Windows での仮想化：

テクノロジ概要

David Chappell、Chappell & Associates

2008 年 8 月

© Copyright Microsoft Corporation 2008. All rights reserved.

目次

仮想化について **3**

仮想化テクノロジ 3

ハードウェアの仮想化 *4*

プレゼンテーションの仮想化 *6*

アプリケーションの仮想化 *8*

その他の仮想化テクノロジ *10*

仮想環境の管理 10

マイクロソフトの仮想化テクノロジ **11**

ハードウェアの仮想化 12

Hyper-V *12*

Virtual Desktop Infrastructure (VDI) *14*

Virtual PC 2007 ..... *16*

今後の展望: Microsoft Enterprise Desktop Virtualization (MED-V) *17*

プレゼンテーションの仮想化: WINDOWS ターミナル サービス 18

アプリケーションの仮想化: MICROSOFT APPLICATION VIRTUALIZATION (APP-V) 20

仮想化された WINDOWS 環境の管理 **23**

SYSTEM CENTER OPERATIONS MANAGER 2007 23

SYSTEM CENTER CONFIGURATION MANAGER 2007 R2 25

SYSTEM CENTER VIRTUAL MACHINE MANAGER 2008 27

仮想化テクノロジの組み合わせ **30**

結論 **31**

著者について **31**

仮想化について

仮想化は、今日の情報技術の中でも、ひときわ注目度の高いトレンドの 1 つです。しかし、こうなったのにも、決して偶然とは言えない背景があります。仮想化テクノロジの範疇に入るテクノロジは多岐に渡りますが、そのどれもが IT 業界の変革に大きく寄与しています。

このドキュメントでは、ハードウェアの仮想化、プレゼンテーションの仮想化、およびアプリケーションの仮想化の 3 つの分野におけるマイクロソフトの仮想化テクノロジを紹介します。また、どのようなテクノロジでもそうですが、仮想化テクノロジも例に漏れず、効率的に管理する必要があるので、仮想環境向けのマイクロソフトの管理製品についても触れます。このドキュメントの目的は、各仮想テクノロジによって何が実現するのかを明らかにし、そのしくみについて簡単に触れ、各仮想テクノロジがどのように連動するのかを示すことです。

仮想化テクノロジ

最新の仮想化テクノロジを理解するために、まず、仮想化テクノロジを導入していないシステムについて考えます。仮に、Microsoft Word などのアプリケーションをスタンドアロンのデスクトップ コンピューターで実行しているとします。この様子を図 1 に示します。

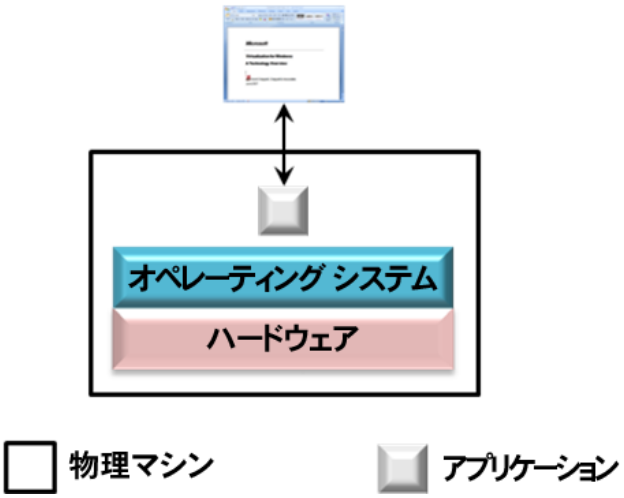

図 1: 仮想化を導入していないシステム

アプリケーションはオペレーティング システム環境に直接インストールされ、そこで実行されます。オペレーティング システムはコンピューターのハードウェア上で直接実行されます。アプリケーションのユーザー インターフェイスは、そのコンピュータに直接接続されているディスプレイを介して表示されます。この例は、Windows を使ったことのある人なら、だれにでもなじみのある単純な例です。

しかし、これが唯一の選択肢ではありません。実際には、最良の選択肢とは言えないケースも多々あります。上記の例では、複数の要素が強固な依存関係の上に成り立っています。つ

まり、オペレーティング システムはハードウェアに、アプリケーションはオペレーティング システムに、そしてユーザー インターフェイスはローカル マシンにそれぞれ強く依存しています。しかし、このような要素間の直接的な依存関係を緩めることが可能です。

依存関係を緩めるということは、環境の各要素を仮想化することです。その実現方法はさまざまです。たとえば、*ハードウェアの仮想化*を行うと、オペレーティング システムが実行されている物理的なハードウェアから、そのオペレーティング システムを切り離すことができます。同様に、*アプリケーションの仮想化*を行うと、オペレーティング システムと、そのオペレーティング システムを使用するアプリケーションとの関係を切り離すことができます。また、*プレゼンテーションの仮想化*を行うと、アプリケーションが実行されている物理マシンから、そのアプリケーションのユーザー インターフェイスを切り離すことができます。このような仮想化のアプローチはすべて、構成要素の間のつながりを柔軟にします。その結果、ハードウェアとソフトウェアの利用方法が多様化し、またどちらも変更が容易になります。企業のほとんどの IT スタッフは、勤務時間の大部分をハードウェアとソフトウェアの新規導入よりもむしろ保守作業に奪われているので、環境の柔軟性が高くなることは望ましいことです。

そのほかにも、仮想化のタイプごとに利点がいろいろとありますが、その利点の内容は、それぞれのタイプで対象となる課題に応じて異なります。利点の内容を理解するには、各テクノロジ自体の知識を深める必要があります。そこで、この後の各セクションでは、各テクノロジを詳しく見ていくことにします。

ハードウェアの仮想化

今日のほとんどの IT スタッフが "仮想化" という言葉を聞いて思い浮かべることは、1 台の物理マシン上で複数のオペレーティング システムを実行することです。これがハードウェアの仮想化です。重要なタイプの仮想化がこれ 1 つだけというわけではありませんが、今日最も多く見受けられるタイプであることに間違いはありません。

ハードウェア仮想化の核となる概念は単純です。それは、ソフトウェアを使って、*仮想マシン (VM)* という、擬似的な物理コンピュータを作成することです。一度に複数の VM を実行できるこのアプローチでは、1 台の物理マシン上で複数のオペレーティング システムを同時に実行できます。この様子を図 2 に示します。

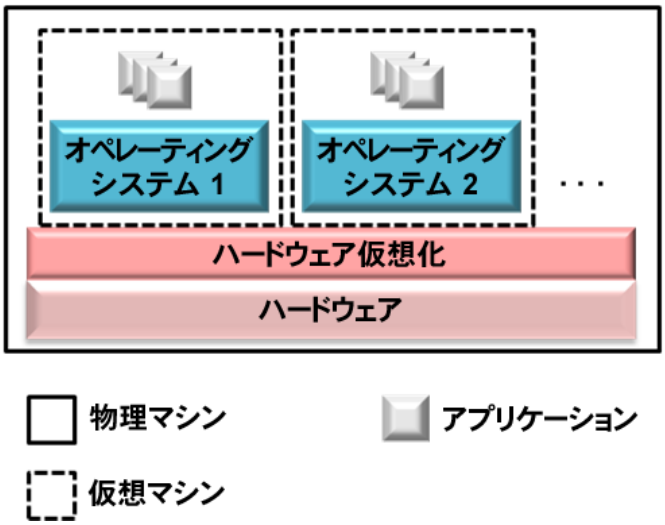

図 2: ハードウェア仮想化の図

このアプローチは、クライアント マシンで利用するときは一般に *デスクトップ*の仮想化と言います。サーバー システムで利用するときは、*サーバー*の仮想化と言います。デスクトップの仮想化はさまざまな状況に役立てることができます。最も一般的なのは、アプリケーションがデスクトップ オペレーティング システムに対応していないという問題に対処する場合です。たとえば、Windows Vista の実行環境で、Windows XP Service Pack 2 でしか実行できないアプリケーションを使用する必要が生じたとします。この場合、この旧バージョンのオペレーティング システムを実行する VM を作成し、その VM にアプリケーションをインストールすれば、問題を解決できます。

デスクトップを仮想化すると確かに便利ですが、ハードウェアの仮想化の対象として本当に魅力的なものは、サーバーに尽きると言っていいでしょう。その主な理由は、経済性にあります。固定的なワークロードを各サーバー マシンに割り振るようにすると、利用率の低いサーバー マシンを何台も投じることになりますが、サーバーを仮想化すれば、各マシンの性能を目一杯使い切るまでワークロードを割り振ることができるため、サーバー マシンの台数が少なくなります。つまり、コンピューターの管理スタッフ、設置スペース、および稼働電力が少なくて済み、結果的にコストの節約につながります。

サーバーの仮想化によって、障害が発生したシステムの復元も容易になります。VM はファイルとして保存されるので、障害が発生したシステムを復元するときは、VM のファイルを新しいマシンにコピーするだけです。また、VM は、その VM を実行している物理マシンとは異なるハードウェア構成にできるので、障害が発生したシステムを空いているマシンに復元できます。物理的に同じシステムを使用する必要はないのです。

ハードウェアの仮想化はさまざまな方法で実現できます。そのため、マイクロソフトは、この分野を対象とした複数のテクノロジを用意しています。次のような製品があります。

- Hyper-V: Windows Server 2008 の一部である Hyper-V は、サーバーのハードウェア仮想化を行います。

- Virtual Desktop Infrastructure (VDI): Hyper-V と Windows Vista をベースとする VDI は、仮想デスクトップの作成方法を規定します。

- Virtual PC 2007: Windows Vista と Windows XP 用に無料でダウンロードできる Virtual PC は、デスクトップ システムのハードウェア仮想化を行います。

- Microsoft Enterprise Desktop Virtualization (MED-V): MED-V を使用すると、1 つまたは複数のアプリケーションが入った Virtual PC ベースの VM を作成し、それをクライアント マシンに配布できます。

これらのテクノロジは、それぞれ適している状況が異なります。このドキュメントで、それぞれの詳細について後述します。

プレゼンテーションの仮想化

一般的に使用されているソフトウェアの多くは、実行もユーザー インターフェイスの表示も同じマシン上で行うことを前提にしています。Microsoft Office のアプリケーションは、その代表的な例に該当しますが、このようなソフトウェアは他にもたくさんあります。通常はこの既定の利用方法で問題ありませんが、マイナス面がないわけでもはありません。たとえば、多数のデスクトップ マシンを管理している企業では、各マシン上の機密データが少しでも漏れないように、セキュリティを厳重に確保しなければなりません。また、各マシンで使用されているアプリケーションの管理に莫大な時間と資金を費やすことを強いられます。このようなケースでは、アプリケーションをリモート サーバーで実行するようにし、そのアプリケーションのユーザー インターフェイスをローカルで表示するようにすれば問題が解決します。このようなアプローチをプレゼンテーションの仮想化と言います。この様子を図 3 に示します。

