

**ORACLE®**

**VM**

Oracleテクニカル・ホワイト・ペーパー  
2014年3月

## Oracle VM Server for SPARCのライブ・ マイグレーション機能を利用した アプリケーション可用性の向上： Oracle Databaseの例

このホワイト・ペーパーでは、稼働中のOracleデータベースを中断せずにシステム間で移行するためのOracle Solaris 11サーバーとストレージ・アレイの構成方法について説明します。また、ライブ・マイグレーション・プロセスの概要と、ライブ・マイグレーションの実行手順についても説明します。

**ORACLE®**

## 概要

このホワイト・ペーパーでは、[Oracle VM Server for SPARC](#)ソフトウェアのライブ・マイグレーション機能を活用して、稼働中のOracle Databaseシングル・インスタンスを中断せずにシステム間で移行する方法について説明します。

**注：**Oracle Database 11g Release 2がOracle VM Server for SPARCシステム上で稼働する構成は、Oracle Solaris 10 OSおよびOracle Solaris 11で認定されています。このホワイト・ペーパーではOracle Solaris 10 9/10 OSおよびOracle Solaris 11 OSを使用して、Oracle VM Server for SPARCのライブ・マイグレーション機能について説明しています。この機能は以前にOracle VM Server for SPARC 2.2を使用して実証されています。このホワイト・ペーパー執筆時のリリース（3.1）を含め、上位互換性があります。

## 基礎

この項では、このホワイト・ペーパーで必要となるシステム構成を作成するための基本的な概念、ソフトウェア、機能について説明します。

- **停止時間：**高可用性について議論する際には、計画停止時間と計画外停止時間を区別する必要があります。

*計画停止時間*は、予防または修正のための保守作業であり、日常的なシステム運用を中断します。このタイプの停止は、修復、バックアップ、アップグレードなどの保守作業の結果生じます。この停止では、システムの可用性に対する影響を最小限に抑えるようにスケジュールを管理する必要があります。サブシステムに対する小さな修正でも中断が発生してビジネス・ニーズに影響を及ぼすことがあるためです。

*計画外停止時間*は、ハードウェアまたはソフトウェアのエラーや外部環境のイベントの結果生じます。そのような停止時間は大至急解消する必要があります。このホワイト・ペーパーでは計画外停止時間については扱いません。

- **Oracle Database：**Oracle Databaseは業界有数のデータベース・アプリケーションであり、サーバー上で頻繁に利用されています。仮想化と統合を利用することで、Oracle VM Server for SPARCなどの特定の認定仮想環境で、Oracle Databaseを実行できます。
- **Oracle VM Server for SPARC：**Oracle VM Server for SPARCは、幅広い[SPARCプラットフォーム](#)でサポートされています。

Oracle VM Server for SPARCを使用すると、1つの物理システムに複数の仮想システムを作成できます。各仮想システムは**論理ドメイン**、または単に**ドメイン**と呼ばれます。各ドメインは、それぞれ独自のOracle Solaris 10 OSまたはOracle Solaris 11 OSインスタンスを実行します。Oracle VM Server for SPARCでは、ワークロードに対してCPUリソースとメモリ・リソースを細かく割り当てることもできます。また、Oracle VM Server for SPARCを利用して、稼働中のドメインをシステム間で移行することもできます。これにより、稼働中のアプリケーションの中断を最小限に抑えられるとともに、本番環境を管理するための強力なプラットフォームが得られます。

ゲスト・ドメインで稼働するOracle Solaris OSソフトウェアのバージョンは、制御ドメインで稼働するOracle Solaris OSバージョンの影響を受けません。つまり、制御ドメインでOracle Solaris 10 OSが稼働する場合にも、ゲスト・ドメインでOracle Solaris 11 OSを稼働させることができます。また、制御ドメインでOracle Solaris 11 OSが稼働する場合にも、ゲスト・ドメインでOracle Solaris 10 OSを稼働させることができます。

- ・ **ライブ・マイグレーション** : Oracle VM Server for SPARC 2.1リリースより、アクティブ・ドメインを稼働させながら、そのドメインをシステム間で移行できます。ライブ・マイグレーションではSSLで論理ドメイン・メモリ構造を暗号化してネットワーク上でセキュアに送信します。そのため、メモリ内容の送信中にも重要な情報が不正アクセスから保護されます。

移行が開始されるホストは「ソース・マシン」、ドメインの移行先のホストは「ターゲット・マシン」と呼びます。移行の進行中、ターゲット・マシンで作成されるドメインは「移行先ドメイン」と呼びます。

ライブ・マイグレーションの要件や、ソース・マシンとターゲット・マシンでのCPU、メモリ、I/Oに関する制約事項については、[Oracle VM Server for SPARC 3.1リリース・ノート](#)の“ライブ・ドメイン・マイグレーションの要件”を参照してください。

ライブ・マイグレーション機能には次のような利点があります。

**システム保守**—ライブ・マイグレーションを利用して、停止時間を最小限に抑えることができます。Oracle VM Server for SPARCには、多くの保守作業中にシステムが稼働を続けることのできる、複数のサービス・ドメインなどの可用性機能が備わっています。しかし、ファームウェアのアップデートなどによって機器をシャットダウンする必要がある場合は、ライブ・マイグレーションを利用することで、サービスを中断せずに稼働中のアプリケーションを他のサーバーに移行できます。

**ハードウェア・リソースのアップグレード**—ライブ・マイグレーションを利用して、物理メモリ容量がより多いマシンやCPU処理能力が高いマシン、あるいはI/O処理に優れたサブシステムを搭載するマシンにアクティブ・ドメインを移行できます。このシステム変更では、アプリケーションを再起動してそのデータ構造をメモリに再ロードすることなく、メモリ構造がウォーム状態で維持されるため、アプリケーション・パフォーマンスが向上します。たとえば、ライブ・マイグレーションの実施後も、Oracle Databaseではシステム・グローバル領域 (SGA) などのメモリ構造がマイグレーション前と同じようにレイアウトされます。メモリ構造の再ビルドやディスクからの再ロードは不要です。容量がより多いサーバーに重要度の高いドメインを移行する他にも、重要度の低いドメインのライブ・マイグレーションによってリソースを解放する方法もあります。また、リソースの動的な再割当てを行って、CPUリソースやメモリ・リソースを、重要度の低いドメインから重要度の高いドメインに移行する方法もあります。場合によっては、ドメインの移行よりも簡単に迅速に実行できます。

**長時間の計算**—マイグレーションの進行中にアプリケーションをシャットダウンする必要がないため、マイグレーション実施中にアプリケーションの可用性を最大限に引き出すことができます。たとえば、金融データ分析や科学計算などの長時間の計算を実行するアプリケーションの移行でも、最初からやり直す必要はなく、中間データを保存してチェックポイント手順をリストアする必要もありません。

