oder es wurden nicht die richtigen BIOS-Einstellungen aktiviert.

---

# Anhang A. Zusätzliche Informationsquellen

Um mehr über Virtualisierung und Linux zu erfahren, werfen Sie einen Blick auf die folgenden Quellen.

## A.1. Online-Informationsquellen

- <http://www.cl.cam.ac.uk/research/srg/netos/xen/> Die Projekt-Website des Xen™-Paravirtualisierungs-Maschinen-Managers, von dem das Fedora *kernel-xen*-Paket abgeleitet ist. Die Seite pflegt die Upstream Xen-Projekt-Binärdateien und den Quellcode und enthält weiterhin Informationen, Überblick über Architekturen, Dokumentation und Links zum Thema Xen sowie damit verbundenen Technologien.
- Die Website der Xen-Gemeinschaft  
<http://www.xen.org/>
- <http://www.libvirt.org/> ist die offizielle Website der **libvirt**-Virtualisierungs-API.
- <http://virt-manager.et.redhat.com/> ist die Projekt-Website für den **Virtual Machine Manager** (*virt-manager*), der grafischen Anwendung zur Verwaltung von virtuellen Maschinen.
- Open Virtualization Center  
<http://www.openvirtualization.com><sup>1</sup>
- Fedora Dokumentation  
<http://docs.fedoraproject.org>
- Überblick über Virtualisierungstechnologien  
<http://virt.kernelnewbies.org><sup>2</sup>
- Red Hat Emerging Technologies Gruppe  
<http://et.redhat.com><sup>3</sup>

## A.2. Installierte Dokumentation

- `/usr/share/doc/xen-<version-number>/` ist das Verzeichnis, das umfassende Informationen über den Xen-Paravirtualisierungs-Hypervisor und damit verbundene Verwaltungstools enthält, inklusive verschiedener Beispielkonfigurationen, hardware-spezifische Informationen sowie der aktuellen Xen-Upstream-Benutzerdokumentation.
- `man virsh` und `/usr/share/doc/libvirt-<version-number>` — Enthält Unterbefehle und -optionen für das `virsh`-Dienstprogramm zur Verwaltung virtueller Maschinen, sowie umfassende Informationen über die **libvirt**-Virtualisierungsbibliothek-API.
- `/usr/share/doc/gnome-applet-vm-<version-number>` — Dokumentation für das grafische Menüleisten-Applet von GNOME, das lokal laufende virtuelle Maschinen überwacht und verwaltet.

- `/usr/share/doc/libvirt-python-<version-number>` — Liefert Details zu den Python-Bindings für die **libvirt**-Bibliothek. Das Paket **libvirt-python** ermöglicht Python-Entwicklern das Erstellen von Programmen, die eine Schnittstelle zur **libvirt**-Virtualisierungs-Management-Bibliothek darstellen.
- `/usr/share/doc/python-virtinst-<version-number>` — Liefert Dokumentation zum Befehl **virt-install**, der beim Starten der Installation von Fedora- und Linux-Distributionen innerhalb virtueller Maschinen behilflich ist.
- `/usr/share/doc/virt-manager-<version-number>` — Liefert Dokumentation zum Virtual Machine Manager, einem grafischen Tool zur Verwaltung von virtuellen Maschinen.

---

# Anhang B. Versionsgeschichte

Version 12.1.3 Mon Oct 12 2009

Christopher Curran [ccurran@redhat.com](mailto:ccurran@redhat.com)

Abspaltung vom Red Hat Enterprise Linux 5 Virtualisierungshandbuch Version 5.4-61.

---



---

# Anhang C. Kolophon

Dieses Handbuch wurden im DocBook XML v4.3 Format verfasst.

Dieses Handbuch basiert auf der Arbeit von Jan Mark Holzer und Chris Curran.

