

Oracle Elite
Engineering Exchange

Oracle SPARC仮想化テクノロジー を使用した集約

Oracleホワイト・ペーパー | 2015年10月



ORACLE®

目次

はじめに.....	1
集約インフラストラクチャの設計.....	3
集約について考慮すべき7つの領域.....	3
セキュリティの独立性.....	3
リソースの独立性.....	3
ワークロードの効率.....	4
可用性.....	4
保守性.....	5
柔軟性.....	5
俊敏性.....	5
要件に基づく集約.....	6
Oracle仮想化テクノロジー.....	6
物理ドメイン (PDoms).....	6
Oracle VM Server for SPARC.....	7
制御ドメイン、I/Oドメイン、サービス・ドメイン、ゲスト・ルート・ドメインのロール.....	8
ゲスト・ドメイン・モデル.....	9
冗長ゲスト・ドメイン・モデル.....	10
SR-IOVまたは直接I/Oドメイン・モデル.....	11
冗長SR-IOVドメイン・モデル.....	12
ゲスト・ルート・ドメイン・モデル.....	14
Oracle Solaris Zones.....	14
ネイティブ・ブランド・ゾーン.....	15
カーネル・ゾーン.....	16
非ネイティブ・ブランド・ゾーン.....	17
仮想化テクノロジーの組合せ.....	19
冗長ゲスト・ドメインとOracle Solaris Zones.....	21
ゲスト・ルート・ドメインとOracle Solaris Zones.....	22
ルート・ドメインとSR-IOVドメイン.....	23
すべてのOracle仮想化テクノロジーの複合組合せ.....	25
仮想化テクノロジーの組合せが持つ特性のまとめ.....	26
結論.....	27

Oracle Elite Engineering Exchangeについて 27

はじめに

このホワイト・ペーパーでは、オラクルの仮想化テクノロジーの概要を説明します。また、次の7つの特性を観察することで、ワークロード要件に合わせて機能を選択できるようにするための、機能評価手法を紹介します。

- » セキュリティの独立性
- » リソースの独立性
- » 効率
- » 可用性
- » 保守性
- » 柔軟性
- » 俊敏性

この手法は、その他のOracle仮想化テクノロジーを評価するためにも使用できます。また、Oracle Database 12cのプラガブル・データベースやOracle WebLogic Serverでのアプリケーション統合などの、本書で扱わないOracle仮想化テクノロジーの組合せに対しても使用できます。

すべての組織が、可能な限りもっとも効率的な方法でITインフラストラクチャを実行するよう求められています。現在は、仮想化テクノロジーを使用することで、多種多様なワークロードを少数の物理サーバー上に集約してコストを削減する傾向があります。この傾向をさらに後押しするのは、一部のワークロードと比べて容量が増加した最近のサーバーと、サーバー・ルートごとに1つのワークロードを割り当てた場合に生じる大規模なサーバーの非有効活用を回避するためにワークロードを集約したいという要件です。

集約を実施する際にもっとも重要となる側面の1つは、コストの発生元を正確に理解し、データセンター・インフラストラクチャの総所有コスト（TCO）の最小化に焦点を合わせたアーキテクチャを構築することです。集約による利点がもたらされる領域は次のとおりです。

- » 運用：監視、管理、保守の必要がある多様なオブジェクトの数が減ることで、インフラストラクチャが簡素化されます。
- » デプロイ：標準ベースのリソース・プールを利用することで、新規プロジェクトごとに新しいインフラストラクチャを調達および実装する必要がなくなるため、新規ワークロードのデプロイが容易になります。
- » インフラストラクチャ：集約によりハードウェアとソフトウェアの利用率が向上するため、インフラストラクチャのフットプリントが縮小し、ひいては、これらの資産の入手に必要な資本コストを削減できます。

集約における一番の目的はコスト削減であり、アーキテクチャの運用および管理コストは、データセンター内での最大のコスト要素の1つです。このため、すべての集約活動では、運用の軽減を進めることに焦点を合わせる必要があります。通常、その方法として、管理対象エンティティの数を減らすとともに、すべてのコンポーネントを全く同様に管理できるように、オペレーティング・システム（OS）やパッチ・レベル、演算処理の構成要素を標準化することで、アーキテクチャを簡素化します。

本書で説明するモデルの一部では、複数のワークロードで同じOSバージョンを共有する必要があるため、この標準化が徹底されます。標準化によるメリットは、個別に管理する必要のあるOSバージョンの数と、OSインスタンスの合計数が減ることで、サポート対象の基盤インフラストラクチャのサイズが小さくなる点です。アプリケーション・バイナリの互換性を保証するOracle Solarisでは、特定のパッチ・レベルやカーネル・リリースにアプリケーションが関連付けられていないため、この統合タイプに非常に適しています。この場合、アプリケーション・スタックごとに固有のOracle Solarisバージョンを多数維持する必要がなくなります。

マシンと仮想マシンが1対1でマッピングされる、従来の画一的な仮想化方式を採用した場合、管理対象のエンティティ数に変わりはないうえに、新たに導入された仮想化インフラストラクチャ・システムを管理するためのオーバーヘッドが加わるため、運用の総合的な複雑さは軽減されません。データセンターの運用効率を上げるのは、ワークロードの集約であるため、単にマシンを集約するのではなく、ワークロードの集約を目標とする必要があります。

最適な仮想化テクノロジーは要件に基づいて選択する必要があります。開発環境の要件は、ミッション・クリティカルな本番環境とは大きく異なる可能性があります。Oracle Solaris 11.2はUnified Archives機能の使用を通じて、任意のデプロイ・オプション間でシームレスにワークロードを移行する機能を提供します。このため、開発、テスト、本番の環境ごとに異なる仮想化モデルを採用しながら、同じワークロードを任意のデプロイ・タイプに移行できます。さらに重要な点として、時間とともに本番環境が発展しても、Unified Archivesが提供する柔軟性を利用することで、ワークロードをもっとも適切なデプロイ・タイプに移行できます。

本書では、Oracleの仮想化テクノロジーを組み合わせることで、ワークロード要件に基づいてもっとも適切なデプロイ・オプションを選択できることを示します。柔軟なモデルとワークロードのデプロイ・オプションのおかげで、時間とともに変化する要件に合わせてアーキテクチャを簡単に変更できます。

このホワイト・ペーパーは、読者がOracle仮想化テクノロジーに関する実務上の知識を有することを前提としています。Oracle仮想化テクノロジーの技術側面を説明するホワイト・ペーパーは、具体的な実装ガイドラインを含めて多数提供されています。

特に、物理ドメイン（PDoms）とOracle VM Server for SPARC（旧称Sun Logical Domains）テクノロジーについて詳しく説明した『[オラクルのSPARC M7サーバーとSPARC T7サーバー：ドメイン構成のベスト・プラクティス](#)』を、本書の前提条件として一読することを推奨します。

集約インフラストラクチャの設計

複数の集約オプションがある場合は、以下に示す、そもそもの集約の動機を思い出し、これらの初期要件を使用して最適なソリューションを導き出すと良いでしょう。

» 運用効率の最大化

- » 集約の利点は、ハードウェア・コストの削減のみから得られる訳ではありません。おもな集約の利点は、運用モデルの標準化と管理対象オブジェクト数の削減による簡素化からもたらされます。
- » できる限り上層のスタックを集約することで、管理対象オブジェクトの合計数が自然に減るとともに、可能な限りの標準化が実現されます。

» ワークロード効率の最大化

- » 多くの区分化を行うことによるトレードオフの1つとして、仮想化オーバーヘッドが高くなる可能性があります。これを念頭に置いて、本当に必要な場合のみ、新しい区分化を作り出すようにします。
- » 最近のOSインスタンスのフットプリントと比べると、一部のワークロードは極めて小さいものとなります。できる限り、1つのOSインスタンスに複数のワークロードを割り当てるようにします。

Oracle仮想化テクノロジーはそれぞれ異なる特徴を持っています。各テクノロジーがどのように異なるかを理解することで、アプリケーションおよびワークロード要件に合った適切なOracleテクノロジーの組合せを選ぶことができます。

集約について考慮すべき7つの領域

集約されたデータセンターは、サイロ化したデータセンターとは大きく異なる特徴を持ちます。そのため、セキュリティ、リソース割当て、可用性、保守性の管理方法を変える必要があります。

セキュリティの独立性

サイロ化した環境では別々のハードウェアが使用されているため、単純に物理的な分離によってワークロードが分離されます。集約環境には多数の共有コンポーネントが含まれるため、必要な分離レベルを確保するためには新しいテクノロジーを導入する必要があります。ほとんどのケースで、既存のセキュリティ・ポリシーを再検討して、仮想化された新しい共有環境に対応する必要があります。それ以外のケースでは、既存ポリシーを遵守するために、特定の仮想化テクノロジーを採用する必要があります。

リソースの独立性

専用ハードウェアを使用するワークロードではリソース割当てが保証されていますが、この場合は、利用されないリソースが大量に生み出されます。集約によるおもなメリットの1つは、すべてのコンピューティング・リソースをプールすることで、より効率的にリソースを使用できるようになることです。この方法でリソースをプールした場合は、必要なサービス・レベルを提供できるだけの十分なリソースを各ワークロードが利用できるようにするため、何らかの基準を導入する必要があります。また、反対に、1つのワークロードが過剰にリソースを消費することで、プール内のその他のワークロードに影響を及ぼすことがないよう防止することが不可欠です。

ワークロードの効率

ほとんどの仮想化テクノロジーはある程度のオーバーヘッドをもたらします。これは通常、次の3種類のオーバーヘッドとして現れます。

- » ハイパーバイザのオーバーヘッド: これは、ハイパーバイザの実行用に割り当てる必要のあるCPUリソースとメモリ・リソースの量を指します。消費されるリソースは、管理される仮想マシン (VM) の数とそのI/O特性によって異なります。一般に、ワークロードの数が増えると、ハイパーバイザに割り当てる必要のあるリソース量も増加します。
- » 仮想化リソースのオーバーヘッド: ハイパーバイザは通常、VMに対して仮想のCPUとメモリを割り当てる役割を果たします。また、VMに対して仮想I/Oリソースを提供します。物理CPUと比べた実際の仮想CPUのパフォーマンスは、オーバーサブスクリプションのレベルによって異なりますが、通常はより低くなります。たいていは、メモリに対しても同じことが当てはまります。ほとんどのケースで、仮想I/Oのスループットと待機時間は、ネイティブI/Oのスループットと待機時間を下回ります。統合されたワークロード用の物理インフラストラクチャのサイジングを行う場合、このパフォーマンス低下を考慮に入れる必要があります。
- » OSオーバーヘッド: 最近のオペレーティング・システムではOS自体が非常に大量のリソースを消費することも、考慮に入れる価値があります。つまり、個々のOSインスタンスに固有のブート・デバイスやディスク、I/Oだけでなく、CPUとメモリのプールを割り当てる必要があり、その一部はOS自体の維持管理のために消費されます。OSインスタンスごとに1つのワークロードをデプロイする場合、OSインスタンス自体がワークロードよりも多くのリソースを消費することはよくあります。これはしばしば、ワークロード対ペイロード比率と呼ばれます。アプリケーション・レベルの統合を使用するか、またはOS仮想化ツールを使用することで、1つのOSインスタンスごとに、できるだけ多くのワークロードを集約することが効率の向上につながります。