**投資収益率の向上**—ライブ・マイグレーションを利用して、テクノロジー更新やアップグレードの一環として旧型のSPARCシステムから新型のSPARCシステムにアプリケーションを移行できます。既存のゲスト・ドメインを単一システムに統合して、電力コストや冷却コスト、スペースを削減しながらパフォーマンスを改善できます。CPU間移行機能を利用すれば、UltraSPARC T2 PlusやSPARC T3といった旧型のCPUアーキテクチャを搭載したシステムから、SPARC T4 CPUを搭載したシステムにゲスト・ドメインを移行できます。SPARC T4、SPARC T5、SPARC M5-32、SPARC M6-32のドメインについても、サーバー間でライブ・マイグレーションを実行できます。詳しくは、『CPU間移行』を参照してください。

## ハードウェアとソフトウェアの要件

はじめに、ソフトウェアとハードウェアが次の要件を満たしていることを確認してください。

### Oracle Databaseの要件

Oracle Solaris 10 Oracle VM Server for SPARCシステム上でOracle Databaseシングル・インスタンス構成を実行するには、Oracle Database 11g Release 2 (Oracle 11.2.0.1)以降を利用する必要があります。Oracle Solaris 11 OSの場合は、Oracle Database 11g Release 2 (Oracle 11.2.0.3)以降を利用する必要があります。特定のバージョンがサポートされているかどうか、また追加要件があるかどうかを確認するには、[Oracle Databaseのドキュメント](#)を参照してください。また、[My Oracle Support](#)も参照してください。

### ハードウェア構成

Oracle VM Server for SPARCをサポートしているすべてのサーバーで、Oracle Databaseを使用できます。選択する構成および配置オプションによって、外部ストレージ・アレイやネットワーク・スイッチなどの追加ハードウェアが必要になる場合があります。

図1に、ライブ・マイグレーション機能に準拠する一般的なハードウェア構成とケーブル配線を示します。このホワイト・ペーパーは、次のハードウェア構成に基づいて記述されています。

- 64GBのメモリと1個のSPARC T4プロセッサを備えたOracle SPARC T4-1サーバー2台。
- 各プロセッサに8CPUコア、各コアに8個のCPUスレッドを搭載。そのため、各システムで64個のCPUスレッド (8×8) が実行されます。サーバーごとに1つの内部ディスクが制御ドメインのシステム・ディスクとして使用されます。
- オラクルのSun Storage 2540-M2ストレージ・アレイ1台。300GBのSASディスク・ドライブ12台が搭載されています。ストレージ・アレイは両方のサーバーに接続されており、各サーバーはLUNがマッピングされた同じアレイ・コントローラ (AまたはB) に接続されています。このストレージ・アレイは、第1 LUN (LUN 0) をゲスト・ドメインのシステム・ディスクとして使用し、第2 LUN (LUN 1) をOracle Databaseのデータ保存に使用します。

- 4GB FC-ALホスト・バス・アダプタ (HBA) 2つ (サーバー1台につき1つ)。これらのHBAはストレージ・アレイを両方のサーバーに接続するために使用されます。
- サーバー1台につき4つのオンボード1ギガビット・イーサネット・ネットワーク・インタフェース。このインタフェースは、ネットワーク通信に使用されます。

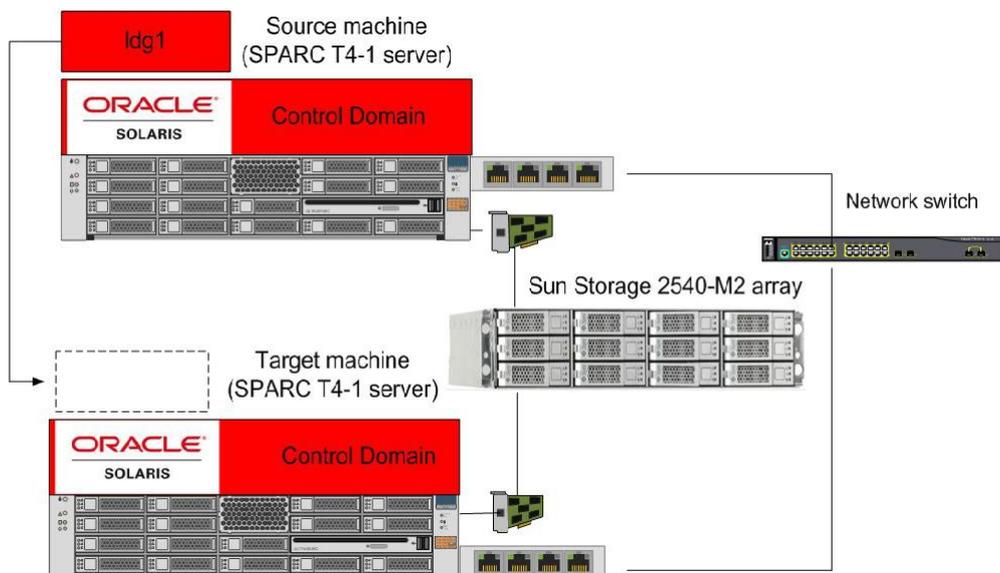


図1—ハードウェア構成とケーブル配線

#### Oracle VM Server for SPARCでのOracle Databaseの実行

Oracle Databaseを実行しているゲスト・ドメインには、最小でも16個の仮想CPU (2CPUコア) と4GBメモリを構成する必要があります。これらのCPUとメモリの割当ては、説明の目的で最小限にとどめていますが、お使いの環境によってはより大規模な設定が必要になります。

Oracleシングル・インスタンス・ゲスト・ドメインに仮想デバイス (仮想ディスクや仮想ネットワーク) を提供する各制御ドメインには、最小でも8個の仮想CPU (1CPUコア) と4GBメモリを構成する必要があります。これらのCPUとメモリの割当ては、説明の目的で最小限のものを使用していますが、お使いの環境によってはより大規模な設定が必要になります。

#### 構成の詳細

このホワイト・ペーパーで使用した例では、各ソース・サーバーに制御ドメインとOracleデータベースをホストする1つのゲスト・ドメイン (ldg1) が構成されています。制御ドメイン (primary) は、I/Oドメインとしても機能します。また、制御ドメインはサービス・ドメインとしても使用され、ゲスト・ドメインに仮想デバイス・サービス (仮想ディスクや仮想ネットワーク) を提供します。

サービス・ドメインとして機能するため、制御ドメインは次の仮想デバイス・サービスを持ちます。

- 物理ディスクを仮想ディスクとしてゲスト・ドメインにエクスポートする、1つの仮想ディスク・サービス (primary-vds0)

- ゲスト・ドメインの仮想コンソールへのアクセスを提供する、1つの仮想コンソール・コンセントレータ・サービス (`primary-vcc0`)。このサービスは5000~5100のポート範囲を使用して、仮想コンソール・アクセスを提供します。
- プライマリ・ネットワーク・インタフェース (`igb0`) に関連付けられた、1つの仮想スイッチ・サービス (`primary-vsw0`)

## ソフトウェアのインストールと構成

以降の項では、ライブ・マイグレーション用の環境を構成する方法について説明します。

### Oracle Solaris OSのインストール

新型のSPARCサーバーには、Oracle Solaris OSが事前にインストールされています。適切なリリースのOracle Solarisがインストールされており、必要なパッチが適用されていることを確認してください。

