



ホワイトペーパー
インテル® Core™ vPro™
プロセッサ
新しいコンピューティング
モデル

デスクトップ仮想化の理解

コスト管理、運用管理、セキュリティ、各種規制の遵守、ビジネスの継続性など、ますます深刻化する多くの問題に直面している企業の IT 部門では、従来の分散型ソフトウェア環境の代替手段として、デスクトップ仮想化モデルの導入を検討しています。

近年のテクノロジーの進化により、IT 部門は、さまざまな集中管理型モデルの中から最適なソリューションを選択できるようになりました。ターミナルサービス、サーバーホスト型デスクトップ仮想化、ブレード PC、アプリケーション / OS ストリーミングなど、各モデルにはそれぞれに利点と制限があります。

1 つのモデルですべてのユーザーとアプリケーションのニーズに十分に対応できることはまれであり、各ユーザーグループのニーズ、IT 部門の要件、既存のインフラストラクチャーについて慎重に検討する必要があります。ほとんどの場合、大企業のソリューションは複数のモデルを組み合わせたものになります。

純粋なサーバーベースのモデルでは、シンターミナルのハードウェアで十分という意見もあります。それに対して、デスクトップ PC およびノートブック PC 向けインテル® Core™ vPro™ プロセッサ・ファミリーを搭載したインテリジェントな PC プラットフォームは、広範囲にわたるソリューションを運用できる高度な運用管理機能とセキュリティを提供します。

このホワイトペーパーでは、利用可能な各種のデスクトップ仮想化モデルとその仕組み、またセキュリティや運用管理機能、消費電力などの主な評価基準から見た各モデルの利点と制限について説明します。また、各モデルの導入に必要なソフトウェア・ソリューションの主な提供方法と端末機器も掲載しています。



目次

デスクトップ仮想化モデルの評価	3	リモート OS ブート	13
表 1. 考慮すべき要因	3	概要	13
表 2. デスクトップ仮想化モデルの比較	4	仕組み	13
ターミナルサービス	5	図 6. リモート OS ブートのアーキテクチャー	13
概要	5	表 11. リモート OS ブートの利点	14
仕組み	5	表 12. リモート OS ブートの制限	14
図 1. ターミナルサービスのアーキテクチャー	5	アプリケーション・ストリーミング / アプリケーション仮想化	15
表 3. ターミナルサービスの利点	6	概要	15
表 4. ターミナルサービスの制限	6	仕組み	15
サーバーホスト型デスクトップ仮想化	7	図 7. アプリケーション・ストリーミング / アプリケーション仮想化のアーキテクチャー	15
概要	7	表 13. アプリケーション・ストリーミング / アプリケーション仮想化の利点	16
仕組み	7	表 14. アプリケーション・ストリーミング / アプリケーション仮想化の制限	16
図 2. サーバーホスト型デスクトップ仮想化のアーキテクチャー	7	仮想コンテナ	17
表 5. サーバーホスト型デスクトップ仮想化の利点	8	概要	17
表 6. サーバーホスト型デスクトップ仮想化の制限	8	仕組み	17
ブレード PC	9	図 8. 仮想コンテナのアーキテクチャー	17
概要	9	表 15. 仮想コンテナの利点	18
仕組み	9	表 16. 仮想コンテナの制限	18
図 3. 1 対 1 ブレードのアーキテクチャー	9	まとめ	19
図 4. 1 対多ブレードのアーキテクチャー	9	関連情報	19
表 7. ブレード PC の利点	10		
表 8. ブレード PC の制限	10		
OS イメージ・ストリーミング	11		
概要	11		
仕組み	11		
図 5. OS イメージ・ストリーミングのアーキテクチャー	11		
表 9. OS イメージ・ストリーミングの利点	12		
表 10. OS イメージ・ストリーミングの制限	12		

デスクトップ仮想化モデルの評価

適切なデスクトップ仮想化モデルを選択するには、相互に関連する複数の要因のバランスをとる必要があります。1つの要因だけを必要以上に重視すると、最適なソリューションは得られません。例えば、IT部門の要件だけを重視すると、IT部門のシステム管理は容易になりますが、ユーザーの操作性は低下する可能性があります。

どのアプリケーションとデータを集中管理し、どれをローカルにインストールするべきかを判断するには、ユーザー全員の用途とビジネスニーズを考慮する必要があります。

まず、遂行する業務と必要なアプリケーションに基づいてユーザーを区分けすることをお勧めします。ユーザーによってアクセスするアプリケーションは異なり、複数のモデルを組み合わせる必要もあります。1つのモデルで特定のユーザーグループのすべてのニーズに対応できることはまれであり、社内のすべてのユーザーのニーズを満たすことはさらに困難です。

どのモデルを使用するかを選択は、多くの場合、そのモデルの導入対象であるクライアント・デバイスとも深く関連しています。モデルの選択とクライアントの選択は、別々に検討することが重要です。

例えば、特定のアプリケーションにサーバーベース・コンピューティングを使用するビジネスシナリオの場合を考えます。このサーバーベース・モデルは「シンクライアント」と呼ばれますが、必ずしもシターミナル上に導入する必要はありません。ユーザーのアプリケーションおよびモビリティに対するニーズによっては、デスクトップ PC またはノートブック PC を使用する方が現実的という場合もあります。また、ユーザーがローカルで実行されるアプリケーションを必要とする場合もあります。そうしたクライアント・デバイスには、対象ユーザーの最も高度な要求に応えられる性能が求められます。

コストの比較については、複数のモデルについてソリューション全体の総コストを比較すべきです。クライアントとデータセンターの間でコストが移動しただけで、実際にはサービスデリバリーの総コストは減っていない場合もあります。また、同等の処理能力およびストレージ容量に必要なインフラストラクチャー・コストは、それらが端末上で実行されるか、データセンターから提供されるかによって大きく異なります。

表 1. 考慮すべき要件

<p>IT 要件 セキュリティ、運用管理機能、ビジネスの継続性など、標準的な IT 問題。</p>	<ul style="list-style-type: none"> ▪ セキュリティ ▪ イメージ管理 ▪ ライセンス管理 ▪ サポート構造 ▪ 災害復旧 ▪ 投資期間
<p>インフラストラクチャー 利用可能なハードウェア、ネットワーク接続、帯域幅。特定のモデルをサポートするインフラストラクチャーが用意されていない場合は、インフラストラクチャーを購入して設置するか、または他のモデルを選択する必要があります。</p>	<ul style="list-style-type: none"> ▪ サーバー ▪ ストレージ ▪ データセンターのスペース、電源、空調設備 ▪ ネットワーク帯域幅 ▪ 予算の優先順位
<p>ユーザー体験 モビリティとパフォーマンスに対するニーズなど、そのシステムのユーザーのワークフロー要件。多くの場合、習慣またはポリシーにより、ユーザーはデバイスを個人的な作業にも使用できることを期待する。これにより、個人識別情報がシステム上に置かれることになる。このような「所有権」と個人情報保護の問題を考慮する必要があります。</p>	<ul style="list-style-type: none"> ▪ モビリティ ▪ 応答性 ▪ カスタマイズ ▪ ネットワーク接続 ▪ 「所有権」と個人情報保護
<p>アプリケーションのワークロード ユーザーが実行するアプリケーションの演算処理需要とグラフィックス処理需要。Voice over Internet Protocol (VoIP) やストリーミング・ビデオなどのアプリケーションでは、ネットワークの遅延は許されない。将来のアプリケーションを見据えた拡張性も考慮する必要があります。</p>	<ul style="list-style-type: none"> ▪ 演算処理負荷 ▪ グラフィックス処理負荷 ▪ 遅延が許されないアプリケーション (ビデオ、モーション・グラフィックス、VoIP) ▪ 拡張性 ▪ Web サーバーの負荷

