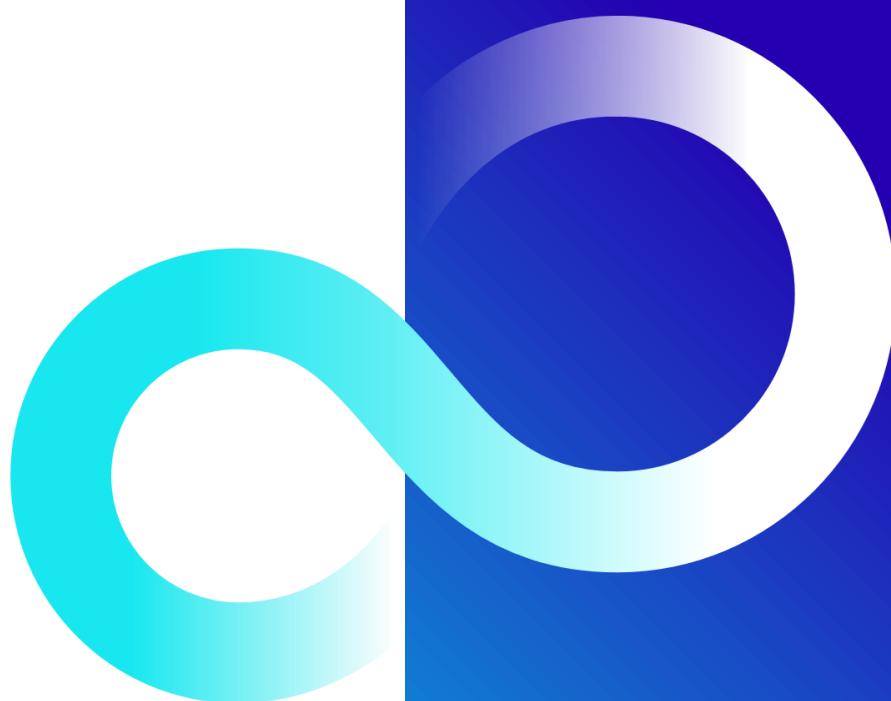


SPARC M12/M10
Building Blockを活用
した高可用性システム
(構築編)



2021年4月

第5版

富士通株式会社

■本資料について

- SPARC M12/M10 で物理パーティションの動的再構成機能を用いた Building Block の構築手順を紹介しています。別冊の「SPARC M12/M10 Building Block を活用した高可用性システム(運用編)」と合わせて、本手順書を活用してください。
- 本書に記載されているコマンドの実行結果は環境によって異なる場合があります。
- 本資料は、SPARC M12/M10, Oracle VM Server for SPARC 3.4 以降, Oracle Solaris11.3 を使用した手順を紹介しています。
- 物理パーティションの動的再構成機能の詳細については、下記 URL に掲載の各マニュアルをご参照下さい。

<https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/downloads/manual/index.html>

「SPARC M12/M10 システム運用・管理ガイド」

「SPARC M12/M10 ドメイン構築ガイド」

「SPARC M12/M10 XSCF リファレンスマニュアル」

■使用条件

- 著作権・商標権・その他の知的財産権について
コンテンツ(文書・画像・音声等)は、著作権・商標権・その他の知的財産権で保護されています。
本コンテンツは、個人的に使用する範囲でプリントアウトまたはダウンロードできます。ただし、これ以外の利用(ご自分のページへの再利用や他のサーバへのアップロード等)については、当社または権利者の許諾が必要となります。
- 保証の制限
本コンテンツについて、当社は、その正確性、商品性、ご利用目的への適合性等に関して保証するものではなく、そのご利用により生じた損害について、当社は法律上のいかなる責任も負いかねます。
本コンテンツは、予告なく変更・廃止されることがあります。

■商標について

- UNIX は、米国およびその他の国におけるオープン・グループの登録商標です。
- SPARC Enterprise、SPARC64 およびすべての SPARC 商標は、米国 SPARC International, Inc.のライセンスを受けて使用している、同社の米国およびその他の国における商標または登録商標です。
- Oracle と Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。
- その他各種製品名は、各社の製品名称、商標または登録商標です。