SPARCシステムにOracle Solaris OSを再インストールする方法については、[Oracle Solaris 10](#)または[Oracle Solaris 11のドキュメント](#)を参照してください。このホワイト・ペーパーの例では、サーバーの最初の内部ディスクにオペレーティング・システムがインストールされます。

Oracle Solaris OSをインストールしたら、Oracle VM Server for SPARCソフトウェアを構成して有効にします。

### Oracle VM Server for SPARCソフトウェアのインストール

Oracle Solaris 11が稼働する新型のSPARCサーバーには、Oracle VM Server for SPARCソフトウェアが事前にインストールされています。Oracle Solaris 10 OSがサーバーにインストールされている場合は、Oracle VM Server for SPARCソフトウェアのインストールが必要になる場合があります。いずれの場合も、[『Oracle VM Server for SPARC 3.1管理ガイド』の“ソフトウェアのインストールおよび有効化”](#)を参照してください。Oracle VM Server for SPARCソフトウェアのインストールに必要な手順や、サーバーに必要なファームウェア・レベルについて詳細に説明しています。

### ストレージ・アレイの構成

Sun Storage 2540-M2ストレージ・アレイは制御ドメインに接続されており、この制御ドメインによって2つのLUN (LUN 0とLUN 1) が仮想ディスクとしてゲスト・ドメインにエクスポートされます。

ゲスト・ドメインは第1LUN (LUN 0) をオペレーティング・システムのシステム・ディスクとして、第2LUN (LUN 1) をOracle Databaseデータファイルを保存するディスクとして使用します。図2に、ソース・マシンとターゲット・マシンのストレージ・レイアウトを示します。

Source machine (SPARC T4-1 server)

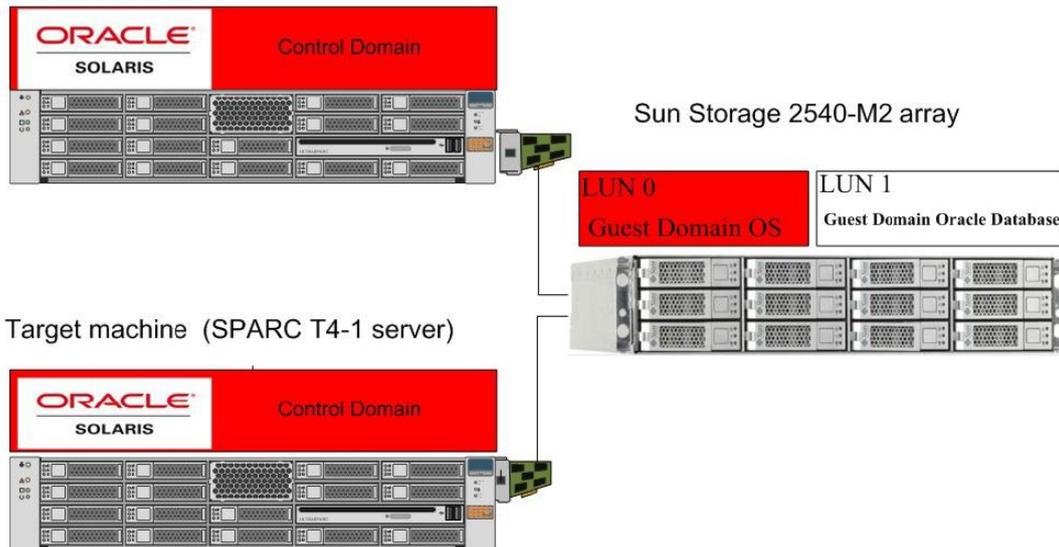


図2-ストレージ構成

### 制御ドメインの構成

Oracle VM Server for SPARCソフトウェアのインストール後、現在のシステムを制御ドメインとして再構成します。この手順については、管理ガイドの“[サービスおよび制御ドメインの設定](#)”の章に記載されています。次に、手順の概要を示します。

それぞれの物理サーバー上で次のタスクを実行します。

1. 仮想ネットワーク端末サーバー・デーモン (vntsd) で使用する仮想コンソール・コンセントレータ (vcc) サービスを、すべての論理ドメイン・コンソールのコンセントレータとして作成します。

```
primary# ldm add-vcc port-range=5000-5100 primary-vcc0 primary
```

2. 仮想ディスク・サーバー (vds) を作成します。

```
primary# ldm add-vds primary-vds0 primary
```

3. 仮想スイッチ・サービス (vsw) を作成し、論理ドメイン内の仮想ネットワーク (vnet) デバイスのネットワーキングを有効にします。Oracle Solaris 10制御ドメインでは、この仮想スイッチ用のネットワーク・インタフェースのデバイス・ドライバ名を使用します。

```
primary# ldm add-vsw net-dev=igb0 primary-vsw0 primary
```

次に、Oracle Solaris 11システム上のプライマリ・ネットワーク・インタフェースの例を示します。このシステムのプライマリ・ネットワーク・インタフェースはnet0であり、net0はigb0上のデータ・リンクのバニティ名です。このネットワーク・インタフェースを使用して、仮想スイッチ・サービス (vsw) を作成します。

```

primary# dladm show-phys
LINK      MEDIA      STATE      SPEED      DUPLEX      DEVICE
net0      Ethernet  up         1000      full       igb0
net1      Ethernet  up         1000      full       igb1
primary# ldm add-vsw net-dev=net0 primary-vsw0 primary

```

4. `ldm list-services`コマンドを使用してサービスが作成されたことを検証します。

```

primary# ldm list-services primary
VCC
  NAME          LDOM      PORT-RANGE
  primary-vcc0  primary  5000-5100
VSW
  NAME          LDOM      MAC          NET-DEV ID DEVICE
LINKPROP DEFAULT-VLAN-ID PVID VID ...
  primary-vsw0  primary  00:14:4f:f8:28:c2 igb0    0  switch@0  1
  1             1500 on
VDS
  NAME          LDOM      VOLUME  OPTIONS  MPGROUP  DEVICE
  primary-vds0  primary

```

Oracle VM Server for SPARCは、SSLを使用して移行トラフィックを暗号化することで、重要なデータを不正な利用から保護します。追加のハードウェアや専用ネットワークは不要です。

ソース・マシンとターゲット・マシンの両方でSPARCハードウェア暗号化アクセラレーションを利用すれば、移行処理の速度が向上します。速度が向上する理由は、CPUパワーが必要なSSL処理を、ソフトウェアの代わりにシリコン内に実装された暗号化アルゴリズムにオフロードできるからです。

**注**：SPARC T3以前のCPUでは、暗号化ユニットによって暗号化アクセラレーションが実装されています。この暗号化ユニットは、利用するドメインに対して割り当てする必要があります。

5. (T3以前のみ) 制御ドメインに暗号化デバイスがインストールされているかどうかを確認します。

```

primary# ldm list -o crypto primary