表 2. デスクトップ仮想化モデルの比較

	ターミナル サービス	サーバーホスト型 デスクトップ 仮想化	ブレードPC	OS イメージ・ ストリーミング	リモート OS ブート	アプリケーション・ ストリーミング/ アプリケーション 仮想化	仮想 コンテナ
アプリケーションの 実行	サーバー	サーバー	サーバー	クライアント	クライアント	クライアント	クライアント
アプリケーション・ データ・ストレージ	サーバー	サーバー	サーバー	サーバー	サーバー	クライアント またはサーバー	クライアント またはサーバー
ローカルデバイスの 接続と同期 (バーコードリーダー、 PDA、携帯電話など)	一部対応 (ベンダー固有) ^a	一部対応 (ベンダー固有) ^a	一部対応 (ベンダー固有) ^a	対応	対応	対応	制限あり
Windows* アプリ ケーションの完全サ ポート (VoIP とリッチ メディアを含む)	一部対応 (ベンダー固有) ^a	一部対応 (ベンダー固有) ^a	一部対応 (ベンダー固有) ^a	対応	対応	対応	対応
Microsoft* Windows* XP、 Windows Vista*、 Windows* 7 の完全 サポート	一部対応 (ベンダー固有) ^a	一部対応 (ベンダー固有) ^a	一部対応 (ベンダー固有) ^a	対応	対応	対応	対応
ネットワーク外の モバイルオプション	非対応	非対応	非対応	非対応	非対応	対応	対応
標準的な クライアント	端末、 デスクトップ PC、 ノートブック PC	端末、 デスクトップ PC、 ノートブック PC	端末、 デスクトップ PC	デスクトップ PC	デスクトップ PC	デスクトップ PC、 ノートブック PC	デスクトップ PC、 ノートブック PC
主なソリューション・ プロバイダー	Citrix、 Microsoft	Citrix、 Microsoft、Red Hat、VMware	ClearCube、HP、Dell、 Devon IT	Citrix、Dell、 Lenovo	Lenovo	Citrix、 Microsoft、 Symantec、 VMware	Citrix、 Microsoft、 VMware
主なソリューション	Citrix* XenApp*、 XenDesktop* (HDX**)	Citrix* XenDesktop* (HDX**)	ClearCube (4 ブレード・ワーク ステーション・モデル)	Citrix* XenDesktop* with Provisioning Server for Desktops	Lenovo* Secure Managed Client (SMC)	Citrix* XenApp* (XenDesktop*)	XenClient* Type 1
	Microsoft* Remote Desktop Services	Microsoft* VDI (RemoteFX**)	HP* Consolidated Client Infrastructure	Dell* On- Demand Desktop Streaming (ODDS)		Microsoft* System Center Config Manager + App-V	Microsoft* MED-V および VPC7
		VMware View* (PCoIP**)	Dell* Dedicated Remote Workstation			Symantec* Workspace Virtualization	VMware View*
			DevonIT HC12 Remote Workstation			VMware* ThinApp	

a 一部のベンダーは拡張された独自規格に基づきリモート接続・プロトコルを提供しています。これらのプロトコルは、さまざまなレベルのメディア、グラフィックス、周辺機器のサポートを提供します。カスタマイズされたハードウェアが必要な場合もあります。

サーバーベース・モデル ターミナルサービス

概要

ターミナルサービスは、メインフレーム・コンピューティング時代からの長期的な実績のある、信頼性の高いサーバーベース・モデルです。このモデルは、サポートしているソフトウェア・アプリケーションに対して、強力なセキュリティと運用管理機能を提供します。しかし、PCのユーザー体験に慣れたユーザーは、このモデルのパフォーマンス、カスタマイズへの対応、柔軟性、モビリティに不満を感じる場合があります。

多くの大企業では、一部のアプリケーションでターミナルサービスを使用しています。特に、セキュリティが不可欠で、ユーザーが決まった場所から常時ネットワークにアクセスしているアプリケーションではよく利用されます。ターミナルサービスに最適なソリューションの例として、銀行窓口係のトランザクション・システム、コールセンター・スタッフの注文処理システム、医療従事者の電子カルテなどが挙げられます。

一部の主要なアプリケーションにのみターミナルサービスを使用し、そ

れ以外のアプリケーションはローカルにインストールする場合は、PCが最適なプラットフォームです。シンクライアントは、100%ターミナルサービスで構成される環境にのみ適しています。この場合でも、既存の大量のPCを新しい端末に置き換えるには、多額の費用がかかります。これらのコストを詳しく分析し、真の投資収益率(ROI)を計算する必要があります。多くのIT部門では、古くなった既存のPCをターミナル・サービス・クライアントとして転用しています。

仕組み

クライアントは単なるディスプレイ、および入力デバイスとして使用されます。すべての処理はサーバー上で集中的に実行され、すべてのデータはデータセンター内に格納されます。クライアント上では処理は実行されず、データも保存されません。通常は、リモート・ディスプレイ・プロトコル(RDP)またはIndependent Computing Architecture(ICA*)を使用して、サーバーベース・アプリケーションのイメージをクライアント上のターミナル・ビューアにプッシュします。

図 1. ターミナルサービスのアーキテクチャー

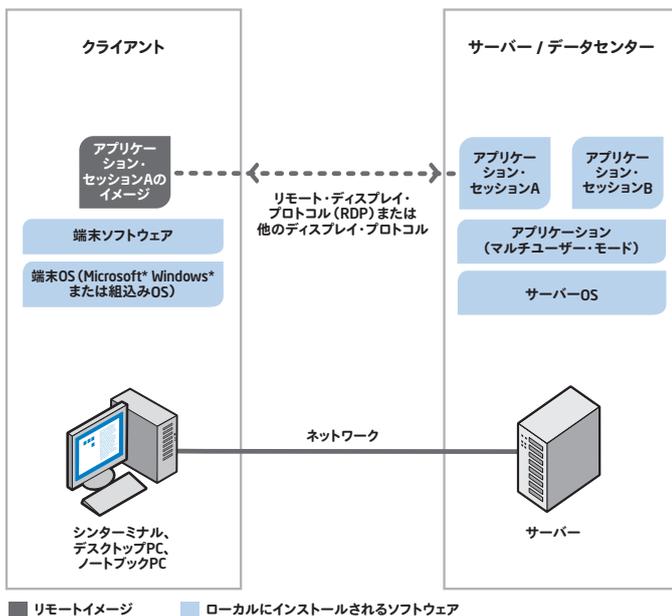


表 3. ターミナルサービスの利点

セキュリティの強化	<ul style="list-style-type: none"> このモデルでは、OS、アプリケーション、データがデータセンター内でロックされるため、クライアント・ベースのモデルに比べてクライアント上でのセキュリティ違反やデータ損失のリスクが軽減される。
運用管理の効率化	<ul style="list-style-type: none"> ターミナルサービスはよく理解された成熟した技術である。 アプリケーションとデータを集中的に管理でき、運用管理の効率化とバックアップの信頼性向上が可能。 ソフトウェア・イメージの管理、検証、サポートが簡略化される。ドライバーとダイナミック・リンク・ライブラリー (DLL) の競合が減る。ユーザーの追加、移動、変更が簡単にできる。
段階的なソフトウェア導入のコスト削減	<ul style="list-style-type: none"> ほとんどの大企業ではすでにターミナルサービスを利用しているため、新しいアプリケーションを追加する際に、新しいインフラストラクチャーやソフトウェアを大量に導入しなくても済む。 サーバーホスト型デスクトップ仮想化などの他のサーバーベース・モデルに比べて、ターミナルサーバー 1 台当たりのサポート可能なユーザー数が多い。
リモートアクセス	<ul style="list-style-type: none"> 集中管理型コンピューティングにより、ネットワークに接続された任意のクライアントからアクセスが可能。端末のユーザーを指定する必要がなく、ユーザーはどの端末でも使用できる。
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> データセンターや職場が災害にあった場合、冗長サーバーとデータストレージが用意されていれば、比較的簡単に他の施設に作業を移転できる。
クライアントの消費電力の削減	<ul style="list-style-type: none"> シンクライアント端末は、大部分のデスクトップ PC よりもクライアントの消費電力が小さい。ただし、データセンター内の消費電力が増えるため、総消費電力はほとんど変わらないことがある。