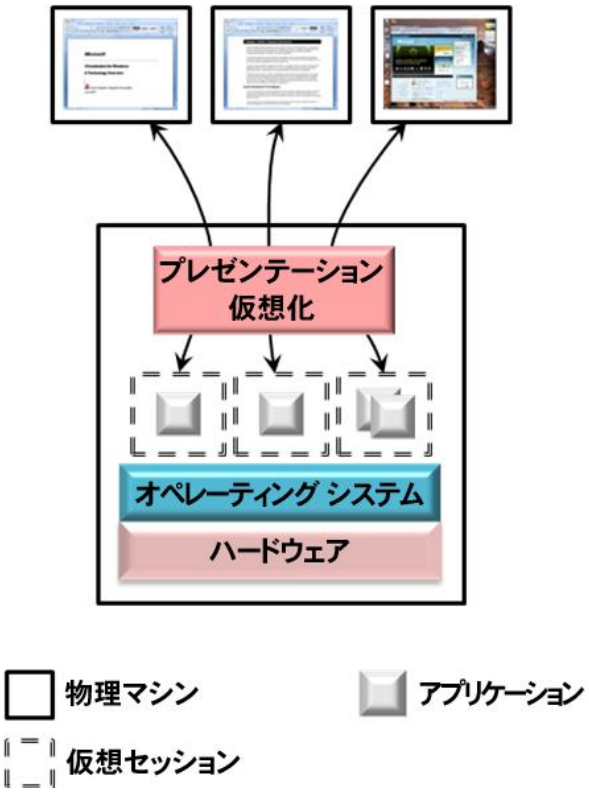

図 3: プレゼンテーション仮想化の図

図からわかるように、このアプローチでは、複数の*仮想セッション*が作成されますが、各セッションと対話するマシンは、同じリモート デスクトップ システムです。セッションで実行されるアプリケーションは、プレゼンテーションの仮想化を介して、ユーザー インターフェイスの表示内容とリモートでの処理を対応付けます。それぞれのセッションで、アプリケーションを 1 つだけ実行することもできれば、現実のデスクトップとまったく同じように、複数のアプリケーションを実行することもできます。どちらの場合でも、インストール済みのアプリケーションを複数の仮想セッションで共有することも可能です。

このように共有サーバー上でアプリケーションを実行すると、次のような利点があります。

- データを一元化して、複数のデスクトップ マシンではなく、中央のサーバーに安全に保存できます。情報がいろいろなシステムでばらばらに管理されることにはならないので、セキュリティが向上します。

- アプリケーションの管理コストを大幅に削減できます。たとえば、各デスクトップでアプリケーションを 1 つずつアップデートする代わりに、サーバー上の共有版のアプリケーションを 1 つだけ変更すればよいことになります。この他にも、プレゼンテーションの仮想化には、使用するデスクトップ オペレーティング システム イメージが軽いもので済み、デスクトップ デバイスの用途を専門化できるという特長があります (一般的に*シン クライアント*と呼ばれます)。どちらも管理コストの削減に役立ちます。

- アプリケーションとデスクトップ オペレーティング システムとの相性を心配する必要がなくなります。前に説明したように、この問題はデスクトップの仮想化でも解決できますが、アプリケーションを中央のサーバーで実行し、プレゼンテーションの仮想化を利用して、任意のオペレーティング システムを実行しているクライアントからそのアプリケーションにアクセスできるようにする方が簡単な場合があります。

- 場合によっては、プレゼンテーションを仮想化にすることで、パフォーマンスを向上できることもあります。たとえば、中央のデータベースからクライアントに大量のデータをダウンロードするクライアント/サーバー アプリケーションがあるとします。クライアントとサーバーを結ぶネットワーク リンクが低速であるか、輻輳が発生している場合は、このアプリケーションの処理も低速になります。アプリケーションのパフォーマンスを向上させる 1 つの方法としては、アプリケーション全体 (クライアントとサーバーの両方) を、データベースに高帯域幅で接続された 1 台のマシン上で実行し、プレゼンテーションの仮想化を利用してアプリケーションのインターフェイスをユーザーに表示させる方法があります。

マイクロソフトのプレゼンテーション仮想化テクノロジは、Windows ターミナル サービスという名前のテクノロジです。最初に Windows NT 4 で導入され、今では Windows Server 2008 に標準で組み込まれています。ターミナル サービスを利用すると、普段使っている Windows デスクトップ アプリケーションを共有サーバー マシン上で実行し、そのユーザー インターフェイスをリモート システム、たとえばデスクトップ コンピューターやシン クライアントに表示できます。仮想化という観点からすると、リモート インターフェイスがその範疇に入るか見方は分かれますが、仮想化の範疇で捉えたほうが、このテクノロジが広く応用されていることの説明が付くと考えられます。

アプリケーションの仮想化

仮想化を行なうと、抽象的に表現された何らかのコンピューティング リソースを利用できるようになります。たとえば、ハードウェアの仮想化では、物理コンピューター上でオペレーティング システムを直接実行せずに、ソフトウェア的に抽象化されたマシンの上でオペレーティング システムを実行します。同様に、プレゼンテーションの仮想化では、アプリケーションのユーザー インターフェイスがリモート デバイスとして抽象化されます。どちらの場合も、本来は緊密な要素間の関係が緩やかになります。

抽象化する手法は、アプリケーションと、そのアプリケーションが実行されているオペレーティング システムとの関係にも効果があります。どのアプリケーションも、メモリ割り当てやデバイス ドライバーなどのさまざまなサービスに応じたオペレーティング システムの各機能に依存しています。アプリケーションとそのオペレーティング システムとの相性が悪くても、前に説明したように、ハードウェアの仮想化またはプレゼンテーションの仮想化によって対処できます。しかし、オペレーティング システムの同一インスタンスにインストールされた 2 つのアプリケーション間の相性が悪い場合はどうしたらよいでしょうか。アプリケーションは一般に、同じシステムにインストールされた他のアプリケーションとさまざまな要素を共有しますが、この共有が問題となることがあります。たとえば、あるアプリケーションが動作するには特定バージョンのダイナミック リンク ライブラリ (DLL) が必要で、別のアプリケーションには同じ DLL の別のバージョンが必要である場合があります。両方のアプリケーションをインストールすると、いわゆる *DLL 地獄*が発生し、一方に必要なバージョンが他方によって上書きされてしまいます。この問題を回避するため、企業では通常、新し

いアプリケーションをインストールする前に広範なテストを実施します。このアプローチは効果があるものの、時間とコストがかかります。

この問題は、アプリケーションの仮想化で解決できます。図 4 に示すように、すべての共有リソースをアプリケーションごとに別々のバージョンとして分けて作成するのです。同一システム上のアプリケーション間で共有したときに、共有が原因となって問題が発生するような要素、たとえばレジストリ エントリや特定の DLL は、共有する代わりに、それぞれのアプリケーションとまとめてパッケージ化し、*仮想アプリケーション*を作成します。仮想アプリケーションを導入すると、そのアプリケーション独自の共有リソースが使われるようになります。

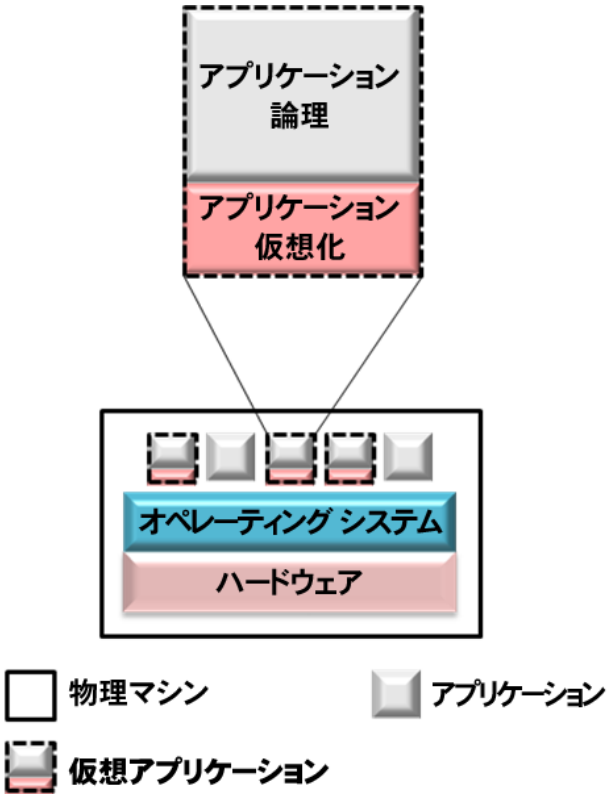

図 4: アプリケーションの仮想化の図

アプリケーションを仮想化することで、運用作業が大幅に簡素化されます。DLL のバージョンや環境内のその他の共有リソースに対するアプリケーション間の競合がなくなるので、新しいアプリケーションを導入する前に既存アプリケーションとの競合をテストする必要がなくなります。また、図 4 からわかるように、仮想アプリケーションは通常のアプリケーションと並行して実行できます。つまり、すべてのアプリケーションを仮想化する必要はありません。

この分野でのマイクロソフトのテクノロジとして、Microsoft Application Virtualization (略称は App-V) があります。App-V の管理者は、仮想アプリケーションを作成したら、必要に応

じて展開することができます。システムの主要要素を抽象的に見せるようにすれば、アプリケーションの導入とアップデートに必要な時間とコストが削減されます。

その他の仮想化テクノロジ

このドキュメントでは、ハードウェア、プレゼンテーション、およびアプリケーションの 3 タイプの仮想化について説明しています。ただし、同じように抽象化する手法は、その他のコンテキストでも使用されています。中でも、ネットワークの仮想化とストレージの仮想化はひときわ重要性が高いものです。

*ネットワークの仮想化*という言葉は、多数の異なる概念を表すために使用されています。おそらく最も一般的なのは、仮想プライベート ネットワーク (VPN) の概念です。VPN はネットワーク接続を抽象的に表現したものです。これにより、リモート ユーザーは、企業の社内ネットワークに物理的に接続しているときと同じ感覚でアクセスすることが可能になります。VPN は幅広く導入されている概念であり、さまざまなテクノロジを利用できます。マイクロソフトで今日の主要な VPN テクノロジは、Internet Security and Acceleration (ISA) Server 2006 と Internet Application Gateway 2007 です。

*ストレージの仮想化*という言葉も、きわめて幅広い意味で使用されています。一般的には、物理的なストレージ デバイスを、抽象化して論理的なストレージ デバイスとして表現することを意味します。したがって、ローカル接続のディスク ドライブ以外のストレージ デバイスはすべて、この概念に当てはまると考えることができます。1 つの簡単な例として、Windows のフォルダー リダイレクトがあります。これは、フォルダー内の情報を、ネットワークアクセス可能な任意のドライブに保存できる機能です。このカテゴリには、ストレージ エリア ネットワーク (SAN) など、さらに強力な (さらに複雑な) アプローチも当てはまります。その方法に関係なく、ストレージの仮想化の利点は他のタイプの仮想化の利点と似ています。つまり、抽象化の度合いが増すほど、要素間の直接的なつながりが弱くなるということです。

仮想環境の管理

仮想化テクノロジにはさまざまな利点があります。しかし、コンピューティング環境の仮想化が進むほど、抽象化の度合いが増していくことになります。抽象化の度合いが増すにつれ、複雑さが増す可能性があり、IT スタッフは環境を制御することがいっそう困難になります。その結果は明らかです。仮想環境が適切に管理されなければ、その利点は実現できません。

たとえば、複数の既存サーバー マシンのワークロードを、1 台のサーバー上で実行している複数の仮想マシンに移す場合を考えてみます。この 1 台の物理コンピューターは、このコンピューターに移す前のすべてのマシンと同じ重要性を持つようになります。このコンピューターで障害が発生すれば、大混乱となるでしょう。適切に管理されていない仮想環境は、仮想化されていない環境よりも信頼性が下がり、コストがかかる可能性があります。

この問題に対処するため、マイクロソフトはシステム管理用の一連のツールを用意しています。仮想環境の管理作業は物理環境の管理作業とほとんど同じであるため、両方に同じツールを使用できます。両方の環境で同じスキルと知識を活用できるということは、環境を管理するスタッフにとって、都合の良いことです。それでも、仮想化に特化したツールのほうが適しているケースはあります。System Center Operations Manager 2007、System Center

Configuration Manager 2007 R2、および System Center Virtual Machine Manager 2008 は、いずれの環境にも使用できるマイクロソフトの製品です。

システム管理における根本的な課題は、分散環境のハードウェアとソフトウェアを監視および管理することです。System Center Operations Manager 2007 は、この問題に対処するためのマイクロソフトの主力製品です。Operations Manager を使うと、物理マシン上で実行されているソフトウェアと物理マシン自体を監視できるため、運用スタッフは環境内で何が起こっているかを把握できます。また、問題が起きたときの対処も適切に実施できるようになり、通常の業務をこなしながら、問題が発生した場合の対処も並行して講じることができます。物理環境と仮想環境は類似性が高いことから、Operations Manager は、仮想マシンおよび仮想環境のその他の要素の監視、管理にも使用できます。

