



# Red Hat Enterprise Linux 7

## 仮想化スタートガイド

---

仮想化の概念について

Dayle Parker  
Scott Radvan

Laura Novich

Jacquelynn East



## 仮想化の概念について

Dayle Parker  
Red Hat Customer Content Services  
dayleparker@redhat.com

Laura Novich  
Red Hat Customer Content Services  
lnovich@redhat.com

Jacquelynn East  
Red Hat Customer Content Services  
jeast@redhat.com

Scott Radvan  
Red Hat Customer Content Services  
sradvan@redhat.com

## 法律上の通知

Copyright © 2011-2015 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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.

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

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack® Word Mark and OpenStack Logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 概要

Red Hat Enterprise Linux 仮想化スタートガイドは、仮想化の基本事項と Red Hat Enterprise Linux で利用可能な仮想化製品および技術について説明しています。

## 目次

<b>第1章 仮想化とは、マイグレーションとは</b> .....	<b>2</b>
1.1. 仮想化とは	2
1.2. マイグレーションとは	2
<b>第2章 仮想化の利点と誤解</b> .....	<b>4</b>
2.1. 仮想化コスト	4
2.2. 仮想化の習熟曲線	4
2.3. パフォーマンス	4
2.4. 障害回復	5
2.5. セキュリティー	5
2.6. サーバー向け仮想化と個人向け仮想化	6
<b>第3章 Red Hat 仮想化製品について</b> .....	<b>7</b>
3.1. KVM および Red Hat Enterprise Linux での仮想化	7
3.2. libvirt および libvirt ツール	9
3.3. 仮想化ハードウェアデバイス	10
3.4. ストレージ	16
<b>第4章 仮想化ツール</b> .....	<b>20</b>
4.1. virsh	20
4.2. virt-manager	20
4.3. virt-install	20
4.4. guestfish	21
4.5. GNOME Boxes	21
4.6. 他の便利ツール	21
<b>第5章 Red Hat Enterprise Linux 7 での仮想化に関するクイックスタートガイド</b> .....	<b>26</b>
5.1. 基本要件および設定	26
5.2. 仮想マシンマネージャーでの仮想マシンのデプロイ	27
<b>付録A 改訂履歴</b> .....	<b>37</b>

# 第1章 仮想化とは、マイグレーションとは

本章では、仮想化とマイグレーションに関連する用語について説明します。

## 1.1. 仮想化とは

仮想化とは、単一システム上で他のプログラムとは切り離して、ソフトウェア（通常、複数のオペレーティングシステム）を同時に実行することを指す広範囲なコンピューター用語です。既存の仮想化実装の多くで、ハイパーバイザーが使用されています。ハイパーバイザーとは、ハードウェアを制御してゲストのオペレーティングシステムが基盤のハードウェアにアクセスできるようにするソフトウェア層またはサブシステムのことです。ハイパーバイザーを使用すると、仮想化ハードウェアをゲストオペレーティングシステムに提供することでゲストと呼ばれる複数のオペレーティングシステムが同じ物理システム上で実行できるようになります。オペレーティングシステムの仮想化には複数の方法があります。

### 仮想化メソッド

#### 完全仮想化

完全仮想化では、プロセッサのハードウェア機能を使用し、基盤の物理システムを完全に抽象化してゲストに提供します。これにより、*仮想マシン*と呼ばれる新規の仮想システムが作成され、変更なしにゲストオペレーティングシステムを実行できます。ゲストオペレーティングシステムとゲスト仮想マシン上のアプリケーションは、仮想化環境を意識せず、正常に実行されます。ハードウェアの仮想化支援機能 (HAV: Hardware-Assisted Virtualization) は、Red Hat Enterprise Linux で KVM (Kernel-based Virtual Machine) を使用した完全仮想化に使用される技術です。

#### 準仮想化

準仮想化は、仮想化ゲストに示されたソフトウェアとデータ構成のコレクションを使用しており、準仮想化環境で使用するにはゲストでソフトウェアの変更が必要になります。Xen 準仮想化ゲストや I/O デバイスを仮想化するドライバーの場合と同様に、準仮想化ではカーネルの領域を使用します。

#### ソフトウェアの仮想化 (またはエミュレーション)

ソフトウェアの仮想化は、より遅いバイナリー変換と他のエミュレーション技術を使用して、変換されていないオペレーティングシステムを実行します。Red Hat Enterprise Linux はソフトウェアの仮想化に対応していません。

### 注記

ゲストのインストールに関する詳細情報および説明は、『Red Hat Enterprise Linux 7 仮想化ホスト設定およびゲストインストールガイド』を参照してください。

## 1.2. マイグレーションとは

マイグレーションとは、ゲスト仮想マシンを任意のホストから別のホストに移行するプロセスです。マイグレーションが可能な理由は、仮想マシンが直接ハードウェア上ではなく仮想化環境内で実行されるためです。仮想マシンのマイグレーションには、ライブとオフラインの2つの方法があります。

### マイグレーションのタイプ

## オフラインマイグレーション

オフラインマイグレーションでは、ゲスト仮想マシンを一時停止またはシャットダウンさせて、仮想マシンのメモリーイメージを移行先のホストに移動します。次に、移行先のホスト上で仮想マシンを再開し、仮想マシンが移行元のホスト上で使用していたメモリーを解放します。

## ライブマイグレーション

ライブマイグレーションは、アクティブな仮想マシンを任意の物理ホストから別の物理ホストに移行するプロセスです。

このマイグレーションプロセスでは仮想マシンのメモリーが移動される点、また仮想マシンに関連付けられているディスクボリュームも移行される点を理解することが重要です。このプロセスはライブブロックマイグレーションを使用して実施されます。ライブブロックマイグレーションに関する情報は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

### 1.2.1. 仮想マシンのマイグレーションの利点

マイグレーションは以下の点で役立ちます。

#### 負荷分散

ホストマシンがオーバーロードとなった場合には、仮想マシンは単独でも多数でも、ライブマイグレーションで他のホストに移行できます。

#### ホストのアップグレードやホストに変更を加える場合

任意のホスト上にあるハードウェアデバイスにアップグレード、追加、削除などの必要性が生じた場合には、仮想マシンを安全に他のホストに移動させることができます。そのため、いずれかのホストへの変更が原因で、ゲスト側でダウンタイムが生じることはありません。

#### 省エネ

仮想マシンを別のホストに再配分することで、アンロードされたホストシステムの電源を電力使用量の少ない時間帯にオフにして節電とコスト削減が可能になります。

#### 地理的なマイグレーション

待ち時間の短縮や他の特別な状況のために、別の物理的場所に仮想マシンを移動することができます。

## 第2章 仮想化の利点と誤解

仮想化には多くの利点がありますが、おそらく同じくらいの誤解があるようです。本章では、これらの点について説明します。

### 2.1. 仮想化コスト

一般的な誤解は、仮想化への変更はコストが高すぎて正当化できない、というものです。仮想化の導入は費用がかかることもありますが、長期的には節約になる場合が多くあります。お使いの環境で仮想化の最適な使用方法を判断するには、投資対効果 (ROI) の分析を行うことが重要です。以下の利点を検討してみてください。

#### 省電力

仮想化により、複数の物理プラットフォームの必要性がほとんどなくなります。これは、マシン稼働や冷却に使われる電力の節約を意味し、エネルギー費用の削減につながります。マシンの消費電力と必要となる冷却を合わせると、複数の物理プラットフォーム購入の初期費用は、仮想化の利用により大幅に削減されます。

#### メンテナンスの省力化

物理システムから仮想化システムへの移行前に十分なプランニングを行うことで、メンテナンスにかかる時間は少なくなります。つまり、部品にかかる費用や人件費が削減されることとなります。

#### インストール済みソフトウェアの使用期限の延長

古いバージョンのソフトウェアは、新しいベアメタルマシン上では直接機能しない可能性があります。しかし、より大きく速い仮想システム上でこれらを稼働することにより、新しいシステムからのパフォーマンスを活用する一方で、古いソフトウェアの使用期限が延長される場合があります。

#### 予測可能なコスト

Red Hat Enterprise Linux のサブスクリプションは、仮想化サポートが固定料金なので、コスト予測が容易になります。

#### スペースの節約

サーバーをより少ないマシンに統合することは、必要な物理的スペースが減ることを意味します。つまり、通常サーバーハードウェアが占拠していたスペースを他の目的に使用できるようになります。

### 2.2. 仮想化の習熟曲線

仮想化を学ぶことは難しい、という誤解があります。実際には仮想化の習熟は、他の新たなプロセスと比べて難しくもなければ簡単でもありません。物理環境の管理とサポートで求められるスキルは、容易に仮想環境に移転することができます。仮想環境は物理環境と同様の動きをするので、学習量がわずかであることは間違いありません。

### 2.3. パフォーマンス

シングル CPU しかサポートしていなかった以前の仮想化バージョンでは、明らかに仮想マシンのパフォーマンスに制約がありました。これにより、仮想化ソリューションは遅い、という長期にわたる誤解が生まれました。



しかし、最近の状況は変わりました。最新の仮想化技術により、仮想マシンのスピードは格段に速くなりました。通常のサーバーアプリケーションの仮想マシン上での効率性は、ベアメタルシステム上とほぼ同一であることがベンチマークで示されています。

- ※ 業界標準である SAP Sales and Distribution (SD) Standard Application Benchmark は、Red Hat Enterprise Linux 6.2 および KVM を同一ハードウェア上で稼働するベアメタルシステムと比較した場合に 85% の仮想化効率を發揮したとしています。
- ※ Red Hat Enterprise Linux 6.1 および KVM は、標準性能評価法人 (Standard Performance Evaluation Corporation) が行った SPECvirt\_sc2010 ベンチマークで記録的な仮想化パフォーマンスを達成しました。これは公開されている SPECvirt 結果のなかでは最高の仮想パフォーマンス得点となりました。SPECvirt\_sc2010 メトリックは、仮想化データセンターサーバー内にあるシステムコンポーネントのコンドツーエンドのパフォーマンスを測定するものです。

## 注記

これら仮想化ベンチマークの詳細については、以下のサイトを参照してください。

- ※ Red Hat ナレッジベース、『SAP-SD Benchmark running in a VM – Leadership Performance using RHEL 6 / KVM』 <https://access.redhat.com/knowledge/articles/216943>
- ※ 『Standard Performance Evaluation Corporation (SPEC)』 <http://www.spec.org>
- ※ 『Red Hat Achieves New Top Virtualization Performance Benchmark with HP』 <http://investors.redhat.com/releasedetail.cfm?ReleaseID=617594>

## 2.4. 障害回復

システムが仮想化されていると、障害からの回復がより迅速で容易になります。物理システムでは、何らかの重大な障害が発生すると、通常オペレーティングシステムを完全に再インストールする必要が生じ、回復時間は何時間にも及びます。しかしシステムが仮想化されていれば、マイグレーション機能のおかげで回復ははるかに速くなります。ライブマイグレーションの要件に従っている場合には、仮想マシンは別のホスト上で再起動が可能になり、最大の遅延の理由として考えられるのは、ゲストデータの復元などでしょう。また、仮想化システムはそれぞれ完全に独立しているため、1つのシステムのダウンタイムが他のシステムに影響することはありません。

## 2.5. セキュリティー

仮想マシンは SELinux および sVirt を使用して、仮想化におけるセキュリティーを強化します。このセクションでは、利用可能なセキュリティーオプションの概要を説明します。

### 2.5.1. 仮想化セキュリティーの特徴

#### SELinux

SELinux は、Linux 用に Mandatory Access Control (MAC) を提供するために米国国家安全保障局などが開発しました。SELinux の管理下では、すべてのプロセスとファイルに **タイプ** が与えられ、アクセスはきめ細かい管理で制限されます。SELinux は攻撃者の能力を限定し、バッファオーバーフロー攻撃や権限エスカレーションといった多くの一般的なセキュリティーエクスプロイトを防ぎます。

SELinux は、Red Hat Enterprise Linux ホストと Red Hat Enterprise Linux の仮想化ゲストのセキュリティーモデルを強化します。SELinux はデフォルトで Red Hat Enterprise Linux 7 に同梱されているすべての仮想化ツールと機能するように設定/テストされています。

## sVirt

sVirt とは、Red Hat Enterprise Linux 7 に導入されている、SELinux と仮想化を統合する技術のことです。仮想マシンの使用時には Mandatory Access Control (MAC) を適用して、ホストや別の仮想マシンへの攻撃経路として使用される可能性のあるハイパーバイザー内のバグに対してシステムを堅牢にするとともにセキュリティを強化します。



### 注記

仮想化のセキュリティに関する詳細は、『Red Hat Enterprise Linux 7 仮想化セキュリティガイド』を参照してください。

## 2.6. サーバー向け仮想化と個人向け仮想化

仮想化はサーバーだけのものではなく、個人にも便利なものとなります。デスクトップの仮想化は、集中型管理やデスクトップソリューションの改良、すぐれた障害回復をもたらします。接続ソフトウェアを使用することで、デスクトップのリモート接続が可能になります。

サーバーの場合、仮想化は大規模ネットワークだけではなく、2 台以上のサーバーがある状況で利用可能です。ライブマイグレーションや高可用性、耐障害性、簡素化されたバックアップなどの機能を提供します。