表 4. ターミナルサービスの制限

パフォーマンスと応答性	<ul style="list-style-type: none"> 多少演算量が多い程度アプリケーションであっても、ユーザー数が増えるとシステムの応答性が低下する。ネットワーク帯域幅と読み込み速度が、クライアント・システムのパフォーマンスの主な決定要因になる。ネットワークには、端末ユーザーの数に対応する十分なピーク処理能力が必要である。 リモート・ディスプレイ・プロトコル (RDP) がグラフィックス性能の重大なボトルネックとなる。ビデオや Adobe* Flash* などのモーション・グラフィックスは、RDP 上では効果的に動作しない。より高性能なサードパーティー製ディスプレイ・プロトコルが必要な場合がある。 ユーザーはマウスのクリックとキー操作に対する迅速な応答を期待している。応答性の要求に応えるには、ターミナルサーバーをクライアントに近い場所に配置しなければならない。
ソフトウェアの互換性	<ul style="list-style-type: none"> ターミナルサービスでは動作しないソフトウェアも存在する。また、VoIP やストリーミング・メディアなど、演算処理やグラフィックス処理の多いアプリケーションは、サーバーベース・コンピューティングには適さない。
モビリティの欠如	<ul style="list-style-type: none"> ターミナルサービスによるアプリケーションの提供には、持続的なネットワーク接続と十分な帯域幅が必要である。ターミナルサービスでもワイヤレス・ノートブック PC やタブレットの使用は可能であるが、デバイスがネットワークから切断されるとセッションは終了する。
新規導入のコスト	<ul style="list-style-type: none"> 多数のユーザーに向けた新規導入には、新しいサーバー、ソフトウェア、ネットワーク・ストレージ、ネットワーク・インフラストラクチャーなどに多額の費用がかかる。 シンクライアント端末の取得コストは、一般的なデスクトップ PC とほぼ同等である。ターミナルサービスの初期コストは、サーバーホスト型デスクトップ仮想化とほぼ同じだが、クライアント・ホスト型デスクトップ仮想化や良好に管理された PC に比べて割高である。¹
単一箇所を原因とするシステム全体の障害	<ul style="list-style-type: none"> 冗長システムのバックアップがない場合、データセンターまたはネットワークの機能が失われると、すべてのユーザーがオフライン状態になる。
ユーザーの満足度が低い	<ul style="list-style-type: none"> ユーザーは PC レベルのパフォーマンス、カスタマイズへの対応、柔軟性、モビリティを期待している。ターミナルサービスの運用が不適切であるか、必要以上に厳格であると、ユーザーは強い不満を感じる。

サーバーホスト型デスクトップ仮想化

概要

サーバーホスト型デスクトップ仮想化（デスクトップ仮想化インフラストラクチャー（VDI）とも呼ばれる）は、多くの企業で評価が進められている新しいモデルです。サーバーホスト型デスクトップ仮想化は、インテリジェントな分散型コンピューティングの応答性に優れたカスタマイズ可能なユーザー体験と、サーバーベース・モデルの優れた運用管理とセキュリティを両立させるように設計されています。このモデルは、デスクトップ・イメージ全体の集中管理を可能にします。

他のサーバーベース・モデルと同じように、このモデルのパフォーマンスと応答性は、ユーザー数、物理的距離、アプリケーションの種類によって異なります。クライアント端末のパフォーマンスは特に重要です。クライアント端末のパフォーマンスが低い場合、メディア・アクセラレーション機能を追加しない限り、ビデオ、Adobe* Flash*、VoIP など、演算処理やグラフィックス処理の多いアプリケーションには適しません。ただし、HDX*、RemoteFX*、PC over IP (PCoIP*) など、最近のリモート・デスクトップ・プロトコルの進化により、インテリジェントなクライアント端末のローカルリソースを利用できるようになり、このソリューションへの期待は高まっています（HDX* と PCoIP* では WAN 経由のユーザー体験が向上し、RemoteFX* では LAN/ 非 WAN ユーザーの体験が向上します。RemoteFX* には、RDP チャンネル上のメディアをエンコードするサーバー GPU カードが必要です）。

サーバーホスト型デスクトップ仮想化には、持続的なネットワーク接続が必要です。したがって、ネットワーク外のモビリティを必要とする用途には、このモデルは不適當です。

クライアント処理、グラフィックス、メモリーのリソースはすべてデータセンター内に構築されている必要があります。またストレージシステムは、各ユーザーの OS、アプリケーション、データを格納する必要があります。このインフラストラクチャーのコストは、運用管理機能の向上による TCO の削減と比較検討する必要があります。

サーバーホスト型デスクトップ仮想化は、アプリケーション・ストリーミングなどの他のデスクトップ仮想化モデルと組み合わせると効果的です。サーバー上の仮想マシン内でアプリケーション・データと OS ストリーミングを組み合わせ使用できます。

仕組み

ターミナルサービスと同じように、すべての処理とストレージは集中化され、アプリケーション・イメージはリモート・ディスプレイ・プロトコル（RDP）または他のディスプレイ・プロトコルによってネットワークを介してクライアントにプッシュされます。ターミナルサービスとの主な違いとしては、サーバーホスト型デスクトップ仮想化は、ユーザー独自の完全な仮想マシン（VM）と、OS、アプリケーション、設定を含むカスタマイズされたデスクトップを各ユーザーに提供できる点が挙げられます。

図 2. サーバーホスト型デスクトップ仮想化のアーキテクチャー

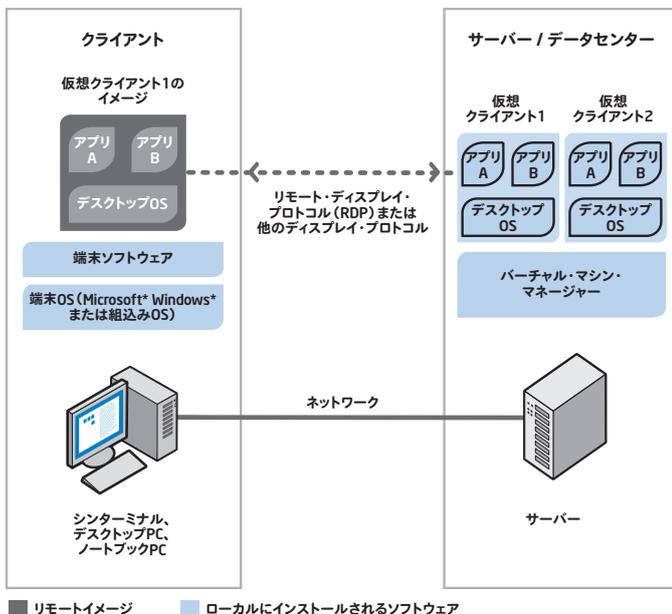


表 5. サーバーホスト型デスクトップ仮想化の利点

パフォーマンス	<ul style="list-style-type: none"> サーバーホスト型デスクトップ仮想化は、クライアントのコンピューティング・リソースを使用して端末のパフォーマンスを改善できる。 HDX* や PCoIP* など、最近のリモート・デスクトップ・プロトコルの進化により、ビデオ、Adobe* Flash*、VoIP などの演算量の多いアプリケーションのローカル・パフォーマンスが向上するが、クライアントの要件も増大している。
セキュリティの強化	<ul style="list-style-type: none"> このモデルでは、OS、アプリケーション、データがデータセンター内でロックされるため、クライアント・ベースのモデルに比べてクライアント上でのセキュリティ違反やデータ損失のリスクが軽減される。
運用管理の効率化	<ul style="list-style-type: none"> アプリケーションとデータを集中的に管理でき、運用管理の効率化とバックアップの信頼性向上が可能。 ソフトウェア・イメージの管理、検証、サポートが簡略化される。ドライバーとダイナミック・リンク・ライブラリー (DLL) の競合が減る。
ユーザーによるカスタマイズ	<ul style="list-style-type: none"> 各ユーザーがサーバー上に完全な仮想マシンを持つため、PC のような設定のパーソナライゼーションが可能。
リモートアクセス	<ul style="list-style-type: none"> 集中管理型コンピューティングにより、ネットワークに接続された任意のクライアントからアクセスが可能。端末の使用者を指定する必要がなく、ユーザーはどの端末でも使用できる。
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> データセンターや職場が災害にあった場合、冗長サーバーとデータストレージが用意されていれば、比較的簡単に他の施設に作業を移転できる。
クライアントの消費電力の削減	<ul style="list-style-type: none"> シンクライアント端末は、大部分のデスクトップ PC よりもクライアントの消費電力が小さい。ただし、データセンター内の消費電力が増えるため、総消費電力はほとんど変わらないことがある。

