



Oracleホワイト・ペーパー
2010年5月

ORACLE[®] VM SERVER FOR SPARC[®]を 使用してデータの信頼性を高めるための ベスト・プラクティス

はじめに	1
このホワイト・ペーパーについて	3
データの信頼性と可用性の概要	3
Oracle VM Server for SPARCを使用した内蔵ストレージの信頼性	4
OracleプラットフォームのI/Oの信頼性と可用性	6
例：オラクルのSun SPARC Enterprise T2000サーバーの I/Oアーキテクチャ	7
I/Oアーキテクチャの概要	8
Oracle VM Server for SPARCのインストールと構成	9
ハードウェアRAIDの構成	10
ハードウェアRAIDのディスク設定	10
ハードウェアRAIDの実装	11
論理ドメインの作成	13
I/OドメインでのOracle Solaris ZFSの構成	14
Oracle Solaris ZFSの実装とゲスト・ドメインのボリュームの クローン化	15
ゲスト・ドメインでのボリューム管理の構成	20
ネットワーク・インストールによるボリューム管理の設定	20
結論	24
著者について	25
謝辞	26
参考資料	26

はじめに

ディスクとネットワークI/Oを対象としたデータセンターのベスト・プラクティスは、比較的明確に定義されています。可用性を向上させるには、ネットワーク・リソースとストレージ・リソースに複数のデータ・パスを構成します。データの信頼性を向上させるには、ハードウェアRAIDやソフトウェアRAIDなどのメディアの冗長性、およびOracle[®] Solaris Zettabyte File System (Oracle Solaris ZFS) などの高度なファイル・システムを使用します。こういった方法は実環境のデータセンター・サーバーの構成やケーブル接続に活用できますが、仮想環境ではどのように適用すればよいのでしょうか。冗長な仮想ディスクとネットワークを使用するとは、どのようなことでしょうか。物理環境と同じレベルの可用性と信頼性のメリットを実現するためには、これらの仮想ディスクとネットワークをどのように利用すればよいのでしょうか。このホワイト・ペーパーでは、まずドメインとの関連でこれらの質問に回答します。

Oracle VM Server for SPARC[®] (旧名Sun Logical Domains) は、CoolThreadsテクノロジーを採用したオラクル・サーバーでサポートされている仮想化テクノロジーで、UltraSPARC[®] T1、T2、およびT2 Plus プロセッサを搭載しています。Oracle VM Server for SPARCでは、サーバー・リソースをパーティション化して仮想マシンに割り当てることができます。リソースはCPUスレッド、暗号化プロセッサ、メモリ、およびPCIバスに割り当てることができます。ドメインのアーキテクチャは、シン・ハイパーバイザー・レイヤー (図1) で実行される4つのクラスの仮想マシンで構成されます。単一の制御ドメインが仮想化環境を管理します。1つ以上のI/Oドメインが実際のI/Oデバイスを保有し、このI/Oデバイスは、サービス・ドメインがゲスト・ドメインに対して仮想I/Oサービスを提供するために使用されます。各ドメインは、サーバー・ハードウェアの独自の専用パーティションで動作します。1つのドメインが複数の役割を果たすことができます。たとえば、I/Oドメインとサービス・ドメインは通常、物理I/Oと仮想I/Oを処理する1つのドメインに統合されます。多くの構成では、制御ドメインはI/Oとサービスのドメインと統合されます。このホワイト・ペーパーでは、I/Oとサービスが統合されたドメインを分かりやすくI/Oドメインと呼んでいます。

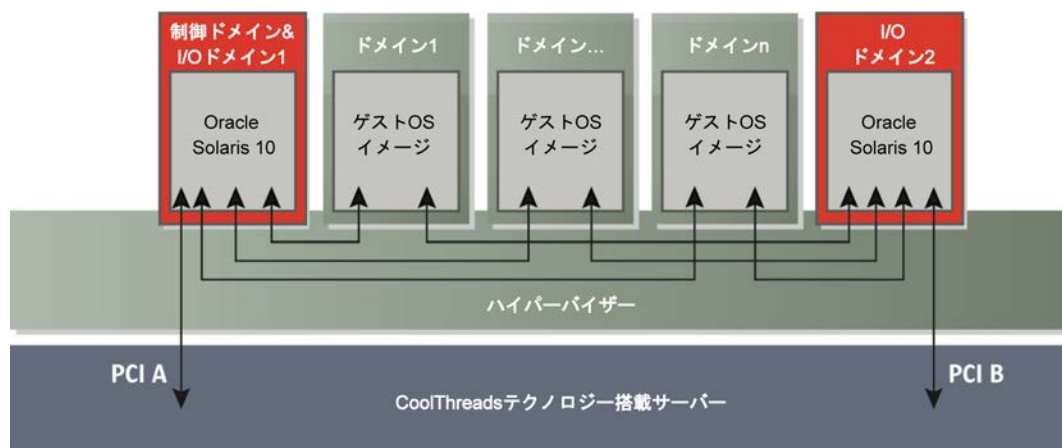


図1. Oracle VM Server for SPARCは、それぞれがサーバー・ハードウェアの独自のセキュア・パーティションを持つ複数のゲスト・ドメインをサポート。

I/Oは、1つ以上のPCIバスを所有するI/Oドメインにより処理

Oracle VM Server for SPARCを使用すると、各I/OドメインがPCIバスおよびPCIバスに接続されたデバイスを'所有'できるよう、個々のPCIルート連結 (nexus) ノード (このホワイト・ペーパーではPCIバスと呼ばれる) をI/Oドメインに割り当てることができます。複数のPCIバスを装備したサーバーでは、ゲスト・ドメインからI/Oリソースに複数のバスを提供するよう、複数のI/Oドメインを構成できます。I/Oドメイン、そのPCIバス、またはその周辺機器に障害が発生した場合や、ドメインを再起動する必要がある場合、ゲスト・ドメインを適切に構成していれば、ゲスト・ドメインでは2番目のI/Oドメインを使用してI/Oリソースに引き続きアクセスできます。

データの信頼性については、CoolThreadsテクノロジーを採用したオラクル・サーバーとOracle Solaris 10オペレーティング・システムを組み合わせることで、冗長性によって信頼性を強化するさまざまなツールを利用できます。サーバーのオンボードのディスク・ドライブとディスク・コントローラを使用してデータの信頼性を高めるために、これらのツールを3つの異なるアーキテクチャ・レベルで使用できます。

- サーバーの組み込みディスク・コントローラのRAID機能を使用して、ハードウェア・レベルの冗長性を確立できます。
- Oracle Solaris ZFSやOracle Solaris Volume Managerなどのソフトウェア機能を使用して、I/OドメインでI/Oの冗長性をサポートできます。I/Oドメインの信頼性を確立することで、ゲスト・ドメインに信頼性の高いストレージを提供できます。
- ゲスト・ドメインで同じソフトウェア機能を使用して、ゲスト自体に冗長性を実装できます。仮想ディスクを使用して、Oracle Solaris ZFSとOracle Solaris Volume Managerを論理ドメインに構成できます。

このテクニカル・ホワイト・ペーパーでは、サーバー自体のリソース、つまりサーバー自体のPCIバス、組み込みのディスク・コントローラ、およびディスク・ドライブを使用して、冗長性によってデータの信頼性を確立するためのアプローチとトレードオフについて説明します。Oracle VM Server for SPARCは、仮想環境にI/Oの可用性技術と信頼性技術を実装する可能性を広げるもので、このホワイト・ペーパーはこの広範なテーマに対処するシリーズの第一弾となります。当然のことながら、このホワイト・ペーパーで説明している技術にはすべてメリットとデメリットがあります。データセンター要件に対する各ソリューションのメリットを慎重に比較した上で、どの技術を実装するか選択する必要があります。

このホワイト・ペーパーについて

このホワイト・ペーパーでは、以下のトピックについて説明します。

- "データの可用性と信頼性の概要"では、Oracle VM Server for SPARCで使用できる可用性技術と信頼性技術の両方について概要を説明します。
- "OracleプラットフォームのI/Oの信頼性と可用性"では、サーバーの論理アーキテクチャに応じて、CoolThreadsテクノロジーを採用したさまざまなオラクル・サーバーでどの技術を使用できるのかを具体的に説明します。
- "Oracle VM Server for SPARCのインストールと構成"では、作業ドメイン環境を設定する方法について概要を簡単に説明します。
- "ハードウェアRAIDの構成"では、サーバーのディスク・コントローラを使用して信頼性の高いディスク・ストレージを構成するための段階的な手順について説明します。
- "I/OドメインでのOracle Solaris ZFSの構成"では、Oracle Solaris ZFSを使用して（特にドメインでスナップショットとクローンを用いて）信頼性を確立するメリットについて説明します。
- "ゲスト・ドメインでのボリューム管理の構成"では、Oracle Solaris JumpStartを使用してゲスト・ドメインでOracle Solaris Volume Managerを設定する方法について説明します。

データの信頼性と可用性の概要

このホワイト・ペーパーの目的は、オンボードのディスク・ストレージとOracle VM Server for SPARCを使用してデータの信頼性を高めることですが、信頼性と可用性は密接に関連しています。信頼性と可用性を実現するために適用できるアプローチは、基盤となるサーバー・アーキテクチャによって異なるため、この関連を理解することが重要です。このホワイト・ペーパーの目的に合わせて、ここでは以下のように定義しています。