### 2.6.1. 仮想化デプロイメントシナリオ

以下は、仮想化における一般的なデプロイメントシナリオとこれらのシナリオのデプロイメントに使用可能なツールです。

#### 最大 3 台の物理ホストおよびゲスト 10 台の小規模デプロイメント: virt-manager

厳密なアップタイム要件やサービスレベルアグリーメントがないサーバー数台を稼働している小規模ビジネスでは、virt-manager のようなツールが役立ちます。この環境では、インフラストラクチャー全体を担当する管理者が 1 人の場合があります。また、コンポーネントの変更などが必要な場合には手順の柔軟性を維持することが重要です。このような環境では、ウェブブラウザーやファイルおよびプリントサーバー、アプリケーションサーバーなどが含まれることがあります。

#### 大規模デプロイメントまたはミッションクリティカルなアプリケーション: Red Hat Enterprise Virtualization (RHEV)

Red Hat Enterprise Virtualization (RHEV) のような完全仮想化プラットフォームは、大規模デプロイメントやミッションクリティカルなアプリケーションを稼働している企業に適しています。この環境では、物理インフラストラクチャーは IT 部門が必要になるほど大きく、ビジネス要件は新たなニーズに対する明確な対応が必要となります。Red Hat Enterprise Virtualization が適する大規模デプロイメントの例としては、ダウンタイムなしに継続的な稼働が必須となるデータベースや商用プラットフォーム、メッセージングシステムなどがあげられます。

#### 管理アプリケーションを作成しているソフトウェア開発者: libvirt

virt-manager と Red Hat Enterprise Virtualization (RHEV) の両方が libvirt を使用して仮想マシンを管理します。libvirt はソフトウェア開発者による管理アプリケーションの作成と適応を可能にする仮想化アプリケーションプログラミングインターフェース (API) です。

## 第3章 Red Hat 仮想化製品について

本章では、Red Hat Enterprise Linux で利用可能な各種仮想化製品をご紹介します。

### 3.1. KVM および Red Hat Enterprise Linux での仮想化

#### KVM とは

KVM (Kernel-based Virtual Machine) は、AMD64 および Intel 64 ハードウェア上の Linux 向けの完全仮想化ソリューションのことで、標準の Red Hat Enterprise Linux 7 カーネルに組み込まれています。変更のない Windows および Linux のゲストオペレーティングシステムを複数実行することができます。Red Hat Enterprise Linux の KVM ハイパーバイザーは、**libvirt** API および (**virt-manager** や **virsh** などの) **libvirt** 向けにビルドされたツールで管理されます。仮想マシンは、これらのツールで管理されたマルチスレッドの Linux プロセスとして実行/稼働されます。

#### オーバーコミット

KVM ハイパーバイザーは システムリソースの オーバーコミットに対応しています。オーバーコミットとは、システム上で利用可能なリソースの容量を超える仮想化 CPU または仮想化メモリーを割り当てることです。メモリーのオーバーコミットにより、ホストはメモリーおよび仮想メモリーを活用してゲストの密度を高めることが可能になります。



#### 重要

オーバーコミットは、システムの安定性へのリスクとなる可能性があります。KVM でのオーバーコミットと必要な予防措置については『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

#### シンプロビジョニング

シンプロビジョニングは柔軟なストレージの割り当てを可能にし、全ゲスト仮想マシンの利用可能なスペースを最適化します。シンプロビジョニングにより、ゲスト上では実際に利用可能なストレージよりも多くの物理ストレージがあるように見えます。これはストレージのみに関することで、CPUやメモリーの割り当てとは関係ないため、オーバーコミットではありません。しかし、オーバーコミットと同様の警告が適用されます。



#### 重要

シンプロビジョニングは、システムの安定性へのリスクとなる可能性があります。KVM のシンプロビジョニングと必要な予防措置については『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

#### KSM

KVM ハイパーバイザーが使用する *Kernel Same-page Merging (KSM)* により、KVM ゲストは同一メモリーページを共有することが可能になります。このような共有ページは通常、共通のライブラリか、使用頻度の高い同一データです。KSM はメモリー重複を防ぐことで同一または類似のゲストオペレーティングシステムのゲスト密度を高めることができます。

**注記**

KSMに関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**QEMU ゲストエージェント**

QEMU ゲストエージェントはゲストオペレーティングシステム上で稼働し、ホストマシンによるゲストオペレーティングシステムへのコマンド発行を可能にします。

**注記**

QEMU ゲストエージェントに関する詳しい情報は『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**Hyper-V エンライト**

Red Hat Enterprise Linux 7 の KVM は、パフォーマンスと安定性を強化するために、Windows ゲストが使用する Hyper-V と互換性のある機能を複数実装します。これにより、Windows ゲストのパフォーマンスは Microsoft Hyper-V ハイパーバイザー上で実行されているかのようになります。

**注記**

Hyper-V 機能に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**ディスク I/O スロットリング**

複数の仮想マシンを同時に実行すると、ディスク I/O を過剰に使用するためシステムのパフォーマンスに影響を与える可能性があります。KVM のディスク I/O スロットリングにより、仮想マシンからホストマシンに送られるディスク I/O 要求に制限を設定できるようになります。これにより、共有リソースを過剰に使用して他の仮想マシンのパフォーマンスに影響を与えないようにすることができます。

**注記**

ディスク I/O スロットリングを使用する方法については、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**NUMA の自動負荷分散**

NUMA の自動負荷分散により、Red Hat Enterprise Linux 7 ゲストで必要とされる手動での微調整をしなくても、NUMA ハードウェアシステムで実行するアプリケーションのパフォーマンスが向上されます。NUMA の自動負荷分散により、スレッドやプロセスなどのタスクをアクセスするメモリーに近い場所に移動します。

**注記**

NUMA の自動負荷分散に関する詳細は、『Red Hat Enterprise Linux 7 仮想化のチューニングと最適化』を参照してください。

**仮想 CPU ホットアド**

仮想 CPU (vCPU) ホットアド機能があれば、ダウンタイムなしに、仮想マシンの実行の際に必要な処理能力を向上できます。必要なワークロードに対応したり、ワークロード関連のサービスレベルアグリーメント (SLA) を確保したりするために、仮想マシンに割り当てられた vCPU を実行中のゲストに追加することができます。

**注記**

仮想 CPU に関する詳しい情報は『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**KVM ゲスト仮想マシンの互換性**

Red Hat Enterprise Linux 7 サーバーには、サポートに関する制限がいくつかあります。

Red Hat Enterprise Linux のプロセッサおよびメモリー容量の制限については、以下のサイトで説明されています。

- ▶ ホストシステムについては、<https://access.redhat.com/site/articles/rhel-limits> を参照してください。
- ▶ KVM ハイパーバイザーについては、<https://access.redhat.com/site/articles/rhel-kvm-limits> を参照してください。

サポートされるオペレーティングシステムとホストおよびゲストの組み合わせの詳細な対応表については、<https://access.redhat.com/site/supported-hypervisors> を参照してください。

**注記**

ご使用のプロセッサが仮想化拡張に対応しているかどうかの確認方法および無効になっている仮想化拡張の有効化に関する情報については、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

## 3.2. libvirt および libvirt ツール

*libvirt* パッケージはハイパーバイザーに依存しない仮想化 API で、幅広いオペレーティングシステムの仮想化機能との対話が可能です。

*libvirt* パッケージが提供する機能

- ▶ ホスト上の仮想マシンを安全に管理する共通、一般的、かつ安定的な層
- ▶ ローカルシステムおよびネットワーク接続されたホストを管理する共通インターフェース



- ▶ 仮想マシンのプロビジョニング、作成、修正、監視、制御、移行、停止に必要なすべての API。ただし、ハイパーバイザーがこれらの操作に対応している場合のみ。**libvirt** で複数のホストに同時アクセスはできますが、API はシングルノード操作に限られています。

**libvirt** パッケージは、**virt-manager** や **virsh** などのコマンドライン管理ツールといった、より高いレベルの管理ツール向けビルディングブロックとして設計されています。移行機能を除き、**libvirt** は単一ホストにフォーカスし、CPU やメモリー、ストレージ、ネットワーク、NUMA (Non-Uniform Memory Access) パーティションを含む管理ノード上で利用可能なリソースを列挙、監視、使用する API を提供します。管理ツールは、セキュアなプロトコルを使用してホストからそれぞれの物理マシンに配置可能です。

Red Hat Enterprise Linux 7 は **libvirt** に対応しており、(Red Hat Enterprise Virtualization Management 同様に) 仮想化管理のデフォルトの方法として **libvirt** ベースのツールが含まれています。

**libvirt** パッケージは、GNU 一般公衆利用許諾契約書 (GPL) の下でフリーソフトウェアとして利用可能です。**libvirt** プロジェクトは、様々なハイパーバイザー技術上で稼働する仮想化管理ツールに長期的な安定性のある C API を提供することを目的としています。**libvirt** パッケージは、Red Hat Enterprise Linux 5 上の Xen と、Red Hat Enterprise Linux 5、6、7 上の KVM に対応しています。

## virsh

**virsh** コマンドラインツールは、**libvirt** 管理 API 上で構築されており、グラフィカルな **virt-manager** アプリケーションの代わりとして機能します。**virsh** コマンドは、権限を持たないユーザーは読み取り専用モードで、ルートアクセスでは完全な管理者機能を使用することができます。**virsh** コマンドは仮想化管理のスクリプト作成に適しており、仮想マシンのインストール、表示、開始、停止などの多数の機能を提供します。

## virt-manager

**virt-manager** は、仮想マシン管理用のグラフィカルなデスクトップツールです。グラフィカルなゲストコンソールへのアクセスが可能になり、仮想化管理や仮想マシン作成、移行、設定タスクが実行できます。仮想マシン、ホスト統計、デバイス情報、パフォーマンスグラフの表示機能も提供されます。ローカルハイパーバイザーとリモートハイパーバイザーは、単一インターフェースで管理可能です。



### 注記

**virsh** および **virt-manager** に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

## 3.3. 仮想化ハードウェアデバイス

Red Hat Enterprise Linux 7 における仮想化は、仮想マシンに 3 種の異なるシステムデバイスを提示します。この 3 つのタイプは以下のとおりです。

- ▶ 仮想化デバイスおよびエミュレートされたデバイス
- ▶ 準仮想化デバイス
- ▶ 物理共有デバイス

これらのハードウェアデバイスはすべて、仮想化マシンに物理的にアタッチされたように表示されますが、デバイスドライバーの機能の仕方が異なります。

### 3.3.1. 仮想化デバイスおよびエミュレートされたデバイス

KVM は、ソフトウェアに多くの仮想マシン向けコアデバイスを実装します。これらのエミュレートされたハードウェアデバイスは、オペレーティングシステムの仮想化に必須のもので、

エミュレートされたデバイスは、ソフトウェア内にのみ存在する仮想デバイスです。

エミュレートされたドライバーは、物理デバイスまたは仮想ソフトウェアデバイスを使用することができます。エミュレートされたドライバーは、仮想マシンと (ソースデバイスを管理する) Linux カーネルの間の交換層です。デバイスレベルの指示は KVM ハイパーバイザーによって完全に交換されます。Linux カーネルで認識される、(ストレージやネットワーク、キーボード、マウスなど) 同じタイプのデバイスはいずれも、エミュレートされたドライバーのバックエンドソースデバイスとして使用可能です。

### 仮想 CPU (vCPU)

ホストシステムはゲストに提示する CPU として、ホスト CPU の数にかかわらず最大 160 の仮想 CPU を保持することが可能です。

### エミュレートされたグラフィックデバイス

エミュレートされたグラフィックデバイスが 2 つ提供されます。これらのデバイスは SPICE プロトコルまたは VNC で接続することができます。

- ※ Cirrus CLGD 5446 PCI VGA カード (**cirrus** デバイスを使用)
- ※ Bochs VESA 拡張を搭載している標準的な VGA グラフィックカード (すべての非標準モードを含むハードウェアレベル)

### エミュレートされたシステムコンポーネント

以下のコアシステムコンポーネントは基本的なシステム機能を提供するためにエミュレートされます。

- ※ Intel i440FX ホスト PCI ブリッジ
- ※ PIIX3 PCI to ISA ブリッジ
- ※ PS/2 マウスおよびキーボード
- ※ EvTouch USB グラフィックタブレット
- ※ PCI UHCI USB コントローラーおよび仮想化 USB ハブ
- ※ エミュレートされたシリアルポート
- ※ EHCI コントローラー、仮想化 USB ストレージ、USB マウス
- ※ USB 3.0 xHCI ホストコントローラー (Red Hat Enterprise Linux 7.1 のホストコントローラー)

### エミュレートされたサウンドデバイス

Red Hat Enterprise Linux 6.1 以降では、エミュレートされた HDA サウンドデバイスである **intel-hda** が提供されます。このデバイスは以下のゲストオペレーティングシステムに対応しています。

- ※ Red Hat Enterprise Linux 7 (x86\_64 アーキテクチャー)
- ※ Red Hat Enterprise Linux 6 (i386 および x86\_64 アーキテクチャー)
- ※ Red Hat Enterprise Linux 5 (i386 および x86\_64 アーキテクチャー)
- ※ Red Hat Enterprise Linux 4 (i386 および x86\_64 アーキテクチャー)