最初の2つによる影響は、一般に仮想化税として知られています。集約活動の一環として、OSインスタンスの数と関連するリソースおよび管理のオーバーヘッドが削減される場合、3番目の影響は税金の払い戻しと見なされます。

可用性

通常、障害は1つのサービスまたはワークロードに対してのみ影響するため、単一ワークロードのサーバーを多数使用する場合、サーバー障害による影響を軽減できるというメリットがあります。より多くのワークロードを少数のサーバー上に統合するにつれて、(計画停止か計画外停止かに関係なく、) サーバー停止による影響は大きくなります。そのため、可用性要件が高くなった場合も確実に要件を満たすことを重視して、全体的なアーキテクチャを慎重に設計する必要があります。これには通常、単一ワークロードでは必要ないと考えられていた高可用性(HA)ソリューションの実装が含まれます。また、ハードウェア・レイヤーの障害が即座に停止を引き起こさないようにするため、基盤ハードウェアで高い信頼性、可用性、保守性(RAS)機能を維持する必要もあります。

HA環境の整備には初期の設計オーバーヘッドが発生しますが、複数の統合仮想環境を使用すると、このような環境を最小限の追加コスト(またはコストなし)で大いに活用できます。さらに、複数の仮想化環境を通じて大幅にアップタイムを向上できます。

ワークロードの集約において常に役立つのは、アーキテクチャ内で障害が発生する可能性のあるエンティティ、障害が発生する可能性、そして障害による実行中ワークロードへの影響を考慮に入れることです。

本書の後半で提示する比較では、クラスタ化テクノロジーによる追加のHAは適用しないという前提のもとに、可用性スコアを決定しています。すべてのケースで、クラスタ化テクノロジーの適用により可用性特性が向上します。

保守性

可用性要件の向上と似てはいますが、コンポーネントの保守に必要な停止時間は、ずっと多くのワークロードに影響します。1つの本番ワークロードに対して停止時間枠を取り決めるだけでも十分に困難ですが、増加するワークロード数に対して同時に停止時間枠を手配することは、劇的に骨の折れる作業です。

この保守性の問題は、実行中ワークロードに影響を与える保守イベントの数を最小化することで解決できます。多くのケースで、上述の可用性アーキテクチャによって保守性要件の多くが満たされます。

柔軟性

どのような環境であっても、システム管理者が、変化するリアルタイムのワークロード要求に備えた計画を立てることは困難ですが、集約された仮想化環境では、この問題は一層深刻になります。現在の管理者は、単一ワークロードに対するサイジングやプランニングではなく、集約されたワークロードが持つすべての要件に注目しなければなりません。

大半のワークロードは、ほとんど負荷をもたらさない休止状態から、すべての割当てリソースを完全に消費する状態まで変動する、動的な特性を持つため、この問題はさらに複雑になります。複数のワークロードを単一プラットフォーム上に集約する場合、適正なサイジングは解決の難しい問題です。ワークロードのピークに対処するために、リソースを超過割当てするという単純なアプローチは非効率的であり、前述の仮想化税を人為的に増やす結果となります。

仮想化の管理に使用するツールおよびテクノロジーは、できる限り高い柔軟性を提供することで、仮想化環境のリソース割当てを動的に変更して、休止状態の負荷に大きなプラットフォーム・リソースが割り当てられている(が、使用されていない)一方で、負荷の高い別の環境がその環境内でのリソース不足で実行できないような状況を回避する必要があります。休止環境から高負荷環境へと動的にリソースを移動することで、全体的なリソース使用率が向上し、使用率が異なるワークロードへのシステム応答も改善されます。

このような柔軟性を実現するために提供されるツールは、使用する仮想化テクノロジーによって異なりますが、全レベルで、一定の粒度を備えた動的なリソース再割当てが可能であり、一部のケースではポリシーに従って事前構成することも可能です。このようなケースでは、システム所有者が規定した事前定義ルールに従って、自動的に環境を拡大または縮小できます。

ワークロードの拡大と縮小(および停止)を可能にする柔軟性は、使用率とスループットを最大化するためにはこの上なく重要です。また、実行中に環境のサイズを変更できる柔軟性により、環境にデプロイするソフトウェアのライセンス要件への潜在的な影響などの結果につながる可能性があります。

どの仮想化ワークロードが仮想化プラットフォームを共有し、どのポリシーを使用してリソース割当てを制御するかを注意深く選ぶことで、全体的なプラットフォーム使用率を卓越したレベルに向上し、使用されないリソースや無駄な資本投資を最小限に抑えることができます。

俊敏性

1つのプールに含まれるサーバー上で多数のワークロードを実行しており、保守のためにこのサーバーのサービスを停止する必要がある場合、保守期間中、サービスの中断を最小限またはゼロに抑えて、これらのワークロードを代替サーバーに移行できることが重要です。ワークロードの数が増えると、同時の停止時間枠を取り決める交渉も困難になります。

俊敏性は、その影響、複雑さ、移行期間といった観点から、ロケーション間でワークロードを移行する際の容易さを評価する特性です。

最近のワークロードの多くはスケーラブルな方法で実行するように設計されているため、Oracle Real Application Clusters (Oracle RAC) やOracle WebLogic Suiteなどを使用することで、サービスに影響を与えることなく、個々のワークロード・インスタンスを容易に無効化できます。俊敏性が重要になるのは、ワークロードがこのような機能を備えていない場合です。

要件に基づく集約

それぞれの仮想化テクノロジーは、分離、オーバーヘッド、効率、柔軟性に対して異なる特徴を持ちます。これらのテクノロジーを使用して実行する必要のある各種ワークロードにもまた、それぞれ異なる要件があります。すべてを満たす万能テクノロジーが存在しないことは明らかであり、もっとも効果的な仮想化の選択肢は、ワークロードに合わせた適切なツール（の組合せ）を使用することです。ここからは、それぞれのOracle仮想化テクノロジーについて、上述した7つの特性の観点から説明します。

Oracle仮想化テクノロジー

オラクルは、それぞれが異なるスタック・レイヤーで動作する、いくつかの仮想化テクノロジーを提供しています。これらが提供する機能を使用して集約されたインフラストラクチャを構築することで、上述したコスト削減を最大化できます。

- » 物理ドメイン (PDoms) : Oracle の SPARC M7-8サーバーおよびM7-16サーバーとSPARC M6-32サーバーは、CPU、メモリ、I/Oの各リソースを複数のPDomsに分離することで、大規模サーバーを複数の小規模サーバーに分割する機能を提供します。各物理ドメインがあたかも物理サーバーのように動作します。
- » Oracle VM Server for SPARCの論理ドメイン (LDoms) : 最新のOracle SPARCサーバー（または上述の物理ドメイン）はすべて、CPUリソースとメモリ・リソースを論理的に分離し、ネイティブまたは仮想化されたI/Oをドメインに提供することで、いくつかの小規模システムに分割できます。これにより、追加の粒度レイヤーが加わりますが、論理ドメイン間でCPUリソースとメモリ・リソースを動的に再割当てできるというメリットがあります。Oracle VM Server for SPARCテクノロジーは論理ドメインの構成に多大な選択肢を提供しますが、本書では一般的なシナリオのいくつかを説明します。現在オラクルが提供するすべての物理ドメインまたはSPARCサーバーには、少なくとも1つの論理ドメインが含まれており、最初はすべてのリソースがここに属します。
- » Oracle Solarisはそれぞれの論理ドメイン内で実行され、Oracle Solaris Zonesと呼ばれる機能を搭載しています。Oracle Solaris Zonesを使用すると、いくつかの仮想化OSインスタンスを追加で作成できます。これにより、非常に効率的な方法でもっともきめ細かいリソース割当てを実行できます。論理ドメインと同様に極めて柔軟なOracle Solaris Zonesテクノロジーは、多数の異なるデプロイ・オプションを提供します。各オプションについて、詳しくは後述します。

物理ドメイン (PDoms)

物理ドメインはサーバー・ハードウェアを電氣的に分離するため、管理者はハードウェアまたはセキュリティの障害を分離して、影響を各ドメイン内に限定できます。その結果、優れたレベルのシステム可用性とセキュリティを実現できます。このテクノロジーは、SPARC M7サーバーとSPARC M6-32サーバーで利用できます。

ソフトウェアとハードウェアのエラーおよび障害が、障害の発生したドメイン以外に広がることはありません。物理ドメイン間の完全な障害分離により、ハードウェアまたはソフトウェアのエラーによるアプリケーションへの影響が制限されます。これにより、多数のアプリケーションを統合する際に不可欠な高水準の可用性がサーバーで維持されます。各物理ドメインは個別に管理されるため、特定のドメイン内のセキュリティ侵害がその他のドメインに影響することはありません。

物理ドメイン・テクノロジーを使用すると、大規模サーバーを分割して、完全に分離された小規模サーバーをいくつか構築できます。そのおもな目的は通常、ワークロードの分離または保守性の向上を目指した、小規模構成要素の作成です。実際、各論理ドメインをスタンドアロンの物理サーバーと見なすことができます。Oracleソフトウェア・ライセンスでは、物理ドメインはハード・パーティションと見なされます。

図1と次のリストに、前述の7つの特性に対して物理ドメインがどのように対応するかを示します。

- » **セキュリティの独立性**：各物理ドメインは事実上、完全に専用のコンポーネントを備え、完全に分離された物理サーバーです。物理ドメインのセキュリティ特性はスタンドアロン・サーバーの特性とまったく同じですが、物理ドメインはOracle Integrated Lights Out Manager (Oracle ILOM) を介して、共通の管理者アクセスを共有します。
- » **リソースの独立性**：各物理ドメインには、完全に専用のCPU、メモリ、I/Oリソースが割り当てられ、サーバー内のその他すべての物理ドメインに割り当てられたリソースからは完全に分離されています。
- » **効率**：前述の“ワークロードの効率”セクションで定義したとおり、ハイパーバイザのオーバーヘッドも仮想化リソースのオーバーヘッドも発生しない場合、すべてのワークロードはベアメタル・パフォーマンス・レベルで実行されます。
- » **可用性**：物理ドメインは一般に、完全に冗長化されたコンポーネント（I/Oリソースを含む）で構成されます。つまり、通常は単一の障害には中断なしで対応できます。
- » **保守性**：SPARC M7サーバーとSPARC M6-32サーバーでは、システム内でのI/Oカードのホットプラグが可能であるため、個々のカードに対して保守作業を実施できます。ただし、CPUコンポーネントとメモリ・コンポーネントの保守には停止が必要です。
- » **柔軟性**：物理ドメインは固定ハードウェア・リソースと見なす必要があります。単一システム内の異なる物理ドメイン間で、CPU、メモリ、I/Oの各リソースを動的に再割り当てすることはできません。
- » **俊敏性**：物理ドメイン内の全リソースが持つ物理的性質のため、システム間で物理ドメインを移行することはできません。

セキュリティの独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
非常に高い	非常に高い	極めて高い	極めて高い	極めて高い	低い	低い

図1. 物理ドメインの7つの特性のまとめ

Oracle VM Server for SPARC

Oracle VM Server for SPARCドメイン（別称論理ドメインまたはLDoms）は、論理的な個別リソース・グループを構成する仮想マシンです。論理ドメインは、単一のコンピュータ・システム内に固有のオペレーティング・システムとIDを持ちます。論理ドメインごとにさまざまなアプリケーション・ソフトウェアを実行して、パフォーマンスとセキュリティのために独立性を維持できます。このテクノロジーは、現在オラクルが提供するすべてのSPARCサーバーで使用できます。