表 6. サーバーホスト型デスクトップ仮想化の制限

パフォーマンスと応答性	<ul style="list-style-type: none"> 多少演算量が多い程度 of アプリケーションであっても、ユーザー数が増えるとシステムの応答性が低下する。ネットワーク帯域幅と読み込み速度が、クライアント・システムのパフォーマンスの主な決定要因になる。ネットワークには、端末ユーザーの数に対応する十分なピーク処理能力が必要である。 リモート・ディスプレイ・プロトコル (RDP) がグラフィックス性能の重大なボトルネックとなる。ビデオや Adobe* Flash* などのモーション・グラフィックスは、RDP 上では効果的に動作しない。HDX* や PCoIP* などの新しいディスプレイ・プロトコルが必要な場合がある。 ユーザーはマウスのクリックとキー操作に対する素早い反応を期待している。応答性の要求に応えるには、ターミナルサーバーをクライアントに近い場所に配置しなければならない。
ネットワーク・パフォーマンス	<ul style="list-style-type: none"> RDP ネットワーク・トラフィックは、メディアと画面上のイメージの動きによって大きく変動する。
運用管理性	<ul style="list-style-type: none"> クライアント上にはソフトウェア・イメージは置かれませんが、IT 部門はデータセンター内に格納されるすべての仮想デスクトップ・イメージに対する管理、更新、パッチ適用を行わなければならない。
ソフトウェアとデバイスの互換性	<ul style="list-style-type: none"> バーチャル・マシン・マネージャー (VMM) が、アプリケーション、デバイス、関連するドライバーと、ハードウェアの間を仲介する。特殊なデバイスや、ハードウェアとの直接の相互作用を必要とするソフトウェアには、互換性の問題が発生する。
モビリティの欠如	<ul style="list-style-type: none"> サーバーホスト型デスクトップ仮想化によるデスクトップの提供には、持続的なネットワーク接続と十分な帯域幅が必要である。サーバーホスト型デスクトップ仮想化でもワイヤレス・ノートブック PC やタブレットの使用は可能であるが、デバイスがネットワークから切断されるとアプリケーション・セッションは終了する。
新規導入のコスト	<ul style="list-style-type: none"> 多数のユーザーに対するサーバーホスト型デスクトップ仮想化の新規導入には、多額の費用がかかる。クライアント処理、グラフィックス、メモリーのリソースはすべてデータセンター内に構築されている必要がある。またストレージシステムは、各ユーザーの OS、アプリケーション、データを格納する必要がある。新しいハードウェア、ネットワーク、ビルド作業スペースが必要になる。 デスクトップ仮想化サーバーによって 1 台あたりにサポートされるユーザー数は、他のサーバーベース・モデルに比べて少ない。 シンクライアント端末の取得コストは、多くのデスクトップ PC とほぼ同等である。 RemoteFX* などの新しいメディア・リダイレクション・テクノロジーには、サーバー GPU カードの増設が必要であり、サポート可能な仮想マシンまたはディスプレイの数が制限される。
単一箇所を原因とするシステム全体の障害	<ul style="list-style-type: none"> 冗長システムのバックアップがない場合、データセンターまたはネットワークの機能が失われると、すべてのユーザーがオフライン状態になる。
ユーザーの満足度が低い	<ul style="list-style-type: none"> ユーザーは PC と同レベルのパフォーマンス、柔軟性、モビリティを期待している。デスクトップ仮想化の運用が不適切であったり、必要以上に厳格過ぎると、ユーザーは強い不満を感じる。

ブレード PC

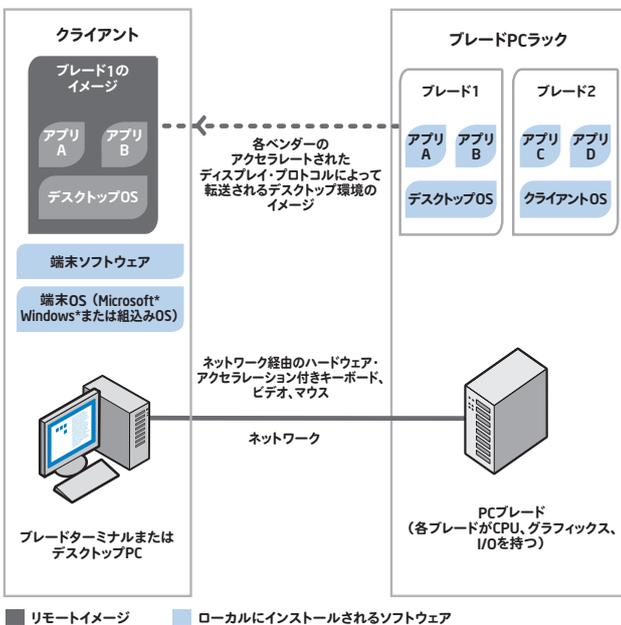
概要

ブレード PC は、クライアント・コンピューティングを中央に配置し、物理的アクセスの制限、ソフトウェア・イメージング・ポリシー、ユーザーの活動の制限により、インテリジェントな分散型コンピューティングを上回る運用管理機能とセキュリティを実現します。

各ユーザーに 1 つの PC ブレードを割り当てる場合 (1 対 1)、このモデルは、インテリジェントな分散型コンピューティングに非常によく似たものになります。相違点は、ブレード PC は決まった場所からのみ利用可能であり、ユーザーはネットワークに常に接続していなければならないという点です。個々の PC ブレードが複数のユーザーに同時にサービスを提供する場合 (1 対多)、このモデルは、サーバーホスト型デスクトップ仮想化によく似たものになります。

ブレード・コンピューティングは独自規格に基づいているため、導入に比較的成本がかかり、IT 部門はベンダー固有のアーキテクチャーに長期的に拘束されることになります。ベンダーまたはモデルを変更する場合、ハードウェアを含めて完全なアップグレードが必要になります。さらに、1 対多のブレード PC には、サーバーホスト型デスクトップ仮想化と同様の短所である、パフォーマンスの問題と高いインフラストラクチャー・コストがあります。

図 3. 1 対 1 ブレードのアーキテクチャー



仕組み

ブレード PC は、PC の機能を再構成して、基本的なディスプレイ、キーボード、マウスの機能だけをクライアント上に残し、プロセッサ、チップセット、グラフィックス・チップを中央のユニットのラックに搭載される小型カード (ブレード) 上に配置します。OS、アプリケーション、データストレージは、ストレージアレイ内に集中化されます。

サーバーブレードとは異なり、PC ブレードは、標準的なデスクトップ・プロセッサまたはモバイル・プロセッサとチップセットで構成されます。多数のブレードをサポートする中央のユニットは、データセンター内、または IT 部門が管理する特別なスペースにおいてセキュリティを確保されます。リモート・ディスプレイと I/O は、データ・ネットワークを介した RDP ではなく、独自規格に基づく専用の接続によって処理される場合もあります。

ブレード PC ベンダーは、当初はユーザー数とブレード数の比率を 1 対 1 にすることを想定していました。この場合、ユーザーは動的に 1 つのブレードを割り当てられ、そのブレードを独占的に使用します。しかし、ブレード・ソリューションと仮想化ソフトウェアの進化とともに、ほとんどのベンダーが 1 対多のブレード PC 機能を提供するようになりました。

図 4. 1 対多ブレードのアーキテクチャー

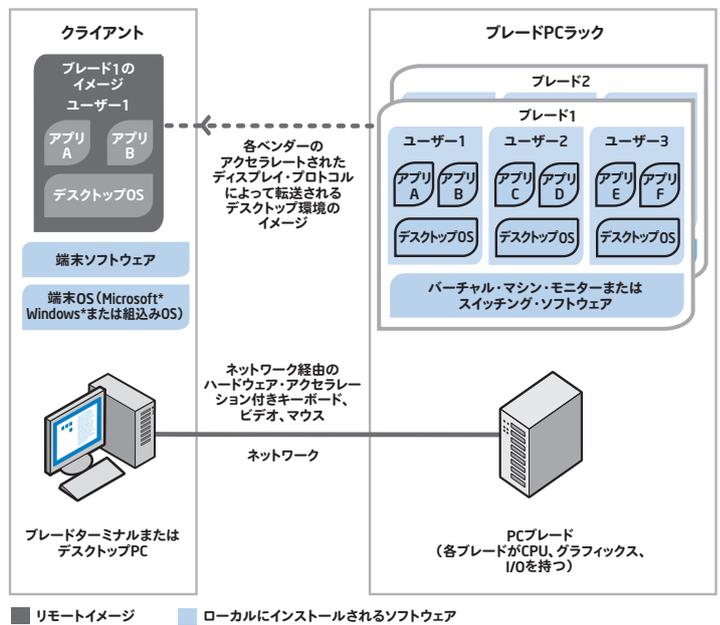


表 7. ブレード PC の利点

セキュリティ	<ul style="list-style-type: none"> このモデルでは、OS、アプリケーション、データがデータセンター内でロックされるため、クライアント・ベースのモデルに比べてクライアント上でのセキュリティ違反やデータ損失のリスクが軽減される。
運用管理性	<ul style="list-style-type: none"> OS、アプリケーション、データを集中的に管理でき、運用管理の効率化とバックアップの信頼性向上が可能。 PC ブレードは共通のハードウェア・プラットフォームを提供するため、検証、イメージ管理、サポートが簡略化される。 任意の利用可能な PC ブレードを動的にユーザーに割り当てることが可能。ユーザーの追加、移動、変更が比較的容易。
ユーザーによるカスタマイズ	<ul style="list-style-type: none"> 各ユーザーが独自の OS/アプリケーション・イメージを持てるため、PC のような設定のパーソナライゼーションが可能。
リモート・ネットワーク・アクセス	<ul style="list-style-type: none"> 集中管理型アーキテクチャーにより、ユーザーは、ブレードに接続された任意のクライアントから、また場合によってはインターネットに接続された任意のクライアントからアクセスできる。端末の使用者を指定する必要がなく、ユーザーはどの端末からでもアクセスできる。
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> ブレードラックや職場が災害にあった場合、冗長インフラストラクチャーが利用可能であれば、他の施設に作業を移転できる。 ユーザーに割り当てられたブレードが故障した場合、簡単に他のブレードに移動できる。
クライアントの消費電力の削減	<ul style="list-style-type: none"> クライアント・デバイスの消費電力、発熱量、ファンノイズは、大部分のデスクトップ PC よりも小さい。ただし、クライアント、ブレード、関連するストレージの総消費電力はほぼ同等になる。