コンピューティング環境の管理スタッフが避けて通ることのできない別の問題として、ソフトウェアのインストールと、ソフトウェアの構成方法の管理があります。このような作業を手作業で行うことも可能ですが、ごく小規模な環境でもない限り、作業を自動化する方がはるかに優れたアプローチです。このアプローチを可能にするためにマイクロソフトが開発した製品が System Center Configuration Manager 2007 R2 です。Operations Manager と同様、Configuration Manager でも、物理環境の場合とほぼ同じ方法で仮想環境が扱われます。このツールもまた、両方の環境に使用できます。

Operations Manager と Configuration Manager はどちらも、専門性の高い IT スタッフを擁する大企業向けの製品です。中規模の企業については、 この 2 つの製品を組み合わせて使用することも確かに可能ですが、マイクロソフトは、それほど複雑でない環境用に、よりシンプルなツールを用意しています。それが System Center Essentials 2007 です。このツールは、Operations Manager と Configuration Manager の最も重要な機能を実装しています。Operations Manager および Configuration Manager と同様、仮想テクノロジが物理システムと同じように認識されるので、両環境の管理に使用できます。

物理環境と仮想環境の両方で使えるツールは魅力があります。しかし、VM が数十台もしくは数百台も存在する環境を想像してください。これだけ多くのマシンをどのようにして作成するのでしょうか。また、いらなくなったときに、どのようにして破棄するのでしょうか。そのほかにもある VM 専用の管理機能はどのようにして実行するのでしょうか。このような問題に対処するには、ハードウェアの仮想化の管理に特化したツールが必要です。Windows の VM に対するマイクロソフトの答えが System Center Virtual Machine Manager 2008 です。このツールの特徴は、運用スタッフが仮想化を行うワークロードを選択し、選択したワークロードを実行する VM を作成し、その VM にアプリケーションを移行できることです。

仮想化の全体像を理解するには、仮想環境をどのようにして管理できるかを理解する必要があります。しかし、仮想化テクノロジ自体の理解も必要です。この理解を深めるため、次のセクションでは、マイクロソフトの各仮想化テクノロジを詳細に見ていきます。

マイクロソフトの仮想化テクノロジ

すべての仮想化テクノロジは、コンピューティング リソースの有益性を高めるために、何らかの抽象化を行います。これが仮想化の核となる概念であり、これは抽象化の対象がコンピューターであっても、アプリケーションのユーザー インターフェイスであっても、アプリケ

ーションを実行する環境であっても変わりません。これらのテクノロジはすべて重要ですが、特にハードウェアの仮想化が、現在最も注目されているテクノロジと言って差し支えないでしょう。そこで、まずはこのテクノロジから始めることにしましょう。

ハードウェアの仮想化

コンピューティングの全体的な傾向のうち、処理性能が加速度的に高くなっているという、大きな流れに乗っているものがその大半を占めます。この流れは、ムーアの法則で表現されています。このような進歩がどの程度のものかわかりやすくするために、1 つ例を挙げます。プロセッサの能力は、2 年後には、コンピューターの黎明期から現在まで向上した分と同じだけ向上すると言われています。このように短期間で能力が向上するとすれば、マシンの利用率をどうやって高くするかが大きな課題になってきます。また、さまざまなアプリケーションのさまざまなワークロードを単一のオペレーティング システム上で実行する難しさもあります。結果的に、利用率の低いサーバーが多数存在するという状況になります。サーバーマシンはそれぞれに購入、設置、稼働、管理のコストがかかります。したがって、サーバーの利用率が高くなるようなテクノロジがあれば、非常に魅力的に映るはずです。

ハードウェアの仮想化がまさにそのテクノロジであり、間違いなく大きな魅力を持っています。ハードウェアの仮想化は 40 年前に生まれた概念ですが、ここに来てようやくコンピューティング環境の主流を成す主要なテクノロジになろうとしています。そう遠くない将来には、ほとんどのアプリケーションが、専用の物理マシンではなく、仮想化サーバーに導入されると推測されます。その利点は、軽んじることができないほど絶大です。

Windows ユーザーがこの利点を享受できるようにするため、マイクロソフトは現在、2 つの基礎的なハードウェア仮想化テクノロジを提供しています。それは、サーバー用の Hyper-V とデスクトップ用の Virtual PC 2007 です。これらのテクノロジは、その他のマイクロソフト テクノロジ、たとえば Virtual Desktop Infrastructure (VDI) や Microsoft Enterprise Desktop Virtualization (MED-V) の基盤にもなっています。この後の各セクションで、各テクノロジについて説明します。

Hyper-V

ハードウェアを仮想化する上で、まず第 1 に課題になることは、どうやってソフトウェアの世界の中で仮想マシンを作成するかです。最も効率的な作成方法は、ハードウェアのすぐ上に、*ハイパーバイザー*という小型のソフトウェアから成るレイヤーをかぶせる方法です。マイクロソフトのハイパーバイザーとしては、Windows Server 2008 の一部である Hyper-V があります。Hyper-V 上の各 VM は、他の VM から完全に独立し、それぞれ独自のゲスト オペレーティング システムを実行します。したがって、各 VM のワークロードは、専用の物理サーバーで実行しているのと同じ状態になります。この様子を図 5 に示します。

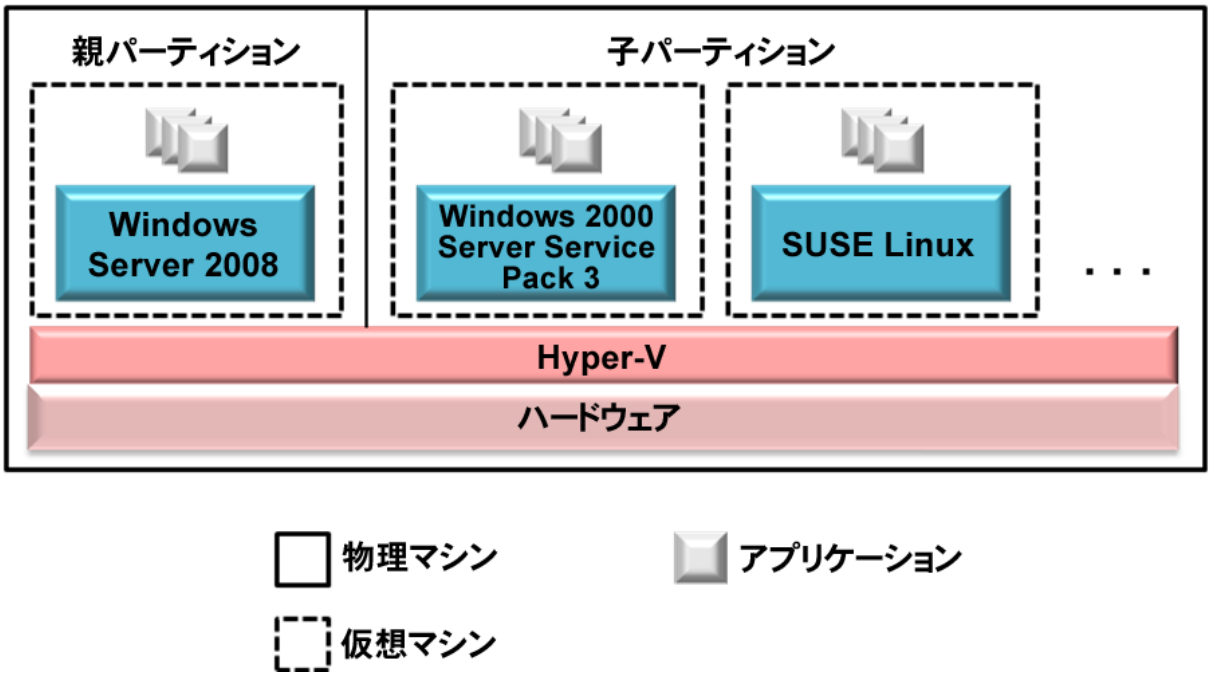

図 5: Windows Server 2008 の Hyper-V の図

図に示すように、Hyper-V の世界では VM は*パーティション*と呼ばれます。このうちの 1 つ、*親*パーティションでは、Windows Server 2008 を実行する必要があります。*子*パーティションでは、サポートされている任意のオペレーティング システム、たとえば、Windows Server 2008、Windows Server 2003、Windows Server 2000、Windows NT 4.0、また SUSE Linux などの Linux ディストリビューションを実行できます。新しいパーティションを作成および管理するときは、親パーティションで実行されている MMC スナップインを使用できます。

このアプローチは、ハードウェア仮想化のためのマイクロソフトのこれまでのサーバー テクノロジとは根本的に異なります。Windows Server 2003 で使用されていた仮想化テクノロジ Virtual Server 2005 R2 は、ハイパーバイザーとして機能するわけではありません。構造的に見ると、オペレーティング システムの上に乗る形になっています。この 2 つのアプローチの重要な違いの 1 つは、Windows ハイパーバイザーによる低レベルのサポートの方が、仮想化を効率良く行うことができ、パフォーマンスも高いことです。

パフォーマンスが高くなるように、Hyper-V は、別の点でも工夫されています。Hyper-V を使用すると、たとえば複数の CPU を 1 つの VM に割り当てることができますが、Hyper-V 自体はネイティブの 64 ビット テクノロジです (Hyper-V は Windows Server 2008 の 3 つの 64 ビット版である Standard、Enterprise、および Data Center に組み込まれていますが、32 ビット版には含まれません)。つまり、広大な物理メモリ領域を扱うことができるので、1 台の物理サーバー上で多数の仮想マシンを実行しなければならない場合に便利です。また、Hyper-V では、サポートされる VM にそれぞれ、最大 64 GB のメモリを割り当てることができます。Hyper-V 自体は 64 ビット テクノロジですが、32 ビットと 64 ビットの両方の VM をサポートしています。どちらのタイプの VM も 1 台の Windows Server 2008 マシン上で同時に実行できます。

実行されているオペレーティング システムに関係なく、すべての VM にはストレージ領域が必要です。VM にストレージ領域を持たせるため、マイクロソフトはバーチャル ハード ディスク (VHD) 形式を開発しました。VHD は実際には単なるファイルですが、仮想マシンからは、接続ディスク ドライブとして認識されます。ゲスト オペレーティング システムとそのアプリケーションは、1 つまたは複数の VHD を介して、ストレージ領域にアクセスします。業界に浸透しやすくなるように、マイクロソフトは Open Specification Promise (OSP) に VHD の仕様を加え、他社もこの形式を自由に実装できるようにしました。また、Hyper-V では Virtual Server 2005 R2 と同じ VHD 形式が使用されているので、この旧テクノロジからの移行の手間はさほどかかりません。

Windows Server 2008 には*サーバー コア*というインストール オプションがあります。このオプションでは、システム機能のサブセットだけがインストールされます。サーバー コアをインストールすると、管理作業が軽減され、システムに潜んでいるかもしれないセキュリティ上の脅威がなくなるので、Hyper-V を導入するサーバーにお勧めです。ただし、このオプションを使用したシステムはグラフィカル ユーザー インターフェイスがサポートされないため、Windows Server Virtualization Management スナップインをローカルで実行することはできません。その代わりに、Virtual Machine Manager を使用して VM の管理をリモートで実行できます。Windows Server 2008 は、従来の仮想化されていない構成に導入することもできます。この場合、Hyper-V はインストールされず、オペレーティング システムはハードウェア上で直接実行されます。

ハードウェアの仮想化は、今日の主流となるテクノロジです。ハードウェアの仮想化を Windows の基本要素にすることをマイクロソフトが決定したという事実だけでも、その重要性がいかに大きいかがわかります。コンピューティングの歴史を振り返ってみると、ハードウェアの仮想化は、長い間くすぶっていた感がありますが、ここに来てようやく、その考え方の実用性が認められ、実を結んだ感があります。