Oracleソフトウェア・ライセンスに関して言えば、適切に構成された論理ドメインはハード・パーティションと見なされます。

ハイパーバイザやゲストVMという概念を持つこのテクノロジーは、従来の仮想化テクノロジーにもっともよく似ていますが、実装方法は大きく異なります。各論理ドメインに許可されているのは、ハイパーバイザを介して利用できるサーバー・リソースの監視と相互作用のみです。ハイパーバイザはサーバー・リソースをパーティション化し、複数のオペレーティング・システム環境に限定的なサブセットを提供します。このパーティション化とプロビジョニングが、論理ドメインを作成するための基本メカニズムとなります。

それぞれの論理ドメインは、次を含む独自のリソースを備え、完全に独立したマシンとして管理できます。

- » 専用のCPUおよびメモリ・リソース
- » カーネル、パッチ、チューニング・パラメータ
- » ユーザー・アカウントと管理者
- » ディスク
- » ネットワーク・インタフェース、Media Access Control (MAC) アドレス、IPアドレス

それぞれの論理ドメインはその他の論理ドメインとは無関係に、停止、開始、再起動することができ、ユーザーがサーバーの電源を入れ直す必要はありません。また、その他の実行中の論理ドメインに影響が及ぶこともありません。

制御ドメイン、I/Oドメイン、サービス・ドメイン、ゲスト・ルート・ドメインのロール

Oracle VM Server for SPARCのデプロイメント内に構成できるドメインの各種ロールには、多種多様な名前が使用されています。1つのドメインには同時に複数のタイプを指定できるため、これはさらに複雑になります。たとえば、制御ドメインは常にI/Oドメインであり、さらにたいていはサービス・ドメインでもあります。本書の目的上、Oracle VM Server for SPARCの各種ドメイン・タイプを表すために、次の用語を使用します。

- » **制御ドメイン** - サーバーの仮想化に対する管理上の制御ポイントであり、ドメイン構成とリソース管理に使用されます。電源投入時に最初に起動されるドメインで、I/Oドメインでもあり、たいていはサービス・ドメインでもあります。制御ドメインは1つだけです。
- » **I/Oドメイン** - PCIeルート・コンプレックスやPCIeデバイス、またはシングル・ルートI/O仮想化 (SR-IOV) 機能のいずれかの物理I/Oデバイスに割り当てられたドメイン。所有するデバイスにネイティブのパフォーマンスと機能性を備えており、仮想化レイヤーによる仲介はありません。I/Oドメインは複数構成できます。
- » **サービス・ドメイン** - 仮想ネットワークとディスク・デバイスをゲスト・ドメインに提供するドメイン。サービス・ドメインは複数構成できます。サービス・ドメインはゲスト・ドメインに対して仮想化する物理的なI/Oリソースを所有する必要があるため、常にI/Oドメインでもあります。ほとんどの場合、これらのサービス・ドメインにはPCIeルート・コンプレックスが割り当てられており、このような場合はルート・ドメインとも呼ばれます。
- » **ゲスト・ドメイン** - すべてのデバイスが仮想で、物理デバイスを持たないドメイン。仮想ネットワーク・デバイスとディスク・デバイスは、1つ以上のサービス・ドメインによってゲスト・ドメインに提供されます。一般的に、ゲスト・ドメインでアプリケーションが実行されます。通常、1つのシステム内に複数のゲスト・ドメインが構成されます。
- » **ゲスト・ルート・ドメイン** - 1つ以上のPCIeルート・コンプレックスが割り当てられたドメイン。サービス・ドメインのようにサービスを提供するためではなく、ドメイン内でアプリケーションを実行するために使用されます。サービス・ドメインとゲスト・ルート・ドメイン間には、用途以外に物理的な違いはありません。多くの場合、ゲスト・ルート・ドメインは単純にルート・ドメインと呼ばれます。

Oracle VM Server for SPARCの実行時に通常使用されるデプロイメント・モデルは大まかには次の4つですが、すべてのゲスト・ドメイン・タイプを組み合わせた混合モデルを使用することもできます。

- » ゲスト・ドメイン・モデル
- » 冗長ゲスト・ドメイン・モデル
- » SR-IOVまたは直接I/Oドメイン・モデル
- » ゲスト・ルート・ドメイン・モデル

上記の各種デプロイメント・モデルについては、[Oracle VM Server for SPARCのWebページ](#)で入手できる多数のホワイト・ペーパーやWebキャストで詳しく説明されています。ここからは、4つの主要デプロイメント・モデルについて概要を説明します。

ゲスト・ドメイン・モデル

このモデルでは、制御ドメインがすべてのルート・コンプレックスを所有しており、すべてのゲスト・ドメインに対する仮想デバイスを作成します。これはもっとも柔軟性の高いモデルであり、比較的小さく、障害による影響の小さいドメインが多数ある場合に適しています。すべてのゲスト・ドメインは、制御ドメインの停止による影響を受けます。ゲスト・ドメイン間のライブ・マイグレーションが可能です。ゲスト・ドメインは、アプリケーション層での水平スケーリングによって可用性が提供される軽量の本番環境に対しても有用です。特に、パフォーマンスと可用性が重要な要件とならないテスト環境や開発環境に適しています。

図2と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：各ゲスト・ドメインは、独自のオペレーティング・システム・バージョンを持つ物理サーバーのように動作します。ただし、すべてのI/Oは制御ドメインを経由し、通常は共有物理デバイスを使用するように構成されます。制御ドメイン内とゲスト・ドメイン自体の内部でセキュリティを構成することで、盗聴や不正アクセスを防止できます。
- » **リソースの独立性**：各ゲスト・ドメインには専用のCPUリソースとメモリ・リソースが含まれるため、リソースに対するアクセスが保証されます。仮想化I/Oは、（通常は共有I/Oデバイスを介して）制御ドメインによって提供されており、このレベルでリソース競合が発生する可能性があります。追加の仮想デバイス・サービスを構成し、制御ドメイン内でリソース制御を使用することで、通常はこの競合を軽減できます。必要に応じて、物理I/Oデバイスごとに異なるゲスト・ドメインを割り当てることもできます。
- » **効率**：Oracle VM Server for SPARCテクノロジーを使用すると、ゲスト・ドメインはCPUリソースとメモリ・リソースを直接利用でき、仮想化オーバーヘッドは発生しません。ただし、このモデルでは、仮想化I/O機能を提供するために、制御ドメインに対してリソースを割り当てる必要があります。仮想化I/Oに対してもまた、小さいパフォーマンス・オーバーヘッドが発生します。このモデルではハイパーバイザと仮想化リソースにオーバーヘッドが発生しますが、これが当てはまるのはワークロードのI/Oコンポーネントに対してのみです。
- » **可用性**：ゲスト・ドメインは完全に制御ドメインに依存しているため、制御ドメインに障害が発生すると、制御ドメインが回復するまでの間はすべてのゲスト・ドメインのI/Oがフリーズします。要件によっては、ゲスト・ドメインをその他の物理サーバー上のドメインとともにクラスタ化できます。I/Oは制御ドメイン内でネットワークやディスクへの冗長パスを構成するため、これらの障害によるゲスト・ドメインへの影響はありません。

- » **保守性**：制御ドメインで保守作業を実施する必要がある場合、ゲスト・ドメインも停止するか、または、ゲスト・ドメインを別のサーバーにマイグレーションする必要があります。ただし、ゲスト・ドメインの保守作業がその他のゲスト・ドメインに影響を与えることはありません。
- » **柔軟性**：ゲスト・ドメインは非常に柔軟性に優れています。動的な作成、サイズ変更、破棄を比較的容易に実施できます。制御ドメインでポリシーを構成し、ワークロードに応じてゲスト・ドメインのサイズを自動的に拡大および縮小できます。
- » **俊敏性**：ゲスト・ドメインはSPARCサーバーに対して、ライブ・マイグレーションまたはコールド・マイグレーションを実行できます。これにより、サービス中断をごくわずかに抑えるか、またはまったく発生させることなく、ワークロードを移行できます。

セキュリティの 独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	非常に高い	中程度	中程度	中程度	高い	高い

図2. ゲスト・ドメイン・モデルの7つの特性のまとめ

冗長ゲスト・ドメイン・モデル

このモデルでは、1対になったサービス・ドメイン（片方は制御ドメイン）によって冗長化されたI/Oサービスが提供されます。ゲスト・ドメイン・モデルに非常によく似た特徴を持ちますが、制御ドメインの障害やサービス・ドメインの障害がゲスト・ドメインに影響をあたえることはありません。このモデルは高い可用性が要求される本番環境に適しています。追加のサービス・ドメイン専用割り当ての必要のあるリソースに対して、追加のオーバーヘッドが発生します。

図3と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：各ゲスト・ドメインは、独自のオペレーティング・システム・バージョンを持つ物理サーバーのように動作します。ただし、すべてのI/Oは制御ドメインとサービス・ドメインを経由し、通常は共有物理デバイスを使用するように構成されます。

制御ドメイン内とサービス・ドメイン内、およびゲスト・ドメイン自体の内部でセキュリティを構成することで、盗聴や不正アクセスを防止できます。
- » **リソースの独立性**：各ゲスト・ドメインには専用のCPUリソースとメモリ・リソースが含まれるため、リソースに対するアクセスが保証されます。仮想化I/Oは、（通常は共有I/Oデバイスを介して）制御ドメインとサービス・ドメインによって提供されており、このレベルでリソース競合が発生する可能性があります。追加の仮想デバイス・サービスを構成し、制御ドメインおよびサービス・ドメイン内でリソース制御を使用することで、通常はこの競合を軽減できます。必要に応じて、異なる物理I/Oデバイスに対してゲスト・ドメインを割り当てることもできます。
- » **効率**：Oracle VM Server for SPARCテクノロジーを使用すると、ゲスト・ドメインはCPUリソースとメモリ・リソースを直接利用でき、仮想化オーバーヘッドは発生しません。このモデルは2つのドメインを使用して冗長化されたI/Oサービスを提供しており、各ドメインにリソースを割り当てる必要があるため、ゲスト・ドメイン・モデルよりも高いハイパーバイザのオーバーヘッドが発生します。また、仮想化I/Oのために小さいパフォーマンス・オーバーヘッドも発生します。このモデルではハイパーバイザと仮想化リソースにオーバーヘッドが発生しますが、これが当てはまるのはワークロードのI/Oコンポーネントに対してのみです。

- » **可用性**：ゲスト・ドメイン・モデルとは対照的に、冗長ゲスト・ドメイン・モデルは制御ドメインに完全には依存しません。そのため、制御ドメインとサービス・ドメインの両方に障害が発生しない限り、ゲスト・ドメインが中断されることはありません。ただし、サーバー全体に障害が発生する障害シナリオでは、すべてのドメインに障害が発生します。制御ドメインとサービス・ドメイン間で I/O を構成することで、ネットワーク・リソースやディスク・リソースへの冗長パスを提供できるため、障害によるゲスト・ドメインへの影響はありません。
- » **保守性**：高い保守性は複数のサービス・ドメインを使用する方式によるおもなメリットです。この方式では、実行中のゲストに影響を与えることなく、制御ドメインとサービス・ドメインの保守作業を実施でき、ゲスト自体も個別に保守作業を実施できます。
- » **柔軟性**：ゲスト・ドメインは非常に柔軟性に優れています。動的な作成、サイズ変更、破棄を比較的容易に実施できます。制御ドメインでポリシーを構成し、ワークロードに応じてゲスト・ドメインのサイズを自動的に拡大および縮小できます。
- » **俊敏性**：ゲスト・ドメインはSPARCサーバーに対して、ライブ・マイグレーションまたはコールド・マイグレーションを実行できます。これにより、サービス中断をごくわずかに抑えるか、またはまったく発生させることなく、ワークロードを移行できます。