表 8. ブレード PC の制限

パフォーマンス	<ul style="list-style-type: none"> 1 対多の構成では、ユーザーの数とワークロードによってアプリケーションのパフォーマンスが低下する。
運用管理性	<ul style="list-style-type: none"> クライアント上にはソフトウェア・イメージは置かれませんが、IT 部門はデータセンター内に格納されるすべての集中型デスクトップ・イメージに対する管理、更新、パッチ適用を行わなければならない。
ベンダーの固定化	<ul style="list-style-type: none"> ブレード PC は標準化されていないため、各ベンダーが独自規格に基づくシステムを提供している。特定ベンダーのブレード・アーキテクチャーを一度選択してしまうと、切り替えのコストは非常に高額になる。ベンダーまたはモデルを変更する場合、通常はハードウェアの完全なアップグレードが必要になる。 利用可能な管理ツールは限られており、IT 部門はブレードベンダーのツールと開発スケジュールに依存せざるを得ない。
モビリティの欠如	<ul style="list-style-type: none"> ブレード PC にはモバイルオプションは存在しない。このモデルは、持続的なネットワーク接続が可能なユーザーにのみ適している。
ユーザー 1 人当たりのコストが高い	<ul style="list-style-type: none"> ブレード PC は非標準アーキテクチャーに基づいているため、ユーザー 1 人当たりのブレード PC の取得コストは他のモデルよりも高くなる。 ストレージシステムは、新規ユーザーの OS、アプリケーション、データを格納する必要がある。新しいハードウェア、ネットワーク、ビルド作業スペースが必要になる。
単一箇所を原因とするシステム全体の障害	<ul style="list-style-type: none"> 冗長システムのバックアップがない場合、ブレードサーバー、ネットワーク・アクセス、またはデータセンターの機能が失われると、すべてのユーザーがオフライン状態になる。

OS イメージ・ストリーミング

概要

OS イメージ・ストリーミングは、サーバーホスト型デスクトップ仮想化と同じように、デスクトップ・イメージ全体の集中管理を可能にします。このモデルは、ローカル実行の高い応答性とパフォーマンスをユーザーに提供する一方、集中管理による優れた運用管理性とセキュリティをIT部門に提供します。サーバーホスト型デスクトップ仮想化とは異なり、大規模なインフラストラクチャーの増設は不要です。

現在のところ、持続的なローカルキャッシュ機能を持つOSイメージ・ストリーミングを提供しているベンダーはありません。したがって、モビリティが要求される場合はこのモデルは使用できません。実行時にデータがクライアント上に置かれるため、最高レベルのデータ・セキュリティを必要とするアプリケーションには適していません。

仕組み

起動時にはクライアントは基本的にベアメタルの状態であり、OSイメージはローカルにインストールされていません。OSイメージはネットワークを介してクライアントにストリーミングされ、クライアントのCPUとグラフィックスを使用してローカルで実行されます。アプリケーション・データはデータセンター内に格納されます。クライアントは、通常はハードディスク・ドライブを搭載せず、RAMのみを使用するPCです。

ストリーミング技術により、OSイメージ・ソフトウェアは、ソフトウェア・ベンダーから提供された状態とは異なる形式でクライアントにストリーミングされます。OSイメージ・ソフトウェアには、まず準備プロセスが適用され、優先順位付きのブロックに分割されて特定の順序で配置され、クライアントにストリーミングされます。起動 / 初期化用の基本ソフトウェアが最初に送信され、需要が大きいサービスと機能が次に送信されます。このような最適化により、すべてのコードがクライアントにストリーミングされる前に、OSを起動して処理を開始できます。ネットワーク・トラフィックを削減するため、一部の使用頻度の低い機能は、要求されるまでデータセンター内にとどまります。

図 5. OS イメージ・ストリーミングのアーキテクチャー

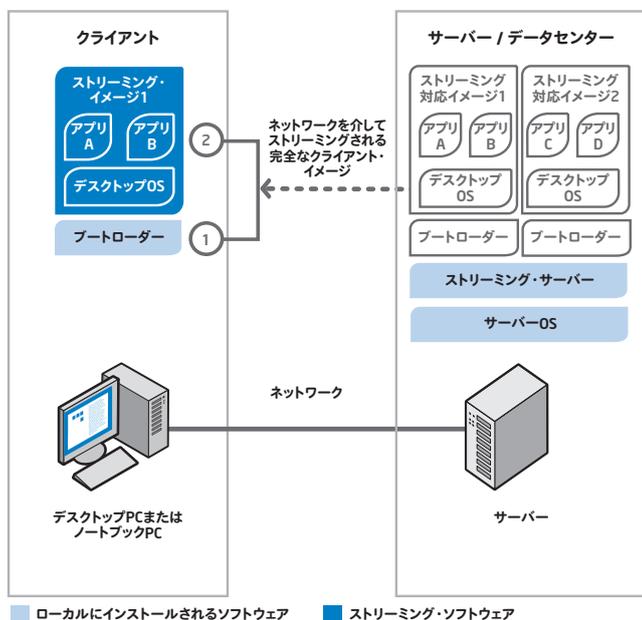


表 9. OS イメージ・ストリーミングの利点

セキュリティ	<ul style="list-style-type: none"> ▪ 重要なアプリケーション・データはデータセンター内に格納される。 ▪ アプリケーションや OS が破壊されることはない。パッチは、システムが起動時に再インストールされた時点で自動的に適用される。
運用管理性	<ul style="list-style-type: none"> ▪ OS、アプリケーション、データを集中的に管理でき、運用管理の効率化、ソフトウェアの簡単な移動、バックアップの信頼性向上が可能。 ▪ OS イメージ・ストリーミングにより、ローカルにインストールされるソフトウェアで発生するイメージ管理とソフトウェア・サポートの問題が大幅に減少。 ▪ ソフトウェア・ライセンスの集中管理が可能。ストリーミングにより、実際のアプリケーション利用率の分析が向上し、ライセンス管理の最適化が可能。 ▪ ステートレス・クライアントにより、ユーザーの追加、移動、変更が非常に簡単に行える。
パフォーマンス	<ul style="list-style-type: none"> ▪ ローカルにインストールされる従来のアプリケーションと、事実上同等のパフォーマンスを実現。ビデオ、Adobe* Flash*、ストリーミング・メディアなど、演算処理やグラフィックス処理の多いアプリケーションにも十分な性能を発揮する。 ▪ ネットワークを介したアプリケーション画面イメージのプッシュにリモート・ディスプレイ・プロトコル (RDP) を使用しないため、最初のブートアップ後のネットワーク負荷はサーバーベースのソリューションより小さい。
インフラストラクチャー・コストの削減	<ul style="list-style-type: none"> ▪ サーバーベースのモデルに比べて、必要なサーバーとネットワーク・インフラストラクチャーは少ない。 ▪ OS イメージ・ストリーミング技術は、すべての集中管理型コンピューティング・モデルの中で最も TCO が小さい。¹
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> ▪ データセンターや職場が災害にあった場合、冗長サーバーまたはデータセンターが用意されていれば、比較的簡単に他の施設に作業を移転できる。