Virtual Desktop Infrastructure (VDI)

Hyper-V で作る VM は、さまざまな用途に応用が利きます。たとえば、Virtual Desktop Infrastructure と呼ばれるアプローチでは、Hyper-V を使って複数のクライアント デスクトップを 1 台のサーバー上で実行できます。この概念を図 6 に示します。

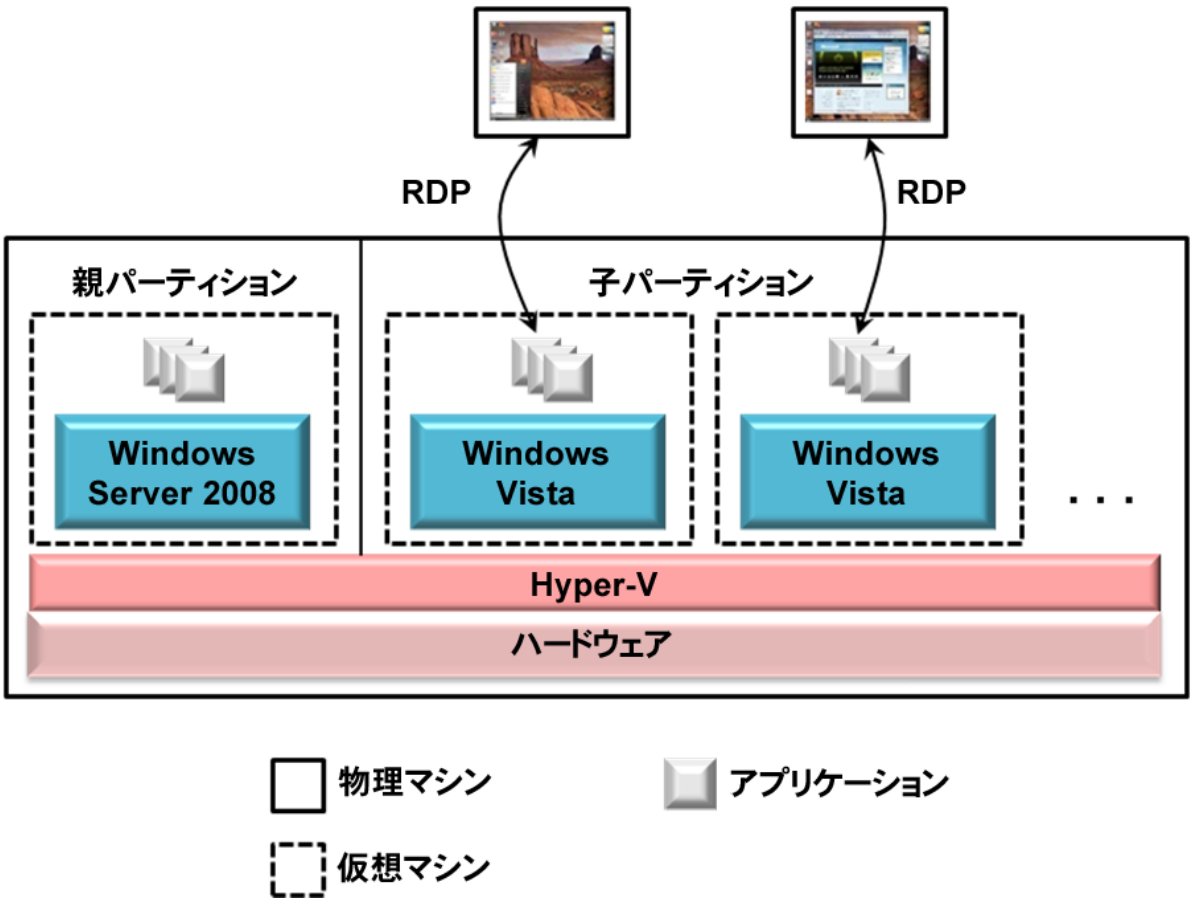

図 6: Virtual Desktop Infrastructure の図

図からわかるように、VDI は Hyper-V のそれぞれの子パーティション (Hyper-V の VM) で Windows Vista のインスタンスを実行します。Vista にはリモート デスクトップ プロトコル (RDP) のサポートが組み込まれており、このサポートにより Vista のユーザー インターフェイスへのリモート アクセスが可能になります。クライアント マシンは、シン クライアント、Macintosh システム、Windows システムなど、RDP をサポートしているものであれば何でもかまいません。これがプレゼンテーションの仮想化に似ていると思われたとしたら、正解です。というのも、RDP は元々 Windows ターミナル サービスのために開発されたからです。しかし、VDI の場合には、プレゼンテーションの仮想化テクノロジを導入する作業は特に必要ありません。その作業は Hyper-V と Vista によって実行されます。

プレゼンテーションの仮想化と同様、VDI を使用すると各ユーザーはそれぞれ専用のデスクトップを利用できます。このとき、クライアント マシンにデスクトップをインストールして管理する場合のコストもセキュリティ上のリスクも発生しません。別の利点として、日中に VDI 用に使用しているサーバーを夜間に別の目的で使い回すこともできます。たとえば、勤務時間が終わってユーザーが退社したら、管理者は Virtual Machine Manager を使って各ユーザーの VM を保存し、他のワークロード (夜間のバッチ処理など) を実行する他の VM をロードできます。翌日の勤務開始時には、各ユーザーのデスクトップを復元できます。このよ

うにデスクトップをホスト方式にすれば、ハードウェアを効率的に使い回すことができるようになるし、分散環境を管理する手間もかからなくなります。

Virtual PC 2007

今日、ハードウェアの仮想化の中で重要になっている要素の中でも、とりわけコスト面で大きな意味を持つ点は、複数の物理サーバーのワークロードを 1 台のマシンに集約できることです。これに加えて、ゲスト オペレーティング システムをデスクトップ マシンで実行できればさらに利便性が上がります。図 7 に示す Virtual PC 2007 は、このような状況を想定して設計されています。

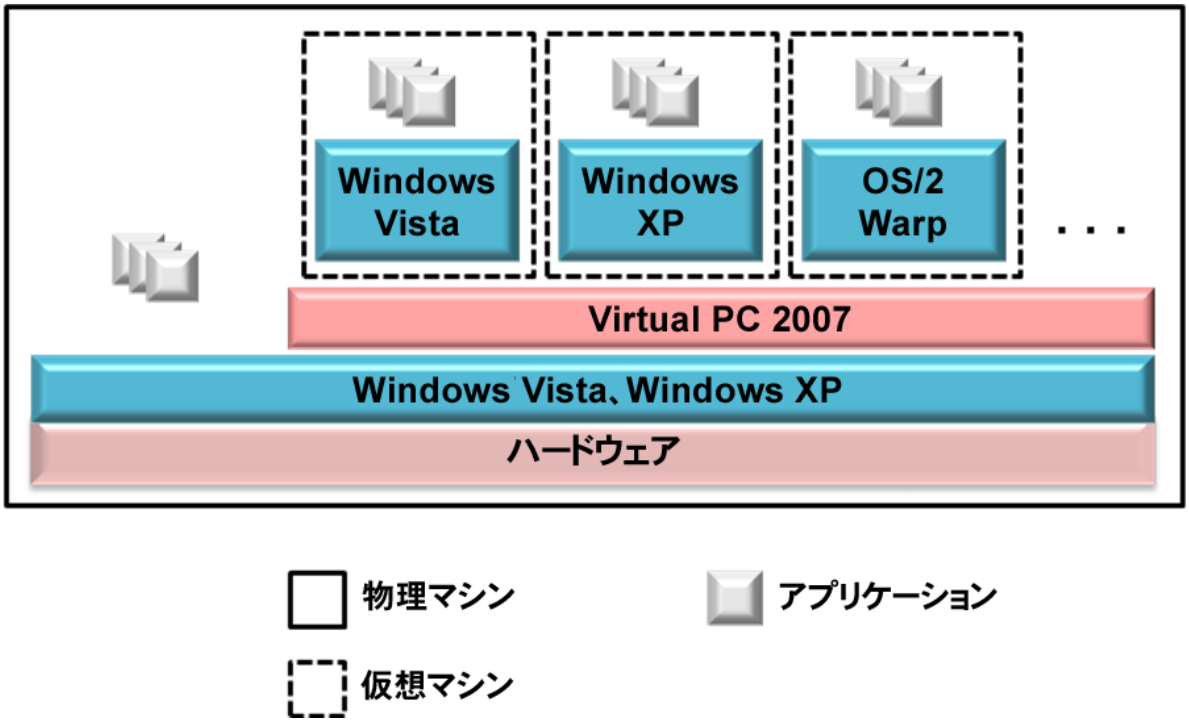

図 7: Virtual PC 2007 の図

Virtual PC は Windows Vista と Windows XP で動作し、x86 ベースのさまざまなゲスト オペレーティング システムを実行できます。サポートされているゲストには、Windows Vista、Windows XP、Windows 2000、Windows 98、OS/2 Warp などがあります。また、Virtual PC は、ストレージに Hyper-V や Virtual Server 2005 R2 と同じ VHD 形式を使用します。

ただし、図 7 からわかるように、Virtual PC は Hyper-V とはアプローチが異なります。Virtual PC ではハイパーバイザーは使用されません。構造的に見ると、この仮想化ソフトウェアは、Virtual Server 2005 R2 と同じように、クライアント マシンのオペレーティング システムの上に乗る形になります。効率面で言えば、このアプローチは、ハイパーバイザーベースの仮想化にはおよそ太刀打ちできませんが、処理速度の面で言えば、多くの (おそらくほとんど) のデスクトップ アプリケーションにとって十分な速度を達成できます。VM 内部で実行されるアプリケーションと並行してネイティブ アプリケーションが実行されることもあるので、そのような場合には、パフォーマンスが落ちるという側面があります。

今後の展望: Microsoft Enterprise Desktop Virtualization (MED-V)

Hyper-V によって実現するサーバー仮想化をさまざまな方法で利用できるのと同じように、Virtual PC によるデスクトップ仮想化もさまざまな用途に利用できます。1 つの例が、2009 年リリースの Microsoft Enterprise Desktop Virtualization です。MED-V を使用すると、Virtual PC がインストールされたクライアントに、事前構成された VM イメージを MED-V サーバーから配信できます。この様子を図 8 に示します。