Weitere mitwirkende Autoren sind:

- Don Dutile trug zur technischen Bearbeitung des Abschnitts über paravirtualisierte Treiber bei.
- Barry Donahue trug zur technischen Bearbeitung des Abschnitts über paravirtualisierte Treiber bei.
- Rick Ring trug zur technischen Bearbeitung des Abschnitts über den Virtual Machine Manager bei.
- Michael Kearey trug zur technischen Bearbeitung der Abschnitte über die Verwendung von XML-Konfigurationsdateien mit Virsh und virtualisierten Floppy-Laufwerken bei.
- Marco Grigull trug zur technischen Bearbeitung des Abschnitts über Software-Kompatibilität und Performance bei.
- Eugene Teo trug zur technischen Bearbeitung des Abschnitts über die Verwaltung von Gästen mit virsh bei.

Jeffrey Fearn schrieb das Publishing-Tool Publican, mit dem dieses Buch erstellt wurde.

Das Red Hat Lokalisierungs-Team sind:

## Ostasiatische Sprachen

- Vereinfachtes Chinesisch
  - Leah Wei Liu
- Traditionelles Chinesisch
  - Chester Cheng
  - Terry Chuang
- Japanisch
  - Junko Ito
- Koreanisch
  - Eun-ju Kim

## Lateinische Sprachen

- Französisch
  - Sam Friedmann
- Deutsch
  - Hedda Peters

- Italienisch
  - Francesco Valente
- Brasilianisches Portugiesisch
  - Glaucia de Freitas
  - Leticia de Lima
- Spanisch
  - Angela Garcia
  - Gladys Guerrero
- Russisch
  - Yuliya Poyarkova

---

# Glossar

Dieses Glossar definiert die Ausdrücke, die in diesem Installationshandbuch verwendet werden.

Bare-Metal	Der Begriff Bare-Metal bezieht sich auf die zu Grunde liegende physische Architektur eines Computers. Ein Betriebssystem auf Bare-Metal laufen zu lassen bedeutet, dass eine unveränderte Version des Betriebssystems auf der physischen Hardware läuft. Beispiele für Betriebssysteme, die auf Bare-Metal laufen, ist <i>dom0</i> oder ein nativ installiertes Betriebssystem.
dom0	Auch als <i>Host</i> bekannt oder Host-Betriebssystem.  <b>dom0</b> bezeichnet die Host-Instanz von Linux, auf welcher der <i>Hypervisor</i> ausgeführt wird, der die Virtualisierung von Gastbetriebssystemen ermöglicht. Dom0 läuft auf der physischen Hardware und verwaltet ebendiese Hardware sowie die Ressourcenzuteilung für sich selbst und Gastbetriebssysteme.
Domains	<i>domU</i> und <i>Domains</i> sind beides Domains. Domains laufen auf dem <i>Hypervisor</i> . Der Ausdruck Domains hat eine ähnliche Bedeutung wie <i>Virtuelle Maschinen</i> und die beiden sind im Prinzip austauschbar. Eine Domain ist eine virtuelle Maschine.
domU	<b>domU</b> bezeichnet das Gastbetriebssystem, das auf dem Host-System ( <i>Domains</i> ) läuft.
Volle Virtualisierung	Xen und KVM können volle Virtualisierung durchführen. Volle Virtualisierung nutzt Hardware-Features des Prozessors, um eine totale Abstrahierung des zu Grunde liegenden physischen Systems ( <i>Bare-Metal</i> ) zu erreichen und so ein neues virtuelles System zu erstellen, in dem das Gastbetriebssystem laufen kann. Es sind keine Anpassungen im Gastbetriebssystem notwendig. Das Gastbetriebssystem und jegliche Anwendungen auf dem Gast sind sich der virtuellen Umgebung nicht bewusst und laufen wie gewohnt. Paravirtualisierung erfordert eine angepasste Version des Linux-Betriebssystems.
Voll virtualisiert	Siehe <i>Volle Virtualisierung</i> .
Gastsystem	Auch Gäste, virtuelle Maschinen oder <i>domU</i> genannt.
Hardware Virtual Machine	Siehe <i>Volle Virtualisierung</i>
Hypervisor	Beim Hypervisor handelt es sich um die Software-Schicht, welche die Hardware von dem Betriebssystem trennt und dadurch ermöglicht, dass mehrere Betriebssysteme auf der gleichen Hardware laufen können. Der Hypervisor läuft auf dem Host-System und ermöglicht es virtuellen Maschinen, ebenfalls auf der Hardware des Hosts zu laufen.
Host	Das Host-Betriebssystem, auch <i>dom0</i> genannt.