表 10. OS イメージ・ストリーミングの制限

セキュリティ	<ul style="list-style-type: none"> ▪ 実行時にデータとアプリケーションがクライアント上に置かれるため、サーバーホスト型モデルに比べて多少クライアント側での攻撃を受けやすい。
ネットワーク	<ul style="list-style-type: none"> ▪ OS のブート時にネットワーク使用率がユーザー 1 人当たり 2 ~ 5Mbps 増加するが、ローカルキャッシュの利用によってその後のトラフィック・レベルは低下する。 ▪ ストリーミングのダウンロード速度は、サーバーからの物理的距離、ネットワーク負荷、ユーザー数の影響を受ける。
ソフトウェアの互換性と実装の問題	<ul style="list-style-type: none"> ▪ 内部アーキテクチャーが原因で、一部の古いアプリケーション・ソフトウェアやカスタム・アプリケーション・ソフトウェアでは準備プロセスが正常に完了せず、OS イメージ・ストリーミングによってサービスを提供できないことがある。 ▪ OS イメージ・ストリーミングでは、ソフトウェア準備プロセスの初期セットアップとデバッグに多くの時間と労力が必要になる。 ▪ 最近のリモート OS ブート・ソリューションは未修正の「ゴールドイメージ」を利用するため、OS ストリーミングに必要な準備プロセスの制約はない。
モビリティの欠如	<ul style="list-style-type: none"> ▪ 本資料の執筆時点では、ネットワーク外またはモバイルで OS イメージ・ストリーミングを使用できる製品は発売されていない。

リモート OS ブート

概要

リモート OS ブートは、OS イメージ・ストリーミングによく似た新しいテクノロジーです。リモート OS ブートは、管理コンソールから管理されるソリューションですが、バックエンド・サーバー・インフラストラクチャーを使用せずに、完全な PC を忠実に再現してエンドユーザーに提供します。このソリューションは、デスクトップ仮想化の利点を提供しながら、IT 部門と経営者にとって非常に魅力的なコストで、完全な PC と Microsoft* Windows* を忠実に再現できます。

仕組み

リモート OS ブートは、インテル® Core™ vPro™ プロセッサ搭載クライアント、インテル® プロセッサ搭載ストレージ・アプライアンス、管理コンソールの 3 つの主なコンポーネントで構成されます。接続ブローカー機能やコンポーネントのプロビジョニングを含むすべての管理作業は、管理コンソールから実行されます。

クライアントの電源をオンにすると、クライアントは内蔵の iSCSI ブート・テクノロジーを使用して、リモート・ストレージ・アプライアンスに接続します。ここで、Xen* に基づくタイプ 1 ハイパーバイザーがリモートストレージからコピーされてクライアントのメモリにロードされ、管理コンソールへの接続が確立されます。管理コンソールは、クライアントの資格情報に基づいて、クライアントとリモートストレージ間の接続を仲介

し、リモートストレージをクライアントにマウントします。クライアントからは、この接続は標準的な SATA ハードディスク・ドライブとして認識されます。ここで、クライアントは標準的なハードディスク・ドライブからブートするときと同じように Windows* をブートします。

このソリューションはシンプルなので、IT 部門とエンドユーザーは、標準的な市販または自社開発の Windows* アプリケーションを使用できます。また、シリアルデバイス、USB デバイス、プリンターなどの標準的な周辺機器も、標準的な PC 上と同じように動作します。こうした特性により、IT 部門は、既存のツールやテクノロジーを変更せずに、新規ユーザーを素早く追加できます。

リモート OS ブートは、ハードディスク・ドライブが無効化または取り外された「ステートレス」PC に完全な OS イメージを提供する点で、OS イメージ・ストリーミングによく似ています。ただし、OS イメージ・ストリーミングとは異なり、クライアントはストレージ・エリア・ネットワーク (SAN) デバイスから直接ブートされます。クライアント PC は、SAN をローカル・ハードディスク・ドライブのように扱います。OS イメージは、ローカルディスク上で使用される「ゴールドイメージ」と同じ状態のまま、修正されずに提供されます。リモート OS ブート・ソリューションは、Pre-Execution (PXE) 環境と PXE サーバーに依存しないため、さらに導入が簡単になり、TCO が削減されます。

図 6. リモート OS ブートのアーキテクチャー

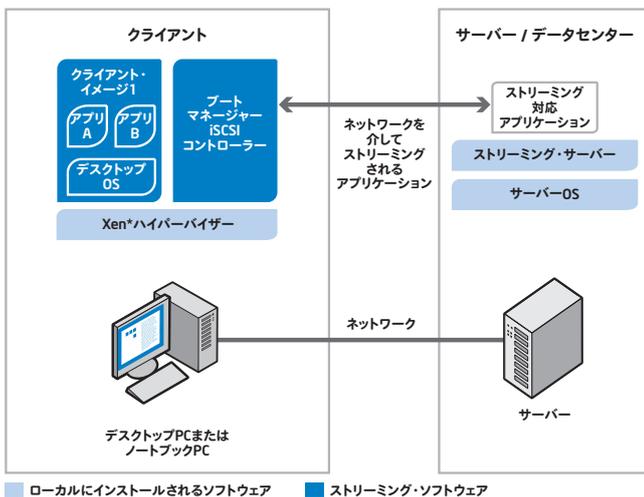


表 11. リモート OS ブートの利点

セキュリティ	<ul style="list-style-type: none"> ▪ 重要なアプリケーションの格納と管理は、中央の管理コンソールから行われる。 ▪ アプリケーションまたは OS のパッチは、クライアントのリポート時に管理コンソールから適用可能。 ▪ Windows* の標準的な暗号化デバイスと認証デバイスをサポートしている。
運用管理性	<ul style="list-style-type: none"> ▪ アプリケーション、データ、エンドユーザー・プロファイルを集中的に管理でき、運用管理の効率化とバックアップの信頼性向上が可能。 ▪ 業界標準規格に基づく IT ツールをサポートし、特殊なツールの必要性を最小限に抑制。 ▪ Active Directory* とローミング・プロファイルのサポートにより、IT 部門は、デスクトップ PC、ノートブック PC、安全なマネージド・クライアント・デバイスと同じプロセスを利用可能。 ▪ ソフトウェア・イメージの管理、検証、サポートが簡略化される。標準的な Windows* ドライバーとダイナミック・リンク・ライブラリーの使用により、ドライバーと DLL の競合を防止。新しいイメージの追加が容易。
互換性	<ul style="list-style-type: none"> ▪ 未修正の Windows* オペレーティング・システムと標準ドライバーの使用により、Windows* アプリケーションおよび標準的な PC オプションとの互換性を確保。
パフォーマンス	<ul style="list-style-type: none"> ▪ VoIP とリッチメディアをサポートし、通常の PC と同等以上のエンドユーザー体験を実現。
新規導入のコスト	<ul style="list-style-type: none"> ▪ 標準的な導入コストは、PC とほぼ同等。 ▪ クライアントの追加コストは、各種のデスクトップ仮想化ソリューションの中で最小。
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> ▪ 停電または災害の発生時には、アクセス機能を他の施設に移転することが可能。 ▪ ローカル・ハードウェアに問題が発生した場合、ユーザーは新しいクライアントに簡単にログオン可能。
クライアントの消費電力	<ul style="list-style-type: none"> ▪ 標準的な PC に比べて、より低い消費電力と発熱量。 ▪ 他のデスクトップ仮想化ソリューションに比べて、より低い総消費電力。 ▪ 標準的な PC の省電力モードは S3 と S4 をサポート。
帯域幅	<ul style="list-style-type: none"> ▪ クライアントのブート後は、ネットワーク負荷が非常に小さい。クライアントが仮想ハードディスクにアクセスする必要があるときのみ、ネットワーク・トラフィックが発生する。

表 12. リモート OS ブートの制限

互換性	<ul style="list-style-type: none"> ▪ 現在のところ、Linux* ネイティブ環境のサポートは提供されていない。
モビリティ	<ul style="list-style-type: none"> ▪ モバイル・コンピューティングはサポートしていない。
帯域幅	<ul style="list-style-type: none"> ▪ ブート時には十分なネットワーク帯域幅が必要である。

アプリケーション・ストリーミング / アプリケーション仮想化

概要

ターミナルサービスと同じように、アプリケーション・ストリーミングは集中型のアプリケーション管理を可能にしますが、応答性やパフォーマンスが犠牲になることはありません。データは、企業のポリシーに基づいて、ローカル・クライアント上または中央のサーバー上に格納できます。さらに、ネットワーク外でも使えるようにストリーミング・アプリケーションをキャッシュすれば、モバイル・コンピューティングをサポートする集中管理型モデルを実現できます。

ただし、実行時にデータがクライアント上に置かれるため、最高レベルのデータ・セキュリティを必要とするアプリケーションには適していません。また、アプリケーション仮想化では、アプリケーション間の相互作用が制限されることがあります。

仕組み

クライアント OS はローカルにインストールされますが、アプリケーションは要求に応じてサーバーからクライアントにストリーミングされ、そこでローカルに実行されます。

「ストリーミング」と「アプリケーション仮想化」はしばしば同じ意味で使用されますが、厳密には同じではありません。ストリーミングとは、クライアント上で実行するためにネットワークを介してソフトウェアを送信するデリバリーモデルです。ストリーミング・ソフトウェアは、クライアン