```

6. (T3以前のみ) 暗号化リソースを制御ドメインに割り当てます。ドメインに割り当てる各コアにつき1つの暗号化ユニットを割り当てます。

```

primary# ldm set-crypto 1 primary

```

7. CPUとメモリを制御ドメインに割り当てます。

ソース・マシンのprimaryドメインにCPUを追加することで、全体的な移行時間を短縮できます。最良のCPU割当て方法としては、SPARCコアの単位（最小でも1CPUコア）で増分しま

す。個々のCPUスレッドを割り当てることもできますが、primaryドメインには少なくとも8個のCPUスレッドを割り当てる必要があります。たとえば、次のコマンドは、制御ドメインに1コア（8個のCPU）と4GBのメモリを割り当てます。これらの値は、説明の目的で最小限のものを使用しています。

```
primary# ldm set-core 1 primary          # or, ldm set-vcpu 8 primary
primary# ldm start-reconf primary
primary# ldm set-memory 4G primary
```

8. システムを再起動します。

```
primary# init 6
```

システムをリブートすると、Oracle VM Server for SPARCが有効になり、システムが1つのドメイン（primary制御ドメイン）として構成されます。制御ドメインから、その他のドメインを作成し構成できます。

9. Logical Domains Manager (ldmd) サービスと仮想ネットワーク端末サーバー・デーモン (vntsd) サービスが有効になっていることを確認します。

```
primary# svcadm enable ldmd
primary# svcadm enable vntsd
```

10. 論理ドメイン・マシン構成をサービス・プロセッサ (SP) に追加して、電源を入れ直した後も構成が維持されるようにします。

たとえば、次のコマンドはinitialという構成を追加します。

```
primary# ldm add-config initial
```

11. 構成を使用する準備が整ったことを検証します。

```
primary# ldm list-config
factory-default
initial [current]
```

## ゲスト・ドメインの構成

制御ドメインの構成が終了した後に、Oracle Databaseノードとして使用するゲスト・ドメインを作成します。ソース・マシンとなる物理サーバー上にゲスト・ドメインを1つだけ作成します。その他の物理サーバーは最終的にターゲット・マシンとなります。

ゲスト・ドメイン (ldg1) の作成時には、次のリソースが割り当てられています。

- 8GBのメモリ
- 24個のCPUスレッド (3コア)
- 仮想スイッチprimary-vsw0に接続された、1つの仮想ネットワーク・インタフェース (vnet0)
- 第1仮想ディスク (ゲスト・ドメインではc0d0として表示される、ストレージ・アレイのLUN) 。ldg1ドメインはストレージ・アレイのLUN 0 (c2t6d0) を使用します。
- 第2仮想ディスク (ゲスト・ドメインではc0d1として表示される、ストレージ・アレイのLUN) 。ldg1ドメインはストレージ・アレイのLUN 1 (c2t6d1) を使用します。この仮想ディスクはOracle Databaseファイルの保存に使用されます。

次の手順を実行して、各ゲスト・ドメインを作成します。

1. ソース・マシン上の制御ドメインからldg1ゲスト・ドメインを作成します。

```
primary# ldm create ldg1
primary# ldm set-core 3 ldg1
primary# ldm set-memory 8G ldg1
primary# ldm add-vnet vnet0 primary-vsw0 ldg1
primary# ldm add-vdsdev /dev/dsk/c2t6d0s2      voll@primary-vds0
primary# ldm add-vdisk voll1 voll1@primary-vds0 ldg1
primary# ldm add-vdsdev /dev/dsk/c2t6d1s2 oradata@primary-vds0
primary# ldm add-vdisk oradata oradata@primary-vds0 ldg1
```

2. ドメインの作成後、次のコマンドを使用して、制御ドメインからゲスト・ドメインをソース・マシンにバインドし、ゲスト・ドメインを開始します。

```
primary# ldm bind ldg1
primary# ldm start ldg1
```

3. ドメインのコンソール・ポートを確認します。

```
primary# ldm ls ldg1

NAME          STATE          FLAGS          CONS   VCPU   MEMORY UTIL
UPTIME

ldg1          active         -n-----     5000   24    8G    0.0% 1h
1m
```

4. telnetコマンドを使用してldg1ドメインのコンソールにアクセスします。

```
primary# telnet localhost 5000
```

ゲスト・ドメインが開始されたら、ゲスト・ドメインに適切なOracle Solaris OSとパッチをインストールします。インストールは、ネットワーク経由またはDVDから、もしくはDVD ISOイメージを使用して実行できます。『Oracle VM Server for SPARC 3.1管理ガイド』を参照してください。

**Oracle Solaris 11のみ** : Oracle Solaris 11自動インストーラを利用して、ネットワーク・インストール中にパッケージをインストールできます。『Oracle Solaris 11システムのインストール』の“自動インストーラの使用方法”を参照してください。

**注** : LUNは多くの場合、サーバーごとに異なる名前が表示されます。制御ドメインではLUN名が一致する必要はありませんが、両方のシステム上の制御ドメインで同じLUN (LUN 0とLUN 1) を同じボリューム名 (vol1およびoradata) で仮想化する必要があります。

#### 例1-ドメインの構成

次の例では、ldg1というドメインが24個のCPU、8GBのメモリ、Sun Storage 2540-M2ストレージ・アレイの2つのLUNでどのように構成されているのかが示されています。

```
primary# ldm ls -l ldg1

NAME          STATE          FLAGS   CONS   VCPU   MEMORY   UTIL   UPTIME
ldg1          active         -n----- 5000   24    8G    0.0% 1h 1m
SOFTSTATE
Solaris running
MAC
    00:14:4f:fb:96:89
HOSTID
```

0x84fb9689

CONTROL

failure-policy=ignore

DEPENDENCY

master=

VCPU

VID	PID	UTIL	STRAND
0	16	0.2%	100%
1	17	0.0%	100%
2	18	0.5%	100%
3	19	0.0%	100%
4	20	0.0%	100%
5	21	0.0%	100%
6	22	0.1%	100%
7	23	0.0%	100%
8	24	0.1%	100%
9	25	0.0%	100%
10	26	0.0%	100%
11	27	0.0%	100%
12	28	0.0%	100%
13	29	0.0%	100%
14	30	0.0%	100%
15	31	0.0%	100%
16	32	0.0%	100%
17	33	0.0%	100%
18	34	0.0%	100%
19	35	0.0%	100%
20	36	0.0%	100%
21	37	0.0%	100%
22	38	0.0%	100%
23	39	0.1%	100%

MEMORY

RA	PA	SIZE
0x8000000	0x408000000	8G

VARIABLES

boot-device=vdisk1

```

keyboard-layout=US-English

NETWORK

      NAME          SERVICE          ID  DEVICE      MAC
MODE  FVID  VID  MTU  LINKPROP
      vnet0      primary-vsw0@primary  0  network@0  00:14:4f:f9:c0:62
1
      1500

DISK

      NAME          VOLUME          TOUT ID  DEVICE  SERVER
MPGROUP
      vdisk1      voll@primary-vds0      0  disk@0  primary
      oradata      oradata@primary-vds0  1  disk@1  primary

VCONS

      NAME          SERVICE          PORT
      ldg1      primary-vcc0@primary  5000