Auf der Betriebssystemumgebung des Hosts läuft die Virtualisierungs-Software für *Voll virtualisiert* und *Paravirtualisiert* Gastsysteme.

I/O

Kurz für Eingabe/Ausgabe (Input/Output). Der Begriff I/O beschreibt alle Programme, Operationen oder Geräte, die Daten von bzw. auf einen Computer oder von bzw. auf ein Peripheriegerät übertragen. Jede Datenübertragung ist eine Ausgabe für ein Gerät und eine Eingabe für das andere Gerät. Geräte wie z. B. Tastatur und Maus sind ausschließlich Eingabegeräte, während Geräte wie z. B. Drucker ausschließlich Ausgabegeräte sind. Eine beschreibbare CD-ROM ist sowohl ein Eingabe- als auch ein Ausgabegerät.

Kernel-based Virtual Machine

KVM (Kernel-based Virtual Machine) ist eine Lösung für *Volle Virtualisierung* für Linux auf AMD64 und Intel 64 Hardware. VM ist ein Linux Kernel-Modul für den standardmäßigen Linux-Kernel. KVM kann mehrere, unmodifizierte virtuelle Windows- und Linux-Gastbetriebssysteme ausführen. KVM ist ein Hypervisor, der die libvirt-Virtualisierungs-Tools (virt-manager und virsh) nutzt.

Bei KVM handelt es sich um eine Gruppe von Linux Kernel-Modulen, die Geräte, Speicher und Management-APIs für das Hypervisor-Modul steuern. Virtualisierte Gäste werden als Linux-Prozesse und Threads ausgeführt, die von diesen Modulen gesteuert werden.

LUN

Logical Unit Number (LUN) ist die Nummer, die einer logischen Einheit (einer SCSI-Protokolleinheit) zugeordnet ist.

Migration

Migration bezeichnet den Vorgang, virtualisierte Gäste von einem Host auf einen anderen zu verschieben. Eine Migration kann "offline" erfolgen (wobei der Gast erst angehalten und dann verschoben wird) oder "live" (wobei der Gast im laufenden Betrieb verschoben wird). Sowohl Xen voll virtualisierte Gäste als auch Xen paravirtualisierte Gäste und KVM voll virtualisierte Gäste können migriert werden.

Migration ist eine Schlüsseleigenschaft der Virtualisierung, da die Software vollständig von der Hardware getrennt ist. Migration ist hilfreich für:

- Load balancing - guests can be moved to hosts with lower usage when a host becomes overloaded.
- Hardware failover - when hardware devices on the host start to fail, guests can be safely relocated so the host can be powered down and repaired.
- Energy saving - guests can be redistributed to other hosts and host systems powered off to save energy and cut costs in low usage periods.
- Geographic migration - guests can be moved to another location for lower latency or in serious circumstances.

---

Zur Speicherung von Gastabbildern wird gemeinsam genutzter Netzwerkspeicher verwendet. Ohne diesen gemeinsamen Speicher wäre eine Migration nicht möglich.

An offline migration suspends the guest then moves an image of the guests memory to the destination host. The guest is resumed on the destination host and the memory the guest used on the source host is freed.

Die Zeit, die eine Offline-Migration dauert, hängt von der Netzwerkbandbreite und Latenz ab. Ein Gast mit 2 GB Speicher sollte über eine 1 Gbit Ethernet-Verbindung einige Sekunden brauchen.