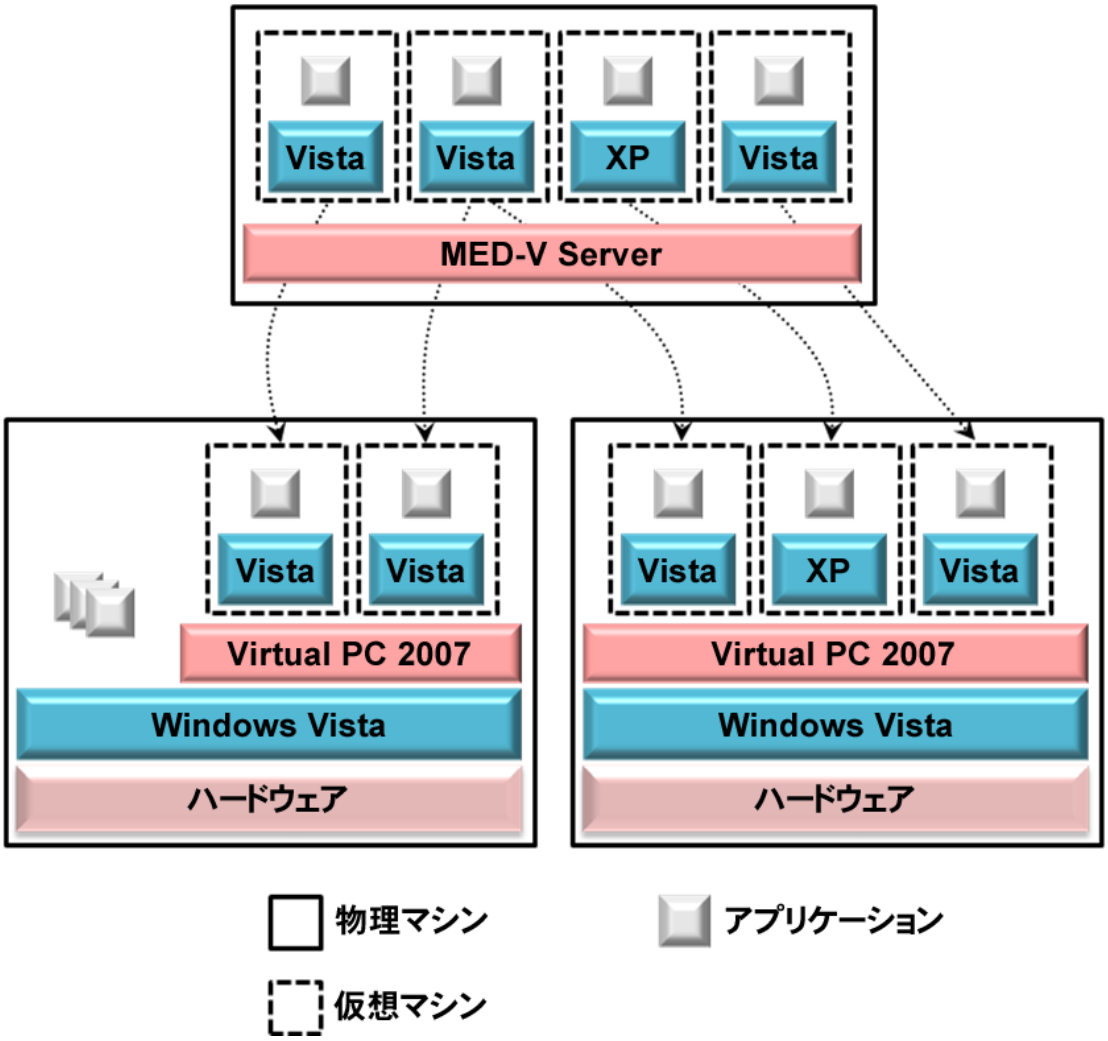

図 8: Microsoft Desktop Virtualization (MED-V) の図

クライアント マシンでは、一部のアプリケーションをネイティブで実行し、その他のアプリケーションを VM 内で実行することも (図の左側)、すべてのアプリケーションを 1 台または複数の VM 内で実行することもできます (図の右側)。どちらの場合も、中央の管理者は、必要な機能をすべて揃えた VM イメージを作成し、クライアントに配信できます。イメージに

は 1 つまたは複数のアプリケーションを含めることができるので、ユーザーのデスクトップのすべてまたは一部をオンデマンドで配信できます。

たとえば、クライアントに Windows Vista をインストールしたものの、Windows XP を必要とするアプリケーションも使用しなければならないとします。管理者は、Windows XP と、該当アプリケーションだけを実行する VM を作成し、MED-V サーバーを使用してその VM を必要なクライアント マシンに配信できます。この方法でパッケージ化されたアプリケーションは、見た目が他のアプリケーションとまったく同じです。ユーザーはアプリケーションを [スタート] メニューから起動して、そのアプリケーションのインターフェイスだけを表示できます。アプリケーションは、実際には専用の仮想マシンの中で安全に実行されます。

プレゼンテーションの仮想化: Windows ターミナル サービス

Windows ターミナル サービスは、登場してから数年が経過していますが、これまで常に仮想化テクノロジとして認識されてきたわけではありません。しかし、仮想化テクノロジの観点から見た方が、実際に起きていることがわかりやすくなるのだとしたら、それだけでも仮想化テクノロジとして考える方が現実的です。現実に、リソースが抽象化され、必要な機能だけがユーザーに渡されているからです。ハードウェアの仮想化ではオペレーティング システムにとって必要なもの、つまり論理的に表現されたハードウェアが提供されるように、プレゼンテーションの仮想化ではユーザーにとって必要なもの、つまりユーザー インターフェイスが提供されます。このセクションでは、ターミナル サービスの最新リリースである Windows Server 2008 ターミナル サービスについて簡単に説明します。

今日のソフトウェアは一般に、画面、キーボード、およびマウスを通してユーザーと対話します。そのために、アプリケーションはローカル ユーザーに対してグラフィカル ユーザー インターフェイスを提供できます。しかし、リモート アプリケーションへのアクセスを、あたかもローカル アクセスのように見せることの方がアプローチとして好ましい状況は数多くあります。アプリケーションのユーザー インターフェイスをリモートで提供すれば、つまりプレゼンテーションの仮想化を利用すれば、このアプローチを効果的に実行できます。図 9 に示すように、Windows Server 2008 ターミナル サービスがこれを可能にします。

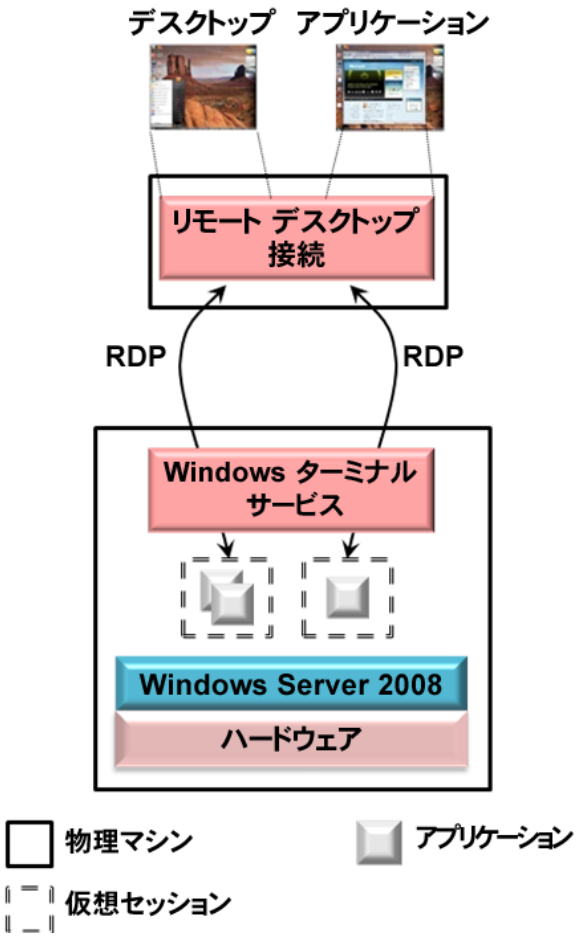

図 9: Windows Server 2008 ターミナル サービスの図

ターミナル サービスは標準の Windows アプリケーションと連動します。変更は一切必要ありません。デスクトップ全体、つまりすべてのアプリケーションのユーザー インターフェイスが揃ったデスクトップをネットワーク経由で表示できます。あるいは、図 9 に示すように、1 つのアプリケーションのインターフェイスだけをユーザーのローカル デスクトップに表示できます。このオプションは、Windows Server 2008 に新しく追加された*ターミナル サービス (TS) RemoteApp* に依存します。TS RemoteApp を使用すると、アプリケーションをローカルで実行しているかのように、アプリケーションのユーザー インターフェイスがデスクトップに表示されます。実際に、TS RemoteApp を通してアクセスしたアプリケーションは、ローカル アプリケーションのようにタスク バーに表示され、[スタート] メニューやショートカットなどから起動することもできます。

完全なデスクトップを表示するオプションも 1 つのアプリケーションだけを表示するオプションも、リモート デスクトップ接続がないと機能しません。クライアント マシンで動作するこのソフトウェアは、前に述べたリモート デスクトップ プロトコルを使用してターミナル サービスと通信し、キーの押下、マウスの動き、および画面データの情報だけを送ります。この必要最低限のアプローチにより、RDP は、ダイヤルアップ回線などの低帯域接続でも動

作します。RDP はトラフィックの暗号化も行うため、アプリケーションへのアクセスがより安全になります。

リモート デスクトップ接続は Windows XP と Windows Vista で動作します。旧バージョンの Windows にもターミナル サービス クライアントがあります。Pocket PC や Apple Macintosh を含め、その他のクライアント オプションも可能です。また、ブラウザー アクセス用に RDP をサポートするクライアントが ActiveX コントロールとして提供されており、アプリケーションへの Web ベースのアクセスが可能です。

Windows ターミナル サービスには、Web 経由でアプリケーションにアクセスするためのその他のサポートもあります。たとえば、完全なリモート デスクトップ接続クライアントがなくても、ターミナル サービス Web アクセス機能を使用すると、1 つのアプリケーション (TS RemoteApp 経由) や完全なデスクトップに Web ブラウザーからアクセスできます。2008 リリースには、RDP トラフィックを HTTPS 内にカプセル化するターミナル サービス ゲートウェイも追加されています。これにより、企業のファイアウォールの外側にいるユーザーは、VPN を使わなくても、ファイアウォールの内側にあるアプリケーションに安全にアクセスできます。

プレゼンテーションの仮想化により、アプリケーションが行う作業のほとんどが、ユーザーのデスクトップから共有サーバーに移ります。ユーザーが期待する応答性を実現するには、特に大規模環境では、膨大な処理リソースが必要になる場合があります。このような応答性の実現を支援するため、ターミナル サービスでは、処理の負荷を複数のマシンに分散するサーバー ファームを作成できます。また、ユーザーの接続先を追跡できるので、接続がユーザーによって解除された場合や、接続が予期せずに失われた場合に、同じシステムへの再接続が可能です。すべての状況に適しているとは言えないまでも、プレゼンテーションの仮想化はかなり多くのシナリオに適切な選択肢となります。

アプリケーションの仮想化: Microsoft Application Virtualization (App-V)

ハードウェアの仮想化とプレゼンテーションの仮想化は、いずれも多くの人にとってなじみのある概念です。アプリケーションの仮想化は比較的新しい概念ですが、決してわかりにくいものではありません。前に説明したように、このテクノロジの最大の目的は、同じマシン上で実行されるアプリケーション間の競合を回避することです。そのために、共有される可能性のあるリソースのコピーをアプリケーションごとに作成し、それを各仮想アプリケーションに組み込みます。Microsoft Application Virtualization (元の名称は *SoftGrid*) によってこの処理が行われる様子を図 10 に示します。

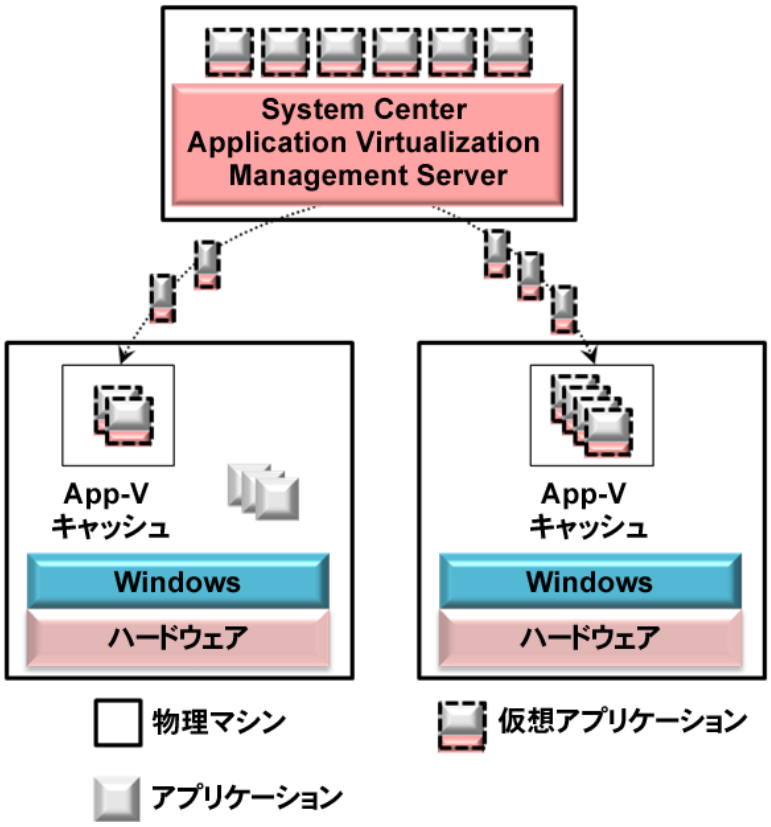

図 10: Microsoft Application Virtualization の図

図 10 からわかるように、仮想アプリケーションは、System Center Application Virtualization Management Server が実行されている中央のマシンに保存できます (後ほど説明しますが、System Center Configuration Manager 2007 R2 でも代用できます。その場合、この特殊なサーバーを使用する必要はありません)。ユーザーが初めて仮想アプリケーションを起動するとき、*ストリーミング*と呼ばれるプロセスによって、このサーバーからユーザーのシステムにアプリケーションのコードが送信されます。その後、仮想アプリケーションの実行が開始します。同じマシンで、他の非仮想アプリケーションも並行して実行できます。この初期ダウンロードが終わると、アプリケーションはマシン上のローカルの App-V キャッシュに保存されます。その後、アプリケーションを実行するときは、このキャッシュされたコードが使用されます。したがって、ストリーミングが必要なのは、アプリケーションに初めてアクセスするときだけです。