```

## 例2-制御ドメインの構成

次の例は、制御ドメインの構成を表示しています。ldm list-servicesコマンドを使用してサービスが作成されたことを検証します。

```

primary# ldm ls-services primary
VCC
      NAME          LDOM          PORT-RANGE
      primary-vcc0  primary      5000-5100
VSW
      NAME          LDOM          MAC          NET-DEV  ID
DEVICE  LINKPROP  DEFAULT-VLAN-ID  PVID  VID          MTU  MODE
      primary-vsw0  primary      00:14:4f:f9:32:b0  igb0  0
switch@0      1          1          1500
VDS
      NAME          LDOM          VOLUME          OPTIONS          MPGROUP
DEVICE
      primary-vds0  primary      voll          /dev/dsk/c2t6d0s2
      oradata      /dev/dsk/c2t6d1s2

```

## ゲスト・ドメインでのOracle Solaris OSのインストール

ゲスト・ドメインに適切なリリースのOracle Solaris OSをインストールし、必要なパッチが適用されていることを確認します。

ゲスト・ドメインでOracle Solaris OSをインストールしたら、システムを構成し、Oracle Databaseソフトウェアの使用を有効にします。

## Oracle Databaseのインストール

Oracle DatabaseソフトウェアのOracle VM Server for SPARCシステムへのインストールは、標準のOracle Databaseインストールと類似しています。はじめにOracle Databaseソフトウェアをインストールし、次に最新のパッチ・セットを適用します。第2LUNを使用して、Oracle Databaseデータファイルを保存するためのUFSまたはZFSファイル・システムを作成します。ZFSは、Oracle Solaris 11 OSのデフォルトのルート・ファイル・システムです。

Oracle DatabaseでZFSを使用する方法について、詳しくは『[Configuring Oracle Solaris ZFS for an Oracle Database](#)』を参照してください。

## ターゲット・マシンの準備

1. 次の各項で説明する手順を実行し、ターゲット・マシンを準備します。
  - “Oracle VM Server for SPARCソフトウェアのインストール”
  - “制御ドメインの構成”
2. 移行元ドメインで使用されるすべての仮想I/Oサービスがターゲット・マシンで使用できることを確認します。
3. ターゲット・マシンが同じストレージからLUN 0とLUN 1の両方にアクセスできることを確認します。

ターゲット・マシンに第1LUN (LUN 0) と第2LUN (LUN 1) を追加するには、ターゲット・マシンの制御ドメインから次のコマンドを実行します。

```
primary# ldm add-vdsdev /dev/dsk/c2t6d0s2 vol1@primary-vds0
```

```
primary# ldm add-vdsdev /dev/dsk/c2t6d1s2 oradata@primary-vds0
```

**注:** LUNは多くの場合、サーバーごとに異なる名前が表示されます。制御ドメインではLUN名が一致する必要はありませんが、両方のマシン上の制御ドメインで同じLUN (LUN 0とLUN 1) を同じボリューム名 (vol1およびoradata) で仮想化する必要があります。

4. 移行元ドメイン内の各仮想ネットワーク・デバイスと対応する仮想ネットワーク・スイッチをターゲット・マシンにインストールします。

```
primary# ldm add-vsw net-dev=igb0 primary-vsw0 primary
```

Oracle Solaris 11 OSの場合は、次のコマンドを実行します。

```
primary# ldm add-vsw net-dev=net0 primary-vsw0 primary
```

## CPU間移行

ソース・マシンとターゲット・マシンのプロセッサ・タイプが異なる場合でも、ドメインを移行できます。たとえば、UltraSPARC T2 PlusやSPARC T3 CPUが搭載されたシステムからSPARC T4 CPUが搭載されたシステムへのゲスト・ドメインの移行、あるいはSPARC T4、SPARC T5、M5-32、M6-32のサーバー間でのゲスト・ドメインの移行を実行できます。

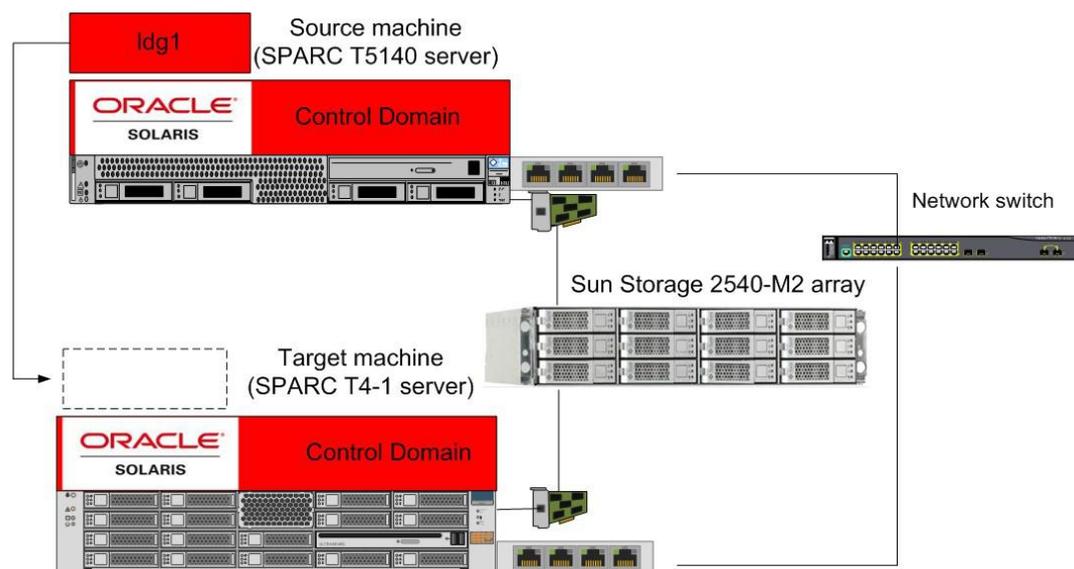


図3-ストレージ構成

**注：**CPU間移行機能を利用するには、ドメインがバインド中または非アクティブのときに、ドメイン上のcpu-archプロパティの値を、次のいずれかの値に設定する必要があります。

- generic: 共通のCPUハードウェア機能を利用して、CPUタイプに依存せずにゲスト・ドメインを移行できます。UltraSPARC T2、UltraSPARC T2+、SPARC T3、SPARC T4またはそれ以降のサーバー間でドメインのライブ・マイグレーションを実行する場合にも利用できます。
- migration-class1: 新型のSPARCサーバーに共通するCPU固有のハードウェア機能を利用します。migration-class1は、SPARC T4プラットフォーム以降のプラットフォーム間で移行する場合に、パフォーマンスと柔軟性のバランスがとれた設定です。SPARC T4プラットフォームは、ハードウェア暗号化命令など、より多くのCPU機能を備えています。SPARC T4以降であれば、他のCPUタイプに移行することもできます。