- **信頼性**とは、データの損失が発生しないようにすることです。ハードウェアRAID、ボリューム・マネージャ（Oracle Solaris Volume Managerなど）に実装されたソフトウェアRAID、またはファイル・システム（Oracle Solaris ZFSなど）を使用して、ディスクに格納されたデータの冗長コピーを保存することで、これを実現します。この冗長性は、論理ドメインと内蔵ディスク・ストレージを使用して、ディスク・コントローラ自体、I/Oドメイン、およびゲスト・ドメインの3つのレベルで確立できます。
- **可用性**とは、できる限りデータにアクセスできるようにすることです。通常、1つのパスに障害が発生してもデータのアクセスに影響しないよう、データへのすべてのパスのうち少なくとも2つのパスを確立することで、可用性を向上します。論理ドメインでは、次のことが可能です。
 - ネットワーク・リソースへの複数のネットワーク接続、およびディスク・ストレージへの複数のPCIバスを構成できます。
 - 2つのI/Oドメインを設定することで、ゲスト・ドメインからI/Oリソースへのデュアル仮想データ・パスを確立できます。I/Oドメインでは専用の各PCIバス上にすべてのデバイスが存在するため、複数のI/Oドメインの使用は複数のPCIバスを装備したサーバーに限定されます。

- 複数のI/Oドメインでゲスト・ドメインのI/Oを処理すると、オペレーティング・システムにパッチを適用した後などに、ゲスト・ドメインでデータへの接続が切断されることなく管理者がI/Oドメインを再起動できるため、可用性が向上します。
- 複数のI/Oドメインを使用して、特定のI/Oドメインで使用されるリソースを他のI/Oドメインから分離できます。これにより、正常に動作していないゲスト・ドメインが1つのI/Oドメインのリソースをすべて使用してしまっても、他のゲスト・ドメインによるデータ・アクセスには影響を与えません。

次の項では、Oracle VM Server for SPARCを使用した信頼性の高いディスク・ストレージをさまざまなレベルで確立する方法について説明します。また、その次の項では、オラクルのCoolThreadsテクノロジーに対応したサーバーのアーキテクチャ、およびそれらのサーバーが可用性を実現するためのさまざまなアプローチをどのようにサポートしているのかについて説明します。

Oracle VM Server for SPARCを使用した内蔵ストレージの信頼性

CoolThreadsテクノロジーを採用したオラクル・サーバーで内蔵ディスク・ドライブを使用して信頼性の高いディスク・ストレージを確立するには、ハードウェアRAID、I/OドメインでのソフトウェアRAID、ゲスト・ドメインでのソフトウェアRAIDの3つのアプローチがあります。ここでは、これらのアプローチについて説明します。構成例については、このホワイト・ペーパーの後半で紹介しています。一般的に、使用するスタックが下位であるほど、RAID実装の効率性が高くなります。ディスク・コントローラに実装されるハードウェアRAIDでは、RAIDの実装によるCPUの負荷が軽減されるため、効率性も高くなります。ゲスト・ドメインに実装されるソフトウェアRAIDでは、すべての書き込みを少なくとも2つの仮想ディスクにレプリケートする必要があるため、書き込みごとにI/Oドメインを通過するというオーバーヘッドが発生します。

CoolThreadsテクノロジーを採用したオラクル・サーバーはすべて、内蔵ディスク・ドライブ用に単一のコントローラを装備しているため、1つのI/Oドメインのみを使用して内蔵ディスクにアクセスできます。そのため、内蔵ストレージへのアクセスでは、信頼性は可用性とは切り離して考慮できます。

ハードウェアRAID

CoolThreadsテクノロジーを採用したオラクル・サーバーはすべて、RAID 0（ストライプ化）とRAID 1（ミラー化）をサポートする内蔵RAIDコントローラを装備しています。RAID 1により、ディスクをペアにしてRAIDグループを構成して、ディスクのミラー化をハードウェアで処理できます。このアプローチは管理が容易で、I/Oドメインとゲスト・ドメインの両方に対して透過的です。

ハードウェアRAIDのミラー化ディスクを使用すると、次の3つの方法で、この信頼性の高いストレージをゲスト・ドメインに提供できます。

1. ミラー化したボリューム全体を1つの仮想ディスクとしてゲスト・ドメインに提供できます。このディスクをパーティション化して、1つのブート・ディスクとして使用できます。
2. ミラー化したディスクの1つのパーティションをゲスト・ドメインに提供できます。ゲスト・ドメインに提供されたパーティションは、仮想ディスクのようにパーティション化することはできません。

このことはOracle VM Server for SPARC 1.0.3
まで適用されますが、今後変更となる可能性があります。
3. ミラー化したディスクで作成したフラット・ファイルを1つの仮想ディスクとしてゲスト・ドメインに提供できます。このディスクをパーティション化して、1つのブート・ディスクとして使用できます。

もっとも一般的な方法は3番目の方法で、フラット・ファイルを使用して1つの仮想ディスクとしてゲスト・ドメインに提供するものです。この方法では、1つのミラー化ディスクに多くの仮想ディスクを作成し、これらの仮想ディスクをパーティション化してゲスト・ドメインからブートできるためです。ゲスト・ドメインごとにミラー化したディスク・セットを1つ割り当てる1番目の方法では、大部分のシステムで、利用可能なドメインの数が大幅に限られてしまいます。ミラー化したディスクのパーティションを割り当てる2番目の方法では、一定の数のパーティションしか物理ディスクに作成できないため、作成できる仮想ディスクの数が限られてしまいます。また、これらの仮想ディスクを論理ドメインのブート・ディスクとして使用することはできません。

I/Oドメインの信頼性

I/Oドメインでは、ソフトウェアRAIDを使用してディスク・ストレージの信頼性を確立できます。ソフトウェアRAIDによって、信頼性の高い仮想ディスクがゲスト・ドメインに提供されます。ハードウェアRAIDと同様に、このアプローチでは、ゲスト・ドメインには詳細が表示されずに、ゲスト・ドメインに信頼性の高いディスク・ストレージが提供されます。この信頼性の高いストレージは、I/Oドメイン自体から一元的に管理されます。Oracle Solaris 10に含まれているソフトウェアを使用した、次の2つのアプローチを利用できます。

- **Oracle Solaris Volume Manager**を使用して、ボリューム全体を仮想ディスクとしてエクスポートする方法、およびフラット・ファイルをボリュームに作成して仮想ディスクとしてエクスポートする方法で、仮想ディスクとしてゲスト・ドメインに提供するRAIDセットを作成できます。繰り返しになりますが、もっとも一般的なアプローチは、パーティション化可能な仮想ディスクをゲスト・ドメインに提供するフラット・ファイルを作成する方法です。
- **Oracle Solaris ZFS**では、ボリューム管理とファイル・システム・テクノロジーが1つのインタフェースに組み込まれており、このインタフェースで両方を管理できます。レプリケーションを確立するOracle Solaris ZFSボリューム管理機能を使用して、Oracle Solaris ZFSプールにディスクを割り当てることができます。Oracle Solaris ZFSでは、ミラー化およびRAID-Z (RAID 5と同様)をサポートしています。Oracle Solaris ZFSを使用したもっとも一般的なアプローチは、パーティション化してブート・ディスクとして使用できるよう、フラット・ファイルをゲスト・ドメインに割り当てるというものです。Oracle VM Server for SPARCバージョン1.0.3では、Oracle Solaris ZFSファイル・システムをゲスト・ドメインに提供できます。

Oracle Solaris Volume ManagerとハードウェアRAIDのいずれかを選択する場合は、ハードウェアRAIDのほうがより透過的な選択肢となります。しかし、IT組織がOracle Solaris Volume Managerを標準としている場合は、IT組織の現在のベスト・プラクティスをOracle VM Server for SPARC環境に直接適用できます。

RAIDテクノロジーとOracle Solaris ZFSのいずれかを選択する場合は、論理ドメインを実装するのに役立つOracle Solaris ZFSのいくつかの重要な機能に注目してください。Oracle Solaris ZFSのスナップショットとクローニングを使用すると、ゲスト・ドメインの仮想ディスクが含まれたフラット・ファイルを任意の時点で固定にして、バックアップとして使用できます。1つ以上のスナップショットを作成して、その時点まで論理ドメインをロールバックするために使用できます。また、スナップショットを特定の時点のボリュームのコピーとして使用して、磁気テープに転送することもできます。ボリューム・コピーとは異なり、スナップショットは即時的で、領域が効率的に使用されます。スナップショットで使用される領域は、ボリュームのメタデータと、スナップショットが作成された後にプライマリ・ボリュームで変更が発生したブロックのみです。

Oracle Solaris ZFSを使用すると、読取り専用のスナップショットをクローンにできます。クローンはスナップショットから作成された読取り/書込みのボリュームで、スナップショットと同様に即時的で領域が効率的に使用されます。クローンを使用すると、さまざまな目的で、多くの論理ドメインに仮想ディスクを作成できます。次にその例を示します。

- ゴールデン・マスター環境を作成して、複数の論理ドメインで使用するようクローン化できます。水平方向に拡張するアプリケーション（Webサーバーなど）が、このテクノロジーを使用するおもしろな用途として挙げられます。
- 1つの環境をクローン化して、プロジェクトのさまざまなフェーズで使用できます。仮想ディスクのクローンは開発、テスト、および本番環境に使用できます。
- 既存の本番環境をクローン化して、オペレーティング・システムやアプリケーションのパッチを本番環境に適用する前のテストに使用できます。