ユーザーの観点からすれば、仮想アプリケーションは、見た目が他のアプリケーションとまったく同じです。Windows の [スタート] メニューやデスクトップ アイコンなどから起動できます。仮想アプリケーションは、タスク マネージャーに表示され、プリンター、ネットワーク接続、およびマシン上のその他のリソースを使用できます。アプリケーションは実際にマシン上でローカルに実行されているので、これはもっともなことです。ただし、仮想アプリケーションが使用する、同じシステム上の他のアプリケーションと競合する可能性のあるリソースはすべて、その仮想アプリケーションの一部になっています。たとえば、仮想アプリケーションによってレジストリ エントリの書き込みが行われる場合、その変更は実際に

は仮想アプリケーション内のエントリに対して行われます。マシンのレジストリは変更されません。

このようなしくみが機能するには、仮想アプリケーションがダウンロードされる前に、*シーケンス処理*と呼ばれるプロセスを使用して、そのアプリケーションをパッケージ化する必要があります。管理者は、App-V のウィザードベースの Sequencer ツールを使用して、通常のアプリケーションから仮想アプリケーションを作成します。Sequencer は、アプリケーションのソース コードを変更しませんが、そのアプリケーションがどのように機能するのかを検証し、アプリケーションが使用する共有構成情報を確認します。その後、アプリケーション固有のこの情報と併せて、アプリケーションを App-V 形式にパッケージ化します。

仮想アプリケーションを一元的に保存し、ユーザーのシステムにオンデマンドでダウンロードされるようになるので、管理が簡単になります。しかし、ユーザーが仮想アプリケーションを起動する前に、アプリケーション全体がダウンロードされまで待たなければならないとしたら、アプリケーションに初めてアクセスするときに長い時間がかかる可能性があります。これを回避するため、App-V のストリーミング プロセスでは、アプリケーションの起動に必要なコードだけがダウンロードされます (起動に必要な要素の正確な判別はシーケンス処理プロセスの一部になっています)。アプリケーションの残りの要素は必要に応じてバックグラウンドでダウンロードできます。

ダウンロードされた仮想アプリケーションは App-V が提供するキャッシュに保存されるので、再度ダウンロードしなくても、何回でも実行できます。キャッシュされた仮想アプリケーションをユーザーが起動すると、App-V によって、そのアプリケーションが、中央のサーバーに現在保存されているバージョンと同じかどうか自動的にチェックされます。サーバー上に新しいバージョンがある場合、変更のある部分はすべてユーザーのマシンにストリームされます。パッチやその他のアップデートを、中央のサーバーに保存されているバージョンの仮想アプリケーションに適用すると、キャッシュに入っているアプリケーションのすべてのコピーに自動的に配布されます。

App-V を使用するとき、仮想アプリケーションをオフラインで使用することもできます。たとえば、クライアントがラップトップ マシンであるとします。ユーザーが必要なアプリケーションにアクセスすると、それらが App-V のキャッシュにダウンロードされます。ダウンロードが終わったら、ラップトップをネットワークから接続解除し、通常どおりに使用できます。仮想アプリケーションはマシンのキャッシュから実行されます。

仮想アプリケーションのコピー先のシステムがデスクトップ マシンであっても、ラップトップであっても、仮想アプリケーションにはライセンスが添付されます。どのアプリケーションがどのマシンで使用されているかがサーバーによって追跡されるので、サーバーでライセンスを一元管理できます。各アプリケーションは最終的にタイム アウトします。したがって、ラップトップにアプリケーションをダウンロードしたユーザーは、中央の App-V サーバーに接続して、それらのアプリケーションのライセンスを再取得しなければなりません。

App-V の開発グループにとって、各ユーザーからアクセスできる仮想アプリケーションをどのように決定するかも課題の 1 つでした。この課題に対処するため、ユーザーが所属する Active Directory グループに基づいて、仮想アプリケーションにユーザーを割り当てることにしました。新しいユーザーがグループに追加されると、そのユーザーは所属先ドメイン内の任意のマシンから自分の App-V 仮想アプリケーションにアクセスできます。

デスクトップ コンピューターやラップトップ コンピューターで仮想アプリケーションを使用することの利点は明白です。また、それほど明白ではないにせよ、このテクノロジにはもう 1 つの重要な使い方があります。単一ユーザーのマシンでアプリケーションが競合することがあるように、Windows ターミナル サービスで使用されているアプリケーションも競合することがあります。たとえば、1 台のターミナル サービス サーバー マシン (通常は単に*ターミナル サーバー*) に、同じ DLL の異なるバージョンを必要とする 2 つのアプリケーションをインストールするとします。この競合は、ユーザーのデスクトップで発生する場合よりも深刻です。なぜなら、このサーバーに依存するすべてのターミナル サービス クライアントに影響が及ぶからです。両方のアプリケーションを使用可能にする必要がある場合、これまでの解決法としては、アプリケーションを別々のターミナル サーバーに展開するのが一般的でした。この解決法はうまくいくものの、サーバーの利用率が下がる傾向があります。

この場合、アプリケーションの仮想化が役立ちます。アプリケーションをターミナル サーバーにロードする前に仮想化すれば、別々のサーバーを使用しなければならなくなるような一般的な競合は回避できます。別々のサーバー サイロを作って、サーバーの利用率を下げるのではなく、任意のターミナル サーバーで仮想アプリケーションを実行できます。そうすれば、使用するサーバー マシンの台数は少なくて済むので、ハードウェアと管理のコストを削減できます。

実際、App-V の仮想アプリケーションの管理は、一般的なインストール済みソフトウェアの管理よりも、Web ページの管理に似ています。仮想アプリケーションは Web ページのようにオンデマンドでサーバーからダウンロードできます。また、Web ページにアクセスする前には競合の可能性をテストする必要がないのと同じように、仮想アプリケーションも展開する前にテストする必要がありません。繰り返しになりますが、根底にある概念は抽象化です。つまり、アプリケーションの構成情報を仮想的に見えるようにすることです。他のタイプの仮想化と同様、コンピューティング環境のさまざまな要素間の分離度を高めることによって、その利点が生まれます。

仮想化された **Windows** 環境の管理

多くの IT 部門にとって最大のコストは人件費です。仮想化によってハードウェアのコストを削減しても、管理作業が増加したとすれば、収支バランス的にはマイナスになる可能性が高くなります。人件費は機器費よりかさむからです。この事実を踏まえると、仮想化テクノロジを効果的に管理することは不可欠です。このセクションでは、Microsoft System Center のツールである Operations Manager、Configuration Manager、および Virtual Machine Manager を使用して、仮想化された Windows 環境を管理する方法について説明します。

System Center Operations Manager 2007

ごく小規模な企業を除くすべての企業にとって、分散環境でシステムを監視および管理するためのツールが必要不可欠です。マイクロソフトの Operations Manager は、Windows 指向の環境でこの課題に対処します。デスクトップ、サーバー、およびその他のデバイス上でのハードウェア/ソフトウェア管理を専門に扱い、システム管理の幅広いアプローチをサポートします。

コンピューティング環境は、クライアント/サーバー マシン、オペレーティング システム、データベース、メール サーバーなど、多数の異なるコンポーネントで構成されます。この多様性に対処するため、Operations Manager では*管理パック (MP)* が使用されます。各 MP には特定のコンポーネントの管理方法に関する知識などがカプセル化されており、それぞれがその分野に関する豊富な経験を持つ人々によって作成されています。たとえば、マイクロソフトは、Windows、SQL Server、Exchange Server など、ほとんどすべてのエンタープライズ製品の管理用 MP を提供しています。HP と Dell は各社のサーバー マシンの管理用 MP を、その他のベンダーも各社製品用の MP を提供しています。適切な MP をインストールすれば、製品の開発グループの知識を活用して、製品をより効果的に管理できます。図 11 に示すように、仮想化を利用した環境の管理にも応用できます。

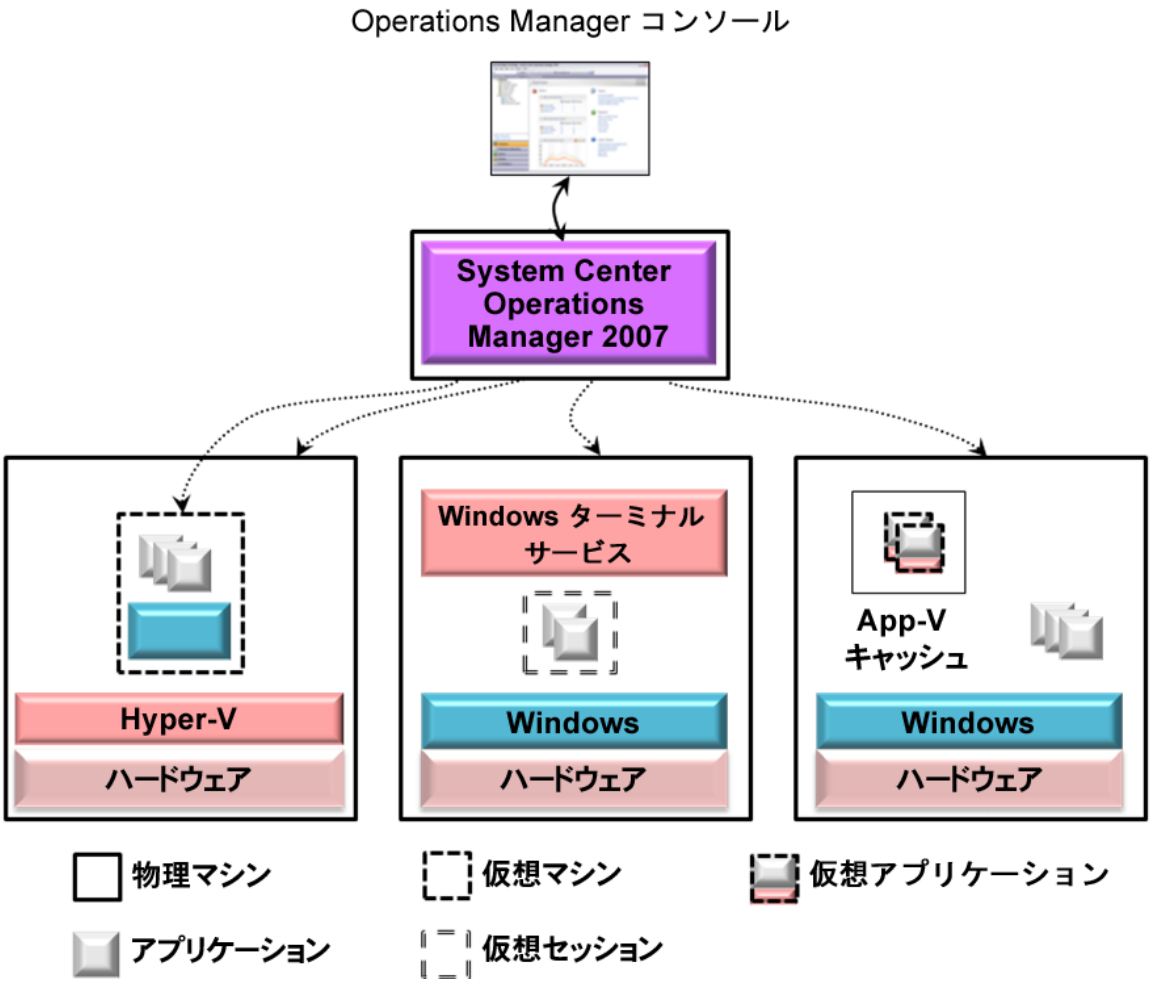

図 11: 仮想環境内の Operations Manager

左側のシステムを見ればわかるように、Operations Manager は物理マシンだけでなく、仮想マシンも管理できます。実際に、この製品はどちらのケースでも同じように機能します。Operations Manager では、管理対象の各マシンで実行されるエージェントが使用されます。