セキュリティの 独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	非常に高い	中程度	非常に高い	高い	高い	高い

図3. 冗長ゲスト・ドメイン・モデルの7つの特性のまとめ

SR-IOVまたは直接I/Oドメイン・モデル

Oracle VM Server for SPARCを使用すると、ドメインから物理デバイスを制御することで、このデバイスに対するアクセスをゲスト・ドメインに提供できます。これを実現する方法は2つあります。直接I/Oを使用する方法では、ローカル・マザーボードまたはPCIeデバイス/カード/スロットがゲストとなるように割り当てます。SR-IOVを使用する方法では、カード自体が仮想化をサポートしており、仮想機能（VF）をゲストに直接割り当てることができます。これらのテクニックを利用すると、ゲスト・ドメインから物理デバイスに直接アクセスできます。その場合もゲスト・ドメインは、物理デバイスを所有するサービス・ドメインに依存することに注意が必要です。複数のサービス・ドメインを構成すると、ゲストSR-IOVドメインの各種グループに対してSR-IOVサービスを提供できます。SPARC M7プロセッサベースのオラクルサーバーでは、直接I/O をサポートしていないことに留意して下さい。ルート・コンプレックスの数が多いこととSR-IOV 技術をサポートする PCIe カードを利用することで直接I/Oの技術が必要としなくなります。

この方式によるおもなメリットは、すべてのゲストがベアメタルに近いI/Oパフォーマンス・レベルで動作する点です。さらに、仮想 I/Oを動作させるために、リソースをそれほど必要としないため、小規模なサービス ドメインが構築可能であるため、ゲストに対してリソースを解放することができます。また、仮想ネットワークとディスク・サービスからは利用できないデバイスタイプへのアクセスも可能です。この方法は多数の論理ドメインをサポートしますが、I/OカードがサポートするVFの数による制限を受けます。また、移行前にSR-IOVまたは直接I/Oの構成を解除しない限り、ライブ・マイグレーションを実行できません。

図4と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：各ゲスト・ドメインは、独自のオペレーティング・システム・バージョンを持つ物理サーバーのように動作します。ほとんどのケースで、“直接”構成されるのは一部のI/Oのみであり、依然として

仮想I/Oも存在します。また、通常は制御ドメインまたはサービス・ドメイン経由のI/Oトラフィックが若干発生します。

制御ドメイン内とサービス・ドメイン内、およびゲスト・ドメイン自体の内部でセキュリティを構成することで、盗聴や不正アクセスを防止できます。

- » **リソースの独立性**：各ゲスト・ドメインには専用のCPUリソースとメモリ・リソースが含まれるため、リソースに対するアクセスが保証されます。SR-I/OVと直接I/Oを使用することで、各ゲスト・ドメインから物理デバイスに直接アクセスできますが、SR-I/OVの場合は物理デバイス自体が仮想化されており、何らかのリソース競合が発生する可能性があります。ただし、ワークロードが同じ共有デバイス上でネイティブ実行されている場合を上回ることはありません。
- » **効率**：Oracle VM Server for SPARCテクノロジーを使用すると、ゲスト・ドメインはCPU、メモリ、I/Oの各リソースを直接利用でき、仮想化オーバーヘッドは発生しません。このモデルでも、サービス・ドメインにリソースを割り当てる必要がありますが、ゲスト・ドメインに仮想化I/Oサービスを提供するためにサービス・ドメインが必要とされる、前の2つのモデル（ゲスト・ドメイン・モデルと冗長ゲスト・ドメイン・モデル）と比べると、必要なリソースは少なくなります。要約すると、このモデルでの仮想化リソース・オーバーヘッドは全面的に減少しますが、いくらかのハイパーバイザ・オーバーヘッドは依然として発生します。
- » **可用性**：直接I/OまたはSR-I/OVのいずれかを使用する場合、物理デバイスを所有するドメインへの依存が発生します。制御ドメインで計画外の障害が発生すると、デバイスに依存するゲストにも障害の影響が及び、不定の結果（通常はパニック）がもたらされます。場合によっては、その他の障害モードよりもリカバリにかかる時間が長くなる可能性があります。
- » **保守性**：物理デバイスを所有するドメインで保守作業を実施する必要がある場合、ゲスト・ドメインも含めて停止するか、またはゲスト・ドメインを別のサーバーに移行する必要があります。ただし、ゲスト・ドメインの保守作業がその他のゲスト・ドメインに影響を与えることはありません。
- » **柔軟性**：SR-I/OVドメインは非常に柔軟性に優れています。動的なサイズ変更、作成、破棄を比較的容易に実施できます。ドメインに対して物理I/Oを直接マッピングしているため、ゲスト・ドメインよりはわずかに柔軟性が低くなります。
- » **俊敏性**：非仮想化I/Oを使用するドメインのライブ・マイグレーションは実行できませんが、その他のSPARCサーバーに対するコールド・マイグレーションを実行できます。これにより、サービス中断を比較的強く抑えたままワークロードを移行できます。移行先サーバーが同一の基盤I/Oインフラストラクチャを提供するため、ゲスト・ドメイン・モデルや冗長ゲスト・ドメイン・モデルよりも若干複雑なプロセスになります。

セキュリティの 独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	非常に高い	高い	中程度	低い	中程度	中程度

図4. SR-I/OVまたは直接I/Oドメイン・モデルの7つの特性のまとめ

冗長SR-I/OVドメイン・モデル

冗長ゲスト・ドメイン・モデルと同様にOracle VM Server for SPARC 3.2より、SR-I/OV物理機能（PF）を提供するI/Oドメインを冗長構成することが可能なため、ゲストに対して仮想機能（VF）を冗長化して割り当てることができます。

このアプローチの主な利点は、I/Oドメインのレジリエンスにより、可用性、保守性を確保できるだけでなく、ゲ

ストがベアメタルに近いI/Oパフォーマンス・レベルで動作する点です。この方法は多数の論理ドメインをサポートしますが、I/OカードがサポートするVFの数に制限があります。

図5と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：各ゲスト・ドメインは、独自のオペレーティング・システム・バージョンを持つ物理サーバーのように動作します。ほとんどのケースで、“直接”構成されるのは一部のI/Oのみであり、依然として仮想I/Oも存在します。また、通常は制御ドメインまたはサービス・ドメイン経由のI/Oトラフィックが若干発生します。制御ドメイン内とサービス・ドメイン内、およびゲスト・ドメイン自体の内部でセキュリティを構成することで、盗聴や不正アクセスを防止できます。
- » **リソースの独立性**：各ゲスト・ドメインには専用のCPUリソースとメモリ・リソースが含まれるため、リソースに対するアクセスが保証されます。SR-IOVを使用することで、各ゲスト・ドメインから物理デバイスに直接アクセスできますが、SR-IOVの場合は物理デバイス自体が仮想化されており、何らかのリソース競合が発生する可能性があります。ただし、ワークロードが同じ共有デバイス上でネイティブ実行されている場合を上回ることはありません。
- » **効率**：Oracle VM Server for SPARCテクノロジーを使用すると、ゲスト・ドメインはCPU、メモリ、I/Oの各リソースを直接利用でき、仮想化オーバーヘッドは発生しません。このモデルでも、2つのサービス・ドメインにリソースを割り当てる必要がありますが、前述のモデルと比べると、必要なリソースは少なくなります。要約すると、このモデルでの仮想化リソース・オーバーヘッドはほぼ完全に排除されますが、いくらかのハイパーバイザ・オーバーヘッドは依然として発生します。
- » **可用性**：前述のモデルとは対照的に、これらのドメイン・モデルはSR-IOV サービスを提供するドメインに完全には依存しません。そのため、依存しているドメインの両方に障害が発生しない限り、ゲスト・ドメインが中断されることはありません。ただし、サーバー全体に障害が発生する障害シナリオでは、すべてのドメインに障害が発生します。レジリエントなI/Oドメイン間でゲストI/Oを構成することで、ネットワーク・リソースやディスク・リソースへの冗長パスを提供できるため、障害によるゲスト・ドメインへの影響はありません。
- » **保守性**：高い保守性はレジリエントなI/Oドメインを使用する方式のおもなメリットの1つであり、この方式では、実行中のゲストに影響を与えることなく、各I/Oドメインの保守作業を順次、実施できます。また、ゲスト自体も個別に保守作業を実施できます。
- » **柔軟性**：SR-IOVドメインは非常に柔軟性に優れています。動的な作成、サイズ変更、破棄を比較的容易に実施できます。ドメインに対して物理I/Oを直接マッピングしているため、ゲスト・ドメインよりはわずかに柔軟性が低くなります。
- » **俊敏性**：非仮想化I/Oを使用するドメインのライブ・マイグレーションは実行できませんが、他のSPARCサーバーへのコールド・マイグレーションを実行できます。これにより、サービス中断を比較的強く抑えたままでワークロードを移行できます。移行先サーバーが同一のI/Oインフラストラクチャを提供する必要があるため、ゲスト・ドメイン・モデルや冗長ゲスト・ドメイン・モデルよりも若干複雑なプロセスになります。

セキュリティの 独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	非常に高い	高い	非常に高い	高い	中程度	中程度

» 図5. 冗長 SR-IOVドメイン・モデルの7つの特性のまとめ

ゲスト・ルート・ドメイン・モデル

» ゲスト・ルート・ドメインには、直接割り当てられたルート・コンプレックスが構成されているため、直接的なI/O所有権があります。複数の仮想ディスクやネットワーク・サービスを作成する必要がないため、これは運用上、もっとも単純なモデルです。これらのゲストはベアメタル・パフォーマンス・レベルで稼働し、完全に互いから独立しています。作成できるゲスト・ルート・ドメインの数は、使用できるルート・コンプレックスとPCIeスロットの数による制限を受けます。このモデルは、パフォーマンスの高い独立したドメインを少数必要とする環境に最適です。