いずれの場合も、Oracle Solaris ZFSのクローンのメリットは同じです。クローンは、ほぼ即時に作成でき、ストレージを効率的に使用でき、投資収益率を向上するのに役立ちます。

ゲスト・ドメインの信頼性

I/Oドメインで信頼性の高いディスク・ストレージを構成できるのと同様に、ゲスト・ドメインでもOracle Solaris Volume ManagerとOracle Solaris ZFSを使用できます。内蔵ストレージを使用して、複数の物理ディスク、物理ディスクのパーティション、またはフラット・ファイルを1つのI/Oドメインを介してゲスト・ドメインに提供できます。Oracle Solaris Volume ManagerまたはOracle Solaris ZFSは、仮想ボリュームを使用するようゲスト・ドメインで構成できます。単一のI/Oドメインを使用している場合、このアプローチでは、I/Oドメインで同じソフトウェアを構成した場合よりも信頼性や可用性が低くなります。このアプローチは、IT組織の現在のベスト・プラクティスでOSレベルのレプリケーションを使用することを規定している場合に、もっとも役立ちます。論理ドメイン内からボリューム管理を使用できるため、これらの標準的な技術やベスト・プラクティスを仮想環境に拡張できます。

外部ストレージに複数のインタフェースを構成している場合、ゲスト・ドメインで信頼性を確立できる可能性は極めて高くなります。この場合、ゲスト・ドメインでは、2つのI/Oドメインを介してストレージにアクセスできます。

外部ストレージを使用した信頼性と可用性の構成については、今後のホワイト・ペーパーで取り上げる予定です。

これにより、I/Oドメインの障害や再起動（またはストレージへのパス上のコンポーネントの障害）のためにアクセスが中断されないよう、ストレージに対して2つの冗長パスを使用することで、可用性が向上します。

OracleプラットフォームのI/Oの信頼性と可用性

複数のI/Oドメインを使用してパラレルの冗長チャネルをI/Oに拡張できるかどうかは、CoolThreadsテクノロジーを採用しているオラクル・サーバーのアーキテクチャによって決まります。このホワイト・ペーパーで説明している構成手順では、単一のPCIバスとディスク・コントローラで接続された内蔵ディスク・ストレージに重点を置いています（CoolThreadsテクノロジーを採用したすべてのオラクル・サーバー）。したがって、内蔵ストレージは1つのI/Oドメインを介して使用できます。今後のホワイト・ペーパーの予備知識として、CoolThreadsテクノロジーを採用したオラクル・サーバーと、ネットワークとディスクI/Oに複数のI/Oドメインを使用できることに関連について概要を説明します。

例：オラクルのSun SPARC Enterprise T2000サーバーのI/Oアーキテクチャ

図2は、I/Oドメインが、オラクルのSun SPARC Enterprise T2000サーバーの2つのPCIバス上のデバイスにどのようにアクセスするのかを示しています。2つのI/Oドメインを作成している場合、一方のドメインではPCIバスAとそのすべての周辺機器にアクセスでき、もう一方のドメインではPCIバスBとそのすべての周辺機器にアクセスできます。

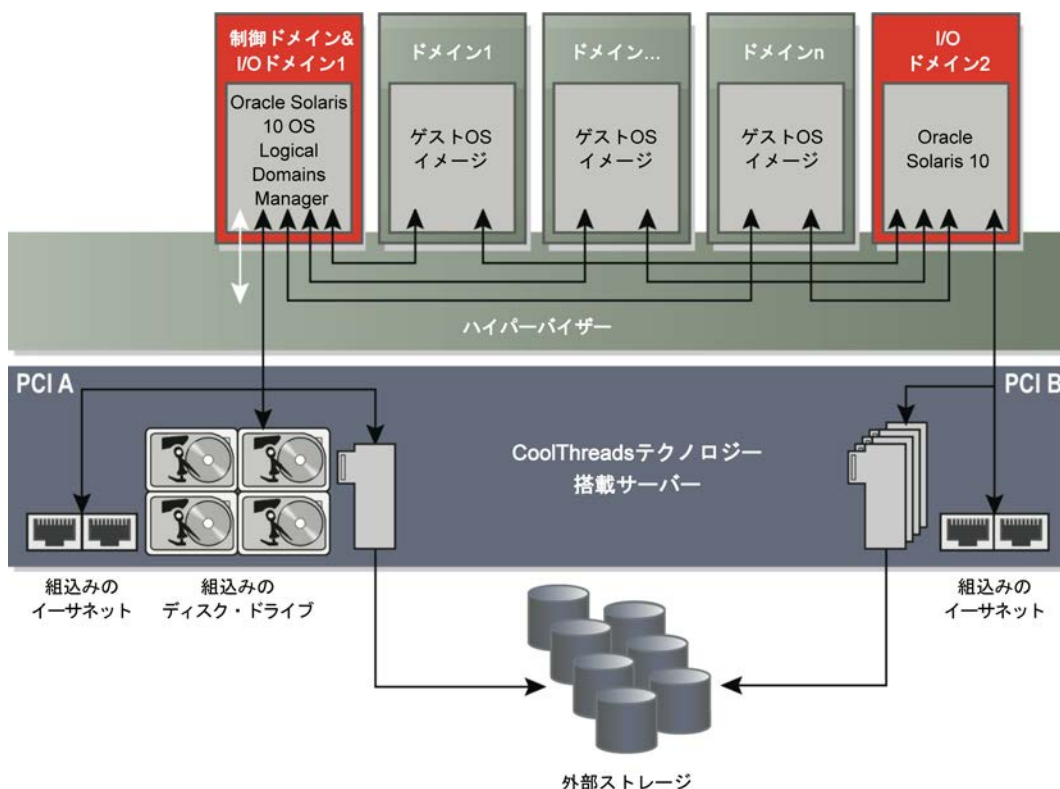


図2. Oracle Sun SPARC Enterprise T2000サーバーのI/O構成を使用して、内蔵I/Oリソースと外部I/Oリソースが2つのI/Oドメインを介してゲスト・ドメインにどのように接続されるのかを説明

この図では、すべての内蔵ドライブが単一のPCIバスに接続されているため、1つのI/Oドメインを介してすべての内蔵ストレージにアクセスできることを示しています。サーバーの2つのPCIバスが2つのI/Oドメインのそれぞれに割り当てられている場合は、2つのI/Oドメインを介して2つのPCI Expressスロットで外部ストレージにアクセスできるよう、2つのホスト・バス・アダプタを構成できます。そのため、Oracle Solaris Volume Managerで提供されているようなソフトウェア・ミラー化を使用して、ゲスト・ドメインに冗長性と耐障害性を確立できます。Sun SPARC Enterprise T2000サーバーは、2つのPCIバスで組込みのイーサネット・ポートが分割されているこのシリーズ唯一のサーバーであるため、ここで興味深い例として使用しています。このアーキテクチャにより、IPマルチパス (IPMP) を使用し、内蔵ポートを用いて2つのI/Oドメインを介して高可用性のネットワーク・アクセスを実現できます。

I/Oアーキテクチャの概要

表1は、このホワイト・ペーパーの作成時点での、CoolThreadsテクノロジーを採用したすべてのオラクル・サーバーのI/Oアーキテクチャを示しています。この表を使用して、ストレージ・リソースとネットワーク・リソースに対して内部冗長パスと外部冗長パスを使用する際の、サーバーの制約を評価できます。

表1. ORACLE VM SERVER FOR SPARCを使用したI/Oドメイン構成に影響するサーバー構成

サーバー	PCIバスの数	PCIバス	各PCIバスに接続されているインターフェースの数		
			ディスク・コントローラ	組み込みイーサネット・ポート	拡張スロット
オラクルのSun SPARC Enterprise T5240サーバー	2	A	1	0	3
		B	0	4	3
オラクルのSun SPARC Enterprise T5140サーバー	2	A	1	0	1
		B	0	4	2
オラクルのSun SPARC Enterprise T5220サーバー	1	A	1	4	6
オラクルのSun SPARC Enterprise T5120サーバー	1	A	1	4	3
		B	1	4	3
オラクルのSun SPARC Enterprise T2000サーバー	2	A	1	2	1
		B	0	2	4
オラクルのSun SPARC Enterprise T1000サーバー	2	A	0	0	1
		B	1	4	0

- PCIバスの数で、構成可能なI/Oドメインの最大数が決まります。デフォルトでは、作成する1つ目のI/OドメインがすべてのPCIバスを所有します。2つ目のI/Oドメインを作成するには、追加の手順がいくつか必要になります。
- 1つのPCIバスを装備しているサーバーでは、すべてのI/Oに1つのI/Oドメインを使用するよう制限されます。
- 2つのPCIバスを装備しているサーバーでは、2つのI/Oドメインを使用できます。たとえば、Sun SPARC Enterprise 5240サーバーに2つ目のI/Oドメインを作成して、PCIバスBをその2つ目のI/Oドメインに割り当てている場合、そのドメインでは、4つの組み込みのイーサネット・ポートと3つのPCI Express拡張スロットにアクセスできます。
- 2つのPCIバスを装備しているサーバーはすべて、バスあたり少なくとも1つの拡張スロットに接続しています。そのため、PCI Expressホスト・バス・アダプタとネットワーク・インターフェース・コントローラをインストールして、2つのI/Oドメインを介してストレージとネットワークへの冗長アクセスを実現できます。このようなサーバーには、Sun SPARC Enterprise T5240、T5140、およびT2000サーバーがあります。
- Sun SPARC Enterprise T2000サーバーは、両方のPCIバスで組み込みのイーサネット・ポートが構成されている唯一の製品です。このサーバーでは、それぞれが2つのイーサネット・ポートに接続された2つ