- ※ Windows 7 (i386 および x86\_64 アーキテクチャー)
- ※ Windows 2008 R2 (x86\_64 アーキテクチャー)

以下のエミュレートされたサウンドドライバも利用可能ですが、特定のゲストオペレーティングシステムとの互換性の問題で推奨されません。

- ※ **ac97** エミュレートされた Intel 82801AA AC97 Audio 互換サウンドカード

## エミュレートされたウォッチドッグデバイス

Red Hat Enterprise Linux 6.0 以降では、2 種類のエミュレートされたウォッチドッグデバイスが提供されます。ウォッチドッグは、仮想マシンがオーバーロードもしくは応答しない場合に自動的に仮想マシンを再起動するために使用できます。

ウォッチドッグパッケージは、ゲスト上にインストールする必要があります。

利用可能なデバイスは以下の 2 つです。

- ※ **i6300esb** エミュレートされた Intel 6300 ESB PCI ウォッチドッグデバイス。Red Hat Enterprise Linux versions 6.0 以降のゲストオペレーティングシステムに対応しており、使用が推奨されているデバイスです。
- ※ **ib700** エミュレートされた iBase 700 ISA ウォッチドッグデバイス。**ib700** ウォッチドッグデバイスは、Red Hat Enterprise Linux 6.2 以降を使用するゲストにのみ対応しています。

これらのウォッチドッグデバイスは両方とも、Red Hat Enterprise Linux 6.2 以降のゲストオペレーティングシステムの i386 および x86\_64 アーキテクチャーに対応しています。

## エミュレートされたネットワークデバイス

エミュレートされたネットワークデバイスは 2 種類あります。

- ※ **e1000** デバイスは、Intel E1000 ネットワークアダプター (Intel 82540EM、82573L、82544GC) をエミュレートします。
- ※ **rtl8139** デバイスは、Realtek 8139 ネットワークアダプターをエミュレートします。

## エミュレートされたストレージドライバー

ストレージデバイスとストレージプールは、これらのエミュレートされたドライバーを使用してストレージデバイスを仮想マシンにアタッチすることができます。ゲストは、エミュレートされたストレージドライバーを使用してストレージプールにアクセスします。

すべての仮想デバイスと同様、ストレージドライバーはストレージデバイスではない点に注意してください。バックিংストレージデバイスやファイル、ストレージプールボリュームなどを仮想マシンにアタッチするために使用するのがドライバーです。対応する任意のタイプのストレージデバイス、ファイル、ストレージプールボリュームをバックিংストレージデバイスにすることができます。

### エミュレートされたIDE ドライバー

KVM は 2 種類のエミュレートされた PCI IDE インターフェースを提供します。エミュレートされた IDE ドライバーを使用して、最大 4 つの仮想 IDE ハードディスクまたは仮想 IDE CD-ROM ドライブの組み合わせを仮想マシンにアタッチすることができます。エミュレートされた IDE ドライバーは、仮想 CD-ROM および DVD-R ドライブにも使用できます。

### エミュレートされたフロッピーディスクドライブドライバー



エミュレートされたフロッピーディスクドライブドライバーは、仮想フロッピードライブの作成に使用されます。

### エミュレートされた AHCI コントローラー

IDE の代わりに、エミュレートされた Advanced Host Controller Interface (AHCI) バスをご利用いただけます。AHCI は Red Hat Enterprise Linux 7.1 ではテクノロジープレビューとして提供されています。



#### 注記

KVM ストレージに関する詳しい情報は、「[ストレージ](#)」を参照してください。

### 3.3.2. 準仮想化デバイス

準仮想化は、ゲストがホストマシン上のデバイスを使用するための高速かつ効率的な通信手段を提供します。KVM は、ハイパーバイザーとゲスト間のレイヤーとして Virtio API を使用し、準仮想化されたデバイスを仮想マシンに提供します。

準仮想化デバイスは、I/O 待ち時間を短縮し、I/O スループットをベアメタルレベル近くにまで高めるものもあれば、その他の方法では利用できない機能を仮想マシンに追加するものもあります。I/O 集約型アプリケーションを稼働している仮想マシンには、エミュレートされたデバイスの代わりに準仮想化ドライバーの使用が推奨されます。

すべての virtio デバイスは、ホストデバイスとゲストドライバーの 2 つの部分から構成されます。準仮想化デバイスドライバーにより、ゲストのオペレーティングシステムはホストシステム上の物理デバイスにアクセスできます。

準仮想化デバイスドライバーは、ゲストオペレーティングシステムにインストールする必要があります。ラフォルトでは、Red Hat Enterprise Linux 4.7 以降、Red Hat Enterprise Linux 5.4 以降、および Red Hat Enterprise Linux 6.0 以降に準仮想化デバイスドライバーが含まれています。Windows ゲストでは、準仮想化デバイスドライバーを手動でインストールする必要があります。



#### 注記

準仮想化デバイスおよびドライバーの使用に関する詳しい情報は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

#### 準仮想化ネットワークデバイス (virtio-net)

準仮想化ネットワークデバイスは、仮想マシンにネットワークアクセスを提供する仮想ネットワークデバイスであり、入出力パフォーマンスを強化し、待ち時間を短縮します。

#### 準仮想化ブロックデバイス (virtio-blk)

準仮想化ブロックデバイスは、ストレージを仮想マシンに提供する高パフォーマンスの仮想ストレージデバイスで、I/O パフォーマンスの強化、レイテンシーの低下を実現します。準仮想化ブロックデバイスはハイパーバイザーによってサポートされ、仮想マシンにアタッチされます (フロッピーディスクドライブは例外で、これにはエミュレートが必要になります)。

#### 準仮想化コントローラーデバイス (virtio-scsi)

準仮想化 SCSI コントローラーデバイスは、virtio-blk に代わる、より柔軟でスケラブルな選択肢を提供します。virtio-scsi ゲストはターゲットデバイスの機能一式を継承でき、わずか 28 デ

バイスしか処理できない virtio-blk に比べて数百ものデバイス処理が可能です。

virtio-scsi は以下のゲストオペレーティングシステムに完全対応しています。

- ✦ Red Hat Enterprise Linux 7
- ✦ Red Hat Enterprise Linux 6.4 以降
- ✦ Windows Server 2008 (32/64 bit)
- ✦ Windows Server 2008 R2
- ✦ Windows 7 (32/64 bit)
- ✦ Windows Server 2012
- ✦ Windows Server 2012 R2
- ✦ Windows 8 (32/64 bit)
- ✦ Windows 8.1 (32/64 bit)

### 準仮想化クロック

タイムスタンプカウンタ (TSC) をクロックソースとして使用するゲストは、時間管理の問題に直面することがあります。一定したタイムスタンプカウンタを持たないホストに関しては、KVM はゲストに準仮想化クロックを提供することで対処します。さらに、準仮想化クロックは、ゲストによる S3 の実行または RAM へのサスペンド操作の後に必要な時間の調整を支援します。準仮想化クロックは、Windows ゲストではサポートされていません。

### 準仮想化シリアルデバイス (virtio-serial)

準仮想化シリアルデバイスは、バイトストリーム指向の文字ストリームデバイスで、ホストのユーザー領域とゲストのユーザー領域をつなぐシンプルな通信インターフェースを提供します。

### バルーンデバイス (virtio-balloon)

バルーンデバイスは仮想マシンの RAM の一部を未使用として指定することが可能で (バルーン膨張)、これによりメモリーが解放されて、ホスト (またはホスト上の他の仮想マシン) が使用できるようになります。仮想マシンが再度メモリーを必要とした際には、バルーンを収縮させて、ホストが RAM を仮想マシンに配分し直すことができます。

### 準仮想化乱数ジェネレーター (virtio-rng)

準仮想化乱数ジェネレーターにより、仮想マシンがホストから直接、エントロピーまたは乱数を収集して暗号化データやセキュリティーに使用することができます。仮想マシンは (ハードウェアの使用状況など) 一般的なインプットを入手できないため、エントロピーが不足することが頻繁にあります。エントロピーの調達には多大な時間を要することがあるため、virtio-rng はホストからゲスト仮想マシンに直接エントロピーを注入することで、このプロセスを高速化します。

### 準仮想化グラフィックカード (QXL)

準仮想化グラフィックカードは、QXL ドライバーと連携して機能し、リモートホストから仮想マシンのグラフィックスを表示する効率的な方法を提供します。SPICE を使用するには QXL ドライバーが必要です。

## 3.3.3. 物理ホストデバイス

一部のハードウェアプラットフォームでは、仮想マシンが直接ハードウェアデバイスやコンポーネントにアクセスすることができます。仮想化におけるこのプロセスは、デバイス割り当てとして知られています。デバイス割り当てはパススルーとも呼ばれます。

### VFIO デバイス割り当て

Virtual Function I/O (VFIO) は、Red Hat Enterprise Linux 7 の新しいカーネルドライバーであり、これにより、仮想マシンが高パフォーマンスで物理ハードウェアにアクセスできるようになります。

VFIO は、ホストシステム上の PCI デバイスを直接仮想マシンにアタッチするため、ゲストはさまざまなタスクで PCI デバイスへの排他的アクセスが可能になります。これにより、PCI デバイスはゲスト仮想マシンに物理的にアタッチされているかのように表示され、動作します。

VFIO は、KVM ハイパーバイザーからデバイス割り当てを移動して、カーネルレベルでデバイスの分離を強制的に行うことで、これまでの PCI デバイス割り当てのアーキテクチャーを向上します。VFIO は、セキュリティが強化されており、セキュアブートとの互換性があります。VFIO は Red Hat Enterprise Linux 7 におけるデフォルトのデバイス割り当てメカニズムです。

VFIO は、Red Hat Enterprise Linux 6 の最大 8 デバイスから、Red Hat Enterprise Linux 7 の割り当てデバイス数を 32 に増やしました。VFIO は、NVIDIA GPU の割り当てもサポートします。



#### 注記

VFIO デバイス割り当てに関する詳しい情報は『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

### USB パススルー

KVM ハイパーバイザーは、ホストシステム上の USB デバイスの仮想マシンへのアタッチをサポートします。USB デバイス割り当てにより、ゲストは様々なタスクで USB デバイスへの排他的アクセスが可能になります。USB デバイスは仮想マシンに物理的にアタッチされているように表示され、動作します。



#### 注記

USB パススルーに関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

### SR-IOV

SR-IOV (シングルルート I/O 仮想化) は、単一の物理 PCI 機能を拡張して PCI リソースを個別の仮想機能 (VF) として共有する PCI Express 規格です。PCI デバイスを割り当てることで、別の仮想マシンで各機能が使用できるようになります。

SR-IOV 対応 PCI-e デバイスは、シングルルート機能 (例えば、単一イーサネットポート) を提供し、複数の個別の仮想デバイスを固有の PCI デバイス機能として提示します。仮想デバイスはそれぞれ、固有の PCI 設定領域、メモリーマップドレジスター、個別の MSI ベースの割り込みを備えている場合があります。

**注記**

SR-IOVに関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**NPIV**

N\_Port ID Virtualization (NPIV) とは、ファイバーチャネルデバイスの一部で利用可能な機能です。NPIV では単一の物理的な N\_Port を複数の N\_Port ID として共有します。SR-IOV が PCIe インターフェースに提供するファイバーチャネル Host Bus Adaptor (HBA) に対して同様の機能を提供します。NPIV を使用すると、ストレージエリアネットワーク (SAN) への仮想ファイバーチャネルイニシエーターを仮想マシンに与えることができます。

NPIV は、エンタープライズレベルのストレージソリューションで高密度の仮想化環境を提供できます。

**3.3.4. ゲスト CPU モデル**

CPU モデルは、ゲストオペレーティングシステムに公開するホスト CPU 機能を定義します。**qemu-kvm** と **libvirt** には、複数の現行プロセッサモデルの定義が含まれ、これによりユーザーは新しい CPU モデルでのみ利用できる CPU 機能を有効にすることができます。ゲストに公開できる CPU 機能のセットは、ホスト CPU、カーネル、および **qemu-kvm** コードのサポートによって異なります。

異なる CPU 機能のセットを持つ複数のホスト間で仮想マシンを安全にマイグレーションできるように、デフォルトでは **qemu-kvm** により、ホスト CPU の全機能がゲストオペレーティングシステムに公開されず、その代わりに、CPU 機能は選択された CPU モデルに基づいて公開されます。仮想マシンで所定の CPU 機能が有効にされている場合には、この機能をゲストへ公開するようにサポートしていないホストに移行することはできません。

**注記**

ゲスト CPU モデルに関する詳しい情報は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**3.4. ストレージ**

仮想マシン向けのストレージは、仮想マシンが使用する物理ストレージから抽出されます。これは、準仮想化またはエミュレートされたブロックデバイスドライバーを使用して仮想マシンにアタッチされます。

**3.4.1. ストレージプール**

ストレージプールとは、仮想マシンにストレージを提供するために **libvirt** が管理するファイルやディレクトリ、ストレージデバイスのことです。ストレージプールは、仮想マシンイメージを保存するストレージボリュームまたは仮想マシンに新たなストレージとしてアタッチされるストレージボリュームに分割されます。複数のゲストが同一のストレージプールを共有でき、ストレージリソースをより適切に割り当てることができます。詳細については、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