■留意事項

- 本ドキュメントの実施例で使用している値は、SPARC M12-2S を使用した場合の例で記載してあります。SPARC M10-4S を使用する場合は、装置仕様を確認の上、ご使用ください。

■本資料の位置づけ



目 次

1. はじめに	1
1.1. BB HA の概要	1
1.2. BB HA の要件	1
1.3. 本書で扱う構成	2
1.3.1. 制御ドメインのみの構成(従来型)	2
1.3.2. 制御ドメインと複数の I/O ルートドメインによる構成(集約型)	5
1.3.3. 制御ドメインと複数のゲストドメインによる構成(高集約型)	8
1.4. 構築の流れ	12
1.4.1. 従来型の構築の流れ	12
1.4.2. 集約型の構築の流れ	13
1.4.3. 高集約型の構築の流れ	14
2. 物理パーティションの設定	15
2.1. マスタ XSCF にログイン	15
2.2. 物理パーティション構成情報の作成	15
2.3. 物理パーティションヘシステムボードを割り当て	16
2.4. CPU コア アクティベーションキーの登録・割り当て	16
2.5. XSCF の時刻補正	18
2.6. 物理パーティションの設定確認	18
2.7. 物理パーティションの電源投入	19
2.8. 物理パーティションのコンソールに接続	19
3. 従来型の構築	20
3.1. Oracle Solaris と Oracle VM Server for SPARC のインストール	20
3.2. 制御ドメイン(グローバルゾーン)の構築	20
3.3. Oracle Solaris ゾーンの構築	26
3.3.1. ノングローバルゾーンの作成	27
3.3.2. Oracle Solaris のインストール	27
3.3.3. ノングローバルゾーンのシステム構成設定	29
3.3.4. ノングローバルゾーンの状態確認	30
4. 集約型の構築	31

4.1. Oracle Solaris と Oracle VM Server for SPARC のインストール	31
4.2. 論理ドメインの構築	31
4.3. I/O ルートドメインの構築	42
4.3.1. Oracle Solaris のインストール	42
4.3.2. I/O ルートドメインの冗長構成設定	43
5. 高集約型の構築	46
5.1. Oracle Solaris と Oracle VM Server for SPARC のインストール	46
5.2. 論理ドメインの構築	46
5.3. ゲストドメインの構築	55
5.3.1. Oracle Solaris のインストール	55
6. 構成情報の保存	57
6.1. Oracle Solaris ゾーン設定の保存	57
6.2. Oracle VM Server for SPARC の設定と構成情報の保存	58
6.2.1. 復旧モードの有効化	58
6.2.2. ハイパーバイザダンプ採取後のシステム自動再起動の有効化	58
6.2.3. 論理ドメイン構成情報を XSCF へ保存	59
6.2.4. 論理ドメイン構成情報のバックアップ	59
改版履歴	61

1. はじめに

1.1. BB HA の概要

SPARC M12/M10 Building Block を活用した高可用性システム(以下、BB High Availability: BB HA)は以下のような特長を持つ高可用性システムです。可用性を維持しつつ、コストを抑えたいシステムに適しています。