のI/Oドメインをサポートできるため、追加のハードウェアを使用せずに、ゲスト・ドメインにIPマルチパス (IPMP) を構成できます。

Oracle VM Server for SPARCのインストールと構成

CoolThreadsテクノロジーを採用したオラクル・サーバーに論理ドメインを設定するためには、必要となるOracle VM Server for SPARCソフトウェアをインストールし、(必要に応じて) サーバー・ファームウェアをアップグレードして、論理ドメインを有効にした状態でサーバーを再起動する必要があります。これらの手順の詳細については、外部ドキュメントを参照してください。ここでは、推奨される手順の概要について説明しています。

1. ネットワーク・インストール・サーバーを設定することをお勧めします。ネットワーク・インストール・サーバーを使用すると、Oracle Solaris JumpStartを使用してゲスト・ドメインにOracle Solarisを迅速かつ容易にインストールしたり、Oracle Solarisフラッシュを使用してお客様独自のOSとアプリケーションの組み合わせをインストールしたりできます。Oracle VM Server for SPARCバージョン1.0.2以前では、ゲスト・ドメインから組込みのDVDドライブへのアクセスが提供されないため、ネットワーク・インストール・サーバーが必要です。Oracle VM Server for SPARC 1.0.3以降では、DVDからゲスト・ドメインを起動してOracle Solarisを対話形式でインストールできます。
2. ハードウェアRAIDの使用を予定している場合、必要に応じてディスク・コントローラのファームウェアを更新し、次の手順に進む前に、次のセクションの説明に従ってハードウェアRAIDを設定します。ファームウェアを入手するには、お使いのサーバーのリリース・ノート・ドキュメント (<http://docs.sun.com/app/docs/prod/coolthreads.srvr#hic>) を参照してください。現在のファームウェア・バージョンとインストールする最新のファームウェアを確認する手順が含まれたパッチを入手できます。ネットワークまたはインストールCDから起動する際に、RAIDセットに構成するディスクの再パーティションが行われるため、ファームウェアを更新する必要があります。
3. Oracle Solaris 10 OS 8/07リリース以降をインストールします。このオペレーティング・システム・バージョンは、CoolThreadsテクノロジーに対応した一部の新しいサーバーで必要となります。また、以前のバージョンで必要となるパッチが組み込まれています。Oracle Solarisオペレーティング・システムおよびOracle VM Server for SPARCの各バージョンで発行されているリリース・ノートで、必要となる、各コンポーネントのサポートされる最小バージョンを確認できます。
4. 最新のOracle VM Server for SPARC Managerソフトウェアを入手するには、<http://www.sun.com/servers/coolthreads/ldoms/get.jsp>を参照し、最新の管理ガイドとリリース・ノートへのリンクを参照してください。Oracle VM Server for SPARCバージョン1.0.2は、Oracle Solaris 10 8/07オペレーティング・システムで動作します。

Oracle VM Server for SPARC 1.0の概要については、Oracleホワイト・ペーパー『Beginner's Guide to Oracle VM Server for SPARC: Understanding and Deploying Logical Domains』(820-0832) (wikis.sun.com/display/BluePrints/Main) を参照してください。このホワイト・ペーパーではOracle VM Server for SPARC 1.0について説明しているため、『Oracle VM Server for SPARC Administration Guide』(docs.sun.com) も参照してください。

ネットワーク・インストール・サーバーを設定するための簡潔かつ必須のガイドについては、Oracleホワイト・ペーパー『Building an Oracle Solaris JumpStart Infrastructure』(816-0428) (wikis.sun.com/display/BluePrints/Main) を参照してください。Oracle Solaris OSインストールガイド『Network-Based Installations』(820-0177) および『Custom Oracle Solaris JumpStart and Advanced Installations』(820-0179) (docs.sun.com) も参照してください。

5. 『Oracle VM Server for SPARC Administration Guide』では、システム・ファームウェアを更新する方法について説明しています。Oracle VM Server for SPARCのハイパーバイザーはファームウェアで動作し、OSに構成されているバージョンと同期する必要があるため、これは重要です。ファームウェアのリビジョン番号は、Oracle VM Server for SPARCのバージョン番号には関連していません。
6. ダウンロードしたパッケージに含まれている手順に従って、Oracle VM Server for SPARC Managerをインストールします。
7. 欄外の記事にあるテクニカル・ホワイト・ペーパーの説明または『Oracle VM Server for SPARC Administration Guide』の説明に従って、制御ドメインとI/Oドメインを設定します。

ハードウェアRAIDの構成

Oracle VM Server for SPARCの一般的な構成では、1つのドメインを使用して、サーバーの制御ドメイン、および1つ目のサービスとI/Oドメインとして機能させます。CoolThreadsテクノロジーを採用したすべてのオラクル・サーバーで利用できるハードウェアRAIDは、統合された制御ドメイン、サービス・ドメイン、およびI/Oドメインに非常に信頼性の高いブート・ディスクを提供するのに最適です。このドメインの信頼性は、きわめて重要です。ドメインに障害が発生すると、すべてのゲスト・ドメインで、オンボードのディスク・ドライブへのすべてのI/Oも使用できなくなります。

このセクションでは、4つのディスクを装備したサーバーで最初の2つのディスク・ドライブにディスクのミラー化を構成する方法について説明します。ここでの構成手順はSun SPARC Enterprise T2000サーバーを使用して実行しましたが、他のプラットフォームでもコマンドは同じです。

ハードウェアRAIDの構成の詳細については、使用しているサーバーのサーバー管理ガイド (docs.sun.com) を参照してください。

ハードウェアRAIDのディスク設定

ハードウェアRAIDコントローラを使用してミラー化を設定すると、同一のディスク・デバイスが2つ使用されて、新しい1つのディスク・ドライブがOpenBootとオペレーティング・システムに提供されます。formatコマンドを使用して、サーバーで使用可能なディスク・ドライブを表示できます。初期化を行うと、使用可能なディスク・ドライブの一覧が表示されます。たとえば、サーバーに2つの72GBドライブと2つの146GBドライブがインストールされているとします。

```
AVAILABLE DISK SELECTIONS:
  0. c1t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
    /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@0,0
  1. c1t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
    /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@1,0
  2. c1t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
    /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@2,0
  3. c1t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
    /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@3,0
```

この例では、ディスク0とディスク1（両方ともオラクルの72GBドライブ）を組み合わせ、1つのミラー化したRAIDセットにします。一連の手順を実行した後で、formatコマンドを実行すると次のドライブ

が表示され、サーバーで使用できるディスク・ドライブが1つ少なく表示されます。つまり、c1t1d0ドライブが表示されなくなります。

```
AVAILABLE DISK SELECTIONS:
  0. c1t0d0 <LSILOGIC-LogicalVolume-3000 cyl 65533 alt 2 hd 16 sec 136>
    /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@0,0
  1. c1t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
    /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@2,0
  2. c1t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
    /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@3,0
```

Oracle Solaris Volume Managerとは対照的に、ハードウェアRAIDでは、すべてのディスクが一度にミラー化されるため、ディスクがどのようにパーティション化されているかに関係なく、それらのディスクのすべてのデータがミラー化されます。次の手順でRAIDセットを構成した後で、c1t0d0で表示されるドライブをパーティション化して、そのドライブにOSをインストールする必要があります。ミラー化されたディスクにそれまで保存されていたデータはすべて失われます。

ハードウェアRAIDの実装

raidctlコマンドを使用すると、コントローラに依存しない方法でハードウェアRAIDを設定できます。RAIDセットを作成すると、そのセットに含まれるドライブに保存されていたすべてのデータが無効になるため、DVDからブートするインストール環境か、またはネットワーク・インストール・サーバーから、このコマンドを実行することをお勧めします。ネットワーク・インストール・サーバーを設定している場合は、ネットワークからシングルユーザー・モードでブートするだけです。

```
{0} ok boot net -s
```

raidctlコマンドを実行すると、標準のコントローラ-ターゲット-ディスク形式 (c1t0d0など) を使用してディスクを指定できます。または、x.y.z形式を使用して、引数を指定せずにraidctlコマンドを入力して可能なx、y、およびzの値を取得できます。ここでは、コントローラ-ターゲット-ディスク形式でディスクを指定できるため、この形式を使用して、ディスクc1t0d0とディスクc1t1d0をミラー化するRAID1セットを作成します。

```
{0} ok boot net -s
```

この相殺的な処理の実行を確認するよう求められた後で、RAIDセットが構成されると、raidctlコマンドにより作成された新しいボリュームのIDを示す状態レポートが表示されます。

```
Creating RAID volume will destroy all data on spare space of member
disks, proceed (yes/no)? y
/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Physical disk 0 created.
/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Physical disk 1 created.
/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Volume 0 created.
```

```

/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Volume 0 is |enabled||optimal|
/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Volume 0 is |enabled||optimal|
/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Physical disk (target 1) is |out of sync||online|
/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Volume 0 is |enabled||degraded|
/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2 (mpt0):
    Volume 0 is |enabled||resyncing||degraded|
Volume clt0d0 is created successfully!