**ローカルストレージプール**

ローカルストレージプールは、ホストサーバーに直接アタッチされています。これに含まれるの

は、ローカルディレクトリー、直接アタッチされたディスク、物理パーティション、ローカルデバイス上のLVM ボリュームグループです。ローカルストレージプールは開発、テスト、移行や多数の仮想マシンを必要としない小規模開発などに役立ちます。ローカルストレージプールは、ライブマイグレーションに対応していないため、多くの本番環境に適切でない可能性があります。

### ネットワーク (共有) ストレージプール

ネットワークストレージプールには、標準プロトコル使用のネットワークにまたがって共有されるストレージデバイスが含まれます。ネットワークストレージは、**virt-manager** を使用したホスト間の仮想マシン移行が必要となりますが、**virsh** を使用した移行の場合はオプションとなります。ネットワークストレージプールは、**libvirt** で管理します。

### 3.4.2. ストレージボリューム

ストレージプールは、さらにストレージボリュームに分割されます。ストレージボリュームとは、物理パーティション、LVM 論理ボリューム、ファイルベースのディスクイメージ、**libvirt** で処理される他のストレージタイプなどを抽象化したものです。ストレージボリュームは、基盤となるハードウェアに関係なく、ローカルストレージデバイスとして仮想マシンに提示されます。

### 3.4.3. エミュレートされたストレージデバイス

仮想マシンには、ホストによりエミュレートされるさまざまなストレージデバイスが提供されます。ストレージデバイスのタイプはそれぞれ、特定のユースケースに適しています。ストレージデバイスの複数の異なるタイプの中から選択できることにより、柔軟性やゲストのオペレーティングシステムとの互換性を最大化することができます。

#### virtio-scsi

virtio-scsi は数多くのディスクや TRIM などの高度なストレージ機能を使用するゲスト用の準仮想化デバイスとして推奨されます。Red Hat Enterprise Linux 7 以外のオペレーティングシステムを使用しているゲストでは、ゲストドライバーのインストールが必要になる場合があります。

#### virtio-blk

virtio-blk はイメージファイルをゲストに公開するのに適した準仮想化ストレージデバイスです。virtio-blk は仮想マシンに対して最高レベルのディスクの I/O パフォーマンスを提供しますが、virtio-scsi と比較すると機能の数が少なくなります。

#### IDE

IDE は virtio ドライバーに対応しないレガシーゲスト用に推奨されます。IDE のパフォーマンスは virtio-scsi または virtio-blk よりも低いですが、複数の異なるシステムに対する互換性の面では優れています。

#### CD-ROM

ATAPI CD-ROM および virtio-scsi CD-ROM は ISO ファイルまたはホスト CD-ROM ドライブをゲストに提示するために使用できます。virtio-scsi CD-ROM は、virtio-scsi ドライバーがインストールされているゲストと共に使用することができます。ATAPI CD-ROM の方が互換性の幅が広いですが、パフォーマンスはより低くなります。

#### USB 大容量ストレージデバイスおよびフロッピーディスク

USB 大容量ストレージデバイスおよびフロッピーディスクは、リムーバブルメディアが必要な場合に利用することができます。フロッピーディスクよりも容量がより大きい USB 大容量ストレージデバイスの使用をお勧めします。

#### AHCI



IDE の代わりに、エミュレートされた Advanced Host Controller Interface (AHCI) バスをご利用いただけます。AHCI バスにより、Serial ATA (SATA) デバイスでの通信など、機能やパフォーマンスが拡張されます。

AHCI は、Red Hat Enterprise Linux 7.1 ではテクノロジープレビューとして提供されています。

### 3.4.4. ホストストレージ

ディスクイメージは、ホストに接続される様々なローカルおよびリモートのストレージテクノロジーの上に格納できます。

#### イメージファイル

イメージファイルはホストファイルシステム上に格納されます。イメージファイルは、ext4 または xfs などのローカルファイルシステム上や、NFS などのネットワークファイルシステム上に格納することができます。

**libguestfs** などのツールを使用して、ファイルの管理、バックアップおよび監視を行うことができます。KVM 上で使用されるディスクイメージの形式には以下が含まれます。

#### Raw

Raw イメージファイルには、ディスクのコンテンツが含まれますが、メタデータは一切追加されません。

Raw ファイルのタイプは事前割り当てファイルまたはスパーズファイルのいずれかになります (ホストのファイルシステムが対応している場合)。スパーズファイルはホストのディスク容量をオンデマンドで割り当てるため、シンプロビジョニングの一種であると言えます。事前割り当てファイルは完全にプロビジョニングされますが、スパーズファイルよりもパフォーマンスが高くなります。

Raw ファイルは、ディスクの I/O パフォーマンスが重要となる場合や、イメージファイルをネットワーク上で転送する必要がほぼない場合に適しています。

#### qcow2

qcow2 イメージファイルは、バッキングファイルや、スナップショット、圧縮および暗号化などの数多くの高度なディスクイメージ機能を提供します。これらのイメージファイルは、テンプレートイメージから仮想マシンのインスタンスを作成するために使用することができます。

qcow2 ファイルは、通常ネットワーク上でより効率的に転送できるファイルです。仮想マシンによって書き込まれるセクターのみがイメージに割り当てられるためです。

Red Hat Enterprise Linux 7.0 以降のバージョンは、qcow2 バージョン 3 のイメージファイル形式をサポートします。

#### LVM ボリューム

論理ボリュームはディスクイメージに使用でき、システムの LVM ツールを使用して管理できます。LVM のブロックストレージのモデルは他のファイルシステムよりもシンプルであるため、LVM はより高いパフォーマンスを提供します。

LVM シンプロビジョニングはスナップショットを提供し、LVM ボリュームの領域のより効率的な使用を可能にします。さらに、qcow2 に対する移行の代替手段として使用することができます。

#### ホストデバイス

物理 CD-ROM および Raw ディスク、または論理ユニット番号 (LUN) などのホストデバイスをゲストに提示することができます。これにより、ゲストはパフォーマンスを維持した状態で、

ネットワーク経由でアクセスできます。これにより、ネットワーク経由で接続された仮想マシンは、SAN や iSCSI LUN、およびローカル CD-ROM メディアを使用できます。

ホストデバイスは、ストレージ管理がホスト上ではなく SAN 上で行なわれている場合に使用することができます。

### 分散ストレージシステム

Gluster ボリュームはディスクイメージとして使用できます。これにより、ネットワーク上で高パフォーマンスのクラスター化ストレージを使用することができます。

Red Hat Enterprise Linux 7 には、GlusterFS 上のディスクに対するネイティブサポートが含まれます。これにより、KVM ホストは GlusterFS ボリュームから仮想マシンイメージを起動し、仮想マシンのデータディスクとして GlusterFS ボリュームのイメージを使用することができます。GlusterFS FUSE と比較すると、KVM のネイティブサポートのほうが高いパフォーマンスを実現します。



#### 注記

ストレージおよび仮想化に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

## 第4章 仮想化ツール

本章では、仮想化で役に立つ多くのツールをご紹介します。

### 4.1. virsh

*virsh* は、ハイパーバイザーとゲスト仮想マシンの管理に使用するコマンドラインインターフェース (CLI) ツールです。**virsh** コマンドラインツールは、**libvirt** 管理 API をベースに構築されており、**qemu-kvm** コマンドおよびグラフィカルな **virt-manager** アプリケーションの代わりとして動作します。**virsh** コマンドは、権限のないユーザーは読み取り専用モードで、ルートアクセスのあるユーザーは完全な管理機能を使用することができます。**virsh** コマンドは仮想化管理のスクリプト作成に適しています。さらに**virsh** ツールは、**virsh** ゲストドメイン向けのメインの管理インターフェースで、現行ドメインのリスト化に加え、ドメインの作成、一時停止、およびシャットダウンに使用できます。このツールは、*libvirt-client* パッケージの一部としてインストールされます。



#### 注記

**virsh** を使用した仮想マシン管理に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

### 4.2. virt-manager

*virt-manager* は、仮想マシン管理用の軽量グラフィカルツールです。このツールでは、既存マシンのライフサイクル管理、新規マシンのプロビジョン、仮想ネットワークの管理、仮想マシンのグラフィカルコンソールへのアクセス、パフォーマンス統計の表示といった機能が提供されます。このツールは、*virt-manager* という独自のパッケージ名で出荷されます。



#### 注記

**virt-manager** を使用した仮想マシン管理に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

### 4.3. virt-install

*virt-install* は、新規仮想マシンをプロビジョニングするコマンドラインツールです。このツールは、シリアルコンソール、SPICE、VNC クライアント/サーバーペアグラフィックスを使用したテキストベースとグラフィカルベースの両インストールに対応します。インストールメディアはローカルにも、NFS、HTTP、FTP サーバー上のリモートにも設置することが可能です。このツールは無人で稼働してインストール完了時にキックスタートを使用したゲストの構成を行うように設定することもでき、インストールの自動化を容易にします。このツールは *python-install* パッケージの一部としてインストールされます。



#### 注記

**virt-manager** の使用に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。



## 4.4. guestfish

*guestfish* は、ホストのファイルシステム検証および修正用のコマンドラインツールです。このツールは *libguestfs* を使用し、**guestfs** API が提供する全機能を公開しており、*guestfish* という独自のパッケージ名で出荷されます。



### 警告

実行中の仮想マシン上で **guestfish** を使用するとディスクイメージが破損する恐れがあります。実行中の仮想マシンがディスクイメージを使用している場合は、**--ro** (読み取り専用) オプションで **guestfish** コマンドを使用してください。



### 注記

**guestfish** に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

## 4.5. GNOME Boxes

**Boxes** とは、仮想マシンおよびリモートシステムの表示やアクセスに使用する軽量のグラフィカルデスクトップ仮想化ツールです。Boxes は、最小の設定で、デスクトップから異なるオペレーティングシステムとアプリケーションをテストする手段を提供します。仮想システムは、手動またはエクスプレスイnstall機能 (最適の設定で仮想マシンを自動で事前設定する機能) を使用してインストールすることができます。このツールは、*gnome-boxes* という名前の独自のパッケージで出荷されています。



### 注記

**GNOME Boxes** の使用に関する詳細は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。

## 4.6. 他の便利ツール

以下のツールは、ゲスト仮想マシンのディスクにホスト経由でアクセスするために使用されます。ゲストのディスクには通常、ホスト上にある **disk-image** ファイル経由でアクセスしますが、**libvirt** ドメイン経由でアクセスすることも時には可能です。以下のコマンドは **libvirt** ドメインの一部で、ゲストのディスクイメージにアクセスするために使用されます。

### guestmount

ホストマシン上に仮想マシンのファイルシステムおよびディスクイメージをマウントするために使用するコマンドラインツール。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

**警告**

ゲストが使用中のディスクに、**--rw** (読み取り/書き込み) モードを指定して **guestmount** でアクセスするとディスクが破損する恐れがあります。ライブの仮想マシン上では、**--rw** (読み取り/書き込み) モードで **guestmount** を使用しないでください。ディスクイメージを使用している場合は、**--ro** (読み取り専用) オプションで **guestmount** コマンドを使用してください。

**virt-builder**

新規仮想マシンを素早く構築およびカスタマイズするコマンドラインツール。このツールは、Red Hat Enterprise Linux 7.1 以降では *libguestfs* パッケージの一部としてインストールされます。

**virt-cat**

指定された仮想マシンのディスクまたはディスクイメージ内にある 1 つ以上のファイルのコンテンツを迅速に表示させる際に使用するコマンドラインツール。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

**virt-customize**

仮想マシンのディスクイメージをカスタマイズするコマンドラインツール。virt-customize は、パッケージのインストール、設定ファイルの編集、スクリプトの実行、パスワードの設定に使用することができます。このツールは、Red Hat Enterprise Linux 7.1 以降では *libguestfs* パッケージの一部としてインストールされます。

**virt-df**

仮想マシンの実際の物理ディスク使用量を表示させるために使用するコマンドラインツールで、**df** コマンドラインツールと同様のもの。このツールは、リモート接続では機能しないことに注意してください。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

**virt-diff**

スナップショット間のファイルの相違点を検出するなど、2 台の仮想マシンのファイルシステム間において差異を表示するコマンドラインツール。このツールは、Red Hat Enterprise Linux 7.1 以降では *libguestfs* パッケージの一部としてインストールされます。

**virt-edit**

指定された仮想マシン上にあるファイルを編集するために使用されるコマンドラインツール。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

**警告**

実行中の仮想マシン上で **virt-edit** を使用すると、仮想マシン内のディスクが破損する恐れがあります。**virt-edit** コマンドは、ユーザーが実行中の仮想マシン上のファイルを編集することを阻止しようとはしますが、全インスタンスの捕捉は保証されません。実行中の仮想マシン上では **virt-edit** を使用しないでください。

**virt-filesystems**

ディスクシステムもしくは仮想マシン内のファイルシステム、パーティション、論理ボリュームおよびそれらのサイズを検出するために使用されるコマンドラインツール。シェルスクリプト内

での一般的な使用方法の一つは、ディスクイメージ内の全ファイルシステムを反復することです。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