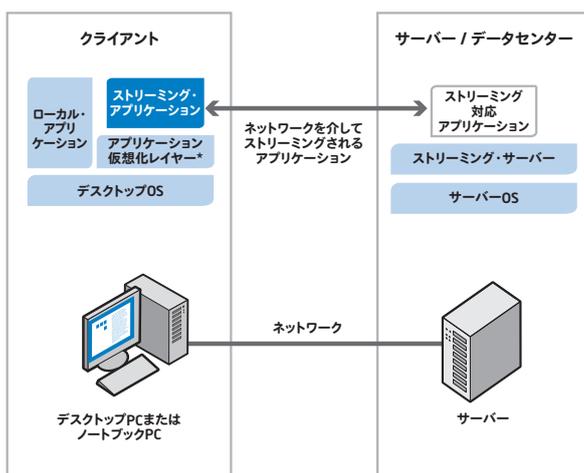
ト OS 内にローカルにインストールすることも可能ですが、ほとんどの場合は仮想化されます。

アプリケーション仮想化では、ストリーミング・ソフトウェアはアブストラクション・レイヤー上で動作し、OS レジストリーやシステムファイル内にはインストールされません。これにより、ストリーミング・アプリケーションと OS 間の相互作用が簡略化され、ソフトウェアの競合やイメー

ただし、ローカルにインストールされている場合、一部の仮想化アプリケーションと他のアプリケーションの間では、データのやり取りが成立しない場合があります。例えば、カット・アンド・ペーストができないことがあります。この問題に対処するには、ストリーミングと仮想化に関連するアプリケーションをバンドルにするか、これらの問題の大部分を解決した仮想化ソフトウェアの最新リリースを使用してください。

他の集中型コンピューティング・モデルとは異なり、ストリーミング・アプリケーションは、ノートブック PC 上にキャッシュすることで、ネットワークから切り離された状態でも使用できます。ノートブック PC がネットワークに再接続した時点で、アプリケーションはサーバーとの同期をとり、ライセンスとパッチ情報をチェックし、アプリケーション・データをデータセンターにダウンロードします。

図 7. アプリケーション・ストリーミング / アプリケーション仮想化のアーキテクチャー



*ストリーミング・ソフトウェア・ベンダーのエージェント。ベンダーとポリシーによって仮想化と隔離の度合は異なる。

表 13. アプリケーション・ストリーミング / アプリケーション仮想化の利点

セキュリティ	<ul style="list-style-type: none"> ▪ 重要なアプリケーション・データは、ポリシーに基づいて保護されたデータセンターに格納できる。ローカル・ストレージの使用を禁止できる。 ▪ アプリケーションが破損されることはない。パッチは、アプリケーションが起動時にロードされた時点で自動的に適用される。 ▪ 仮想化アプリケーションを互いに隔離し、他のアプリケーションおよび OS からのデータアクセスを制限できる。
運用管理性	<ul style="list-style-type: none"> ▪ アプリケーションと(場合によっては)データを集中的に管理でき、運用管理の効率化、ソフトウェアの簡単な移動、バックアップの信頼性向上が可能。 ▪ アプリケーション仮想化により、ローカルにインストールされるソフトウェアで発生するイメージ管理とソフトウェア・サポートの問題が大幅に減少。 ▪ アプリケーションをローカルにインストールすると互換性の問題が発生する場合でも、アプリケーション仮想化により、古いアプリケーションを新しい OS 上で実行可能。 ▪ ソフトウェア・ライセンスの集中管理が可能。ストリーミングにより、実際のアプリケーション利用率の分析が向上し、ライセンス管理の最適化が可能。
パフォーマンス	<ul style="list-style-type: none"> ▪ ローカルにインストールされる従来のアプリケーションと、事実上同等のパフォーマンスを実現。ビデオ、Adobe* Flash*、ストリーミング・メディアなど、演算処理やグラフィックス処理の多いアプリケーションにおいても十分な性能を発揮。 ▪ キャッシュオプションの設定により、アプリケーション・データをローカルに格納しないようにすることで、クライアントの起動とアプリケーションの起動を高速化。 ▪ リモート・ディスプレイ・プロトコル (RDP) などのプロトコルを使用してネットワークを介してアプリケーションの画面イメージをプッシュする必要がないため、サーバーベースのソリューションに比べて、ネットワーク負荷が小さい。 ▪ アプリケーションだけをストリーミングするため、OS イメージ全体をストリーミングするのに比べて、ネットワーク負荷は軽減される。 ▪ ユーザー体験はインテリジェントなローカル・クライアントと同等。
インフラストラクチャー・コストの削減	<ul style="list-style-type: none"> ▪ サーバーベースのモデルに比べて、必要なサーバーとネットワーク・インフラストラクチャーは少ない。
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> ▪ ネットワークまたはデータセンターがオフラインの場合でも、ユーザーはキャッシュされたアプリケーションを使用してローカル・クライアント上で作業を継続することが可能。
モビリティ	<ul style="list-style-type: none"> ▪ 他のサーバーベースのコンピューティング・モデルとは異なり、ストリーミング・アプリケーションをモバイル・クライアント上でキャッシュし、ネットワークから切り離された状態でも使用可能。

表 14. アプリケーション・ストリーミング / アプリケーション仮想化の制限

セキュリティ	<ul style="list-style-type: none"> ▪ 実行時にデータとアプリケーションがクライアント上に置かれるため、サーバーホスト型モデルに比べて多少クライアント側での攻撃を受けやすい。
パフォーマンス	<ul style="list-style-type: none"> ▪ ストリーミングのダウンロード速度は、サーバーからの物理的距離、ネットワーク負荷、ユーザー数の影響を受ける。 ▪ 仮想化により、アプリケーション間の相互作用が一部制限される(例えば、アプリケーション間のカット・アンド・ペーストは使用できない)。
ソフトウェアの互換性と実装の問題	<ul style="list-style-type: none"> ▪ 内部アーキテクチャーが原因で、一部の古いアプリケーション・ソフトウェアやカスタム・アプリケーション・ソフトウェアでは準備プロセスが正常に完了せず、アプリケーション・ストリーミングによってサービスを提供できないことがある。 ▪ アプリケーション仮想化では、ソフトウェア準備プロセスの初期セットアップとデバッグに多くの時間と労力が必要になる。
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> ▪ アプリケーション・データは中央のサーバーに格納されるが、OS またはローカルデータ、あるいはその両方がローカル・クライアント上に格納されるため、他のモデルに比べて新しい施設での作業再開に多少手間がかかる。

仮想コンテナ

概要

仮想コンテナは新しいモデルであり、急速に進化しています。このモデルは、OS およびアプリケーション・イメージ全体の集中型管理を可能にします。ローカルにインストールされる OS やストリーミング OS とは異なり、仮想コンテナは、バーチャル・マシン・マネージャー (VMM) によってクライアント・ハードウェアから抽象化されます。VMM の存在により、コンテナ内の OS から見て各種のプラットフォーム・ハードウェアの違いが隠蔽されるため、IT 部門の検証作業の負担が軽減されます。また、仮想コンテナにより、契約社員用の管理されたイメージや、ユーザーが所有する PC の利用プログラムなど、新しい利用モデルが可能になります。仮想コンテナはクライアント・ホスト型で実行されるため、データセンターの大規模な増設は不要です。サーバーベースで実行の方が合理的な場合は、同じ仮想コンテナをサーバーホスト型デスクトップ仮想化サーバー上で実行することもできます。ユーザーの利点として、仮想コンテナは、応答性に優れたローカルでの実行とネットワーク外のモビリティを可能にします。

仮想コンテナにより、標準の Windows* OS とビジネス・アプリケーションなどで構成される汎用ユーザー環境を作成できます。あるいは、単一の機能を持つ専用の「仮想プライアンス」を作成して、法令準拠モニタリングなどのサービスや、高度なセキュリティが必要なアプリケーションを提供することも可能です。

仕組み

このモデルでは、OS とアプリケーションを含む仮想マシンイメージが、IT 部門によって一元的に作成され、管理されます。サーバーホスト型デスクトップ仮想化モデルとは異なり、仮想マシンはサーバー上で動作するのではなく、クライアントにストリーミングされ、クライアント・ベースの VMM 上でローカルで実行されます。クライアント上で実行されるため、演算処理やグラフィックス処理の多いアプリケーションであっても高い応答性が維持され、ネットワーク外のモビリティも確保できます。現在、グラフィックス仮想化、ワイヤレス通信、省電力機能など、業界全体で取り組んでいる技術的な問題がいくつかあります。