```

ディスクをフォーマットする場合、ドライブをパーティション化してラベル付けする前に、formatコマンドを実行してディスクのタイプを自動構成することが重要です。ドライブ0とドライブ1を1つのボリュームに統合している場合は、formatコマンドを入力してディスク0を選択します。typeコマンドを使用して、使用可能なディスク・タイプの選択肢を表示します。

```

format> type

AVAILABLE DRIVE TYPES:
    0. Auto configure
    1. Quantum ProDrive 80S
    2. Quantum ProDrive 105S
    3. CDC Wren IV 94171-344
    4. SUN0104
    ...

```

0を選択してドライブを自動構成すると、formatコマンドで新しく作成したRAIDセットが認識されます。ディスクにラベル付けします。

```

Specify disk type (enter its number)[19]: 0
clt0d0: configured with capacity of 68.00GB
<LSILOGIC-LogicalVolume-3000 cyl 65533 alt 2 hd 16 sec 136>
selecting clt0d0
[disk formatted]
format>
format> label
Ready to label disk, continue? y

```

ディスク上にRAIDコントローラがメタデータを保存する領域を確保するため、ディスク・サイズがやや小さくなっています。OSインストール・プロセスの一環としてディスクをパーティション化できます。また、いつでもraidctl -lと入力してRAIDセットの状態を確認できます。RAIDセットを作成した直後にこのコマンドを実行すると、コントローラが2つのドライブを同期していることが出力として示されます。同期が完了すると、状態がOPTIMALに変化します。

```
# raidctl -l c1t0d0
```

Volume	Sub	Disk	Size	Stripe Size	Status	Cache	RAID Level
c1t0d0			68.3G	N/A	SYNC	N/A	RAID1
		0.0.0	68.3G		GOOD		
		0.1.0	68.3G		GOOD		

この一連の手順で行ったように、ハードウェアRAIDの設定は容易です。これで、論理ドメインに指定したRawパーティションまたはフラット・ファイルにはすべて、バックアップの冗長ストレージがあります。

論理ドメインの作成

RAIDセットにOracle Solaris 10 OSをインストールしたら、Oracle VM Server for SPARCをインストールして構成する必要があります。ゲスト・ドメインにストレージとネットワークを提供する、仮想ディスク・サービスと仮想スイッチを作成します。コンテンツがミラー化されたフラット・ファイルを作成し、それらのフラット・ファイルをブート・ディスクとしてゲスト・ドメインに接続して、ネットワーク・インストール・サーバーを使用して各ドメインにOracle Solaris OSをインストールできます。この例では、2つの仮想CPU、2GBのメイン・メモリ、および8GBのブート・ディスクを備えたゲスト・ドメインguest2の作成について説明します。ゲスト・ドメインのネットワークは、primary-vsw0という名前の既存の仮想スイッチに接続されています。ドメインは、自動的に起動しないよう設定されています。

論理ドメインの作成手順については、10ページ上部の欄外の記事に記載されているリソースを参照してください。

```
fraser# mkfile 8g /domains/s2image.img
fraser# ldm create guest2
fraser# ldm set-vcpu 2 guest2
fraser# ldm set-mem 2g guest2
fraser# ldm add-vnet vnet0 primary-vsw0 guest2
fraser# ldm add-vdsdev /domains/s2image.img vol2@primary-vds0
fraser# ldm add-vdisk vdisk0 vol2@primary-vds0 guest2
fraser# ldm bind guest2
fraser# ldm set-variable auto-boot?=false guest2
```

このドメインにネットワーク・インストールを行う予定の場合、Oracle Solaris JumpStartを設定するために必要となる重要な情報の1つに、サーバーを物理ネットワークに接続させる仮想スイッチのドメインのネットワークMACアドレスがあります。

```
fraser# ldm list-bindings guest2
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL
guest2	bound	-----	5001	2	2G	

```
MAC
00:14:4f:fb:08:50

VCPU
VID    PID    UTIL STRAND
```

```

0      8      100%
1      9      100%

MEMORY
  RA          PA          SIZE
  0x8000000  0x188000000  2G

VARIABLES
  auto-boot?=false

NETWORK
  NAME          SERVICE          DEVICE          MAC
  vnet0         primary-vsw0@primary  network@0  00:14:4f:fa:2b:e4
  PEER          MAC
  primary-vsw0@primary  00:14:4f:fb:ae:95
  vnet0@guest1  00:14:4f:fb:58:90

DISK
  NAME          VOLUME          TOUT DEVICE  SERVER
  vdisk0        vol2@primary-vds0  disk@0  primary

VCONS
  NAME          SERVICE          PORT
  guest2        primary-vc0@primary  5001

```

使用する必要があるMACアドレスは、上記に赤で示されている00:14:4f:fa:2b:e4です。これは、仮想ネットワークvnet0に表示されるサーバーのMACアドレスです。MACの下に表示されるMACアドレスではありません。

list-bindingsコマンドで、ドメインのコンソールがポート5001にあることが示されています。telnet localhost 5001を使用してそのポートのコンソールに行き、ネットワーク・インストールを開始できます。

```

fraser# telnet localhost 5001
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Connecting to console "guest2" in group "guest2" ....
Press ~? for control options ..

{0} ok boot vnet0 - install

```

ネットワーク・インストールは、物理サーバーの場合と同じように進行します。Oracle Solaris JumpStartの構成ファイルの例は、このホワイト・ペーパーの後半の"ルール・ファイルの設定"以降に示しています。

I/OドメインでのOracle Solaris ZFSの構成

Oracle Solaris Volume ManagerやOracle Solaris ZFSなどの製品を使用したソフトウェア・ボリューム管理によって、ゲスト・ドメインで使用する信頼性の高いストレージを構成できます。信頼性の高いディスク・ストレージをI/Oドメインで構成すると、ストレージ・ソ

Oracle Solaris ZFSの詳細については、『Oracle Solaris ZFS管理ガイド』（819-6260）
<http://www.oracle.com/technetwork/jp/indexes/documentation/index.html> を参照してください。

ソフトウェアを一元管理でき、信頼性の側面がゲスト・ドメインに対して透過的になります。Oracle Solaris ZFSは、単純なボリューム管理とハードウェアRAIDで実現できる範囲を超えた機能を備えているため、I/Oドメインで冗長性を確立する際に特に役立つテクノロジーです。スナップショットでは、ゲスト・ドメインの仮想ディスクのポイント・イン・タイムのコピーを作成でき、クローンは、他の仮想マシンで使用するゲスト・ドメインの仮想ディスクの永続コピーを作成できます。この方法は、さまざまな開発フェーズ（開発、テスト、本番）でゲスト・ドメインの複数のコピーを作成するのに適しています。また、水平方向の拡張（Webサーバーなど）にも適しています。目的に関係なく、Oracle Solaris ZFSのスナップショットとクローンは事実上、即時的であり、そのcopy-on-writeセマンティクスによりストレージを効率的に使用できます。

Oracle Solaris ZFSの実装とゲスト・ドメインのボリュームのクローン化

このセクションでは、既存のアプリケーションのブート・ディスクのコピーを作成して、複数の論理ドメインで使用するようそのコピーの複数のインスタンスをクローン化する方法について、仮定の例を用いて説明します。このプロセスは、たとえば、水平方向に拡大するアプリケーションの複数のインスタンスを作成する必要がある場合に役立ちます。図3に一連のスナップショットとクローンを示します。

1. 作業する構成のブート・ディスクのイメージをOracle Solaris ZFSファイル・システムにコピーします。

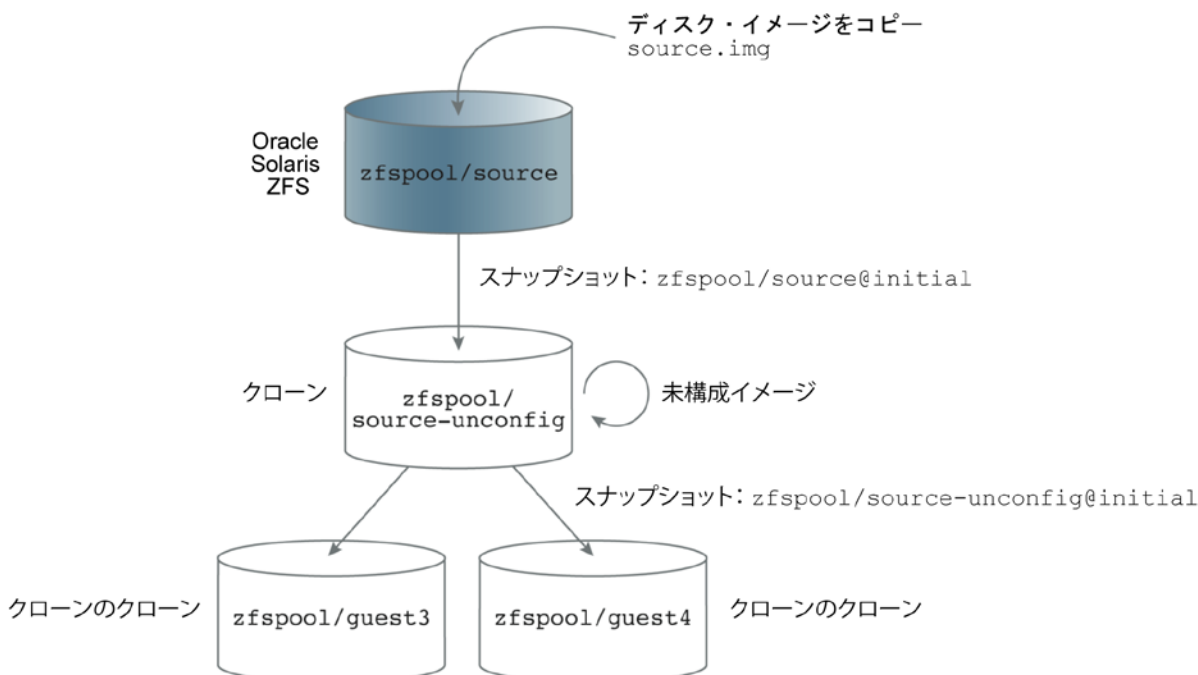


図3. 一連のスナップショットとクローンで、複数の論理ドメインで使用されるアプリケーション環境の複数のインスタンスを作成。

青色の元のOracle Solaris ZFSファイル・システムでのみ、大規模なディスク領域を使用

2. スナップショットとクローンを作成し、構成解除します。この手順により、別の目的で再度使用したり、誤りがあった場合にロールバックしたりできるよう、元のブート・ディスクのイメージを保持します。
3. 構成解除したイメージが含まれたファイル・システムのスナップショットを作成し、そのスナップショットの複数のクローンを作成します。つまり、クローンのクローンを作成できます。

Oracle Solaris ZFSプールの作成

まず、Oracle Solaris ZFSプールを作成します。Sun Fire T2000サーバーの4つのドライブのうちの最初の2つをハードウェアRAIDの実証に使用したため、ソフトウェア・ミラー化を有効にした残りの2つのドライブをOracle Solaris ZFSプールに割り当てます。もっとも簡単な方法は、次のコマンドを使用して2つのドライブ全体をOracle Solaris ZFSプールに割り当てる方法です。