このツールは **virt-list-fileSYSTEMS** および **virt-list-partitions** を置き換えます。

### virt-inspector

仮想マシンまたはディスクイメージを検証して当該オペレーティングシステムおよび他の情報を測定することができるコマンドラインツール。XML 出力の生成も可能で、これは他のプログラムにパイプすることができます。**virt-inspector** は一度に 1 ドメインしか検査できないことに注意してください。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

### virt-ls

仮想マシン内のファイルおよびディレクトリーをリスト化するコマンドラインツール。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

### virt-log

Linux および Windowsの仮想マシンからのログファイルを表示するコマンドラインツール。このツールは、Red Hat Enterprise Linux 7.1 以降では *libguestfs* パッケージの一部としてインストールされます。

### virt-make-fs

ディレクトリー内の tar アーカイブもしくはファイルに基づいてファイルシステムを作成するコマンドラインツール。**mkisofs** や **mk squashfs** に類似したツールですが、ext2、ext3、NTFS といった一般的なファイルシステムの作成が可能で、作成されるファイルシステムのサイズは元となるファイルサイズと同等もしくはそれよりも大きいものになります。このツールは、*libguestfs-tools* パッケージの一部として提供されます。

### virt-rescue

ブート不可能な仮想マシンおよびディスクイメージ用のレスキューシェルとシンプルなりカバリーツールを提供するコマンドラインツール。**libvirt** が認識する仮想マシン上であれば実行可能です。また、ディスクイメージ上で直接実行することもできます。このツールは、*libguestfs-tool* パッケージの一部としてインストールされます。



#### 警告

実行中の仮想マシン上で **virt-rescue** を使用すると仮想マシン内のディスクが破損する恐れがあります。**virt-rescue** は、実行中の仮想マシン上での使用を阻止しようとしませんが、すべてのケースをカバーできる訳ではありません。--ro (読み取り専用) オプションでこのコマンドを使用してもディスクは破損しませんが、見たことのない、または一貫性を欠く結果となる場合があります。

実行中の仮想マシンでの **virt-rescue** の使用は避けてください。

### virt-resize

仮想マシンディスクのサイズ変更、また仮想マシンディスク上のすべてのパーティションのサイズ変更もしくは削除に使用するコマンドラインツール。ゲストイメージをコピーして、オリジナルのディスクイメージをそのまま残すことで機能します。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

**重要**

実行中の仮想マシンで **virt-resize** を使用すると、一貫性のない結果をもたらす可能性があります。サイズ変更前に仮想マシンをシャットダウンすることが最良の方法です。

**virt-sparsify**

仮想マシンディスク (またはディスクイメージ) をシンプロビジョニング化するコマンドラインツール。このツールを使用して、ディスクイメージの空き容量をホスト上の空き容量に変換することができます。

**virt-sysprep**

クローン作成用テンプレートを準備するために仮想マシンのリセット、カスタマイズまたは設定解除を行うためのコマンドラインツール。このツールは *libguestfs-tools* パッケージの一部としてインストールされます。

**重要**

仮想マシンは **virt-sysprep** を実行する前にシャットダウンしている必要があります。仮想マシンの既存のコンテンツを保持するには、**virt-sysprep** を実行する前にディスクのスナップショットを取得するか、またはディスクのコピーまたはクローン作成を行ってください。

**virt-tar-in**

圧縮した tar ファイルを、仮想マシンのディスクイメージまたは指定の libvirt ドメインに展開するコマンドラインアーカイブツール。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

**警告**

実行中の仮想マシン上で **virt-edit** を使用すると、仮想マシン内のディスクが破損する恐れがあります。このコマンドを使用する前に、仮想マシンをシャットダウンするようにしてください。

**virt-tar-out**

仮想マシンディスクイメージのディレクトリーを tar ファイルに圧縮するコマンドラインアーカイブツール。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。

**virt-top**

仮想化ドメインに関連する統計を表示する **top** に類似したコマンドラインユーティリティ。このツールは、*virt-top* という独自のパッケージ名で出荷されます。

**virt-v2v**

異種のハイパーバイザーからの仮想マシンを libvirt、Red Hat Enterprise Linux OpenStack Platform、または Red Hat Enterprise Virtualization 管理の KVM で実行するように変換するコマンドラインツール。現在、**virt-v2v** は、Xen および VMware ESX 上で実行する Red Hat Enterprise Linux および Windows ゲストを変換することができます。**virt-v2v** ツールは、Red

Hat Enterprise Linux 7.1 以降では、*libguestfs* パッケージの一部としてインストールされます。

### い **virt-viewer**

仮想マシンのグラフィカルコンソールを VNC および SPICE プロトコル経由で表示させる最小限のツール。このツールは、*virt-viewer* というパッケージ名称で出荷されます。

### **virt-what**

プログラムが仮想マシンで実行中かどうかを検出するシェルスクリプト。このツールは、*virt-what* というパッケージ名称で出荷されます。

### **virt-who**

*virt-who* パッケージは、ゲスト UUID を **libvirt** に問い合わせる Red Hat Enterprise Linux ホストエージェントです。次に、そのデータを証明書発行目的でローカルエンタイトルメントサーバーに渡します。このツールは、*virt-who* というパッケージ名称で出荷されます。

### **virt-win-reg**

Windows 仮想マシンから Windows レジストリーのエントリーをエクスポートし、マージして、シンプルなレジストリー操作を実行するコマンドラインツール。このツールは、*libguestfs-tools* パッケージの一部としてインストールされます。



#### 警告

実行中の仮想マシン上で **virt-win-reg** を使うと、仮想マシン内のディスクに復元不可能な破損をもたらします。**virt-win-reg** は、実行中の仮想マシン上での使用を回避しようとはしますが、すべてのケースをカバーする訳ではありません。



#### 警告

Windows レジストリーの修正は、フォーマットが意図的に不明瞭でドキュメントに未記載であることから、本質的にリスクのある操作です。レジストリーの変更でシステムが起動できなくなる可能性があるため、信頼できるバックアップがあることを確認してから、**--merge** オプションを使用するようにしてください。

### **virt-xml-validate**

**libvirt** XML ファイルの発行済スキーマとのコンプライアンスを検証するコマンドラインツール。このツールは、*libvirt-client* パッケージの一部としてインストールされます。

## 第5章 Red Hat Enterprise Linux 7 での仮想化に関するクイックスタートガイド

本章では、Red Hat Enterprise Linux 7 における仮想化について確認いただけます。

本章では、まず、仮想化を使用するための最小システム仕様および必要なパッケージについて説明します。これらの基本パッケージをインストールしたら、[「仮想マシンマネージャーでの仮想マシンの作成」](#)の手順に従い、**仮想マシンマネージャー** で基本的な仮想マシンを構築します。



### 注記

このチュートリアルでは、仮想マシンマネージャーを使用して素早く仮想マシンを構築し、KVM 仮想化を体験します。実稼動環境に必要な機能を備えた仮想マシンを設定する場合は、仮想マシンの実行に関するシステム要件および推奨事項に関する詳しい情報を『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』で参照してください。

### 5.1. 基本要件および設定

仮想化は、Intel 64 および AMD64 アーキテクチャーをベースにした Red Hat Enterprise Linux 7 で利用できます。KVM ハイパーバイザーは、Red Hat Enterprise Linux 7 で提供されています。

Red Hat Enterprise Linux 7 システム上で仮想マシンの設定を行う前に、特定のパッケージをお使いのシステムにインストールしておく必要があります。

#### システム要件

お使いのシステムで仮想化を機能させるには、最小構成として以下が必要になります。

- ▶ 6 GB の空きディスク領域
- ▶ 2 GB のメモリー

#### 仮想化に必要なパッケージ

仮想化を使用する前に、コンピューターに仮想化パッケージをインストールする必要があります。仮想化パッケージは、**yum** コマンドおよび Red Hat カスタマーポータルを使用してホストのインストール時またはインストール後にインストールすることができます。本章では、機能する Red Hat Enterprise Linux 7 システム上に KVM ハイパーバイザーをインストールする手順について説明します。

パッケージをインストールするには、お使いのホストマシンを登録しておく必要があります。Red Hat サブスクリプションマネージャーで登録するには、**subscription-manager register** コマンドを実行してプロンプトに従います。

有効な Red Hat サブスクリプションをお持ちでない場合は、[Red Hat オンラインストア](#) で取得してください。

#### 手順5.1 yum を使用して仮想化パッケージをインストール

Red Hat Enterprise Linux 上で仮想化を使用するには、少なくとも **qemu-kvm** と **qemu-img** パッケージが必要になります。これらのパッケージは、ホストの Red Hat Enterprise Linux システム上にユーザーレベルの KVM エミュレーターとディスクイメージマネージャーを提供します。



1. 以下のコマンドを使用して `qemu-kvm` と `qemu-img` パッケージをインストールします。

```
# yum install qemu-kvm qemu-img
```

2. 以下のコマンドを使用して、`virt-manager` パッケージをインストールします。

```
# yum install virt-manager
```

3. また、このチュートリアルでは、Red Hat Enterprise Linux 7 ISO イメージをシステムにダウンロードして、ゲスト仮想マシンのオペレーティングシステムを作成します。

Red Hat Enterprise Linux ISO ファイルは Red Hat カスタマーポータル (<https://access.redhat.com/downloads/>) にあります。

## 5.2. 仮想マシンマネージャーでの仮想マシンのデプロイ

仮想マシンマネージャー (別称 `virt-manager`) は、Red Hat Enterprise Linux で素早く仮想マシンをデプロイするためのグラフィカルツールです。本チュートリアルでは、このツールの基本機能に慣れ、**仮想マシンマネージャー**を使用して仮想マシンを作成できるようになります。

### 5.2.1. 仮想マシンマネージャーの概要

アプリケーションメニューのシステムツールサブメニューから **仮想マシンマネージャー** アプリケーションを開きます。

以下の画像は **仮想マシンマネージャー** インターフェースを示します。このインターフェースは、単一の中央ロケーションから全仮想マシンを制御することができます。

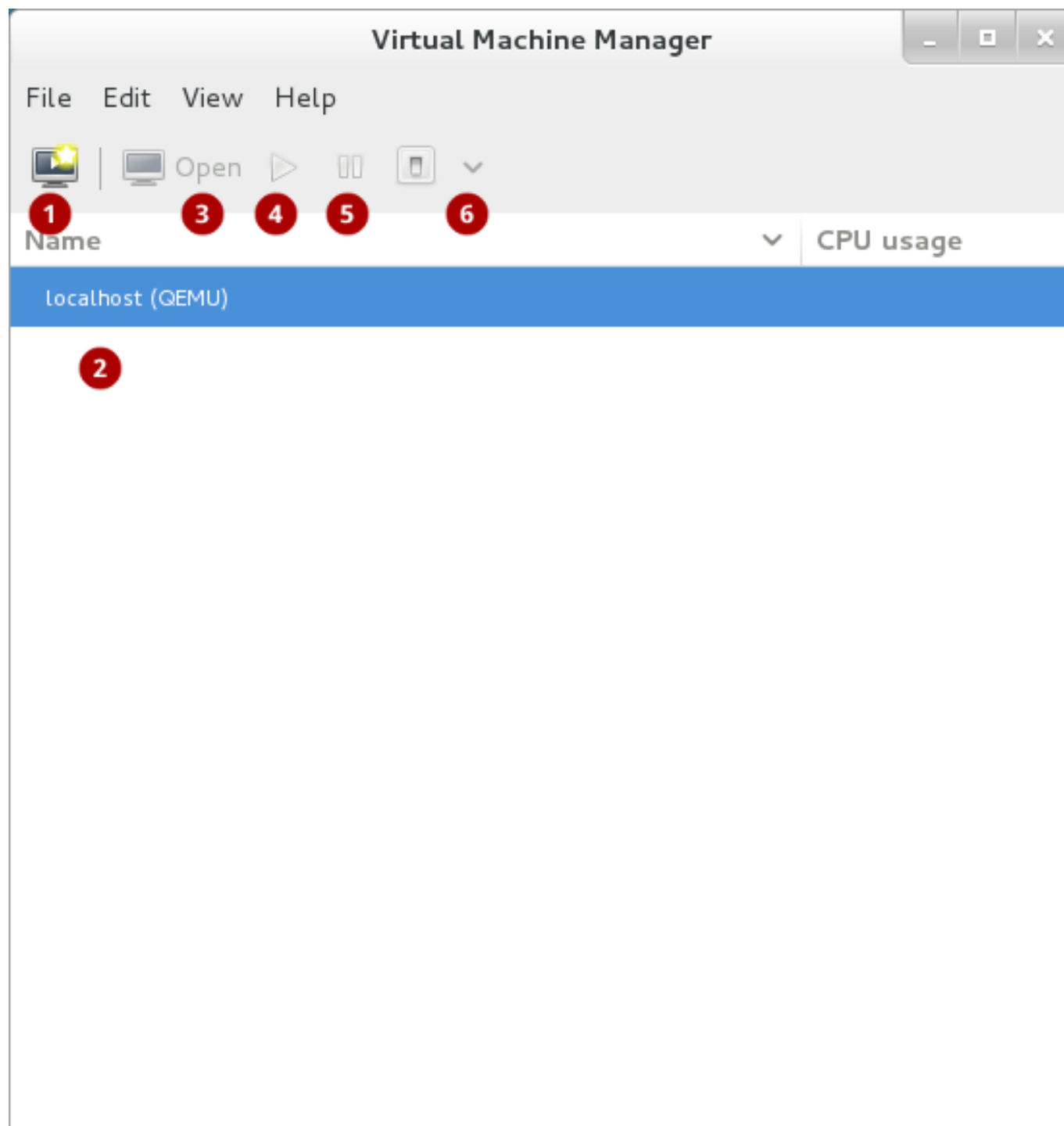


図5.1 仮想マシンマネージャーのインターフェース

一般的に使用する機能

- ※ **1** **新しい仮想マシンを作成**: 新規仮想マシンの作成はここから開始します。
- ※ **2** **仮想マシン**: 全仮想マシンまたはゲストの一覧。仮想マシンが作成されると、ここに表示されます。ゲストが実行中の場合には、**CPU 使用率** 部分でゲストの CPU 使用率がアニメーショングラフで表示されます。