したがって、すべてのマシンは、物理的か仮想的かにかかわらず、独自のエージェントを持ちます。たとえば上の図で、左側のシステムには 2 つのエージェントがあることになっています。1 つは物理マシン用、もう 1 つは Hyper-V 上の VM 用です。Operations Manager コンソールを操作するオペレーターから見ると、どちらも通常の Windows マシンのように表示され、どちらも同じ方法で管理します。Operations Manager を使用すると、物理環境と仮想環境の管理に別々のツールを導入しなくても、同じユーザー インターフェイス、同じ MP が両方の環境に適用されます。

物理マシンと仮想マシンの管理は同じ MP を使用して行いますが、仮想化テクノロジを管理するための特別な MP もあります。たとえば、Hyper-V の MP を使用すると、特定の物理マシンで実行されている VM をエミュレートしたり、それらの VM の状態を監視したりできます。Windows ターミナル サービスの MP を使用すると、このプレゼンテーション仮想化テクノロジのパフォーマンスと可用性を追跡できます。App-V の MP も同じような管理作業をサポートしています。Operations Manager では、物理環境と仮想環境に同じテクノロジが適用されているので、この 2 つの環境を一貫したアプローチで管理できます。

System Center Configuration Manager 2007 R2

適切なマシンに適切なソフトウェアを導入し、そのソフトウェアを常に最新の状態に維持していくとなると、大変な労力を要する作業になりかねません。それに加え、ソフトウェア資産の最新の記録を維持する作業もあります。自動化ツールが必要になるのも当然の成り行きです。これらの課題に対処するため、マイクロソフトは System Center ファミリの別のメンバーとして Configuration Manager を用意しています。

ソフトウェア構成の管理は物理環境においても困難ですが、仮想化を導入するとさらに困難になる可能性があります。たとえば、仮想マシンを増やせば、ソフトウェアのアップデートが必要になるマシンの数が増えることになります。このような環境では、効果的な構成管理が不可欠です。

Operations Manager と同様、Configuration Manager も物理環境と仮想環境が同じ方法で処理されます。2 つの別個の環境でソフトウェア構成を管理するツールを別々に用意しなくても、Configuration Manager によって同じテクノロジが両環境に適用されます。この様子を図 12 に示します。

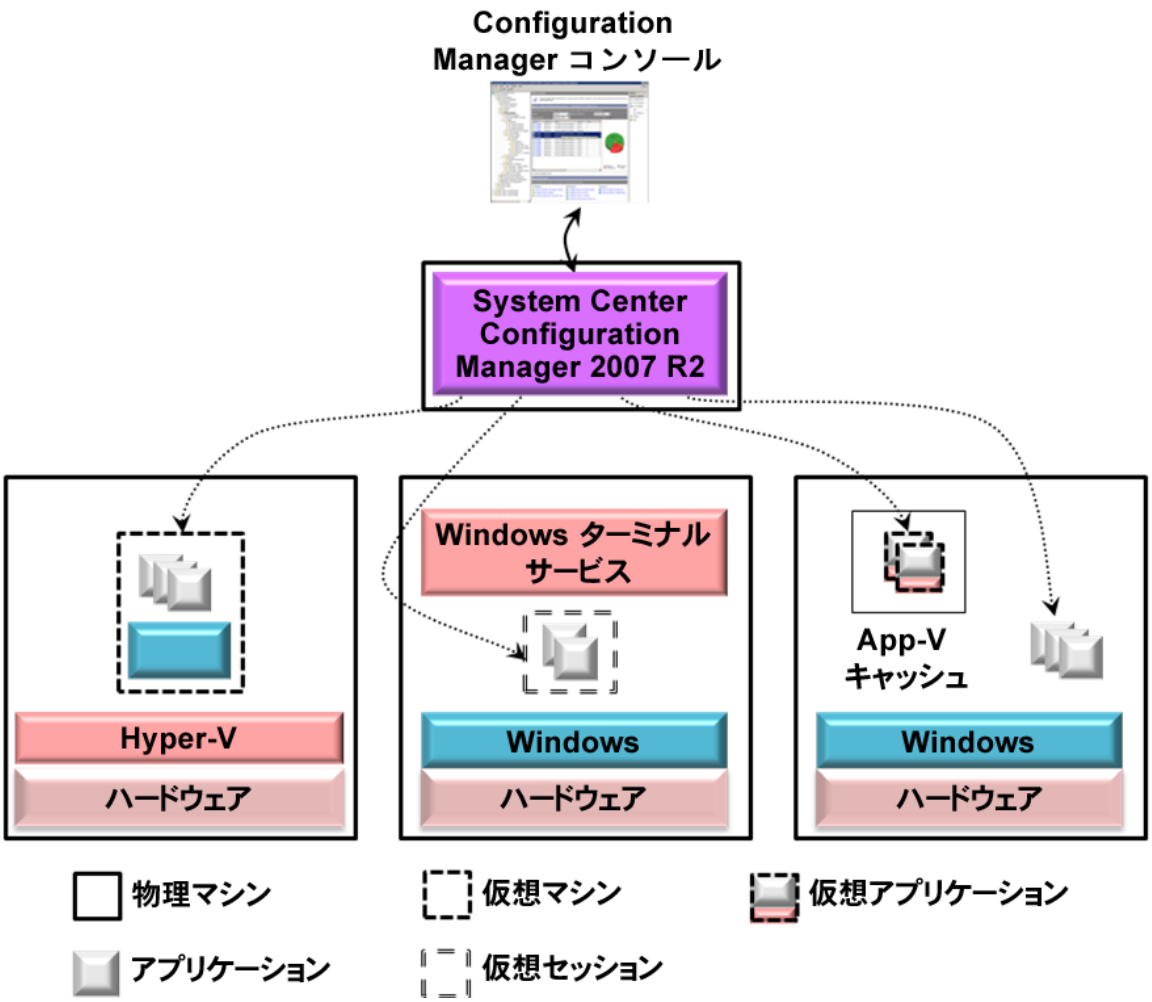

図 12: 仮想環境内の Configuration Manager 2007 R2

この図の左端のシステムからわかるように、Configuration Manager は、Hyper-V 上の VM を物理マシンと同様に扱っています。このマシンにソフトウェアをインストールし、必要に応じてアップデートできます。また、ソフトウェアは、Configuration Manager によって維持される資産インベントリの一部として表示されます。同様に、図 12 の中央のシステムからわかるように、このツールは、その他のアプリケーションのときと同じように、ターミナル サーバー上で実行されているアプリケーションに対しても機能します。

また、図の右端のシステムでは、Configuration Manager が App-V に対して機能しています。前に説明したように、App-V を使用している企業には選択肢があります。App-V の一部である System Center Application Virtualization Management Server を使用して仮想アプリケーションを配布する方法もあれば、System Center Configuration Manager 2007 R2 を使用する方法もあります。Configuration Manager を使用する場合、仮想アプリケーションを実行用のシステムにオンデマンドでストリームすることはできませんが、同一のサーバーから仮想アプリケーションと非仮想アプリケーションの両方を展開することはできます。

ソフトウェア構成の管理はすべての企業にとって重要です。IT 業界全体に仮想化の波が押し寄せ続ける中で、仮想化されたソフトウェアの管理がますます重要になってきています。Configuration Manager の目的は、物理環境と仮想環境に共通の問題の解決法を提供することです。

System Center Virtual Machine Manager 2008

仮想環境を管理するための要件の多くは、純粋な物理環境での要件と同じです。Operations Manager と Configuration Manager はこの事実を踏まえて、両環境を大差なく扱います。しかし、仮想化することで、仮想化に固有の管理上の課題も出てきます。最も重要な課題は、ハードウェアの仮想化と、仮想化によって実現する過剰なまでの数の仮想マシンが原因となって発生します。作成し、使用する仮想マシンの数が増えるにつれて、仮想マシンの管理に特化したツールの必要性が高まります。

たとえば、Hyper-V には VM を管理するためのツールがありますが、このツールは 1 台の物理マシンでのみ動作します。数台以上の VM が、複数の物理マシンに分散されている場合は、それらの VM を一元管理するためのアプローチが最適です。Virtual Machine Manager が、このニーズに対するマイクロソフトの回答です。名前から推察できるように、このツールは VM を管理することを唯一の目的としています。最新リリースである Virtual Machine Manager 2008 の使用例を、図 13 に簡単に示します。