```
fraser# zpool create zfspool mirror c1t2d0 c1t3d0
```

ここでは、以降の使用に備えて物理ディスクの一部のパーティションを確保しておく必要があるため、2つのディスク・パーティションをOracle Solaris ZFSプールに割り当てます。まず、formatコマンドを使用して、ドライブ3であるc1t2d0のパーティション表を表示します。

Part	Tag	Flag	Cylinders	Size	Blocks	
0	root	wm	0 - 1030	10.01GB	(1031/0/0)	20982912
1	swap	wu	1031 - 1237	2.01GB	(207/0/0)	4212864
2	backup	wu	0 - 14086	136.71GB	(14087/0/0)	286698624
3	unassigned	wm	0	0	(0/0/0)	0
4	unassigned	wm	0	0	(0/0/0)	0
5	unassigned	wm	0	0	(0/0/0)	0
6	usr	wm	1238 - 14086	124.69GB	(12849/0/0)	261502848
7	unassigned	wm	0	0	(0/0/0)	0

4つ目のドライブをまったく同じようにパーティション化するため、強力な（ただしリスクを伴う）一連のコマンドを使用して、ドライブ3のパーティションを4つ目のドライブにコピーします。この一連のコマンドを使用すると作業が容易ですが、間違ったドライブを誤ってパーティション化すると既存のデータにアクセスできなくなるため、引数が正しいことを必ず確認してください。

```
fraser# prtvtoc /dev/rdisk/c1t2d0s2 | fmthard -s - /dev/rdisk/c1t3d0s2
```

各ドライブのusrスライスを使用する、zfspoolと呼ばれるOracle Solaris ZFSプールを作成します。

```
fraser# zpool create zfspool mirror c1t2d0s6 c1t3d0s6
```

ゲスト・ドメインに仮想ディスクとして提供するファイルを保存できる、sourceという名前のOracle Solaris ZFSファイル・システムを作成します。

```
fraser# zfs create zfspool/source
```

別のゲスト・ドメインのブート・ディスクが含まれているフラット・ファイルをsourceファイル・システムにコピーします。このファイル・システムで作業するソース・ファイルは、source.imgです。

```
fraser# cp /domains/s2image.img /zfspool/source/source.img
```

zfs listコマンドを使用して、イメージに8GBを使用していることを確認します。このイメージは、Oracle Solaris ZFSプールに格納されている唯一のファイルです。

```
fraser# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
zfspool              8.00G  114G   25.5K  /zfspool
zfspool/source       8.00G  114G   8.00G  /zfspool/source
```

クローン化と構成解除

Oracle Solaris ZFSファイル・システムのzfspool/sourceに格納されたファイルsource.imgには作業イメージが含まれているため、作業を進める前にそのクローンを作成して、いつでもそのイメージにロールバックできるようにします。このセクションの手順は必要なものではありません。これらの手順は、実際の状況でどのようなことができるのか、単に例を示すために紹介しています。

クローンを作成するため、まず、Oracle Solaris ZFSファイル・システムのスナップショットを作成し、そのスナップショットのクローンを作成します。クローン化プロセスで、スナップショットの永続コピーが作成されます。スナップショットはファイルではなくファイル・システムで機能するため、多くのディスク・イメージを1つのファイル・システムにロードするのではなく、1つのイメージまたは関連する一連のイメージに1つのファイル・システムを作成することをお勧めします。この例では、initialという名前のソース・ファイル・システムのスナップショットを作成し、そのスナップショットをクローン化して、それと同時に新しいファイル・システムzfspool/source-unconfigを作成しています。Oracle Solaris ZFSプールに2つのファイル・システムを作成しました。

```
fraser# zfs snapshot zfspool/source@initial
fraser# zfs clone zfspool/source@initial zfspool/source-unconfig
fraser# ls
source                source-unconfig
```

8GBのディスク・イメージのクローンでは、クローンのすべてのブロックは既存のzfspool/sourceファイル・システムを参照するだけであるため、22.5KBのストレージ領域が使用されています。

```
fraser# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
zfspool              8.00G  114G   27.5K  /zfspool
zfspool/source       8.00G  114G   8.00G  /zfspool/source
zfspool/source@initial  0      -      8.00G  -
zfspool/source-unconfig 22.5K  114G   8.00G  /zfspool/source-unconfig
```

source.imgのクローンは、ホスト名、IPアドレス、タイムゾーンなどで定義されたシステムIDが保持されないよう、構成解除するディスク・イメージです。このファイルは一時的にだけ使用するため、構成解除の手順を実行するときこのファイルをブート・ディスクとして接続して、既存の論理ドメインをハイジャックしてみます。この例では、まず、ゲストの既存の仮想ディスクのデバイス (vol2@primary-vds0) を切り離し、クローン化したファイル・システムから新しい仮想ディスクのデバイス (source.img ファイル) を同じ名前 (vol2@primary-vds0) で追加します。最初に、停止状態になっているguest2を切り離して、guest2のディスクを切り替えたらguest2を起動します。

```
fraser# ldm unbind-domain guest2
fraser# ldm remove-vdsdev vol2@primary-vds0
fraser# ldm add-vdsdev /zfspool/source-unconfig/source.img
                vol2@primary-vds0
fraser# ldm bind guest2
fraser# ldm start guest2
LDom guest2 started
```

localhostの正しいポート番号に対してtelnetを実行してguest2のコンソールに行き、必要に応じてドメインを起動して、sys-unconfigコマンドを実行します。誤った操作を行った場合は、いつでもクローンを削除して、zfspool/sourceファイル・システムのsource.imgのコピーに戻すことができます。

```
# sys-unconfig

                          WARNING

This program will unconfigure your system.It will cause it
to revert to a "blank" system - it will not have a name or know
about other systems or networks.

This program will also halt the system.

Do you want to continue (y/n) ? y
```

sys-unconfigコマンドを実行してシステムをシャットダウンします。guest2の元のディスク・イメージをリストアできるようになりました。

```
fraser# ldm unbind-domain guest2
fraser# ldm remove-vdsdev vol2@primary-vds0
fraser# ls /domains
guestlimage s2image.img
fraser# ldm add-vdsdev /domains/s2image.img vol2@primary-vds0
fraser# ldm bind guest2
```

構成解除したシステムのクローン化と実行

構成解除したシステムを使用できるようになったため、ブート・ディスクが含まれたファイル・システムを何度でもクローン化して、クローン化した各ファイル・システムのsource.imgファイルを新しい論理ドメインに接続できます。この例では、構成解除したディスク・イメージが含まれているファイル・システムのinitialという名前のスナップショットを作成します。guest3とguest4の2つのクローンを作成します。Oracle Solaris ZFSプールで使用されているストレージは、まだ約8GBのみです。

```
fraser# zfs snapshot zfspool/source-unconfig@initial
fraser# zfs clone zfspool/source-unconfig@initial zfspool/guest3
fraser# zfs clone zfspool/source-unconfig@initial zfspool/guest4
fraser# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
zfspool	8.00G	114G	27.5K	/zfspool
zfspool/guest3	0	114G	8.00G	/zfspool/guest3
zfspool/guest4	0	114G	8.00G	/zfspool/guest4
zfspool/source	8.00G	114G	8.00G	/zfspool/source
zfspool/source@initial	0	-	8.00G	-
zfspool/source-unconfig	31.3M	114G	8.00G	/zfspool/source-unconfig
zfspool/source-unconfig@initial	0	-	8.00G	-