一覧から仮想マシンを選択した後は、以下のボタンを使用して、選択した仮想マシンの状態を制御します。



- **3** 開く: 新しいウィンドウでゲスト仮想マシンコンソールと詳細を開きます。
- **4** 仮想マシンの電源を入れる: 仮想マシンの電源をオンにします。
- **5** 仮想マシンの一時停止: 仮想マシンを一時停止します。
- **6** 仮想マシンをシャットダウン: 仮想マシンをシャットダウンします。矢印をクリックすると、仮想マシンの電源をオフにするためのオプションが複数含まれたドロップダウン (再起動、シャットダウン、強制的にリセット、強制的に電源OFF、保存) が表示されます。

仮想マシンを右クリックすると、以下のような他の機能が含まれるメニューが表示されます。

- ✦ クローン: 仮想マシンのクローンを作成します。
- ✦ 移行: 仮想マシンを別のホストに移行します。
- ✦ 削除: 仮想マシンを削除します。

### 5.2.2. 仮想マシンマネージャーでの仮想マシンの作成

以下の手順に従い、**仮想マシンマネージャー** で Red Hat Enterprise Linux 7 の仮想マシンを作成します。

#### 手順5.2 仮想マシンマネージャーを使用したゲスト仮想マシンの作成

1. 仮想マシンマネージャーを開きます。

アプリケーションメニューのシステムツールサブメニューから **仮想マシンマネージャー** アプリケーションを起動します。

2. 新しい仮想マシンを作成します。

**新しい仮想マシンの作成** ボタン ([図5.2 「新しい仮想マシンの作成ボタン」](#)) をクリックして **新しい仮想マシン** ウィザードを開きます。



図5.2 新しい仮想マシンの作成ボタン

3. 名前とインストールの方法を指定します。

仮想マシンの名前を入力して、ゲスト仮想マシンのオペレーティングシステムをインストールする際のインストールタイプを選択します。

仮想マシンの作成のプロセスではまず、仮想マシンに名前を付け、インストール方法を選択します。仮想マシンの名前にはアンダースコア (\_)、ピリオド (.), ハイフン (-) を使用することができません。

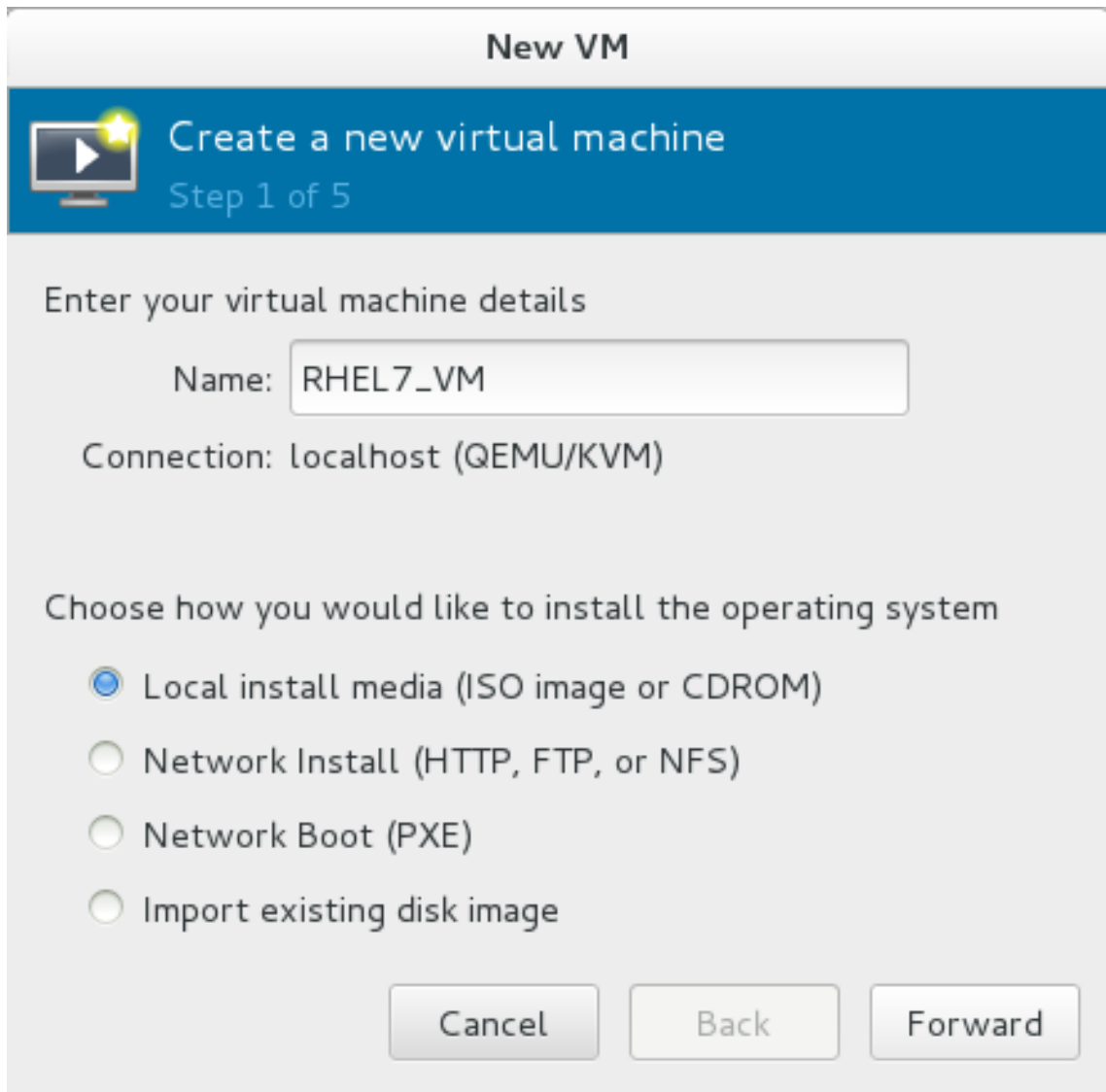


図5.3 仮想マシンに名前を付けてインストール方法を選択

このチュートリアルでは、**ローカルインストールメディア (ISO イメージ)** を選択します。本インストールの手順は、インストールディスクのイメージを使用します (例: `.iso` ファイル)。**進む** ボタンを押して続行します。

#### 4. インストールメディアの場所を指定します。

ゲスト仮想マシンにインストールするオペレーティングシステムの ISO の取得元を指定します。

たとえば [参照](#) を使用して既存のストレージパスを指定して、[手順5.1「yum を使用して仮想化パッケージをインストール」](#) でダウンロードした ISO をお使いのマシン上で特定します。

インストールする **OS の種類** と **バージョン** を選択して、仮想マシンに適した OS の種類が選択されていることを確認します。たとえば、**OS の種類** から **Linux** を選択して、**バージョン** から **Red Hat Enterprise Linux 7** を選び、**進む** ボタンを押します。