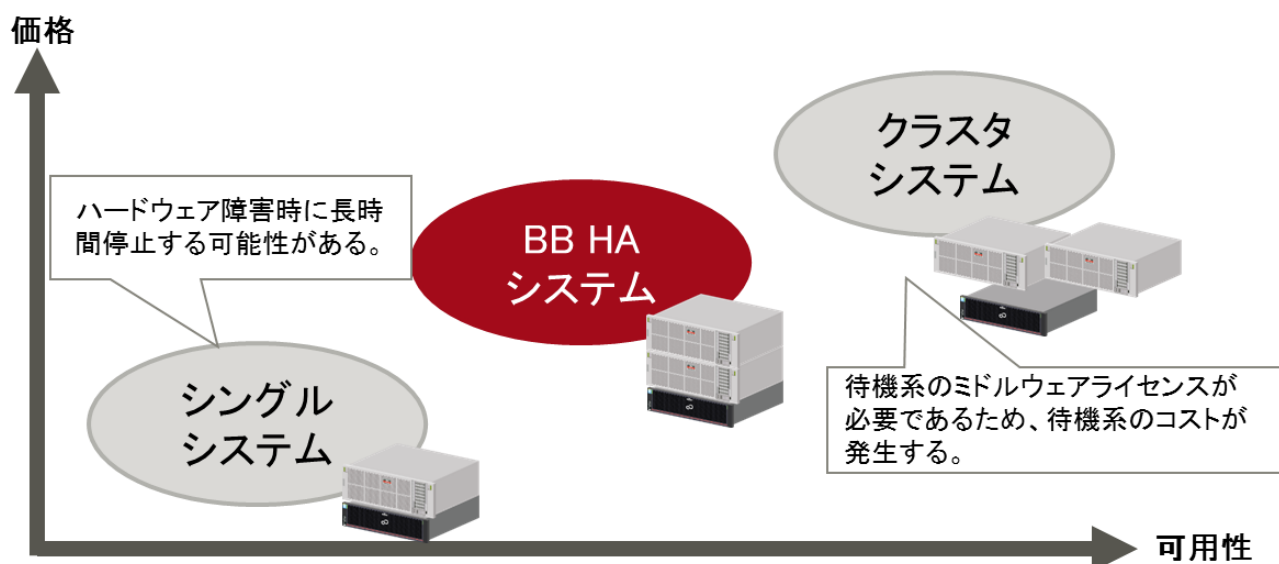
- ハードウェア故障時にリブート後、故障部品を切り離して業務復旧。故障部品の活性保守が可能

⇒ 可用性向上

- 待機系のミドルウェアライセンスが発生しない

⇒ コスト削減

図 1. BB HA の特長



1.2. BB HA の要件

BB HA を実現するためには、以下の条件を満たす構成にする必要があります。

- 各ドメインの使用する CPU コア/メモリの合計サイズに 1BB 分以上の余剰を持たせることで、BB 縮退時に残りの CPU コア/メモリで動作を継続できるように構築します。
- 各ドメインは、各 BB のディスクボリュームやネットワークインターフェースを接続し冗長構成にすることで、BB を削除した際に残りの BB で動作を継続できるように構築します。

1.3. 本書で扱う構成

ここでは、3 つのタイプによる BB HA の構成を以下それぞれの項にて解説します。それぞれの構成の特徴は、『SPARC M12/M10 Building Block を活用した高可用性システム(概要編)』をご参照ください。いずれも 2 つのシステムボード(以下 2BB)で構成される物理パーティションでの構築手順となります。なお、本書で記載している CPU コア、メモリ、物理 I/O などは構成の一例です。

1.3.1. 制御ドメインのみの構成(従来型)