図6と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：各ゲスト・ルート・ドメインは、独自のオペレーティング・システム・バージョンを持つ物理サーバーのように動作します。注意すべき点として、いずれかのゲスト・ルート・ドメインは制御ドメインの役割も果たすため、サーバー上のその他すべてのゲスト・ルート・ドメインの物理構成を操作できます。
- » **リソースの独立性**：各ゲスト・ルート・ドメインにはCPU、メモリ、I/Oの各専用リソースが含まれるため、リソースに対するアクセスが保証されます。それぞれのゲスト・ルート・ドメインは互いから完全に分離されています。各ゲスト・ルート・ドメインは、割り当てられたI/Oに基づいて、ブート・デバイスとネットワーク・リソースにアクセスすると予想されます。
- » **効率**：このモデルでは、すべてのドメインがベアメタル・パフォーマンス・レベルで動作し、追加のサービス・ドメインに割り当てる必要のあるリソースもないため、もっとも高いレベルの効率を実現します。ハイパーバイザと仮想化によるリソース・オーバーヘッドは発生しません。
- » **可用性**：ゲスト・ルート・ドメインはその他すべてのドメインから完全に独立しているため、物理サーバーと同様の可用性レベルを提供し、冗長性レベルは冗長I/Oカードなどの設置する物理構成に依存します。
- » **保守性**：サーバー内の各ゲスト・ルート・ドメインはその他のドメインに影響することなく、個別に保守作業を実施できます。また、I/Oコンポーネントに対して通常のホットスワップを実行できます。
- » **柔軟性**：ゲスト・ルート・ドメインはかなり高い柔軟性を備えています。CPUとメモリの割当てを動的にサイズ変更でき、比較的容易に作成と破棄を実行できます。ただし、ルート・コンプレックス・レベルで物理的にI/Oを割り当てる必要があるため、通常は比較的静的なI/O構成で運用されます。
- » **俊敏性**：ゲスト・ルート・ドメインは物理ドメイン(PDoms)や物理サーバーと同様に、システム間のマイグレーションを実行できません。

セキュリティの独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
非常に高い	非常に高い	極めて高い	非常に高い	高い	高い	低い

図6. ゲスト・ルート・ドメイン・モデルの7つの特性のまとめ

Oracle Solaris Zones

Oracle Solarisには、Oracle Solaris Zonesと呼ばれる仮想化機能が組み込まれています。この機能を使用すると、柔軟なソフトウェア定義の境界を使用して、ソフトウェア・アプリケーションやサービスを分離できます。ハイパー

バイザに基づく仮想化とは異なり、Oracle Solaris ZonesテクノロジーはOSレベルの仮想化を提供するため、複数の物理マシンというよりも複数のOSインスタンスのように見えます。Oracle Solaris Zonesテクノロジーを使用すると、単一のオペレーティング・システム・インスタンスから多数のプライベート実行環境を作成して、環境全体と個々のゾーンのリソースを完全に管理できます。Oracleソフトウェア・ライセンスに関して言えば、制限付きのCPU (capped-cpu) または専用CPU (dedicated-cpu) として構成されたOracle Solaris Zonesは、ハード・パーティションと見なされます。

OSの仮想化という性質上、Oracle Solaris Zonesテクノロジーはオーバーヘッドが非常に低く、低レイテンシの環境を提供します。このため、1つのシステム上に数百または数千のゾーンを作成できます。Oracle Solaris ZFSおよびネットワーク仮想化を完全に統合しているため、その他のVM実装では問題となりうる領域である、実行と保存のオーバーヘッドも低く抑えられます。Oracle Solaris Zonesテクノロジーは、I/Oに対してベアメタル・パフォーマンス・レベルに近いパフォーマンスを提供するため、ゾーンは卓越したI/Oパフォーマンスの提供に最適です。

Oracle Solaris 11は完全に仮想化されたネットワーク・レイヤーを提供します。仮想化されたネットワーク、ルーター、ファイアウォール、ネットワーク・インタフェース・カード (NIC) を使用して、データセンター全体のネットワーク・トポロジを単一OSインスタンス内に作成できます。これらの仮想化されたネットワーク・コンポーネントには、高い可観測性、セキュリティ、柔軟性、リソース管理機能が備わっています。これにより、柔軟性が高くなるとともに、一部の物理的なネットワーク・ハードウェアが不要になることで、コストが削減されます。ネットワーク仮想化ソフトウェアはサービス品質 (QoS) をサポートしているため、重要アプリケーションには適切な帯域幅が確保されます。

また、Oracle Solaris Zonesテクノロジーは、ゾーン内で古いOracle Solarisバージョンを実行する機能を提供します。この機能はブランド・ゾーンと呼ばれています。Oracle Solaris 10のグローバル・ゾーンを実行する場合、内部でOracle Solaris 8およびOracle Solaris 9のゾーンを実行できます。これにより、より最新のプラットフォームにレガシー・アプリケーションを簡単に統合できます。また、Oracle Solaris 11のグローバル・ゾーン上でOracle Solaris 10ゾーンを実行することで、Oracle Solaris 10のワークロードはOracle Solaris 11が持つネットワーク仮想化機能を活用できるようになります。

Oracle Solaris 11で利用できる不変ゾーンは、ゾーンを読み取り専用にすることでセキュリティ・レベルを向上します。これにより、悪意のある構成変更やルートキット型の攻撃による影響を防止できます。

Oracle Solaris 11.2のカーネル・ゾーン機能は新しいタイプのゾーンであり、OSレベルの仮想化によるメリットをすべて継承した上に、独自のカーネルを実行できるという追加のメリットがあります。これにより分離レベルが向上し、特に、1つのグローバル・ゾーン内で異なるOracle Solarisの更新を実行できるようになります。

また、Oracle Solaris ZonesはOracle Solaris DTraceと統合されています。DTraceは、アプリケーションとカーネルの両方の処理に対して動的な計測とトレースを提供するOracle Solarisの機能です。たとえば、管理者はDTraceを使用して、ソフトウェア・スタック全体でJavaアプリケーションのパフォーマンスを調査できます。DTraceにより、Oracle Solaris Zones内とグローバル・ゾーン内の両方が可視化されるため、ボトルネックを特定および解消し、パフォーマンスを最適化することが容易になります。

注目すべきは、特定のグローバル・ゾーン (論理ドメイン/物理ドメイン/サーバー上で実行中の基盤Oracle Solaris インスタンス) で、次に示す任意のゾーン・タイプを同時にホストできるという点です。これにより、ワークロード要件に合わせて適切なゾーン・タイプを選択できます。

Oracle Solaris Zonesについて、詳しくは[Oracle Solaris 11 Virtualization Technologyのページ](#)を参照してください。

ネイティブ・ブランド・ゾーン

本書では、ネイティブ・ブランド・ゾーンという用語は、そこで動作しているグローバル・ゾーンと同じOracle

Solarisバージョンを実行するゾーンを指します。このゾーンではグローバル・ゾーン内で実行するカーネルを直接使用するため、ゾーン内のアプリケーションのパフォーマンスは、グローバル・ゾーン内で直接実行する場合と変わりなく、オーバーヘッドがもっとも低くなります。ネイティブ・ブランド・ゾーンは、セキュリティとパフォーマンスが分離された、アプリケーションのプライベート実行環境を提供します。

同じカーネルを共有するため、グローバル・ゾーンの更新時にはすべてのネイティブ・ブランド・ゾーンも更新されます。このため、運用上の複雑さがいくらか増すとともに、単一システム内で同時に複数のOS更新レベルを実行できる機能が制限されます。この問題は通常、停止による実質的な影響がないスケラブルなアプリケーション・アーキテクチャを使用するか、または更新前に一部（または全部）のゾーンを移行することで軽減されます。

図7と次のリストに、前述した7つの特性に対してネイティブ・ブランド・ゾーンがどのように対応するかを示します。

- » **リソースの独立性** : Oracle Solaris Zonesは、極めて柔軟なリソース制御機能を幅広く提供します。CPUリソースの制御については、専用CPU(dedicated-cpu)を指定するか、制限付きのCPU(capped-cpu)を使用するか、または公平配分スケジューラ(FSS: Fair Share Scheduler)を使用し、重み付けされたシステムを使用して使用可能なCPUをゾーン間で共有できます。また、rcapを使用して、ゾーン内のメモリ使用を制限することもできます。ネットワーク使用は、各リンクに対する帯域幅設定を使用して制御できます。
- » **セキュリティの独立性** : Oracle Solaris Zonesの各ゾーンはその他のゾーンから管理上分離されており、必要に応じて変更不可能にすることもできるため、極めて高度なセキュリティを提供します。
- » **効率** : Oracle Solaris Zonesは極めて軽量であり、仮想化オーバーヘッドはほとんど発生しません。
- » **可用性** : Oracle Solaris Zonesは、グローバル・ゾーンで実行される基盤のOracle Solarisインスタンスとして使用できます。多数のゾーンがデプロイされるシステムでは通常、高水準のRASと冗長I/Oを備えた基盤システムを構成します。ただし、すべてのゾーンが基盤のグローバル・ゾーンに完全に依存するため、グローバル・ゾーンで障害が発生すると、ホストされているすべてのゾーンにも障害が発生します。
- » **保守性** : ネイティブ・ブランド・ゾーンのOracle Solaris Zonesは、グローバル・ゾーンと同時に更新されません。同じOSバージョンを実行するグローバル・ゾーン間では、ゾーンの移行が可能です。
- » **柔軟性** : Oracle Solaris Zonesは高い柔軟性を備えています。動的なサイズ変更、作成、破棄を極めて容易に実施できます。Oracle Solaris Zonesは、あらかじめインスタンス化されたOS上のプロセス・セットであるため、迅速に開始および停止できます。
- » **俊敏性** : このモデルのOracle Solaris Zonesは、SPARCサーバー上で稼働するその他のグローバル・ゾーンに対してコールド・マイグレーションを実行できます。これにより、サービス中断を比較的強く抑えたままでワークロードを移行できます。ゾーンの再起動にかかる時間は、物理サーバーまたはVMと比べて大幅に迅速です。

セキュリティの独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	高い	非常に高い	中程度	低い	非常に高い	中程度

図7. ネイティブ・ブランド・ゾーンの7つの特性のまとめ

カーネル・ゾーン

Oracle Solaris 11.2の新機能であるカーネル・ゾーンは、それぞれが独自のカーネルを実行します。これにより、システム上でOracle Solaris 11.2の異なる更新を実行できるようになり、それぞれのカーネル・ゾーンを個別に更新できます。

図7と次のリストに、前述した7つの特性に対してカーネル・ゾーンがどのように対応するかを示します。

- » **リソースの独立性**：カーネル・ゾーンは標準ゾーンと同じリソース割当て機能を提供しますが、各ゾーンに専用メモリを割り当てる機能が追加されています。
- » **セキュリティの独立性**：Oracle Solaris Zones の各ゾーンはその他のゾーンから管理上分離されているため、極めて高度なセキュリティを提供します。
- » **効率**：カーネル・ゾーンは軽量であり、仮想化オーバーヘッドはほとんど発生しません。各カーネル・ゾーンが独自のカーネルを実行するため、インスタンスあたりで、従来のゾーンよりも多くのメモリを必要とします。これにより、従来のゾーンと比べるとオーバーヘッドが高くなります。
- » **可用性**：カーネル・ゾーンは、グローバル・ゾーンで実行されるOracle Solarisインスタンスと同じように使用できます。多数のゾーンがデプロイされるシステムでは通常、高水準のRASと冗長I/Oを備えた基盤システム上に構成します。ただし、すべてのゾーンがそこで動作するグローバル・ゾーンに完全に依存するため、グローバル・ゾーンで障害が発生すると、ホストされているすべてのゾーンにも障害が発生します。
- » **保守性**：カーネル・ゾーンはそこで動作するグローバル・ゾーンとは無関係に更新できます。言うまでもありませんが、更新後のグローバル・ゾーンを再起動するときは、カーネル・ゾーンも同様に再起動が必要になります。同じOSバージョンを実行するグローバル・ゾーン間でゾーンを移行することは可能です。カーネル・ゾーンは、あるサーバー上でゾーンを一時停止して、別のサーバー上で再開できるウォーム・マイグレーション機能を提供しています。
- » **柔軟性**：Oracle Solaris Zonesは高い柔軟性を備えています。動的なサイズ変更、作成、破棄を極めて容易に実施できます。Oracle Solaris Zonesは、あらかじめインスタンス化されたOS上のプロセス・セットであるため、極めて迅速に開始および停止できます。
- » **俊敏性**：このモデルのOracle Solaris Zonesは、SPARCサーバー上で稼働するその他のグローバル・ゾーンに対してウォーム・マイグレーションを実行できます。これにより、サービス中断を比較的強く抑えたままでワークロードを移行できます。ゾーンの再起動にかかる時間は、物理サーバーまたはVMと比べて大幅に迅速です。