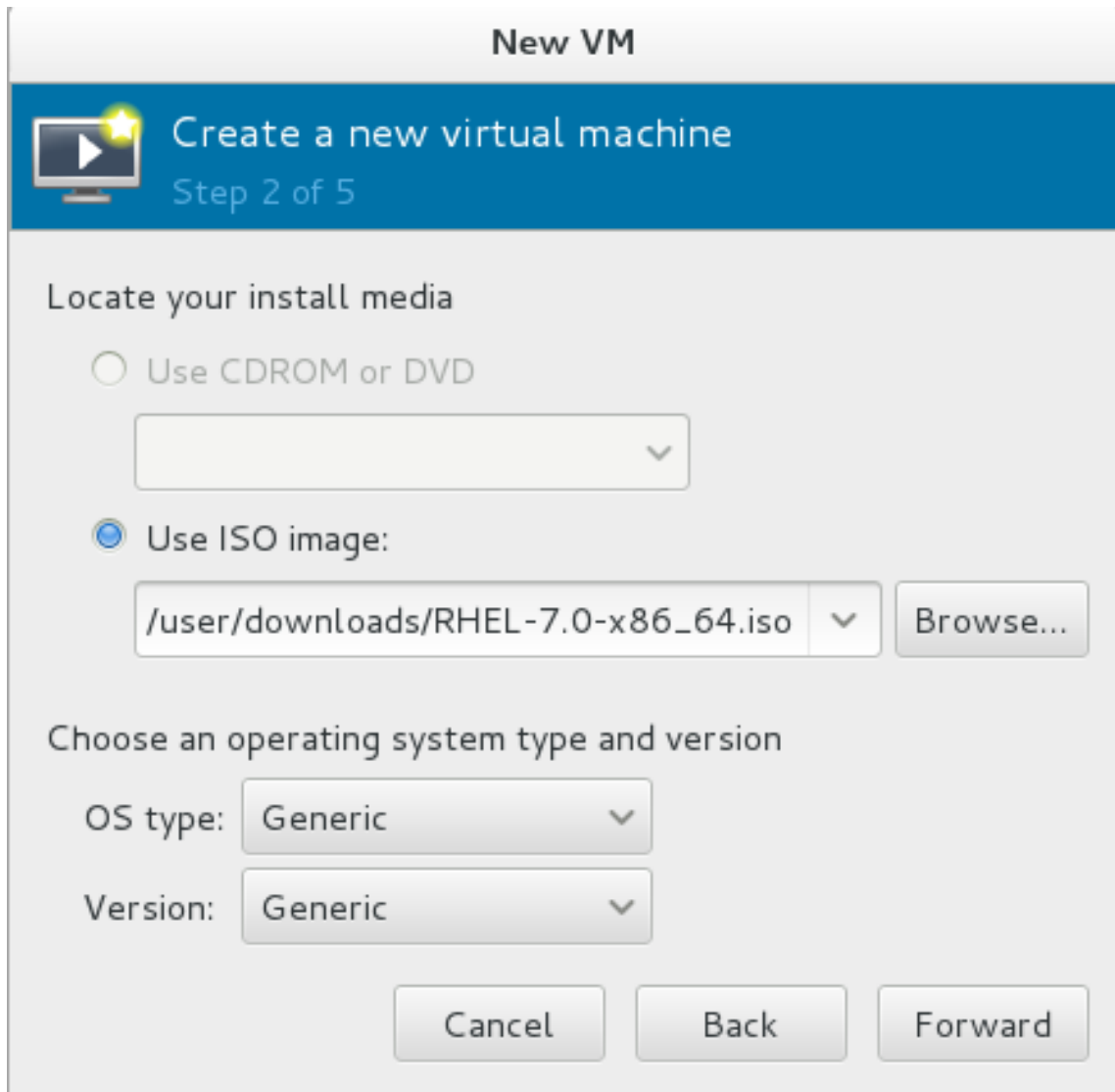


図5.4 ローカル ISO イメージのインストール

5. メモリーおよび CPU を設定します。

仮想マシンに割り当てる CPU 数とメモリー量を選択します。ウィザードでは、割り当て可能な CPU 数とメモリー量が表示されます。

本チュートリアルでは、デフォルト設定を選択して **進む** ボタンをクリックします。

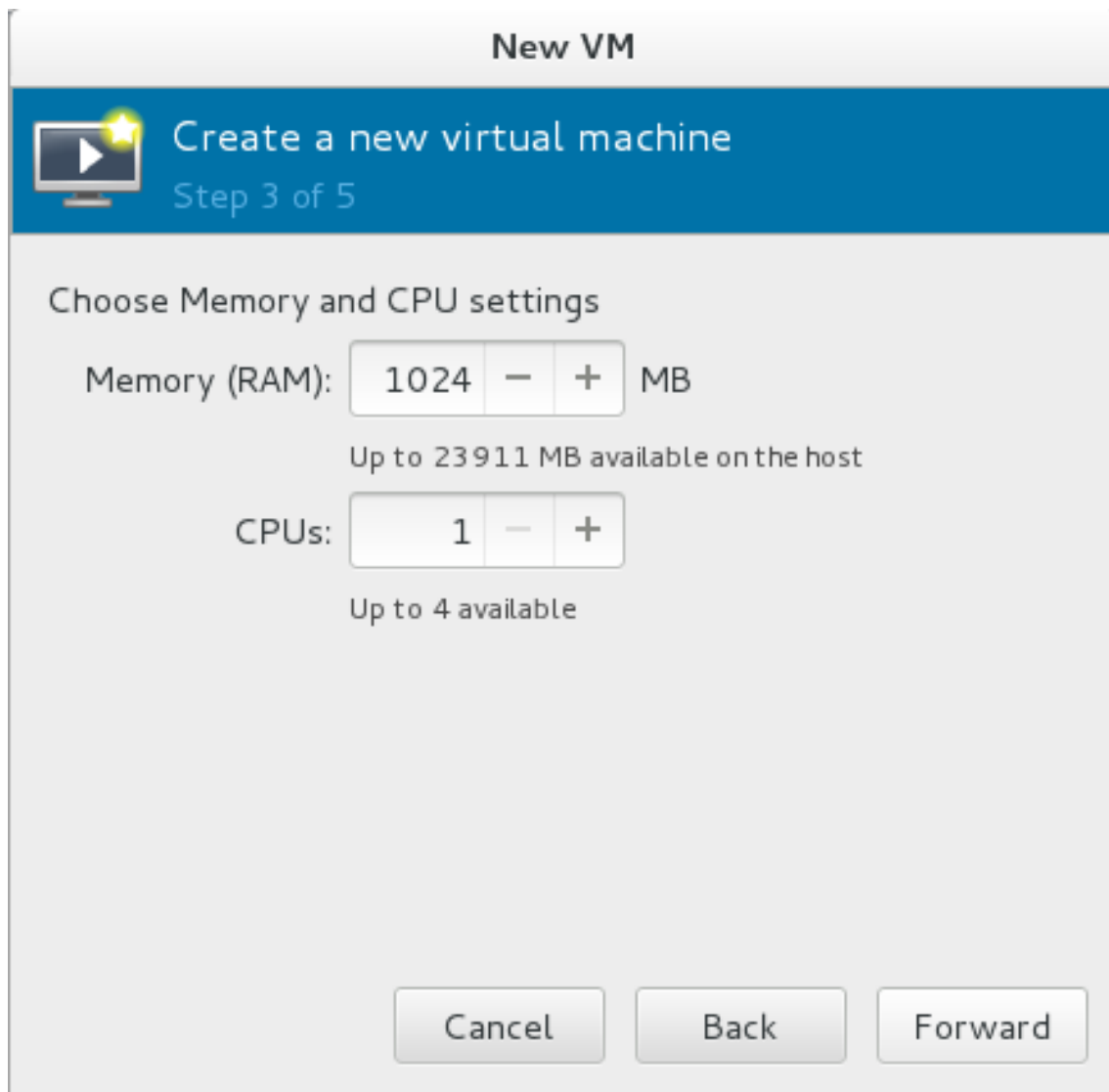


図5.5 CPU およびメモリーの設定

6. ストレージを設定します。

ゲスト仮想マシンにストレージを割り当てます。ウィザードでは、ホストマシンでどこに仮想マシンを保存するかなど、ストレージのオプションが表示されます。本チュートリアルでは、デフォルト設定を選択して **進む** ボタンをクリックします。

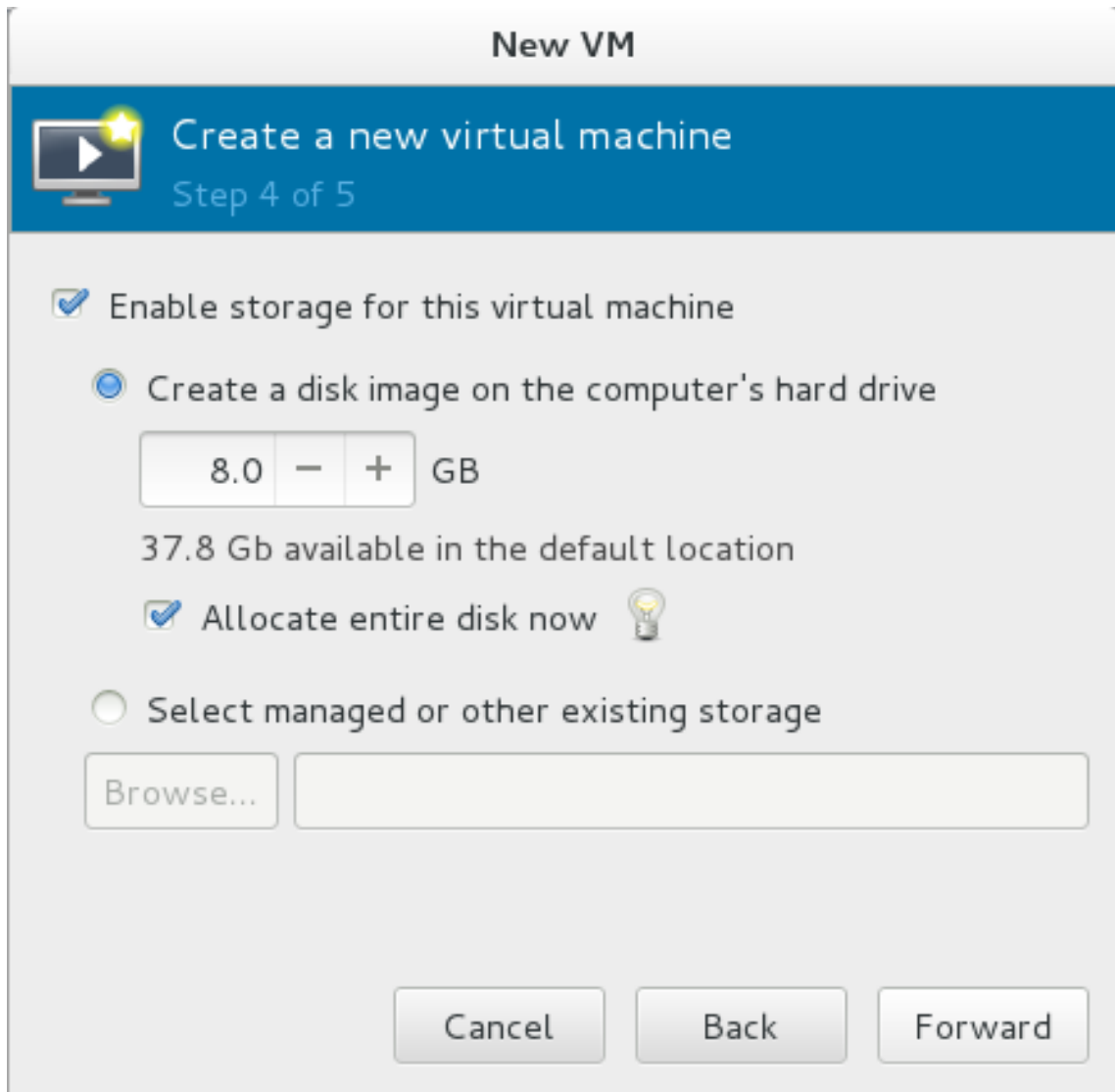


図5.6 ストレージの設定

#### 7. 最終的な設定を行います。

仮想マシンの設定を確認して、完了 ボタンをクリックします。次に、仮想マシンマネージャーで、指定のハードウェア設定で仮想マシンを作成します。

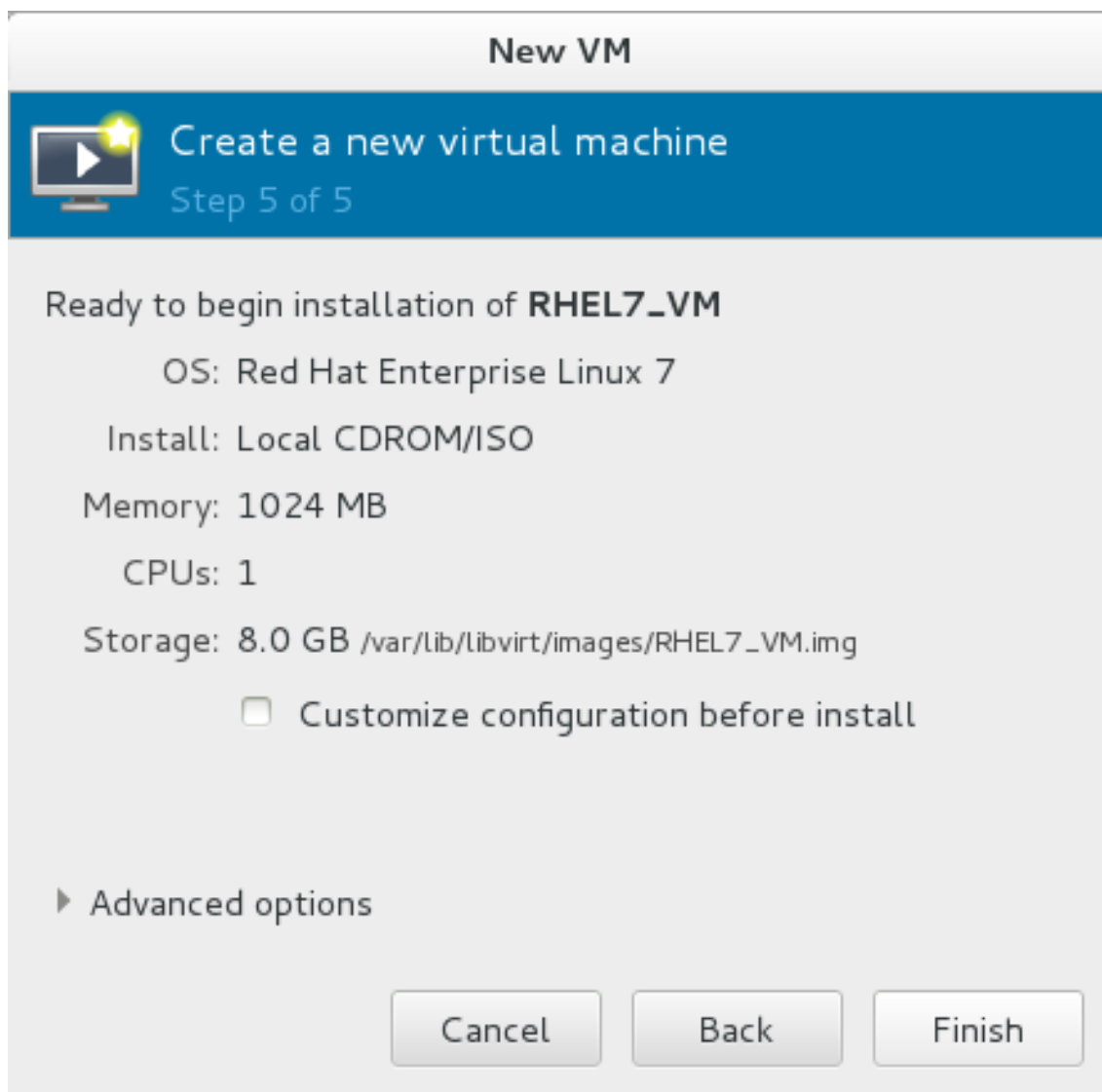


図5.7 設定の確認

仮想マシンマネージャーで Red Hat Enterprise Linux 7 の仮想マシンを作成した後に、Red Hat Enterprise Linux 7 インストーラーの説明に従い、仮想マシンのオペレーティングシステムのインストールを完了します。



### 注記

Red Hat Enterprise Linux 7 のインストールのヘルプに関する情報は、『Red Hat Enterprise Linux 7 インストールガイド』を参照します。

### 5.2.3. ゲスト仮想マシンの検証

仮想マシンマネージャーウィンドウで仮想マシンを選択して、**開く** をクリックすることで、仮想マシンのコンソールを表示できます。物理システムと同じ方法で、コンソールから Red Hat Enterprise Linux 7 仮想マシンを操作できます。