次に、2つの仮想CPU、2GBのメイン・メモリ、およびブート・ディスク（イメージはOracle Solaris ZFSファイル・システムzfspool/guest3に格納されたsource.img）を備えた、論理ドメインguest3を作成する手順を示します。本番環境を実行しているわけではないため、ドメインは自動的に起動しないように設定します。

```
fraser# ldm create guest3
fraser# ldm set-vcpu 2 guest3
fraser# ldm set-mem 2g guest3
fraser# ldm add-vnet vnet0 primary-vsw0 guest3
fraser# ldm add-vdsdev /zfspool/guest3/source.img vol3@primary-vds0
fraser# ldm add-vdisk vdisk0 vol3@primary-vds0 guest3
fraser# ldm set-variable auto-boot?=-false guest3
fraser# ldm set-variable boot-device=/virtual-devices@100/channel-
devices@200/disk@0:a guest3
fraser# ldm bind guest3
```

ldmの起動を使用して、論理ドメインを起動できます。guest3が作成した3つ目の論理ドメインである場合、localhostポート5002に対してtelnetを実行し、コンソールに接続してドメインを起動できます。起動すると、システム構成プロセスが開始されて、新しいサーバーを未設定の状態を使い始める場合と同じように、システムID情報を指定するように求められます。実際の環境では、ソースの論理ドメインで構成した"ゴールドン・マスター"のアプリケーション・インスタンスの新しいインスタンスが作成されます。

クローンを使用すると、非常に大きな領域を節約できます。2つのドメインを作成および構成した後で、ldm listコマンドを実行すると、ドメインあたり約50GBしか使用していないことが示されます。

```
fraser# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
zfspool                             8.13G  114G  29.5K  /zfspool
zfspool/guest3                       50.4M  114G  8.00G  /zfspool/guest3
zfspool/guest4                       51.7M  114G  8.00G  /zfspool/guest4
zfspool/source                       8.00G  114G  8.00G  /zfspool/source
zfspool/source@initial                0      -  8.00G  -
zfspool/source-unconfig              31.3M  114G  8.00G  /zfspool/source-unconfig
zfspool/source-unconfig@initial       0      -  8.00G  -
```

構成の保存

論理ドメイン構成を変更したら、電源を入れ直しても構成が保持されるよう、サービス・プロセッサに状態を保存することを忘れないでください。次の一連のコマンドで、既存の構成fourguestsを削除して現在の構成に置き換えます。

```
fraser# ldm remove-spconfig fourguests
fraser# ldm add-spconfig fourguests
```

ゲスト・ドメインでのボリューム管理の構成

I/Oの冗長性を構成できる3つ目のアーキテクチャ・レベルは、I/Oドメイン自体に存在します。Oracle Solaris Volume ManagerやOracle Solaris ZFSなどのボリューム管理ソフトウェアでは、複数の仮想ディスクを管理し、ゲスト・ドメインから必要なレプリケーションを実行します。この2つのボリューム管理ソフトウェアのうち、ミラー化したブート・ディスクをサポートしているOracle Solaris Volume Managerについてここで説明します。

Oracle Solaris Volume Managerの詳細については、『Oracle Solaris Volume Manager Administration Guide』(docs.sun.com)を参照してください。

2つのI/Oドメインを使用して外部ストレージへの複数のパスをサポートする場合、サーバー自体以外に通信パスにシングル・ポイント障害がないため、ゲスト・ドメインのボリューム管理が推奨されるアプローチとなります。このホワイト・ペーパーのトピックである、内蔵ディスク・ストレージにアクセスする単一のI/Oドメインの場合、ゲスト・ドメイン内のストレージをレプリケートするのは、ゲスト内で仮想デバイスから物理デバイスへのデュアル・パスのオーバーヘッドが発生するため、効率的ではありません。すべてのディスク書込みが仮想デバイスへのパスを2回通過する必要がありますが、I/Oドメインでレプリケーションを構成すると、I/Oが物理デバイスの近くに配置されるため、より効率的です。

ネットワーク・インストールによるボリューム管理の設定

ゲストOSでボリューム管理を使用することを既存のベスト・プラクティスで規定している場合や、実環境の再作成する開発環境を作成する必要がある場合は、物理システムの場合と同様にゲスト・ドメインでOracle Solaris Volume ManagerとOracle Solaris ZFSを使用できます。データセンターのベスト・プラクティスの1つは、Oracle Solaris JumpStartを使用したインストール中にディスクのミラー化を確立することです。そのため、この項では、ネットワーク・インストール・プロセスの一部としてOracle Solaris Volume Managerを使用してルート・ディスクのミラー化を設定する方法を説明します。この項は読者がOracle Solaris JumpStartの設定プロセスに精通していることを前提としているため、ここではプロセスの要点のみ説明します。

ネットワーク・インストール・サーバーを設定するための簡潔かつ必須のガイドについては、テクニカル・ホワイト・ペーパー『Building a JumpStart Infrastructure』(816-0428) (wikis.sun.com/display/BluePrints/Main)を参照してください。Oracle Solaris OSインストールガイド『Network-Based Installations』(820-0177) および『Custom JumpStart and Advanced Installations』(820-0179) (docs.sun.com)も参照してください。

ゲスト・ドメインの設定

まず、Oracle Solaris Volume Managerを使用してミラー化する、2つの仮想ディスクを備えたゲスト・ドメインを設定します。Oracle Solaris Volume Managerではクォーラム・アルゴリズムを使用しているため、ミラー化した一連のディスクを使用するには、半分と1つの状態データベースを加えたものを使用可能にして有効にする必要があります。つまり、2つのディスクを使用して各ディスクに半分の状態データベースを保持しているシステムで、一方のディスクに障害が発生した場合、ブートするには、障害が発生したディスクの状態データベースを手動で削除する必要があります。仮想ディスクを使用できるというメリットがあるため、安価であり、タイ・ブレイカーとして機能する小容量16MBの3つ目のディスクを作成できます。

実際の環境では、コンテンツが3つの異なる物理ディスクに格納される、3つの仮想ディスクを作成する必要があります。この例では、既にサーバーのディスクの半分をRAIDセットに割り当てて、残りの半分

をOracle Solaris ZFSプールに割り当てているため、既にミラー化した領域から仮想ディスクを作成します。これを行うのは効率的ではありませんが、この例の目的にはこのコマンドが適切です。

前のセクションでクローン化したディスクを使用して設定した、論理ドメインguest4を使用します。2つ目のフラット・ファイルをミラーとして作成し、3つ目のフラット・ファイルをOracle Solaris Volume Managerのクォーラム・ディスクとして作成します。サーバーの電源を入れ直しても保持されるよう、構成を保存します。

```
fraser# mkfile 8g /zfspool/guest4/mirror.img
fraser# mkfile 16M /domains/guest4quorum.img
fraser# ldm stop guest4
fraser# ldm unbind guest4
fraser# ldm add-vdsdev /zfspool/guest4/mirror.img vol5@primary-vds0
fraser# ldm add-vdisk vdisk1 vol5@primary-vds0 guest4
fraser# ldm add-vdsdev /domains/guest4quorum.img vol6@primary-vds0
fraser# ldm add-vdisk vdisk2 vol6@primary-vds0 guest4
fraser# ldm bind guest4
fraser# ldm remove-sponconfig fourguests
fraser# ldm add-sponconfig fourguests
```

ドメインのコンソールに接続し、3つのディスクが接続されていることを確認します。

```
{0} ok show-disks
a) /virtual-devices@100/channel-devices@200/disk@2
b) /virtual-devices@100/channel-devices@200/disk@1
c) /virtual-devices@100/channel-devices@200/disk@0
q) NO SELECTION
Enter Selection, q to quit: q
```

Rules File Setup

ネットワーク・インストール・サーバーのルール・ファイルは、インストール・クライアントのタイプを検出すると、インストール・プロセスを開始して、ルール・ファイルで指定されている特定のプロファイル・ファイルに従います。次のルール・ファイルの行で、CoolThreadsテクノロジーを採用したサーバー（sun4vアーキテクチャ）へのすべてのインストールに使用するprofile.mirror.s4プロファイルを指定し、インストール前またはインストール後にスクリプトを実行しないことを指定します。

```
arch sun4v - Profiles/profile.mirror.s4 -
```

プロファイル・ファイルの設定

次に、既存のOracle Solarisフラッシュ・アーカイブをインストールするために使用した、プロファイル・ファイルを示します。

```
#
# Jumpstart Profile to install a Solaris Flash archive in a LDom
# Use a third, small virtual disk as a quorum disk for SVM
#
install_type flash_install
archive_location nfs js:/jumpstart/Archives/s4mirror.flar
#
partitioning explicit
#
# Mirror root and swap
#
filesystem mirror:d10 c0d0s0 c0d1s0 free /
filesystem mirror:d20 c0d0s1 c0d1s1 1024 swap
#
# Create three sets of state databases so there should
# always be a quorum of databases if a single disk fails.
# Two databases per volume for media redundancy.
#
metadb c0d0s7 size 8192 count 2
metadb c0d1s7 size 8192 count 2
metadb c0d2s2 size 8192 count 2
```

Sysidcfgファイルの設定

Oracle Solaris JumpStartを設定する便利な方法があるとすれば、無人インストールを可能にするsysidcfgファイルを作成することが挙げられます。要件はOracle Solarisオペレーティング・システムのリリースごとに変化するため、通常、変化するターゲットが作成されます。次のsysidcfgファイルはOracle Solaris 10 8/07オペレーティング・システムのインストールで機能します。このファイル特有の操作として、対話型のインストールにならないよう、タイム・サーバーをlocalhostとして指定する必要があります。

```
timezone=US/Mountain
timeserver=localhost
root_password=678B3hGfBa01U
name_service=DNS {domain_name=loneagle.com search=loneagle.com}
name_server=ins(192.168.0.134)}
network_interface=vnet0 {protocol_ipv6=no hostname=s4 ip_address=192.168.0.58
default_route=192.168.0.1 netmask=255.255.255.192}
security_policy=NONE
system_locale=en_US
terminal=xterms
keyboard=US-English
service_profile=open
nfs4_domain=loneagle.com
```