セキュリティの 独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	非常に高い	高い	中程度	中程度	非常に高い	高い

図8. カーネル・ゾーンの7つの特性のまとめ

非ネイティブ・ブランド・ゾーン

非ネイティブ・ブランド・ゾーンという用語は、グローバル・ゾーンで実行されるバージョンよりも古いOracle Solarisバージョンを実行するゾーンを指します。たとえば、Oracle Solaris 10では、Oracle Solaris 8およびOracle Solaris 9のブランド・ゾーンを実行でき、Oracle Solaris 11では、Oracle Solaris 10のブランド・ゾーンを実行できます。これ以外の点では、非ネイティブ・ブランド・ゾーンは従来のゾーンと同じ特性を持ちます。

図9と次のリストに、前述した7つの特性に対して非ネイティブ・ブランド・ゾーンがどのように対応するかを示します。

- » **リソースの独立性**：Oracle Solaris Zonesは、極めて柔軟なリソース制御機能を幅広く提供します。CPUリソースの制御については、専用CPU (dedicated-cpu) を指定するか、制限付きのCPU (capped-cpu) を使用するか、または公平配分スケジューラ (FSS) を使用し、重み付けされたシステムを使用して使用可能なCPUをゾーン間で共有

”できます。また、rcapを使用して、ゾーン内のメモリ使用を制限することもできます。ネットワーク使用は、各リンクに対する帯域幅設定を使用して制御できます。

- » **セキュリティの独立性** : Oracle Solaris Zonesの各ゾーンはその他のゾーンから管理上分離されているため、極めて高度なセキュリティを提供します。
- » **効率** : Oracle Solaris Zonesは極めて軽量であり、仮想化オーバーヘッドはほとんど発生しません。
- » **可用性** : ゾーンは、グローバル・ゾーンで実行されるOracle Solarisインスタンスと同じように使用できます。多数のゾーンがデプロイされるシステムでは通常、高水準のRASと冗長I/Oを備えた基盤システム上に構成します。ただし、すべてのゾーンがそこで動作するグローバル・ゾーンに完全に依存するため、グローバル・ゾーンで障害が発生すると、ホストされているすべてのゾーンにも障害が発生します。
- » **保守性** : 非ネイティブ・ブランド・ゾーンは、グローバル・ゾーンと同時に更新されます。同じOSバージョンを実行するグローバル・ゾーン間では、ゾーンのマイグレーションが可能です。
- » **柔軟性** : Oracle Solaris Zonesは高い柔軟性を備えています。動的なサイズ変更、作成、破棄を極めて容易に実施できます。Oracle Solaris Zonesは、あらかじめインスタンス化されたOS上のプロセス・セットであるため、極めて迅速に開始および停止できます。
- » **俊敏性** : このモデルのゾーンは、SPARCサーバー上で稼働するその他のグローバル・ゾーンに対してコールド・マイグレーションを実行できます。これにより、サービス中断を比較的強く抑えたままでワークロードを移行できます。ゾーンの再起動にかかる時間は、物理サーバーまたはVMと比べて大幅に迅速です。

セキュリティの独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	高い	非常に高い	中程度	低い	非常に高い	中程度

図9. 非ネイティブ・ブランド・ゾーンの7つの特性のまとめ

仮想化テクノロジーの組合せ

表1に示すとおり、すべての特性に対して最高のスコアを持つ単一仮想化テクノロジーは存在しません。これは、多くの特性が相互排他的な性質を持つためです。

表1: すべての仮想化テクノロジーに対する7つの特性のまとめ

	セキュリティ の独立性	リソースの 独立性	効率	可用性	保守性	柔軟性	俊敏性
フル・システム	極めて高い	極めて高い	極めて高い	極めて高い	極めて高い	低い	低い
物理ドメイン(PDoms)	非常に高い	非常に高い	極めて高い	極めて高い	極めて高い	低い	低い
Oracle VM ゲスト・ルート・ドメイン・モデル	非常に高い	非常に高い	極めて高い	非常に高い	高い	高い	低い
Oracle VM 冗長ゲスト・ドメイン・モデル	高い	非常に高い	中程度	非常に高い	高い	高い	高い
Oracle VM SR-IOVまたは 直接I/Oドメイン・モデル	高い	非常に高い	高い	中程度	低い	中程度	中程度
Oracle VM 冗長SR-IOVドメイン・モデル	高い	非常に高い	高い	非常に高い	高い	中程度	中程度
Oracle VM ゲスト・ドメイン・モデル	高い	非常に高い	中程度	中程度	中程度	高い	高い
Oracle Solarisカーネル・ゾーン	高い	非常に高い	高い	中程度	中程度	非常に高い	高い
Oracle Solaris Zones	高い	高い	非常に高い	中程度	低い	非常に高い	中程度
Oracle Solaris ブランド・ゾーン	高い	高い	非常に高い	中程度	低い	非常に高い	中程度

テクノロジーごとの特性を確認したことで、2つ以上のOracle仮想化テクノロジーを組み合わせる階層化アプローチを集約に対して採用すると、1つの仮想化テクノロジーが満たす要件よりも大幅に多くの要件を満たすソリューションを作成できることが明らかです。

このセクションの目的上、物理サーバーと物理ドメイン (PDoms) を基本的に同等として扱います。物理ドメインの構成とサイジングに関する考察が当てはまるのは、SPARC M7サーバーとSPARC M6-32サーバーのみです。また、物理ドメインの数とサイズはおもに、ワークロードのサイジング要件と保守性要件に基づいて決定されます。

16ソケットのSPARC M7-16サーバーにおける選択肢は、16ソケットの物理ドメインを1つ、8ソケットの物理ドメインを2つ、4ソケットの物理ドメインを4つ、4ソケットの物理ドメインを2つと8ソケットの物理ドメインを1つのいずれかとなります。SPARC M7-8サーバーの場合、2~8ソケットの物理ドメインを1つあるいは1対の物理ドメイン(各々2~4ソケットを選択可)を工場出荷時に選択可能です。

一般的に物理ドメインに関連する仮想化オーバーヘッドは発生しませんが、サイジングに関しては、保守性や複雑さへの要件から判断する必要があります。物理ドメインを構成すると、それらの物理ドメインは独立した物理サーバーと同様に扱うことができます。

集約アーキテクチャを定義するための次のステップは、論理ドメイン (LDoms) またはOracle Solaris Zones、も



しくはこれらの組合せを使用して、物理ドメインまたはSPARCサーバー内のワークロードをデプロイする最適な方法を選択することです。

デプロイに必要なワークロードが64個ある場合に実施できる方法はいくつかあります。極端に言えば、1つの論理ドメイン内に64個のゾーンを構成することも、その反対に64個の論理ドメインを構成して、論理ドメインあたり1つのワークロードを割り当てることもできます。

1つの大きい論理ドメイン内の単一グローバルOSインスタンス上で64個のゾーンを実行するソリューションは、もっとも高い効率を提供し、構成および管理するエンティティの数が大幅に減ります。ただし、多数のゾーンがあると保守性に影響が及ぶ可能性があり、ワークロードに対する管理、セキュリティ、リソースが十分に分離されない場合があります。64個の論理ドメインをデプロイする場合、それぞれの論理ドメインに独自のブート・ディスクと冗長化された仮想ネットワークが必要になります。また、論理ドメインを64個の独立したサーバーとして更新および管理する必要があるため、インフラストラクチャ運用の複雑さに影響が及びます。

最善の選択肢は、2つのソリューションの間のどこかにある可能性がもっとも高いでしょう。

ここからは、潜在的に有効な組合せのサブセットについて概説し、それぞれの効果を組み合わせることで、異なるワークロード要件をどのように満たせるかを示します。次に挙げる例だけにはとどまらず、他にも複合組合せなど多数の組合せが存在します。また、デプロイの容易な複合組合せもあります。

冗長ゲスト・ドメインとOracle Solaris Zones

図10に示した仮想化テクノロジーの組合せは、Oracle Solaris Zonesを使用して、OSレベルで各論理ドメインを仮想化することで、作成する必要がある論理ドメインの数を削減し、論理ドメインごとに複数の分離ワークロードを実行できるようにします。これにより、論理ドメインによるおもな分離のメリットと、サーバー間での論理ドメイン全体のライブ・マイグレーション機能を維持したままで、論理ドメイン構成を簡素化し、運用上の複雑さを軽減できます。このモデルでは、前述した3つのゾーン・タイプのいずれもデプロイ可能であり、ワークロード要件によってゾーン・タイプを選択できます。

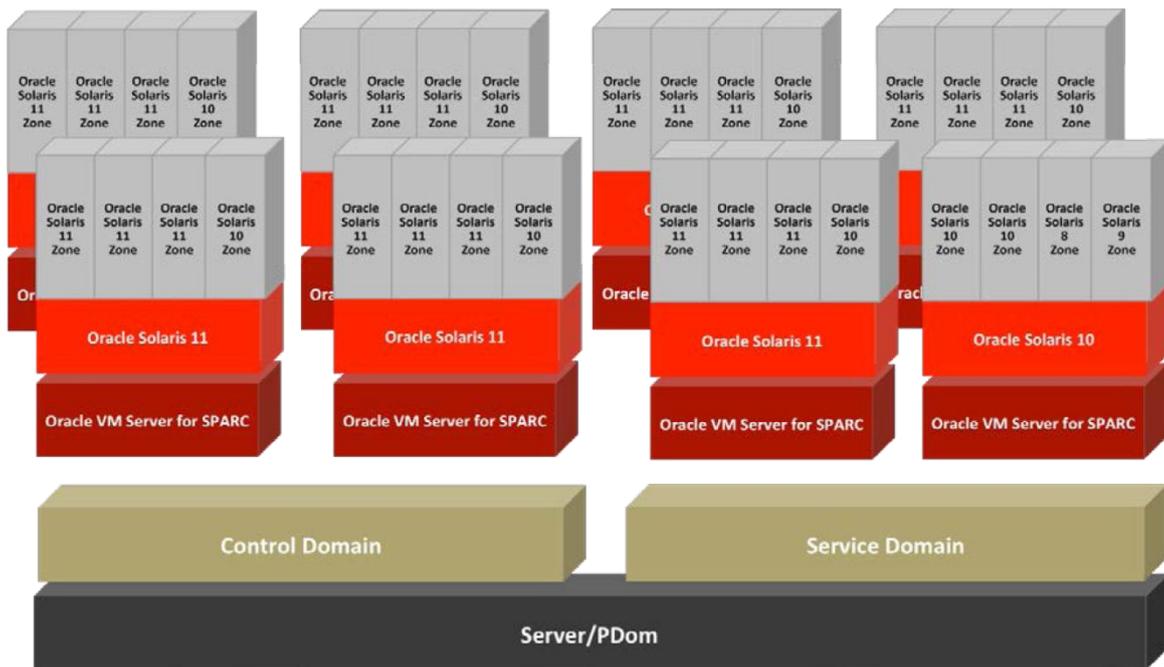


図10. 冗長ゲスト・ドメインとOracle Solaris Zonesを組み合わせた構成例

図11と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：このモデルには2つのセキュリティ分離レベルがあります。論理ドメイン・レベルでは、論理ドメイン間に高度な分離が存在しており、論理ドメイン内で実行されるゾーン間では、さらなる分離レベルが存在します。同じ論理ドメイン内にある同じセキュリティ・レベル内のゾーンを通常グループ化します。
- » **リソースの独立性**：前述のとおり、各ゲスト・ドメインには専用のCPUリソースとメモリ・リソースが含まれるため、リソースに対するアクセスが保証されます。ただし、論理ドメイン内で実行されるゾーン間では、これらのリソースをさらに分割して、より動的な方法で共有できます。
- » **効率**：ゾーンを使用してドメイン内のワークロード分離レベルを上げることで、作成する必要があるゲスト・ドメイン数を減らせるため、このモデルの効率は冗長ゲスト・ドメインのみのモデルよりも事実上高くなります。
- » **可用性**：I/Oサービスを提供するドメインのいずれかが停止した場合も利用できるという観点から言えば、このモデルはドメインの可用性を維持します。
- » **保守性**：実行中のゲストに影響を与えることなく制御ドメインとサービス・ドメインで保守作業を実施できるため、高水準の保守性は複数サービス・ドメインを使用する方式のおもなメリットの1つとなっています。