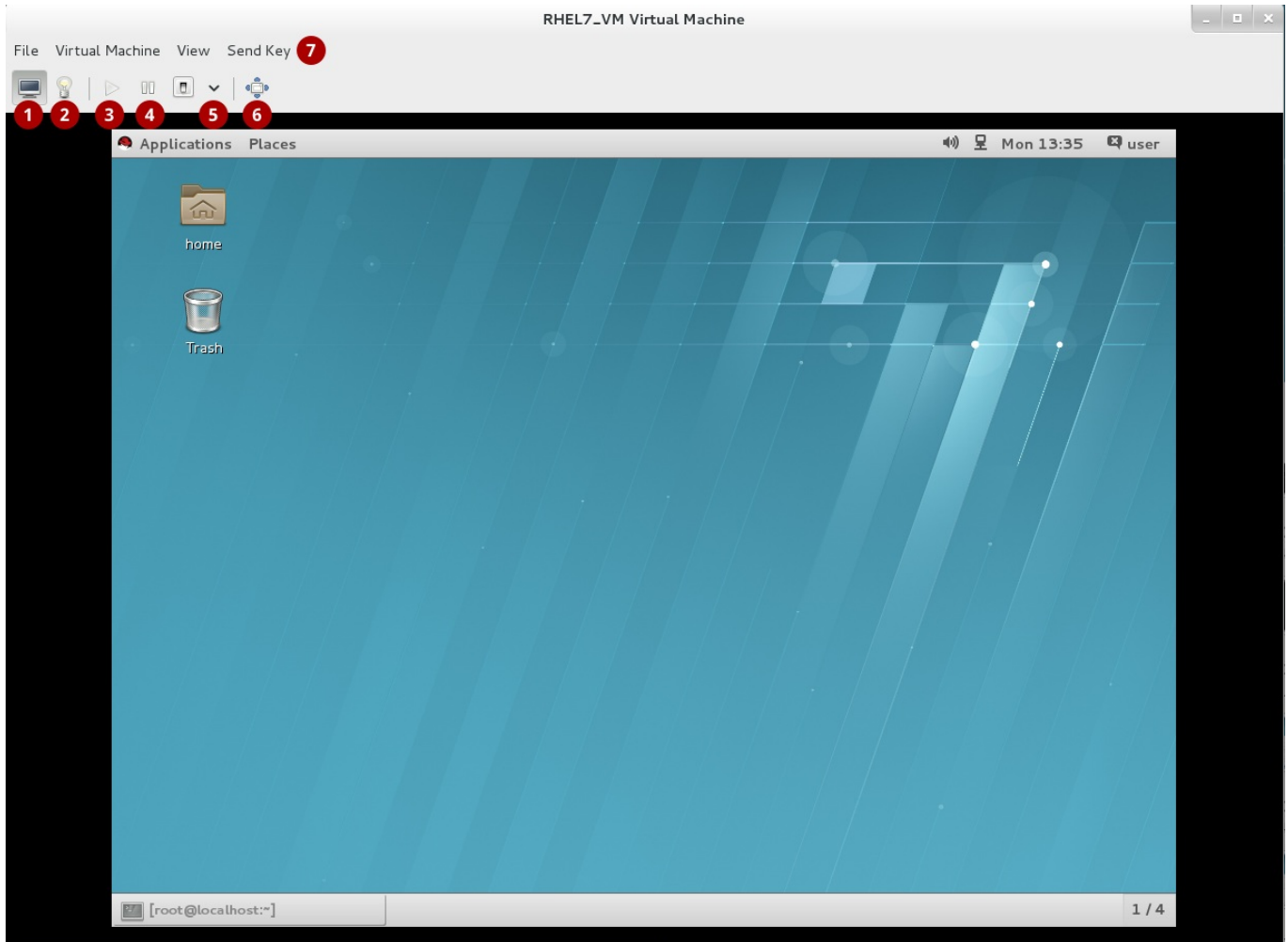


図5.8 ゲスト仮想マシンコンソール

- ※ **1** **グラフィカルコンソールの表示**: これは、仮想マシンの画面上のものを表示します。仮想マシンはコンソールから、物理マシンと同じように操作することができます。
- ※ **2** **仮想ハードウェアの詳細を表示**: このウィンドウは、ゲストが使用する仮想ハードウェアの詳細を表示します。これにより、基本システムの詳細、パフォーマンス、プロセッサ、メモリー、起動設定、システムの仮想デバイスの詳細に関する概要が提示されます。
- ※ 以下のボタンは、仮想マシンの状態を制御します。
  - **3** **仮想マシンの電源を入れる**: 仮想マシンの電源をオンにします。
  - **4** **仮想マシンの一時停止**: 仮想マシンを一時停止します。
  - **5** **仮想マシンをシャットダウン**: 仮想マシンをシャットダウンします。矢印をクリックすると、仮想マシンの電源をオフにするためのオプションが複数含まれたドロップダウン (再起動、シャットダウン、強制的にリセット、強制的に電源OFF、保存) が表示されます。

- ※ **6** フルスクリーン: 仮想マシンを全画面表示に切り替えます。
- ※ **7** キーの送信: Ctrl+Alt+Backspace、Ctrl+Alt+Delete、Ctrl+Alt+F1、PrintScreen などのキーの組み合わせを仮想マシンに送信します。



## 注記

**Virtual Machine Manager** を使用して仮想マシンを実行/作成する方法についての情報は、『Red Hat Enterprise Linux 7 仮想化の導入および管理ガイド』を参照してください。



## 付録A 改訂履歴

改訂 1.0-35.1	Tue Aug 25 2015	Chester Cheng
翻訳ファイルを XML ソースバージョン 1.0-35 と同期		
改訂 1.0-35	Wed Feb 25 2015	Dayle Parker
7.1 GA リリース向けのバージョンを更新 7.1 向けのテクノロジープレビューのステータスを更新 ( <a href="#">BZ#1064610</a> )		
改訂 1.0-34	Wed Feb 18 2015	Dayle Parker
7.1 GA リリース向けのバージョン		
改訂 1.0-33	Tue Feb 17 2015	Dayle Parker
新規 libguestfs ツール (virt-builder、virt-customize、virt-diff、virt-log、virt-v2v) の説明を追加 ( <a href="#">BZ#1134630</a> )		
改訂 1.0-31	Tue Jan 13 2015	Scott Radvan
ガイドの更新バージョンを公開		
改訂 1.0-30	Mon Jan 12 2015	Scott Radvan
ツールの章に virt-sparsify を追加		
改訂 1.0-29	Mon Dec 08 2014	Scott Radvan
スプラッシュページを更新し、新しい sort_order を実装		
改訂 1.0-27	Fri Dec 05 2014	Scott Radvan
RHEL 7.1 Beta リリースの公開		
改訂 1.0-25	Tue Nov 25 2014	Scott Radvan
新製品のセクションを追加		
改訂 1.0-24	Wed July 16 2014	Dayle Parker
SME のフィードバックをもとに QXL ドライバーの説明を編集 ( <a href="#">BZ#1097974</a> )		
改訂 1.0-22	Tues June 3 2014	Dayle Parker
7.0 GA リリース向けバージョン		
改訂 1.0-21	Tues June 3 2014	Dayle Parker
「KVM ゲストマシンの互換性」にあるサポート制限の URL を更新 ( <a href="#">BZ#1064610</a> ) guestmount ツールの警告を更新 ( <a href="#">BZ#1064611</a> )		
改訂 1.0-19	Mon May 19 2014	Dayle Parker
QE フィードバックに合わせてオフラインマイグレーションの説明に「シャットダウン」に関する言及を追加 ( <a href="#">BZ#1064608</a> ) QE フィードバックに合わせて製品の章を若干編集 ( <a href="#">BZ#1064610</a> ) QE レビューで指摘されたツールリストを編集 ( <a href="#">BZ#1064611</a> ) 概要および前書き部分を削除		
改訂 1.0-18	Fri May 9 2014	Dayle Parker
スタイル変更のための再ビルド		

<b>改訂 1.0-14</b>	<b>Tues May 6 2014</b>	<b>Dayle Parker</b>
製品の章にある仮想マシン CPU のホットアドに関する説明を編集		
<b>改訂 1.0-12</b>	<b>Mon April 28 2014</b>	<b>Dayle Parker</b>
クイックスタートの章の画像に吹出しを追加 ( <a href="#">BZ#919320</a> )		
<b>改訂 1.0-10</b>	<b>Tues April 16 2014</b>	<b>Dayle Parker</b>
virsh のセクションを編集 ( <a href="#">BZ#1056848</a> )		
<b>改訂 1.0-09</b>	<b>Mon April 14 2014</b>	<b>Dayle Parker</b>
libvirt のセクションを編集 ( <a href="#">BZ#1056848</a> ) ゲスト CPU モデルのセクションを編集 ( <a href="#">BZ#952964</a> )		
<b>改訂 1.0-08</b>	<b>Wed April 9 2014</b>	<b>Dayle Parker</b>
ベータ版の更新		
<b>改訂 1.0-07</b>	<b>Mon April 7 2014</b>	<b>Dayle Parker</b>
他の仮想化ガイドへの参照を更新 ( <a href="#">BZ#872060</a> ) SME フィードバックに合わせて vCPU ホットアドの説明を編集 ( <a href="#">BZ#953006</a> )		
<b>改訂 1.0-06</b>	<b>Thurs April 3 2014</b>	<b>Dayle Parker</b>
QE フィードバックに合わせてクイックスタートの章を編集 ( <a href="#">BZ#919320</a> ) KVM 製品一覧に vCPU ホットアドの説明を追加 ( <a href="#">BZ#953006</a> )		
<b>改訂 1.0-04</b>	<b>Wed April 2 2014</b>	<b>Dayle Parker</b>
クイックスタートの章を編集 エミュレーションされたシステムのコンポーネントの一覧に PCI Express バスを追加 ( <a href="#">BZ#952442</a> )		
<b>改訂 1.0-03</b>	<b>Wed April 2 2014</b>	<b>Dayle Parker</b>
libvirt セクションに ACL の説明を追加 ( <a href="#">BZ#1056848</a> ) 「Red Hat Enterprise Linux の KVM および仮想化」の最後に KVM の互換性情報を移動 ( <a href="#">BZ#895819</a> )		
<b>改訂 1.0-02</b>	<b>Mon March 31 2014</b>	<b>Dayle Parker</b>
仮想化のクイックスタートの章を追加 ( <a href="#">BZ#919320</a> および <a href="#">BZ#919282</a> ) 追加のディスク I/O スロットルの情報について正しいガイドが参照されるように更新 QE フィードバックに合わせて準仮想化デバイスのセクションを編集 ( <a href="#">BZ#1056429</a> )		
<b>改訂 0.1-23</b>	<b>Mon March 17 2014</b>	<b>Dayle Parker</b>
新たな Red Hat Enterprise Virtualization ドキュメントが含まれるように仮想化ドキュメントスイートを新 NUMA の自動負荷分散に関する説明を追加 ( <a href="#">BZ#1071730</a> ) パフォーマンスのセクションを編集 ( <a href="#">BZ#1056535</a> )		
<b>改訂 0.1-22</b>	<b>Wed March 12 2014</b>	<b>Dayle Parker</b>
他の Red Hat Enterprise Linux 7 ガイドへの参照を更新 ( <a href="#">BZ#872060</a> ) QXL ドライバーの説明を追加 ( <a href="#">BZ#1056429</a> ) ストレージの概念のセクションを更新 ( <a href="#">BZ#1010806</a> ) エミュレートされたストレージドライバーのセクションを編集 ( <a href="#">BZ#1056424</a> )		
<b>改訂 0.1-21</b>	<b>Thurs Feb 27 2014</b>	<b>Dayle Parker</b>

仮想化およびエミュレートされたデバイスのセクションを編集 ([BZ#1056424](#))  
 AHCI の説明を追加  
 ストレージの概念のセクションを更新 ([BZ#1010806](#))

<b>改訂 0.1-20</b>	<b>Fri Feb 7 2014</b>	<b>Dayle Parker</b>
virt-install のセクションを編集 ( <a href="#">BZ#1056896</a> ) Boxes の説明を編集 ( <a href="#">BZ#921813</a> )		
<b>改訂 0.1-19</b>	<b>Mon Jan 13 2014</b>	<b>Dayle Parker</b>
ストレージの概念の説明を追加 ( <a href="#">BZ#1010806</a> ) Boxes の説明を編集 ( <a href="#">BZ#921813</a> ) ガイド全体に QE のフィードバックを適用		
<b>改訂 0.1-18</b>	<b>Thurs Dec 12 2013</b>	<b>Dayle Parker</b>
SME フィードバックをもとに VFIO の説明を編集 ( <a href="#">BZ#952004</a> )		
<b>改訂 0.1-17</b>	<b>Fri Dec 6 2013</b>	<b>Dayle Parker</b>
ベータ版		
<b>改訂 0.1-16</b>	<b>Wed Dec 4 2013</b>	<b>Dayle Parker</b>
VFIO の説明を追加 ( <a href="#">BZ#952004</a> )		
<b>改訂 0.1-15</b>	<b>Thurs Nov 28 2013</b>	<b>Dayle Parker</b>
GlusterFS の説明を追加 ( <a href="#">BZ#1010826</a> )		
<b>改訂 0.1-14</b>	<b>Wed Nov 27 2013</b>	<b>Dayle Parker</b>
7.0 ベータの初期版をビルド		
<b>改訂 0.1-12</b>	<b>Wed Oct 2 2013</b>	<b>Dayle Parker</b>
ディスク I/O スロットリングの説明を追加 ( <a href="#">BZ#1010821</a> ) 準仮想化デバイスのセクションの用語を編集 ( <a href="#">BZ#928129</a> ) 他の仮想化ガイドへの参照を更新 ( <a href="#">BZ#872060</a> )		
<b>改訂 0.1-11</b>	<b>Mon June 10, 2013</b>	<b>Dayle Parker</b>
準仮想化デバイスのセクションに virtio-rng の説明を追加 ( <a href="#">BZ#923008</a> ) GNOME Boxes の説明をツールのセクションに追加		
<b>改訂 0.1-10</b>	<b>Mon May 6, 2013</b>	<b>Dayle Parker</b>
xHCI ホストコントローラーをエミュレートされたシステムのコンポーネント一覧に追加		
<b>改訂 0.1-7</b>	<b>Mon Apr 29 2013</b>	<b>Dayle Parker</b>
著者グループファイルに著者名を追加		
<b>改訂 0.1-6</b>	<b>Fri Apr 26 2013</b>	<b>Dayle Parker</b>
ドキュメントスイート一覧を第 1 章に追加		
<b>改訂 0.1-3</b>	<b>Thurs Jan 31 2013</b>	<b>Dayle Parker</b>
4.4.1 ストレージプールを更新		
<b>改訂 0.1-2</b>	<b>Wed Jan 23 2013</b>	<b>Dayle Parker</b>
初版を作成するために Red Hat Enterprise Linux 6 の言及を Red Hat Enterprise Linux 7 に更新		