- `sparc64-class1`: SPARC64プラットフォーム向けのCPU間移行ファミリーです。この値は、Fujitsu M10システムにのみ対応します。
- `native`: CPU固有のハードウェア機能を利用して、同じタイプのCPUが搭載されたプラットフォーム間でのみゲスト・ドメインを移行できます。`native`はデフォルト値であり、常に最適なCPUパフォーマンスが発揮されます。

`generic`を使用する場合は、他の値と比較してパフォーマンスが低下する可能性があります。これは、新型のCPUタイプにのみ存在する機能をゲスト・ドメインが利用しない場合に発生します。`generic`設定では、新型タイプのCPUを利用するシステムと、旧型タイプのCPUを利用するシステムの間でドメインを移行できる柔軟性が得られます。

ドメインがバインド中または非アクティブ状態のときに、`cpu-arch`プロパティの値を変更するには、`ldm set-domain`コマンドを次のように使用します。

```
primary# ldm stop ldg1

primary# ldm set-domain cpu-arch=migration-class1 ldg1

primary# ldm start ldg1
```

以上の前提条件が満たされれば、ライブ・マイグレーションのプロセスに進むことができます。

## ライブ・マイグレーションのプロセス

ソース・マシン上のLogical Domains Managerは、ドメイン移行リクエストを受け取り、ターゲット・マシン上で稼働するLogical Domains Managerとのセキュアなネットワーク接続を確立します。この接続の確立後、移行が実行されます。移行は次の5つのフェーズに分かれます。

- **フェーズ1**: ターゲット・マシン上で稼働するLogical Domains Managerと接続した後に、ソース・マシンおよび移行元ドメインに関する情報がターゲット・マシンに送信されます。この情報を利用して、移行が可能かどうかを判別する一連のチェックが実行されます。

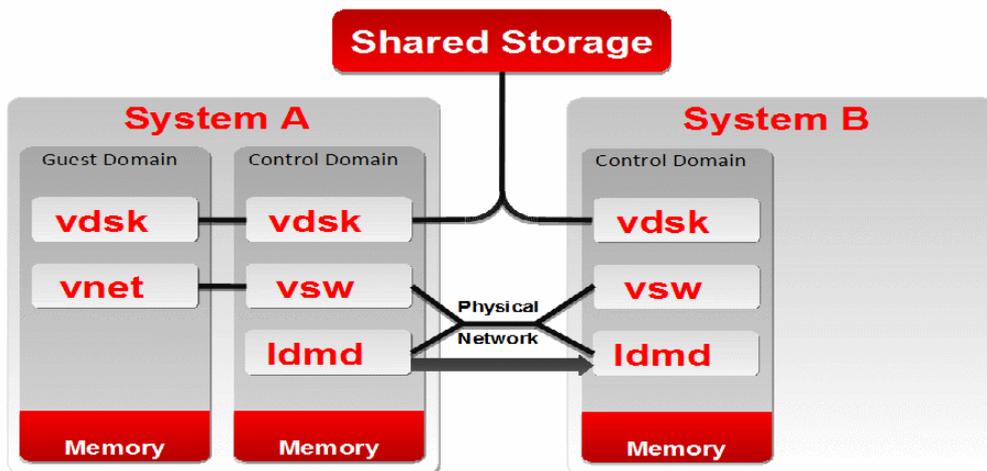


図4-ライブ・マイグレーションのフェーズ1

- ・ **フェーズ2**：フェーズ1のすべてのチェックを通過したら、ソース・マシンとターゲット・マシンで移行の準備が行われます。ターゲット・マシン上に、ソース・ドメインを受け取るための非アクティブなドメインが作成されます。

ソース・マシンで、アクティブなゲスト・ドメインのどのメモリ・ブロックが移行処理中に変更されるのかについて追跡を開始します。移行元ドメインが非アクティブまたはバインド中の場合、移行処理はフェーズ5に進みます。

- ・ **フェーズ3**：ドメインのランタイム状態に関する情報がすべてターゲットに送信されます。この情報には、CPUの状態とドメインのメモリが含まれます。ソース・マシンはターゲット・マシンにすべてのメモリ・ブロックを送信します。ゲスト・ドメインのメモリがターゲット・マシンに送信されると、ソース・マシンは変更中としてマークされているブロックの送信を開始します。ゲスト・ドメインの稼働が続いているため、この送信処理は複数回反復されます。
- ・ **フェーズ4**：移行元ドメインが一時停止されます。この時点で、ソース・ドメインのその他の変更済みメモリ・ブロックと状態情報がターゲット・マシンに送信されます。
- ・ **フェーズ5**：すべての状態情報が送信された後に、ターゲット・ドメインが実行を再開する際のハンドオフが行われ、ソース・ドメインが破棄されます。この時点から、ターゲット・ドメインが唯一の実行中ドメインのバージョンとなります。

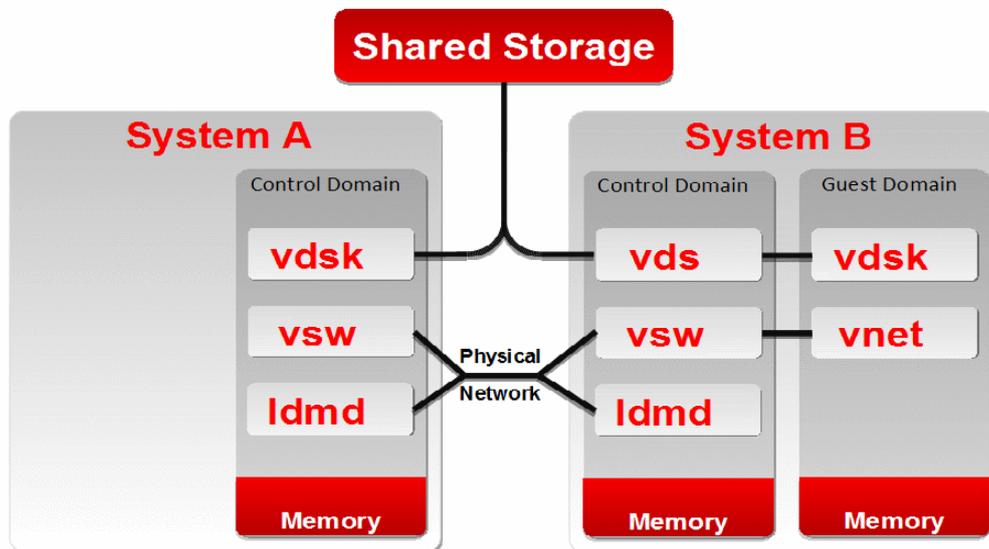


図5-ライブ・マイグレーションのフェーズ5

## ライブ・マイグレーションの例

この項の例では、`ldm migrate-domain`コマンドを使用して、ゲスト・ドメインをシステム間で移行します。

オプションとして、ターゲット・マシン上で認証するための代替ユーザーを指定できます。『*Oracle VM Server for SPARC管理ガイド*』の“[ユーザーへの役割の割り当て](#)”を参照してください。