- » **柔軟性**：ゲスト・ドメインは非常に柔軟性に優れています。動的な作成、サイズ変更、破棄を比較的容易に実施できます。ゾーンは動的に作成、サイズ変更、破棄できるため、論理ドメイン内のゾーンによる追加のメリットとして、さらにきめ細かいレベルの柔軟性が得られます。
- » **俊敏性**：ゲスト・ドメインはSPARCサーバーに対して、ライブまたはコールド・マイグレーションを実行できます。これにより、サービス中断をごくわずかに抑えるか、またはまったく発生させることなく、ワークロードを移行できます。もう1つの方法として、必要に応じて単一ゾーンをゲスト・ドメイン間で移行することもできます。

セキュリティの 独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
高い	非常に高い	高い	非常に高い	高い	非常に高い	高い

図11. 冗長ゲスト・ドメインとOracle Solaris Zonesを組み合わせた構成に対する7つの特性のまとめ

ゲスト・ルート・ドメインとOracle Solaris Zones

図12に示すモデルは、仮想化オーバーヘッドがなく、高度な分離を実現するゲスト・ルート・ドメインと、柔軟性と俊敏性を備えたOracle Solaris Zonesを組み合わせたものであり、オーバーヘッドを最小レベルに抑えながら、可能な限り最高の効率を提供します。特定のワークロードに対してカーネル・ゾーンを使用することで、特にメモリ・リソース管理の分野におけるゾーン・レベルでの追加の分離が可能になり、ゾーン・ワークロードのウォーム・マイグレーションも実行できるようになります。このモデルには通常、多数のゾーンを実行する少数の論理ドメインが含まれます。

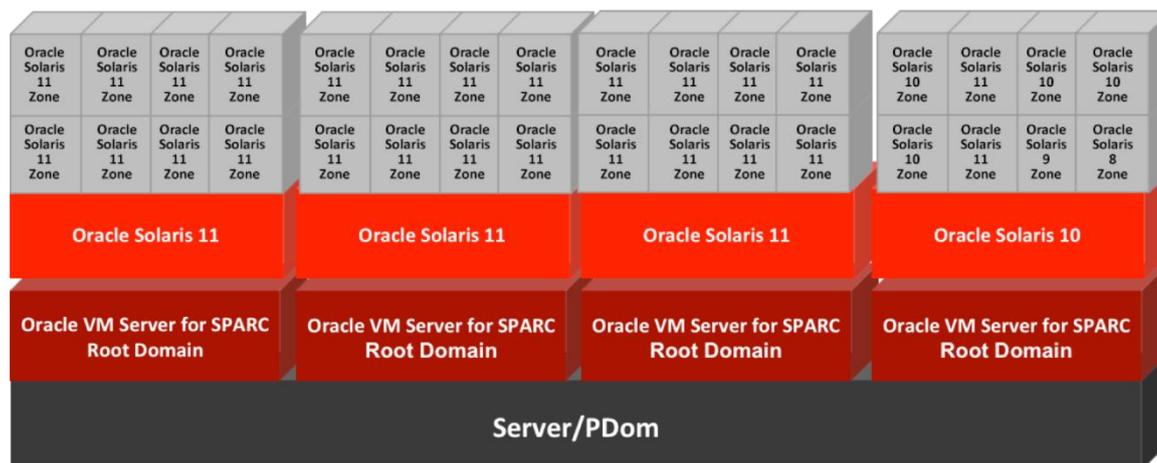


図12. ゲスト・ルート・ドメインとOracle Solaris Zonesを組み合わせた構成例

図13と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：前のモデルの例とは違い、ゲスト・ルート・ドメインは専用ハードウェア上で実行されますが、このモデルにも2つのセキュリティ分離レベルが存在します。論理ドメイン・レベルでは、論理ドメイン間に高度な分離が存在しており、論理ドメイン内で実行されるゾーン間では、さらなる分離レベルが存在します。同じ論理ドメイン内にある同じセキュリティ・レベル内のゾーンを通常グループ化します。

- » **リソースの独立性**：前述のとおり、各ゲスト・ルート論理ドメインにはCPU、メモリ、I/Oの各専用リソースが含まれるため、リソースに対するアクセスが保証されます。ただし、論理ドメイン内で実行されるゾーン間では、これらのリソースをさらに分割して、より動的な方法で共有できます。
- » **効率**：ハイパーバイザによるオーバーヘッドは発生せず、仮想化オーバーヘッドもないため、このモデルはもっとも効率の高いモデルです。また、管理対象エンティティの数をもっとも少ないため、運用上もっともシンプルなモデルでもあります。このように、このモデルは前述した3つの仮想化税すべてを回避しています。
- » **可用性**：それぞれのゲスト・ルート論理ドメインは完全に独立したドメインであり、冗長I/Oが構成されている場合は、前述の冗長ゲスト・ドメイン・モデルよりも高い可用性を提供します。言うまでもありませんが、ゾーンはこれまでと同様に、そこで動作するグローバル・ゾーンとゲスト・ルート・ドメインに完全に依存します。
- » **保守性**：このモデルでは、冗長I/O接続により、ルート・ドメイン自体が一定の保守性を有しています。ただし、多くのケースで、作業を実施するためには、通常、ゲスト・ルート・ドメインのサービスを停止します。ゾーンを使用することで、これらのワークロードをよりきめ細かく移行できるようになります。
- » **柔軟性**：ゲスト論理ドメインは非常に柔軟性に優れています。動的なサイズ変更、作成、破棄を比較的容易に実施できます。論理ドメイン内のゾーンによる追加のメリットは、より詳細レベルの柔軟性です。
- » **俊敏性**：ルート・ドメイン自体をサーバー間で移行することはできませんが、Oracle Solaris Zonesを追加することで、これらのワークロードを容易にルート・ゲスト・ドメインから移行できるようになります。カーネル・ゾーンの一時停止機能と再開機能を使用すると、ウォーム・マイグレーションを実行できます。

セキュリティの 独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
非常に高い	非常に高い	極めて高い	非常に高い	高い	非常に高い	高い

図13. ゲスト・ルート・ドメインとOracle Solaris Zonesを組み合わせた構成に対する7つの特性のまとめ

ルート・ドメインとSR-IOVドメイン

このメカニズムは両方のレイヤーに対して同じテクノロジーを使用していますが (Oracle VM Server for SPARC)、もう1つの便利なモデルを提供します。ゲスト・ルート・ドメインとOracle Solaris Zonesを組み合わせたモデルとよく似ていますが、多数のSR-IOVドメイン・ゲストに対して、単純にSR-IOVサーバーとしてルート・ドメインを使用します。ルート・ドメイン内で実際に実行されるアプリケーション・ワークロードはないため、ルート・ドメインはゲスト・ルート・ドメインとは異なります。ルート・ドメインには大量のI/Oが割り当てられますが、CPUとメモリの割当て量は中程度です。(図12に示すように、) これらのゲスト上にOracle Solaris Zonesレイヤーを重ねることも可能ですが、ここでは分かりやすくするため、2層モデルのみについて考察します。

SR-IOVドメインは制御ドメインに依存しているため、単一サーバー内に複数の障害ドメインを作成することが重要になります。こうすることで、サービス・ドメインの障害による影響が、サービスを提供しているゲストのみに限定されます。また、ゲスト・ドメインに対してレジリエントな I/O サービスを提供するために、ルート・ドメインはペアでの構成が推奨されます。

このモデルは、単一システム内で比較的多数のSR-IOVドメインを使用できるようにすることで、卓越したパフォーマンス・レベルを維持しながら最高レベルの分離を実現します。このモデルの可用性特性が前のモデルと似ているのは、両方のケースでルート・ドメインに依存しているためです (ゾーンまたはSR-IOVゲストのホスティング)。