Bei einer Live-Migration läuft der Gast auf dem Quell-Host weiter ohne anzuhalten, während der Speicher verschoben wird. Alle modifizierten Speicherseiten werden nachverfolgt und zum Ziel gesendet, nachdem das Abbild übertragen wurde. Der Speicher wird dann mit den modifizierten Speicherseiten aktualisiert. Dieser Prozess läuft so lange, bis eine Heuristik erreicht wurde: Entweder wurden alle Seiten erfolgreich übertragen, oder die Quelle ändert sich zu rasch und der Ziel-Host kann keine Fortschritte erzielen. Sobald die Heuristik erreicht wurde, wird der Gast auf dem Quell-Host kurzzeitig angehalten, so dass die Register und Puffer übertragen werden können. Die Register werden nun auf dem neuen Host geladen und der Gast wird schließlich auf dem Ziel-Host wieder gestartet. Falls der Gast auf diese Weise nicht übertragen werden kann (was bei extrem hoher Auslastung des Gasts vorkommen kann), so wird er angehalten und stattdessen eine Offline-Migration eingeleitet.

Die Zeit, die eine Offline-Migration dauert, hängt von der Netzwerkbandbreite und Latenz sowie der Aktivität auf dem Gast ab. Bei hoher CPU-Beanspruchung oder erheblichem umfassenden I/O-Vorgängen dauert die Migration deutlich länger.

#### MAC-Adressen

Die Media-Access-Control-Adresse ist eine Hardware-Adresse eines Netzwerk-Interface-Controllers. Im Zusammenhang mit der Virtualisierung müssen MAC-Adressen für die virtuellen Netzwerk-Schnittstellen generiert werden, wobei jede MAC-Adresse einzigartig auf ihrer lokalen Domain sein muss.

#### Paravirtualisierung

Paravirtualisierung verwendet einen speziellen Kernel, den so genannten Xen-Kernel oder auch *kernel-xen*-Paket. Paravirtualisierte Gast-Kernel werden gleichzeitig auf dem Host ausgeführt und nutzen dabei die Bibliotheken und Geräte des Hosts. Eine paravirtualisierte Installation besitzt vollständigen Zugriff auf alle Geräte im System. Paravirtualisierung ist deutlich schneller als die volle Virtualisierung und kann effektiv für Lastverteilung, Provisioning, Sicherheit und andere Vorzüge genutzt werden.

Seit Fedora 9 wird kein spezieller Kernel mehr benötigt. Sobald dieser Patch im Haupt-Linux-Baum akzeptiert ist, werden alle Linux-Kernel

	nach dieser Version Paravirtualisierung aktiviert oder verfügbar haben.
Paravirtualisiert	Siehe <a href="#">Paravirtualisierung</a> ,
Paravirtualisierte Treiber	Paravirtualisierte Treiber sind Gerätetreiber, die auf voll virtualisierten Linux-Gästen laufen. Diese Treiber steigern die Leistungsfähigkeit von Netzwerk- und Blockgerät-I/O für vollvirtualisierte Gäste.
Security Enhanced Linux	Security Enhanced Linux, oder kurz SELinux, verwendet Linux Security Modules (LSM) im Linux-Kernel, um mit Hilfe einer Reihe von Sicherheitsrichtlinien minimal notwendige Privilegien zu implementieren. bieten
Universally Unique Identifier	Universally Unique Identifiers (UUIDs) sind Teil eines standardisierten Verfahrens zur Identifizierung von Systemen, Geräten und bestimmten Software-Objekten in verteilten Rechnernetzen. Bei der Virtualisierung verwendete UUID-Typen sind u. a.: ext 2 und ext 3-Dateisystem-Identifizier, RAID-Gerät-Identifizierers, -iuCSI an-Geräte-lice identifisowie Identifizier für ers-AAC addn undeand elle Rechnerferät.
Virtualization	<p>Virtualisierung ist der Überbegriff für die gleichzeitige Ausführung von Software (in der Regel Betriebssysteme) isoliert von anderen Programmen auf einem einzigen System. Die meisten derzeitigen Virtualisierungsimplementierungen nutzen einen Hypervisor, d. h. eine Software-Schicht auf einem Betriebssystem, um die Hardware zu abstrahieren. Der Hypervisor erlaubt es mehreren Betriebssystemen, auf demselben physischen System zu laufen, indem dem Gastbetriebssystem virtualisierte Hardware zur Verfügung gestellt wird. Es gibt verschiedenen Verfahren zur Virtualisierung von Betriebssystemen:</p> <ul style="list-style-type: none"><li>• Hardware-unterstützte Virtualisierung wird bei der vollen Virtualisierung mit Xen und KVM verwendet (Definition: <a href="#">Volle Virtualisierung</a>)</li><li>• Paravirtualisierung wird von Xen verwendet, um Linux-Gäste auszuführen (Definition: <a href="#">Paravirtualisierung</a>)</li><li>• Software-Virtualisierung oder Emulation. Software-Virtualisierung nutzt Binärübersetzung und andere Emulationstechniken, um unmodifizierte Betriebssysteme auszuführen. Software-Virtualisierung ist deutlich langsamer als hardwareunterstützte Virtualisierung oder Paravirtualisierung.</li></ul>
Virtuelle CPU	Ein System besitzt eine Anzahl virtueller CPUs (auch: VCPUs) relativ zur Anzahl der physischen Prozessorkerne. Die Anzahl der VCPUs ist begrenzt und repräsentiert die maximale Anzahl von virtuellen CPUs, die virtuellen Gastmaschinen zugewiesen werden können.
Virtuelle Maschinen	Eine virtuelle Maschine ist eine Software-Implementation einer physischen Maschine oder Programmiersprache (z. B. Java Runtime Environment oder LISP). Virtuelle Maschinen im Zusammenhang mit