インストール・クライアントの追加

ネットワーク・インストール・サーバーの/etc/hostsファイルにゲスト・ドメインのIPアドレスを設定していることを確認します。14ページの"論理ドメインの作成"で説明している手順を使用して、ゲスト・ドメインのMACアドレスを指定します。次に、add_install_clientコマンドを実行することで、ブート・ブロックを設定し、/etc/bootparamsに正しいパラメータを設定し、/etc/ethersにMACアドレスを設定します。

```
js# ./add_install_client -e 00:14:4f:f8:2b:7e -s
js:/jumpstart/OS/Solaris10-2007-08 -c js:/jumpstart -p
js:/jumpstart/Sysidcfg.ldom-mirror s4 sun4v
Adding Ethernet number for s4 to /etc/ethers
updating /etc/bootparams
```

インストール

Oracle Solaris OSをインストールするか、または、論理ドメインのコンソールに接続してコマンドを発行し、Oracle Solarisフラッシュ・アーカイブをインストールします（ここでは、後者を実行）。

```
#{0} ok boot vnet0 - install
```

インストール・プロセスで、ミラー化されたrootファイル・システムとswapファイル・システムが設定されます。インストール時に、状態のレプリカが作成されるときに、エラー・メッセージが表示される場合があります。

```
metadb:s4:network/rpc/meta:default:failed to enable/disable SVM service
```

また、ブート時に、次のようなエラーがいくつか表示される場合があります。

```
NOTICE:mddb:unable to get devid for 'vdc', 0x7
```

これらのエラーは既知の問題に関連するもので、Oracle Solaris Volume Managerの操作には影響しません。

ハウスキーピング

ミラー内の2つ目のディスクにルート・ブロックを設定したりOpenBootで代替のブート・デバイスを設定したりするなど、Oracle Solaris Volume Managerを使用してミラー化したルート・パーティションを設定する際に必要な、通常のハウスキーピングを行う必要があります。

1つ目のディスクに障害が発生した場合に2つ目のディスクからブートできるよう、ゲスト・ドメイン内からミラーの2つ目のディスクにブート・ブロックをインストールします。

```
s4# cd /usr/platform/`/usr/bin/uname -i`/lib/fs/ufs
s4# /usr/sbin/installboot ./bootblk /dev/rdsk/c0d1s0
```

OpenBootプロンプトで、プライマリとセカンダリのブート・ディスクにデバイス・エイリアスを作成し、それらのデバイス・エイリアスから順番にブートするようブート・デバイスを設定します。

```
{0} ok nvalias primary_root /virtual-devices@100/channel-devices@200/disk@0
{0} ok nvalias backup_root /virtual-devices@100/channel-devices@200/disk@1
{0} ok setenv boot-device primary_root backup_root
boot-device = primary_root backup_root
{0} ok nvstore
```

テスト

構成をテストするには、ディスクを事実上削除します。つまり、ゲスト・ドメインをシャットダウンして、いずれかのディスクを削除します。この例では、disk1を削除します。

```
fraser# ldm unbind guest4
fraser# ldm remove-vdisk vdisk1 guest4
fraser# ldm bind guest4
fraser# ldm start guest4
```

OpenBootからdisk1がなくなっていることを確認できます。

```
{0} ok show-disks
a) /virtual-devices@100/channel-devices@200/disk@2
b) /virtual-devices@100/channel-devices@200/disk@0
```

状態データベースのクォーラムがあるため、ドメインは正常に起動します。テスト後、'障害が発生'したディスクを交換して、両方のディスクを使用できる状態でリブートできます。

```
fraser# ldm unbind guest4
fraser# ldm add-vdisk vdisk1 vol5@primary-vds0 guest4
fraser# ldm bind guest4
fraser# ldm start guest4
LDom guest4 started
```

これで障害をシミュレートしたドライブをリストアしました。ゲスト・ドメインで実行したmetastatコマンドで、ミラーを再同期する必要があることが示されます。ミラーを再同期するには、metareplaceコマンドを使用します。

```
s4# metareplace -e d10 c0d1s0
s4# metareplace -e d20 c0d1s1
```

結論

CoolThreadsテクノロジーを採用したオラクル・サーバーでOracle VM Server for SPARCを使用すると、仮想環境でデータの信頼性メカニズムと可用性メカニズムを物理サーバー上と同様に実装する必要がある組織では、さまざまな可能性が広がります。ネットワーク・リソースやディスクI/Oリソースに複数のパ

スを構成すると、データの可用性を向上できます。ミラー化技術とRAID技術を使用して冗長なストレージ・メディアを構成すると、データの信頼性を向上できます。

このテクニカル・ホワイト・ペーパーの目的は、オラクル・サーバーで実行されている論理ドメインと内蔵ディスク・リソースを使用して信頼性を高める技術に取り組むことです。CoolThreadsテクノロジーを採用したオラクル・サーバーはすべて、オンボード・ディスク・ストレージ用の単一のコントローラを使って構成されているため、使用可能な技術は、次の3つのアーキテクチャ・レベルのいずれかでミラー化とRAIDを実装するというものになります。

- ハードウェアで、ディスク・コントローラ自体を使用する
- I/Oドメインで、Oracle Solaris Volume ManagerまたはOracle Solaris ZFSを使用する
- ゲスト・ドメインで、Oracle Solaris Volume ManagerまたはOracle Solaris ZFSを使用する

単一のI/Oドメインを使用するという制限があるため、できるだけ下位のレベルで冗長性を実装することをお勧めしていますが、次のように、より高いレベルのアーキテクチャを使用する理由も多々あります。

- Oracle Solaris ZFSは、ミラー化とRAIDをサポートし、スナップショットとクローンもサポートし、ボリューム管理とファイル・システムを統合しています。これらのメカニズムを使用して、開発やテストに使用したり、水平方向の拡張に対応するよう多くの同一の論理ドメインを実装したりできる、新しい論理ドメインを作成できます。スナップショットをバックアップに使用したり、スナップショットを使用して、障害時リカバリの目的でリモートの場所に移動できるポイント・イン・タイムのコピーを作成したりできます。Oracle Solaris ZFSのもっとも優れている点は、スナップショットとクローンがほぼ即時に作成されて、copy-on-writeセマンティクスによりストレージ・リソースを効率的に使用できる点です。
- ゲスト・ドメインでOracle Solaris Volume ManagerまたはOracle Solaris ZFSを使用すると、既存の物理構成をほぼ正確に、仮想環境に再作成できます。これらのボリューム管理メカニズムでは、物理ディスクにアクセスする代わりに、フラット・ファイル、物理ドライブ、またはディスク・パーティションによって下位でサポートされる仮想ディスクにアクセスします。物理サーバーの環境を非常に正確に仮想環境に再作成できるため、IT組織では、既存のベスト・プラクティスを引き続き使用できます。また、ディスク障害のシナリオを作成したり、それらの障害シナリオに対処する手順を検証したりするために、低コストで負荷が少ないテスト環境を構築できます。

このホワイト・ペーパーは、物理環境で使用されているI/Oの信頼性技術と可用性技術を論理ドメイン環境に適用する際に生じる問題に対処するために作成された、シリーズの第一弾となります。今後のホワイト・ペーパーでは、複数のPCIバスを使用した高可用性のネットワーク構成とディスクI/O構成、およびさまざまなアーキテクチャ・レベルでの技術の実装のトレードオフについて説明します。

著者について

Peter A. Wilsonは16年間を超える業界経験があります。うち、12年間はSunとOracleで勤務しており、ハードウェア、ソフトウェア、システム、および製品マーケティングでさまざまな役割を果たしてきました。2000年に英国から米国に異動し、SunのNetraとフォルト・トレラント・サーバーのお客様向けテストを担当しました。現在、テクニカル・マーケティングのエンジニアとして、CoolThreadsテクノロジーを採用したオラクル・サーバーのすべての側面を担当しています。Peterは、英国のニューカッスル・アポン・

タイン大学で、マイクロエレクトロニクスとソフトウェア・エンジニアリングのM.Eng（工学修士）の学位を取得しています。

謝辞

このホワイト・ペーパーの著者は、このホワイト・ペーパーで説明している問題と構成に取り組み、それらに基づいてこのホワイト・ペーパーを作成したことに対して、フリーのテクニカル・ライターでありエンジニアであるSteve Gaedeに感謝の意を表明しています。また、Oracle Solaris Volume Managerの使用に関するニュアンスを理解できるよう支援したことに対して、Maciej Browarski、Alexandre Chartre、James MacFarlane、およびNarayan Venkatにも感謝の意を表明しています。

参考資料

関連するOracleのホワイト・ペーパーについては、このホワイト・ペーパーで紹介しています。



Oracle VM Server for SPARCを使用してデータの
信頼性を高めるためのベスト・プラクティス
2010年5月
著者 : Peter A. Wilson

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

海外からのお問い合わせ窓口 :
電話 : +1.650.506.7000
ファクシミリ : +1.650.506.7200
oracle.com



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

Copyright © 2010, 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、ここに記載される内容は予告なく変更されることがあります。本文書は一切間違いがないことを保証するものではなく、さらに、口述による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含み、いかなる他の保証や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。本文書はオラクル社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。

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。
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。UNIXはX/Open Company, Ltd.によってライセンス提供された登録商標です。
0310

Hardware and Software, Engineered to Work Together