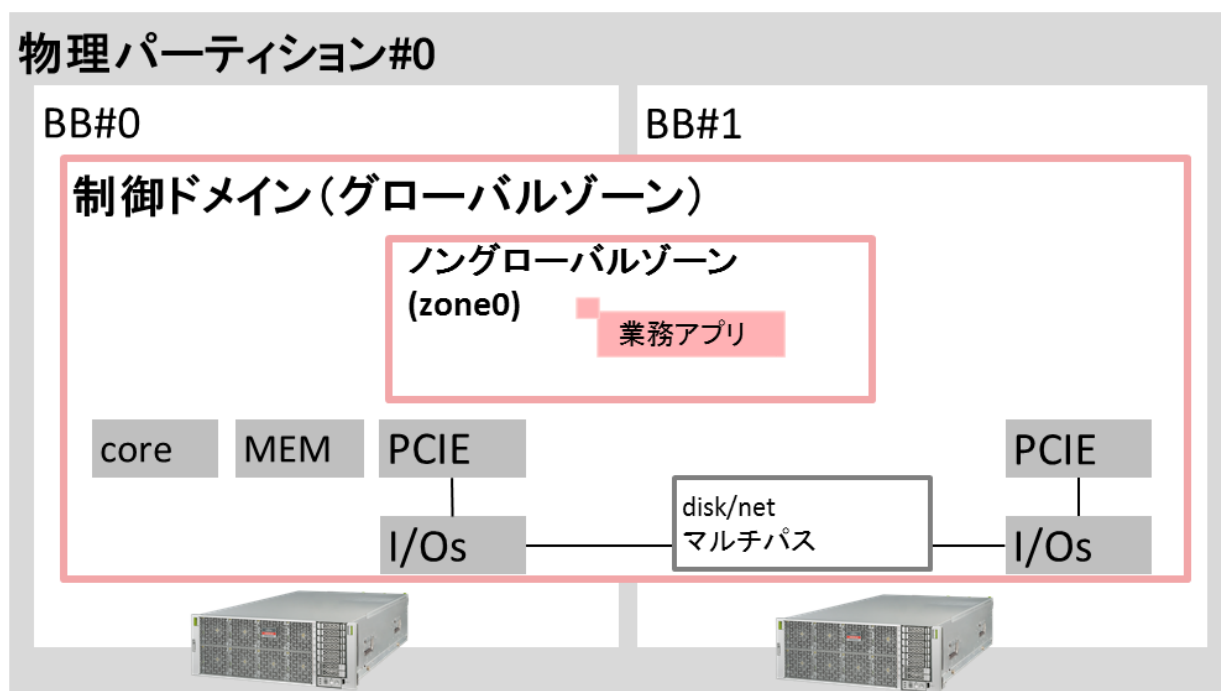
1.3.2. 制御ドメインと複数の I/O ルートドメインによる構成(集約型)

1.3.3. 制御ドメインと複数のゲストドメインによる構成(高集約型)

1.3.1. 制御ドメインのみの構成(従来型)

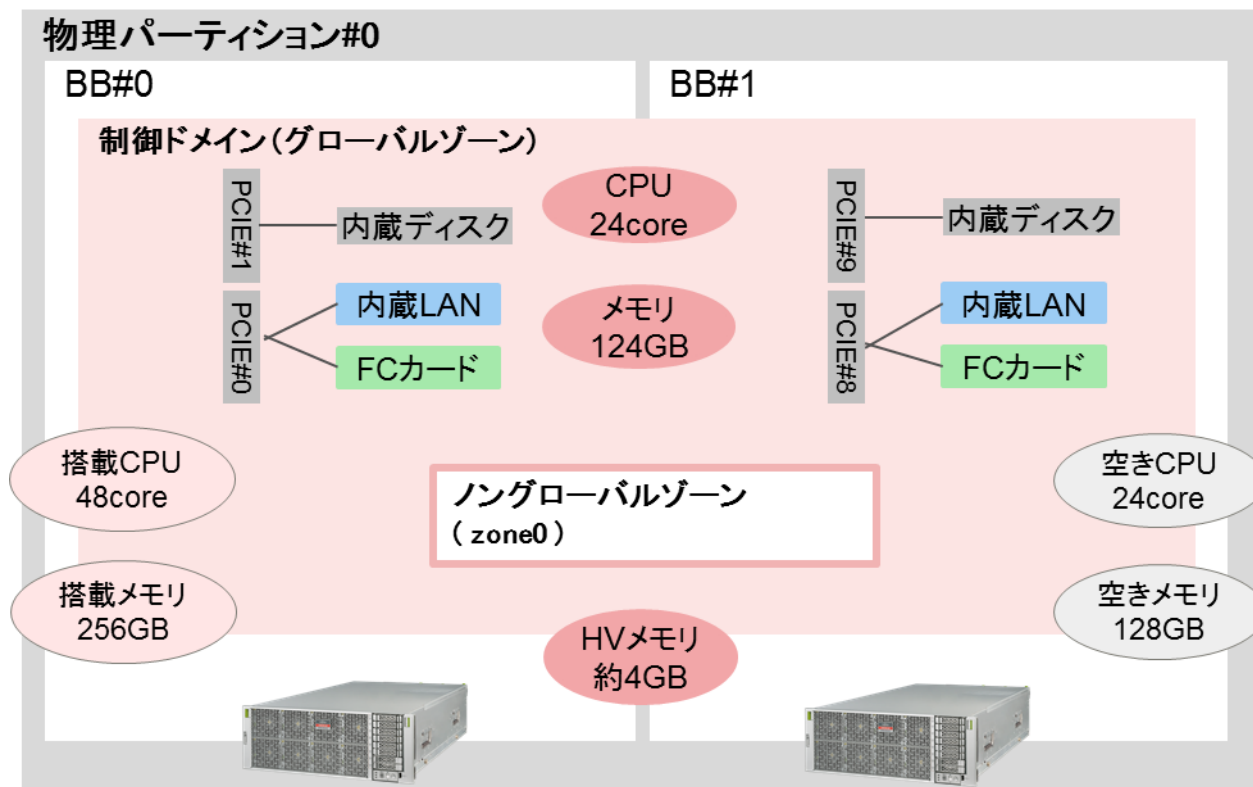
本構成手順では図 2 のように制御ドメイン上に Oracle Solaris ゾーンを構築します。また、業務アプリケーションはノングローバルゾーン上に構築します。