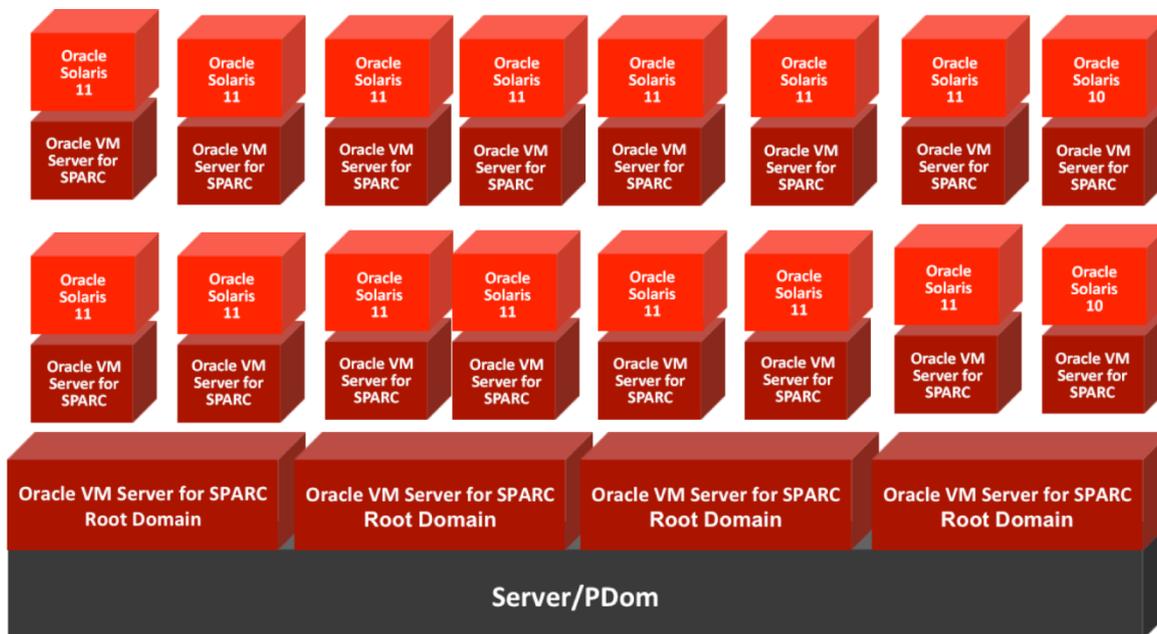


図14. ルート・ドメインとSR-IOVゲストを組み合わせた構成例

図15と次のリストに、前述した7つの特性に対してこのモデルがどのように対応するかを示します。

- » **セキュリティの独立性**：各ゲスト・ドメインは、独自のオペレーティング・システム・バージョンを持つ物理サーバーのように動作し、単ルート・ドメインが提供するサービスを利用します。同じセキュリティ・レルム内にあるSR-IOVゲスト・ドメインは、同じルート・ドメインが提供するサービスを利用できます。
- » **リソースの独立性**：各ゲスト・ドメインには専用のCPUリソースとメモリ・リソースが含まれるため、リソースに対するアクセスが保証されます。SR-IOVを使用することで、各ゲスト・ドメインは、複数の仮想接続を直接サポートする物理デバイスに直接アクセスできるようになります。何らかのリソース競合が発生する可能性はありますが、同じ共有デバイス上でワークロードがネイティブ実行されている場合を上回ることはありません。
- » **効率**：Oracle VM Server for SPARCテクノロジーを使用すると、ゲスト・ドメインはCPU、メモリ、I/Oの各リソースを直接利用でき、仮想化オーバーヘッドは発生しません。ただし、このモデルでは、サービス・ドメインに対してリソースを割り当てる必要がありますが、仮想化I/Oを使用するモデルよりも必要なリソースは少なくなります。要約すると、このモデルでは仮想化リソースのオーバーヘッドが全面的に削減されますが、ハイパーバイザ・オーバーヘッドは引き続き発生します。
- » **可用性**：SR-IOVを使用する場合、物理デバイスを所有するドメインへの依存が発生します。このドメインで計画外の障害が発生すると、デバイスに依存するゲストにも障害の影響が及び、不定の結果（通常はパニック）がもたらされます。SR-IOVサービスを提供する複数のドメインをゲスト・ドメイン・グループに追加すると、単一障害による影響を軽減できます。
- » **保守性**：物理デバイスを所有するドメインで保守作業を実施する必要がある場合、ゲスト・ドメインも含めて停止するか、またはゲスト・ドメインを別のサーバーに移行する必要があります。ただし、ゲスト・ドメインでは、その他のゲスト・ドメインに影響を与えることなく保守作業を実施できます。ルート・ドメインでの保守作業の影響が及ぶのは、これらのルート・ドメインのサービスを利用しているゲストのみです。
- » **柔軟性**：ゲスト・ドメインは非常に柔軟性に優れています。動的なサイズ変更、作成、破棄を比較的容易に実

施できます。

- » **俊敏性**：非仮想化I/Oを持つゲスト・ドメインは、その他のSPARCサーバーに対するコールド・マイグレーションを実行できます。これにより、サービス中断を比較的強く抑えたままでワークロードを移行できます。

セキュリティの独立性	リソースの独立性	効率	可用性	保守性	柔軟性	俊敏性
非常に高い	非常に高い	高い	高い	高い	非常に高い	中程度

図15. ルート・ドメインとSR-I/Oドメインを組み合わせた構成に対する7つの特性のまとめ

すべてのOracle仮想化テクノロジーの複合組合せ

Oracle仮想化テクノロジーの柔軟性は、相互に排他的でないということです。つまり、すべてのOracle Solarisインスタンス上で任意のタイプのゾーンをいつでも実行できます。また、Oracle VM Server for SPARCテクノロジーを使用することで、システム内のルート・ドメインごとに異なる機能を実行できます。これにより、幅広いワークロード・タイプに対応できる最大の柔軟性が提供されます。

図16に示す最初の2つのルート・ドメインは、冗長化されたドメイン・ペアとして4つの冗長ゲスト・ドメインにI/Oサービスを提供し (A)、3番目のドメインは、ルート・ドメインとして4つのゲストSR-I/OドメインにSR-I/Oサービスを提供します (B)。最後に、4番目のルート・ドメインは単純にOracle Solaris Zonesモデルを使用しています (C)。

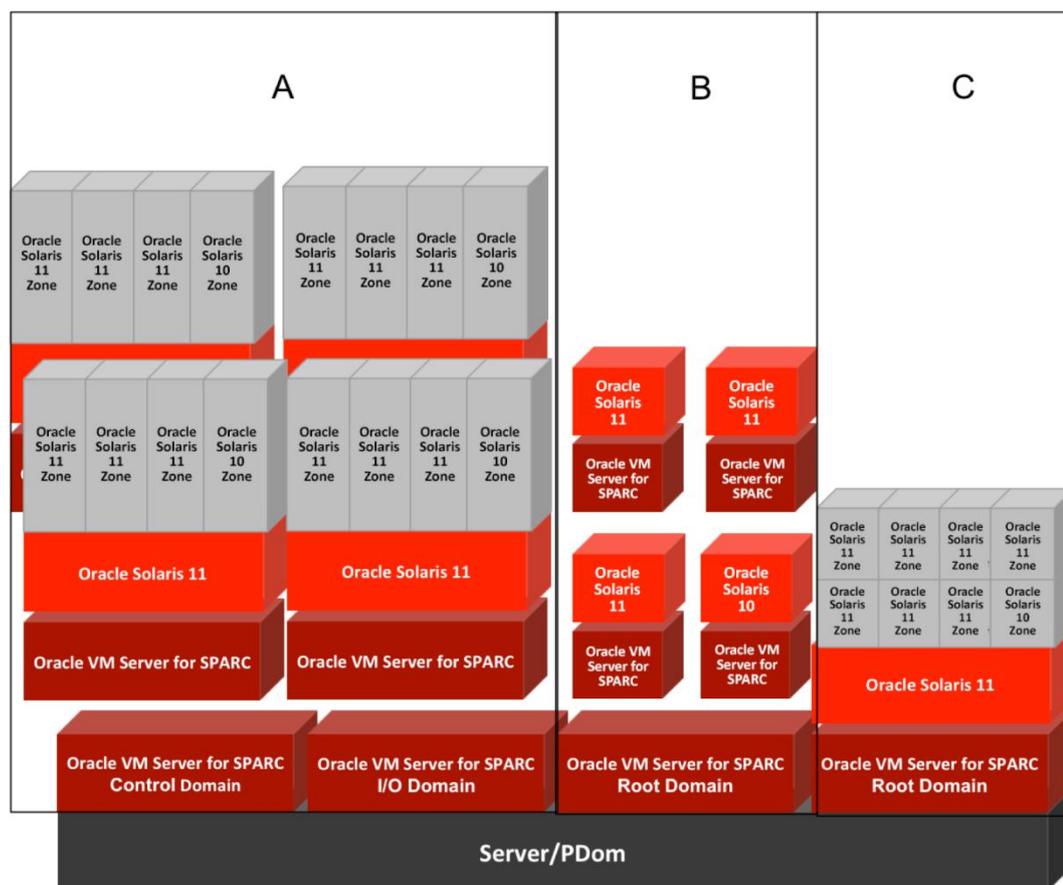


図16. すべてのOracle仮想化テクノロジーを組み合わせた複合組合せの構成例

図から分るとおり、1つのサーバー/物理ドメイン内に各モデルを組み込むことが可能であり、モデルの仮想化特

性は前述した組合せモデルとまったく同一になります。

仮想化テクノロジーの組合せが持つ特性のまとめ

1つのテクノロジーのみを使用する場合と比べると、それぞれの組合せ方式が、すべての仮想化特性でより高いスコアを達成します。表2に示すとおり、仮想化に対して階層化アプローチを採用することで、より広範なワークロード要件を満たすアーキテクチャの構築が可能になります。

表2：組合せ仮想化モデルに対する7つの特性のまとめ

	セキュリティの 独立性	リソースの 独立性	効率	可用性	保守性	柔軟性	俊敏性
冗長ゲスト・ドメインとOracle Solaris Zones	高い	非常に高い	高い	非常に高い	高い	非常に高い	高い
ゲスト・ルート・ドメインとOracle Solaris Zones	非常に高い	非常に高い	極めて高い	非常に高い	高い	非常に高い	高い
ルート・ドメインとSR-IOVドメイン	非常に高い	非常に高い	高い	高い	高い	非常に高い	中程度

結論

本書では、それぞれのOracle仮想化テクノロジーの特性を明らかにしました。また、1つの仮想化テクノロジーをデプロイする代わりに、複数のテクノロジーを組み合わせることで、代表的なワークロード要件をより厳密に満たすアーキテクチャを構築できることを示しました。

1つのテクノロジーですべての要件を満たすことはできません。集約を成功に導く鍵となるのは、あらゆるタイプのワークロードに容易に対応できる柔軟なインフラストラクチャを構築することです。

また、コスト削減という、本来の集約の動機を見失わないことが重要です。コスト削減を可能にする要因は次のとおりです。

- » ハードウェアおよびソフトウェア・インフラストラクチャの規模の縮小
- » 効率の向上
- » 運用上の複雑さの軽減

物理ドメインとOracle VM Server for SPARC、および各種のOracle Solaris ZonesというOracle仮想化テクノロジーを組み合わせることで、上記の目標の達成を目的とする俊敏かつ柔軟なアーキテクチャを構築できるようになります。

Oracle Elite Engineering Exchangeについて

Elite Engineering Exchange (EEE) は部門の枠を越えたグローバル組織であり、オラクル選り抜きのセールス・コンサルタント (SC) とシステムズのエンジニア (Product Engineering) で構成されています。EEEは共同の取組みを通じて、お客様対応するセールス・コンサルタントの中にある最高の専門家とシステムズのエンジニアを直接つなげることで、顧客と市場傾向に関する双方向のコミュニケーションを可能にし、次世代製品の技術的方向性に対する深い知見をもたらします。EEEを通じて、実際の顧客体験が直接システムズのエンジニアに伝えられ、エンジニアリングに関する技術的な詳細事項や知見がセールス・コンサルタントに届けられることにより、オラクルの顧客のニーズの変化に対して、より効果的なソリューションを提供することができます。


ORACLE®

お問い合わせ窓口

Oracle Direct

TEL 0120-155-096

URL oracle.com/jp/direct

Oracle is committed to developing practices and products that help protect the environment

CONNECT WITH US

blogs.oracle.com/oraclefacebook.com/oracletwitter.com/oracleoracle.com

Copyright © 2015, Oracle and/or its affiliates. All rights reserved. 本文書は情報提供のみを目的として提供されており、ここに記載されている内容は予告なく変更されることがあります。本文書は、その内容に誤りがないことを保証するものではなく、また、口頭による明示的保証や法律による黙示的保証を含め、商品性ないし特定目的適合性に関する黙示的保証および条件などのいかなる保証および条件も提供するものではありません。オラクル社は本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクルの書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。

OracleおよびJavaはOracleおよびその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標です。

IntelおよびIntel XeonはIntel Corporationの商標または登録商標です。すべてのSPARC商標はライセンスに基づいて使用されるSPARC International, Inc.の商標または登録商標です。AMD、Opteron、AMDロゴおよびAMD Opteronロゴは、Advanced Micro Devicesの商標または登録商標です。UNIXは、The Open Groupの登録商標です。0914

Hardware and Software, Engineered to Work Together