`ldm migrate-domain -n`コマンドを使用すると、移行の予行演習を行うことができます。このコマンドは移行のチェックを実行しますが、指定したドメインの移行は実施しません。満たしていない要件があればエラーとして報告されるため、実際の移行を試みる前に構成上のエラーを修正できます。

次の例では、ドメイン移行を実行するさまざまな方法を示します。

- t4-1というターゲット・マシンに対するldg1ドメインの移行の予行演習を行います。

```
primary# ldm migrate-domain -n ldg1 t4-1
```

```
Target Password: password
```

- t4-1というターゲット・マシンに対するldg1ドメインの移行を実行します。

```
primary# ldm migrate-domain ldg1 t4-1
```

```
Target Password: password
```

- t4-1というターゲット・マシンに対するldg1ゲスト・ドメインの移行を、非対話形式で実行します。-pオプションで、t4-1ターゲット・マシンにアクセスするためのターゲット・マシンのスーパーユーザーのパスワードをプレーン・テキストとして含むファイルの名前(pfile)を指定します。

```
primary# ldm migrate-domain -p pfile ldg1 t4-1
```

**注：**スーパーユーザーのパスワードを含むファイルは、root所有者または権限を持つユーザーだけが読み取れるように、必ず保護してください。

```
primary# chmod 400 pfile
```

- t4-1というターゲット・マシンに対するldg1ドメインの移行を実行し、その際にt4-1マシン上の認証に使用する代替ユーザー(tracy)を指定します。

```
primary# ldm migrate-domain ldg1 tracy@t4-1
```

ユーザーへの認証の割当てについては、[『Oracle VM Server for SPARC 3.1管理ガイド』の“ユーザー権利プロファイルの管理”](#)を参照してください。

- 移行処理の進捗を監視します。

```
primary# ldm list -o status ldg1
```

**注：**移行にかかる時間は、割り当てられたメモリ容量やネットワーク利用率によって異なります。

#### ワークロードの説明とパフォーマンスの結果

[SwingBenchキット](#)には、Oracleデータベースのストレス・テストを目的として設計された負荷生成ツールとベンチマークが含まれています。SwingBenchの同梱コードには、OrderEntry、SalesHistory、CallingCircle、StressTestの各ベンチマークが含まれます。

このホワイト・ペーパーでは、SwingBench OrderEntryワークロードを利用して、移行中のゲスト・ドメインに対するワークロードを生成しました。OrderEntryはOracle Database 11gに含まれるoeスキーマをベースとしています。このワークロードは少数の表に対する重度の競合を引き起こすものであり、システムのインターコネクトとメモリに対してストレスをかけることを目的としています。

ゲスト・ドメインに対して生成されるワークロードは次のとおりです。

- ディスク上のデータベース・サイズ : 30GB
- システム・グローバル領域 (sga\_target) : 18GB

SGAは1つのOracle Databaseインスタンスのデータおよび制御情報を含む共有メモリ構造のグループです。

- ・ ゲスト・ドメインに割り当てるCPU数：24個のCPUスレッド（3コア）
- ・ ゲスト・ドメインに割り当てるメモリ容量：24GB
- ・ ワークロード用のワークロード・ユーザー数（users）：50ユーザー
- ・ 思考時間（ユーザーが行う、あるアクションから次のアクションまでの時間）：100ミリ秒
- ・ ワークロードの実行時間：30分間

表1に、この特定のワークロードを利用する2台のSPARC T4-1マシン間で実行したライブ・マイグレーション処理の結果を示します。

**表1：SPARC T4-1ライブ・マイグレーションの結果**

制御ドメイン上のCPU数	移行時間合計	一時停止時間	ゲスト・ドメインのCPU利用率
8 CPU	8分12秒	26秒	70.00%
16 CPU	4分2秒	13秒	80.00%
24 CPU	2分3秒	7秒	85.00%

#### ドメインの移行を開始する前に

マシン間でドメインを移行する際に一般的に見受けられるライブ・マイグレーション処理について次に説明します。移行にかかる時間は制御ドメインに割り当てられたCPU数とメモリ容量、ワークロード・タイプによって変化するため、実際の結果は少し異なることがあります。

- ・ Oracle VM Server for SPARCライブ・マイグレーションは、共有ネットワークやセキュアでないネットワークなど、あらゆるタイプのネットワークで動作するように設計されています。このホワイト・ペーパー向けにメインのNICで実行された移行では、専用NICで実行された移行と比較して、パフォーマンスは低下しませんでした。最大のパフォーマンスを発揮するためには、10ギガビット・イーサネットなど、飽和状態にない高速ネットワーク・インタフェースを使用します。
- ・ ゲスト・ドメインのCPUに対するシステム・メモリ追跡結果の観察には、mpstatコマンドを使用します。各CPUの利用率はUSR列で確認します。
- ・ ゲスト・ドメインのCPU負荷はライブ・マイグレーション中に上昇します。これは、ゲスト・ドメインのCPUスレッドがハイパーバイザ内でメモリ変化状況の追跡を実行するためです。この追跡は、アプリケーション・プロセスのCPU時間として報告されます。
- ・ 最適な構成としては、データベースを実行するゲスト・ドメインと、ライブ・マイグレーションを統合的に実施する制御ドメインの両方に十分なCPUリソースを割り当てます。このホワイト・ペーパーと同様のデータベース・ワークロードの場合、ゲスト・ドメイン（データベース）にSPARC T4システムの24個のCPU（3コア）を割り当てて、制御ドメインに少なくとも1個のCPUコア（8個のCPUスレッド）を割り当てます。制御ドメインにさらにCPUコアを追加すると、ライブ・マイグレーションの実行時間が短縮します。最良の方法としては、自分の環境固有のワークロードを利用した独自のベンチマークを実行して、適切なCPUリソースの割当てを決定します。制御ドメインにCPUを追加する方法については、表1と“CPUの動的な再構成の利用”を参照してください。

- 移行にかかる時間は、次の要因にも依存します。ゲスト・ドメインに割り当てたメモリ容量、データベースSGAのサイズ、データベース・ブロック変更数、ディスクに対する読取り/書込みI/Oの量です。
- 自動ワークロード・リポジトリ (AWR) ツールを使用して、データベース・ブロック変更数を監視できます。生成されるAWRレポートのInstance Activity Statsセクションにある `db_block_changes` 情報を確認します。『[Oracle Databaseパフォーマンス・チューニング・ガイド](#)』の5.2“自動ワークロード・リポジトリの概要”の項を参照してください。
- ネットワーク時刻の同期プロトコル (NTP) による時間同期サービスを使用するようにゲスト・ドメインを構成して、ゲスト・ドメイン上の時刻が移行処理の完了後に正常に再設定されるようにします。Oracle Solaris 10システムまたはOracle Solaris 11システムでのNTPセットアップの例については、“NTPの構成”の項を参照してください。
- ライブ・マイグレーションは緊急ツールとしては使用しないでください。それでも使用する必要がある場合は、事前にストレージ・パスを設定して環境を準備してください。また、本番環境にできるだけ近い環境でライブ・マイグレーション処理を実行してください。このような準備を事前に実施することで、移行にどの程度の時間がかかるかを把握できます。