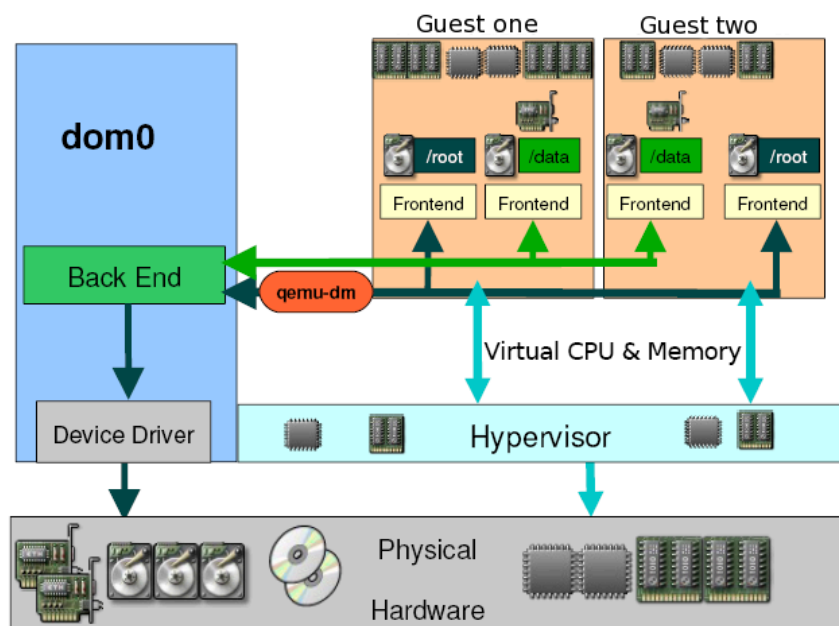
Virtualisierung sind Betriebssysteme, die auf virtualisierter Hardware laufen.

Xen

Fedora unterstützt den Xen-Hypervisor und den KVM-Hypervisor (siehe [Kernel-based Virtual Machine](#)). Beide Hypervisoren haben verschiedene Architekturen und Herangehensweisen der Entwicklung. Der Xen-Hypervisor läuft unterhalb eines Linux-Betriebssystems, das als Host fungiert und Systemressourcen und Virtualisierungs-APIs steuert. Der Host wird manchmal auch als *dom0* oder Domain0 bezeichnet.

## Xen Full Virtualization Architecture

With the para-virtualized drivers



# Xen Para-virtualization Architecture