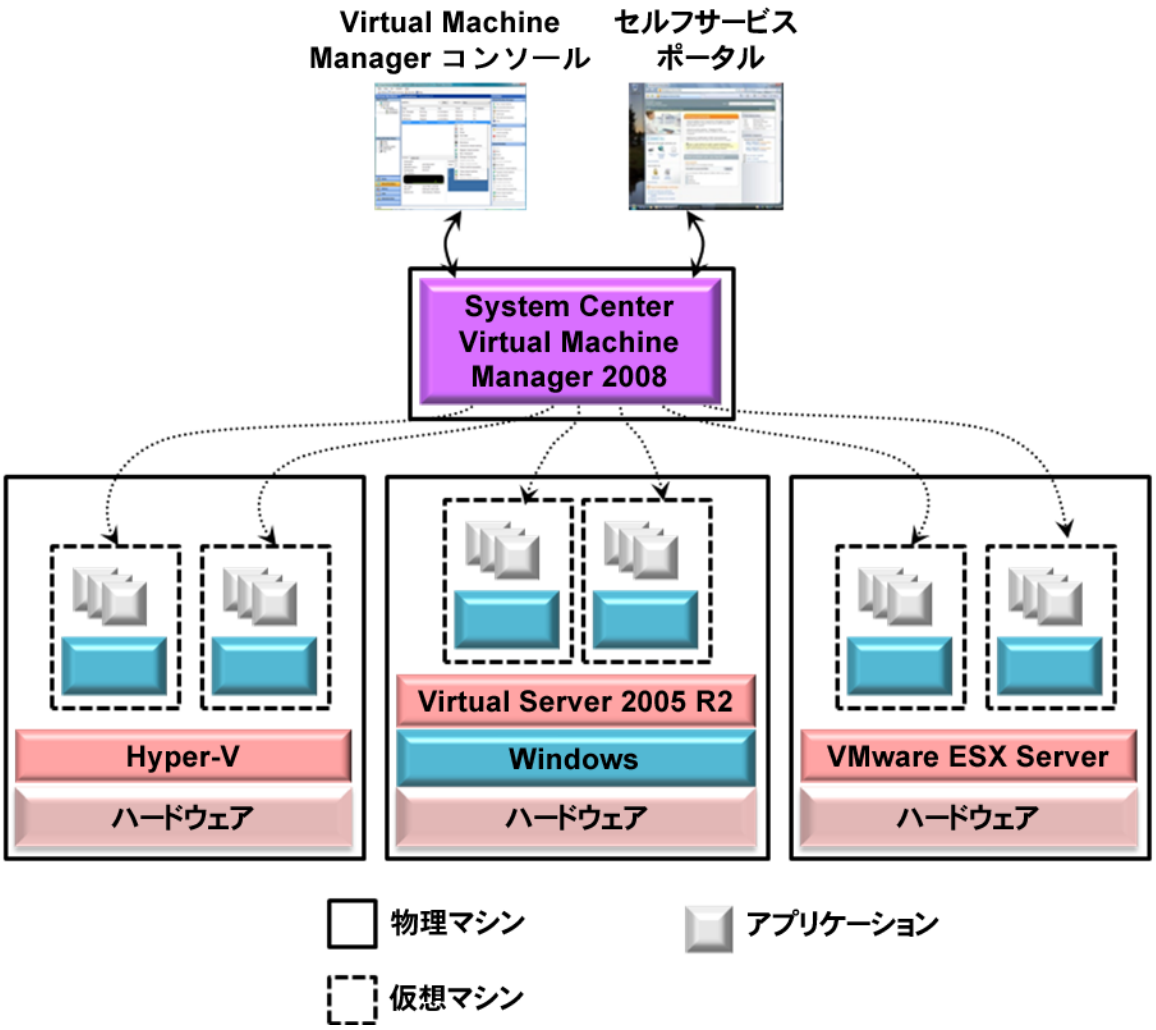

図 13: Virtual Machine Manager 2008 の図

図からわかるように、Virtual Machine Manager は中央のコンソールとして機能し、多数の VM を一元管理できます。管理者は、このコンソールを使用して、VM のステータスのチェック、VM 内で実行されているアプリケーションの正確な把握、物理マシン間での VM の移動などといった管理タスクの実行が可能です。このコンソールはグラフィカルなインターフェイスを備えていますが、このインターフェイスはマイクロソフトの PowerShell スクリプティング ツールに完全に基づいて構築されています。グラフィカル インターフェイスを使って実行できる操作は、コマンド ラインからこの言語を使って実行することもできます。

さらに図 13 からわかることは、Virtual Machine Manager の 2008 リリースを使うと、Hyper-V、Microsoft Virtual Server 2005 R2 SP1、および VMware ESX Server の 3 つの異なるテクノロジを使用して作成された VM を管理できるということです。利用できる管理機能は 3 つのテクノロジですべて同じです。たとえば、Virtual Machine Manager の新しいバーチャル マシン ウィザードを使用すると、3 つの中から任意のテクノロジを使用して VM を作成できます。このツールは、新しい VM を定義するために次のオプションを揃えています。

- VM を新規作成します。CPU の種類やメモリ サイズなどを指定します。
- 物理マシン環境を新しい VM に変換します。*P2V* と呼ばれるプロセスです。
- 既存の VM をもとに新しい VM を作成します。
- VMware を使用して作成された既存の VM をマイクロソフトの VHD 形式に変換します。
- *テンプレート*を使用します。各テンプレートは、仮想マシンです。その中には、すぐにでも導入が可能な Windows が入っていますが、必要に応じて、管理者がカスタマイズすることもできます。

どのオプションを選択した場合でも、新しい VM のホスト先に適した物理マシンがどれか判断しやすくなるように、ウィザードによってパフォーマンス データが検査されます。このプロセスを*インテリジェント配置*といいます。空き容量やその他の基準に基づき、サーバー候補が 1 位から 5 位まで挙げられます。管理者は、サーバーを選択したら、今度はツールを使って新しい仮想マシンをそのシステムにインストールできます。

管理者の作業を軽減するため、Virtual Machine Manager では、テンプレート、VHD、およびその他の情報がライブラリ化されています。管理者は、このライブラリの内容を利用した新しい VM の作成に加えて、既存の VM をオフラインにし、それをライブラリに保存して、後で復元することもできます。Virtual Machine Manager のセルフサービス ポータルを使用すると、このライブラリのテンプレートからユーザーが自分で VM を作成することもできます。VM の作成に管理者が制限をかけられるように、ユーザーごとにポリシーを定義できます。1 人のユーザーが作成できる VM の数を制限するクォータなどの設定を指定できます。

仮想環境の管理における別の課題として、VM の管理に使用するツールと、システムとアプリケーションの監視に使用するツールとの接続があります。たとえば、複数の VM をホストする物理マシンでディスク空き容量が少なくなったとします。監視ツールに警告を送ることはできますが、問題を解決するには一部の VM を別の物理マシンに移動しなければならない場合があります。VM の移動は監視ツールではできませんが、VM の管理ツールならできます。この問題の解決には、この 2 つのツールを連動させることが必要です。

これに対処するため、Virtual Machine Manager 2008 には、Operations Manager との連動を可能にする Performance and Resource Optimization (PRO) という機能が組み込まれています。たとえば、ディスク空き容量が少なくなったマシンから Operations Manager に警告が送られると、その情報は Operations Manager から Virtual Machine Manager に渡されます。その後には、Virtual Machine Manager を使用して、そのマシンから別のマシンに VM を移動できます。同様に、Operations Manager と Virtual Machine Manager が連動して、マシン上で実行されている VM や、各 VM で実行されているアプリケーションなどを把握できます。VM の実装がマイクロソフトのテクノロジによって行われているのか、VMware ESX によって行われているのかといったことには一切関係なく、このすべてが機能します。マイクロソフトの目標は、顧客がどのオプションを使用していても、同社の管理ツールを使うメリットがあるようにすることです。

ハードウェアの仮想化が、特にサーバー上で急速に一般的になってきています。VM を 1 台ずつ別々のツールで管理するような方法は、単純なケースでは問題がないでしょうが、今日

行われているような広範囲にわたる仮想化には十分ではありません。一元化されたコンソール、リソース源としてのライブラリ、およびその他のツールを備えた Virtual Machine Manager は、企業全体の Windows VM を一元的に管理することを目指しています。

仮想化テクノロジの組み合わせ

個々の仮想化テクノロジを他と切り離したほうが、テクノロジの見通しがよくなるという側面はあります。しかし、テクノロジ同士を組み合わせて使用することも実際には必要です。ハードウェアの仮想化、プレゼンテーションの仮想化、そしてアプリケーションの仮想化を組み合わせたサンプル シナリオを図 14 に示します。

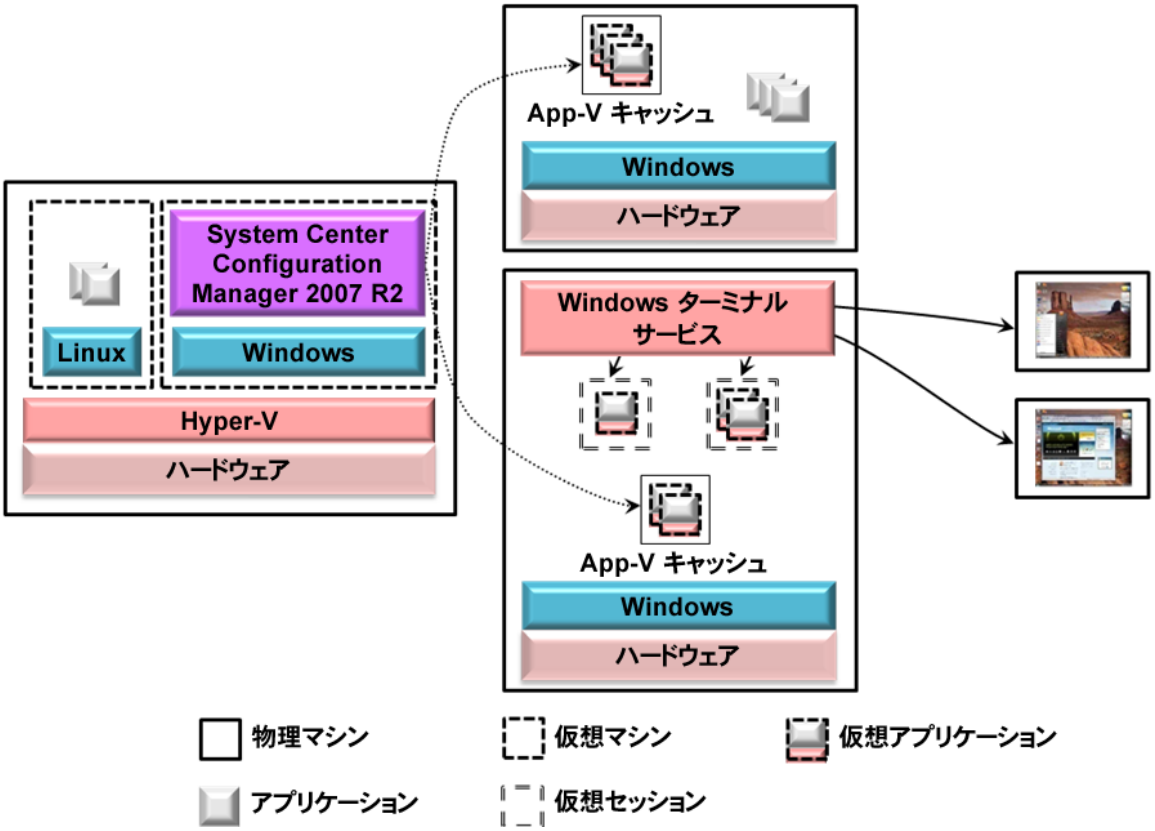

図 14: さまざまな仮想化テクノロジを組み合わせた例

この例の場合、左側のシステムでは Hyper-V を使用してハードウェアの仮想化を行っています。1 台の VM では Linux 上でワークロードを実行し、もう 1 台の VM では Windows 上で System Center Configuration Manager 2007 R2 を実行しています。同じ組織の別のシステムが、このサーバーから App-V 仮想アプリケーションをストリームできるようにしています。図の一番上にあるマシンは、デスクトップ、ラップトップ、サーバー マシンのどれであってもよく、オンデマンドで App-V 仮想アプリケーションを数種類ほどストリームしています。一番下のシステムは、ターミナル サービスを使用してプレゼンテーションの仮想化を行って

います。実行されているすべてのアプリケーションが仮想アプリケーションとしてパッケージ化されています。さまざまなタイプの仮想化の普及に伴い、ここに示すような複合的にテクノロジを組み合わせる事例がますます一般的になってきています。

この記事で取り上げていない重要な問題の 1 つに、仮想化テクノロジがライセンスに及ぼす影響があります。従来のライセンス体系は、ハードウェアと組み合わせた形態のものがほとんどですが、このような考え方は、仮想環境にはそぐいません。別の考え方が必要になるため、仮想化テクノロジのライセンス要件を理解しておく必要があります。たとえば、VDI には Vista Enterprise Centralized Desktop 製品ライセンスが必要です。状況に応じて、それぞれに独自のライセンス要件があります。

結論

仮想化の魅力は絶大です。経済的な効果の高さは、大きな魅力として映ります。実際のところ、導入の妨げになるような事情がある企業もほとんどないでしょう。仮想化テクノロジの管理がうまくいけば、環境の改善に大きく寄与することでしょう。

マイクロソフトはこの分野を大局的な視点で見つめ、ハードウェアの仮想化、プレゼンテーションの仮想化、アプリケーションの仮想化といったテクノロジを提供しています。また、管理についても幅広い視野から考え、物理的なテクノロジと同等に仮想化テクノロジを重視しています。仮想化の人気が拡大を続ける中、これらのテクノロジは近代のコンピューティングの基盤になろうとしています。

著者について

David Chappell 氏は、カリフォルニア州サンフランシスコに拠点を置く Chappell & Associates (www.davidchappell.com) 社の社長です。ソフトウェア ユーザーが新しいテクノロジを理解して使用し、的確な判断を下せるように、世界を股にかけて講演、著作、コンサルティングなどの活動を行なっています。