## 関連情報

ここでは、NTPの構成やCPUの動的な再構成などの、Oracle VM for SPARCライブ・マイグレーション機能に関連する情報を示します。

### NTPの構成

NTPを使用して、移行元ドメイン上のシステム・クロックを同期します。

この例では、ソース・マシン上の制御ドメインをタイム・ソースとして使用し、さらにNTPサーバーとして構成しています。最良の方法としては、専用の時間同期ソースとなるNTPサーバーを選択します。こうすれば、マシンが計画保守のためにダウンした場合に他のサービスに悪影響を及ぼすことはありません。

#### 例3-NTPサーバーの構成

次の例で、NTPサーバーの構成方法を示します。

```
# grep -v ^# /etc/inet/ntp.conf
server 127.127.1.0 prefer
broadcast 224.0.1.1 ttl 4
enable auth monitor
driftfile /var/ntp/ntp.drift statsdir /var/ntp/ntpstats/
filegen peerstats file peerstats type day enable
filegen loopstats file loopstats type day enable
filegen clockstats file clockstats type day
enable keys /etc/inet/ntp.keys
trustedkey 0
requestkey 0
```

```
controlkey 0
# touch /var/ntp/ntp.drift
# svcadm enable ntp
```

次の例で、NTPクライアントの構成方法を示します。

```
# grep -v ^# /etc/inet/ntp.conf
server source prefer
slewalways yes
disable pll
# svcadm enable ntp
```

#### CPUの動的な再構成の利用

Oracle VM Server for SPARCでは、CPUの動的な再構成がサポートされています。ldm add-core、ldm rm-core、ldm set-core、ldm add-vcpu、ldm rm-vcpu、ldm-set-vcpuの各コマンドを使用すると、制御ドメインなどの任意のアクティブ・ドメインからCPUを動的に追加または削除できます。

これらのコマンドを制御ドメインから実行することで、CPUの割当て数の増減が可能です。制御ドメインでCPU数を増やすと、特にソース・システムにおいて、移行にかかる時間を短縮できます。UltraSPARC T2、UltraSPARC T2 Plus、SPARC T3プラットフォームなどの個別の暗号化ユニットのあるシステムでは、CPUリソースを動的に追加する際に、制御ドメインに暗号化ユニットを追加する必要があります。

次の例で、制御ドメインに1CPUコア（8個のCPUスレッド）を追加する方法を示します。

```
primary# ldm add-core 1 primary
```

移行が完了した後に、制御ドメインから仮想CPUを削除して別のドメインにそのCPUを割り当てることができます。次の例では制御ドメインから1CPUコアを削除し、ldg1ゲスト・ドメインにそのCPUコアを割り当てます。

```
primary# ldm rm-core 1 primary
```

```
primary# ldm add-core 1 ldg1
```

**注：**CPU数を削減するときは、ドメインのワークロードを効率的に処理できるだけの十分な数のCPUを残すようにしてください。たとえば、Oracle Databaseシングル・インスタンスのゲスト・ドメインは常に1CPUコア以上が必要です。

## 結論

IT部門は、適正なハードウェアとソフトウェアを選択し、アプリケーション可用性の最適な管理を行い、データセンターのリソースを最適化することで、戦術的、戦略的な目標を達成できます。

このホワイト・ペーパーでは、Oracle Database 11g Release 2シングル・インスタンス・データベースの本番環境ライフ・サイクルを管理するためにOracle VM Server for SPARCライブ・マイグレーション機能を利用する利点について説明しました。また、それらの利点を発揮するためのドメインの作成と構成、ストレージの構成、ソフトウェアなどの完全な構成プロセスについて示しました。ライブ・マイグレーション機能の利点を示すことに加え、本番環境での動的なハードウェア・リソース管理によって、計画停止中のアプリケーション可用性を高める方法や、ハードウェアおよびソフトウェアのコストを削減する方法について説明しました。

オラクルの仮想化ソリューションについて、詳しくは[オラクルの仮想化](#)を参照してください。

## 著者について

Orgad Kimchiは2007年9月にSun/Oracleに入社し、現在は独立系ソフトウェア・ベンダー（ISV）エンジニアリング部門で、オラクルのハードウェアおよびソフトウェアのパフォーマンスを向上させるOracleテクノロジーの導入に貢献しています。[Orgadのブログ](#)をご覧ください。

Roman Ivanovは2006年1月にSun/Oracleに入社し、現在はISVエンジニアリング部門で、オラクルのハードウェアのパフォーマンスを向上させるOracleテクノロジーの導入に貢献しています。[Romanのブログ](#)をご覧ください。

Jeff Savitは2000年2月にSun/Oracleに入社し、現在はOracle VM Product Management部門に属しています。[Jeffのブログ](#)をご覧ください。

## 参考資料

次に、このホワイト・ペーパーの参考資料を示します。

- Oracle VM Server for SPARC Webページ  
<http://www.oracle.com/jp/technologies/virtualization/oracle-vm-server-for-sparc/overview/index.html>
- Oracle VM Server for SPARCのドキュメント  
<http://www.oracle.com/technetwork/documentation/vm-sparc-194287.html>
- Oracle Databaseのドキュメント  
[http://docs.oracle.com/cd/E16338\\_01/index.htm](http://docs.oracle.com/cd/E16338_01/index.htm)
- My Oracle Support（登録が必要）  
<https://support.oracle.com>
- Oracle Solaris 10のドキュメント

<http://www.oracle.com/technetwork/documentation/solaris-10-192992.html>

- Oracle Solaris 11のドキュメント

<http://www.oracle.com/technetwork/documentation/solaris-11-192991.html>

- Oracle VMのダウンロード

<http://www.oracle.com/technetwork/jp/server-storage/vm/downloads/index.html>

- 『Configuring Oracle Solaris ZFS for an Oracle Database』

<http://www.oracle.com/technetwork/server-storage/solaris/config-solaris-zfs-wp-167894.pdf>

- SwingBench Webサイト

<http://dominicgiles.com/swingbench.html>

- 『Oracle Databaseパフォーマンス・チューニング・ガイド』

[http://docs.oracle.com/cd/E16338\\_01/server.112/b56312/toc.htm](http://docs.oracle.com/cd/E16338_01/server.112/b56312/toc.htm)

- Oracleの仮想化ソリューションのWebページ

<http://www.oracle.com/jp/technologies/virtualization/overview/index.html>

- Orgadのブログ

<https://blogs.oracle.com/vreality>

- Romanのブログ

<https://blogs.oracle.com/pomah>



Oracle VM Server for SPARCを利用した  
アプリケーション可用性の向上：  
Oracle Databaseの例  
2014年3月、バージョン1.3  
著者：Orgad Kimchi、Roman Ivanov、  
Jeff Savit

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

お問い合わせ窓口  
Oracle Direct

TEL 0120-155-096  
URL [oracle.com/jp/direct](http://oracle.com/jp/direct)



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

Copyright © 2014, 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。

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。0114

**Hardware and Software, Engineered to Work Together**