ソフトウェア・ベンダーが提供している主な仮想化手法は以下の 2 種類です。各手法の利点の比較については、次の表で説明します。

- タイプ 1** : 仮想化ソフトウェアは PC ハードウェア上で直接動作し、仮想マシンまたは仮想 OS はハイパーバイザーの上で動作します。タイプ 1 の仮想化ソフトウェアは、ハイパーバイザーと呼ばれます。
- タイプ 2** : 仮想化ソフトウェアは Windows* などの OS 内でアプリケーションのように動作し、仮想マシンまたは仮想 OS は仮想化ソフトウェアの上で動作します。タイプ 2 の仮想化ソフトウェアは、バーチャル・マシン・マネージャー (VMM) と呼ばれます。

図 8. 仮想コンテナのアーキテクチャー

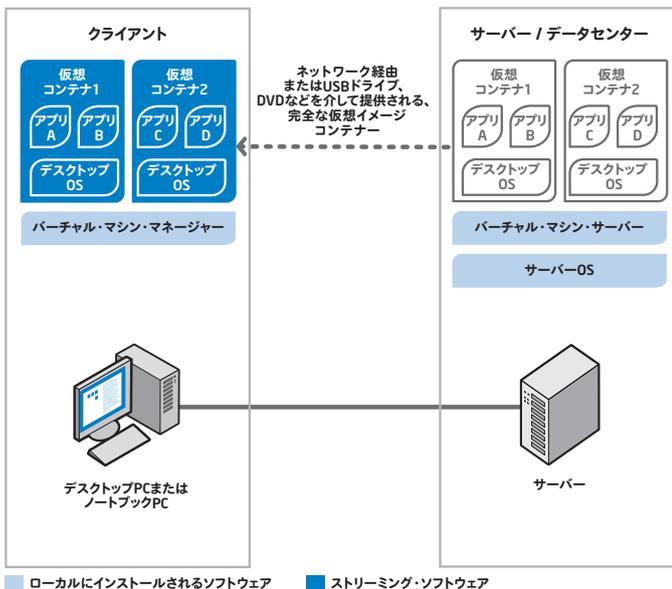


表 15. 仮想コンテナの利点

セキュリティ	<ul style="list-style-type: none"> 仮想コンテナにより、ウイルスや侵入の脅威はコンテナ化され、高度なセキュリティを必要とするアプリケーションは個別のコンテナに簡単に分離することが可能。 IT 部門が仮想コンテナのセキュリティ・ポリシーを設定し、ロックダウン、アクセスポリシー、データストレージ、認証 / 失効の管理を強化することが可能。 専用の仮想アプライアンスを使用して、ユーザーの環境の外側から強力なセキュリティ・サービスを提供することが可能。 タイプ 1 とタイプ 2 の比較: タイプ 1 は、仮想マシン同士が分離されるため、より安全と考えられる。タイプ 2 では、基盤となる OS が攻撃されると、仮想マシンも被害を受ける。
運用管理性	<ul style="list-style-type: none"> OS イメージ、アプリケーション、データを集中的に管理でき、運用管理の効率化、ソフトウェアの簡単な移動、バックアップの信頼性向上が可能。 仮想コンテナは物理ハードウェア上ではなく VMM 上で動作するため、異なる多数のハードウェア構成に対してイメージの検証を行う必要はなく、VMM に対する検証だけで済む。 仮想コンテナの移動が非常に簡単。クライアントにファイルをストリーミングするか、USB ドライブを挿入するだけでインストールが完了。 タイプ 1 とタイプ 2 の比較: タイプ 2 は、すべての Windows* ネイティブ環境のデバイスドライバーを正常に実行できる。タイプ 2 はタイプ 1 よりも複雑でなく、PC 上での動作が実証されている。タイプ 1 では、しばしば非 Windows* ドライバーを組み込む必要がある。
パフォーマンス	<ul style="list-style-type: none"> ローカルにインストールされる従来のアプリケーションと、事実上同等のパフォーマンスを実現。ビデオ、Adobe* Flash*、ストリーミング・メディアなど、演算処理やグラフィックス処理の多いアプリケーションにも十分な性能を発揮する。 タイプ 1 とタイプ 2 の比較: タイプ 2 では OS と VMM が必要なのに対して、タイプ 1 のハイパーバイザーはコードのサイズが小さくオーバーヘッドが小さいため、タイプ 1 はタイプ 2 より高性能と考えられる。
モビリティ	<ul style="list-style-type: none"> 他のサーバーベースのコンピューティング・モデルとは異なり、仮想コンテナはキャッシュを利用してネットワーク外のモバイル・クライアント上で使用することが可能。 ユーザーは、USB ドライブを利用して各自の仮想コンテナイメージを持ち運び、任意の PC (自宅、オフィス、提携先施設など) で実行することが可能。
インフラストラクチャー・コストの削減	<ul style="list-style-type: none"> サーバーベースのモデルに比べて、必要なサーバーとネットワーク・インフラストラクチャーは少ない。
災害復旧とビジネスの継続性	<ul style="list-style-type: none"> ネットワークまたはデータセンターがオフラインの場合でも、ユーザーはキャッシュされた仮想コンテナを使用してローカル・クライアント上で作業を続行することが可能。

表 16. 仮想コンテナの制限

セキュリティ	<ul style="list-style-type: none"> 実行時にデータとアプリケーションがクライアント上に置かれるため、サーバーホスト型モデルよりも多少クライアント側で攻撃を受けやすい。 VMM はソフトウェア・スタック内の新しいレイヤーであり、保護を必要とする。
パフォーマンス	<ul style="list-style-type: none"> VMM 上で複数の仮想マシンを動作させると、パフォーマンスが低下する可能性がある。ネイティブ環境に近いアプリケーション性能を実現するには、VMM の効率とハードウェア支援機能がカギとなる。 仮想化によってアプリケーション間の相互作用が制限される可能性がある。この制限は意図的な場合があるが、意図しない影響をもたらす場合もある。
成熟度	<ul style="list-style-type: none"> 仮想コンテナは比較的新しいモデルである。テクノロジー、管理ツール、IT プロセスは十分に成熟していない。
業界全体の技術的課題	<ul style="list-style-type: none"> 仮想コンテナの潜在的な可能性をフルに実現するには、業界全体でいくつかの技術的課題に取り組む必要がある。これには、グラフィックスの仮想化、ワイヤレス通信、省電力機能、ドッキング・ステーション、周辺機器などが含まれる。

まとめ

ソフトウェア・デリバリー・モデルの進化とともに、IT 部門の選択肢はかつてないほど広がっています。適切なデスクトップ仮想化モデルを選択するには、多くの要因のバランスを考える必要があります。限定されたユーザーセグメントに限っても、対象となるユーザーと IT 部門、両者のニーズをすべて満たすソリューションを見つけるには、複数のデリバリーモデルを組み合わせる必要があります。

インテル® Core™ vPro™ プロセッサ・ファミリー搭載 PC は、インテリジェントなパフォーマンスと独自のハードウェア支援によるセキュリティ機能および運用管理機能を利用して、企業のコスト削減と効率の向上を支援します。これらの PC は、あらゆるデスクトップ仮想化モデルに対応する、高度な運用管理機能とセキュアなプラットフォームを提供します。

関連情報

Principled Technologies が発行している無料のホワイトペーパーには、本資料の裏付けとなる、第三者が定量化したデータが多数記載されています。これらのホワイトペーパーは、Principled Technologies の Web サイトで入手できます。<http://www.principledtechnologies.com/> (英語)

■ ホワイトペーパー

『Total Cost of Ownership for Various Computing Models』 : <http://www.principledtechnologies.com/Clients/Reports/Intel/CompModelsTCO1107.pdf> (英語)

■ スプレッドシート

総保有コスト (TCO) 試算ツール: <http://www.principledtechnologies.com/Clients/Reports/Intel/ComputeModelTCOCalc1107.xls> (英語)

¹ 『Total Cost of Ownership for Various Computing Models』(Principled Technologies)

Intel、インテル、Intel ロゴ、Intel Core、Core Inside、Intel vPro、vPro Inside は、アメリカ合衆国およびその他の国における Intel Corporation の商標です。

Microsoft、Active Directory、Windows、Windows Vista、Windows ロゴは、米国 Microsoft Corporation および / またはその関連会社の商標です。

* その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

インテル株式会社

〒100-0005 東京都千代田区丸の内 3-1-1

<http://www.intel.co.jp/>

©2011 Intel Corporation. 無断での引用、転載を禁じます。
2011年6月

319175-005JA
JPN/1106/PDF/SE/IT/NT