図 2. BB HA の従来型の要件を満たすシステム構成の概要例



制御ドメイン(グローバルゾーン)に割り当てる CPU コア、メモリ、物理 I/O を図 3 に示します。

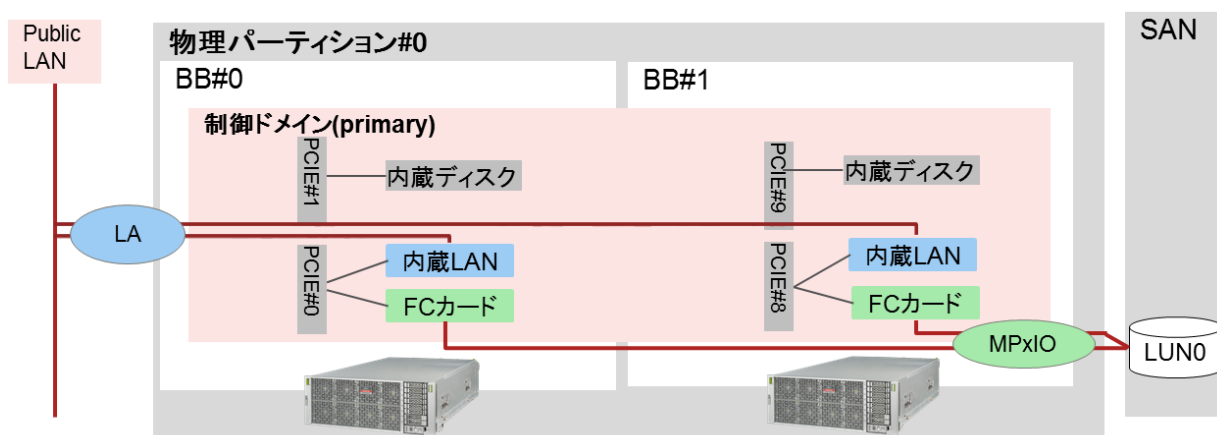
図 3. 制御ドメイン(グローバルゾーン)の CPU/メモリ/物理 I/O 割り当て



BB HA の要件に示したように、制御ドメイン(グローバルゾーン)の CPU コア/メモリの合計サイズに 1BB 分以上の余剰を持たせる必要があります。

制御ドメイン(グローバルゾーン)で使用する I/O 構成および冗長構成を図 4 に示します。

図 4. 制御ドメイン(グローバルゾーン)の I/O 構成



BB HA の要件に示したように、各 BB のディスクボリュームやネットワークインターフェースを接続し冗長構成にする必要があります。

本構成例では各 BB のネットワークインターフェースでリンクアグリゲーション(LA)を構成し、ネットワーク接続を冗長化しています。また、ディスクボリュームはストレージエリアネットワーク(SAN)上の LUN を使用し、各 BB のファイバーチャネルカードでマルチパスを構成し、ディスクボリュームの冗長化を実現しています。

表 1 では、制御ドメイン(グローバルゾーン)に割り当てる CPU コア、メモリ、物理 I/O をまとめています。

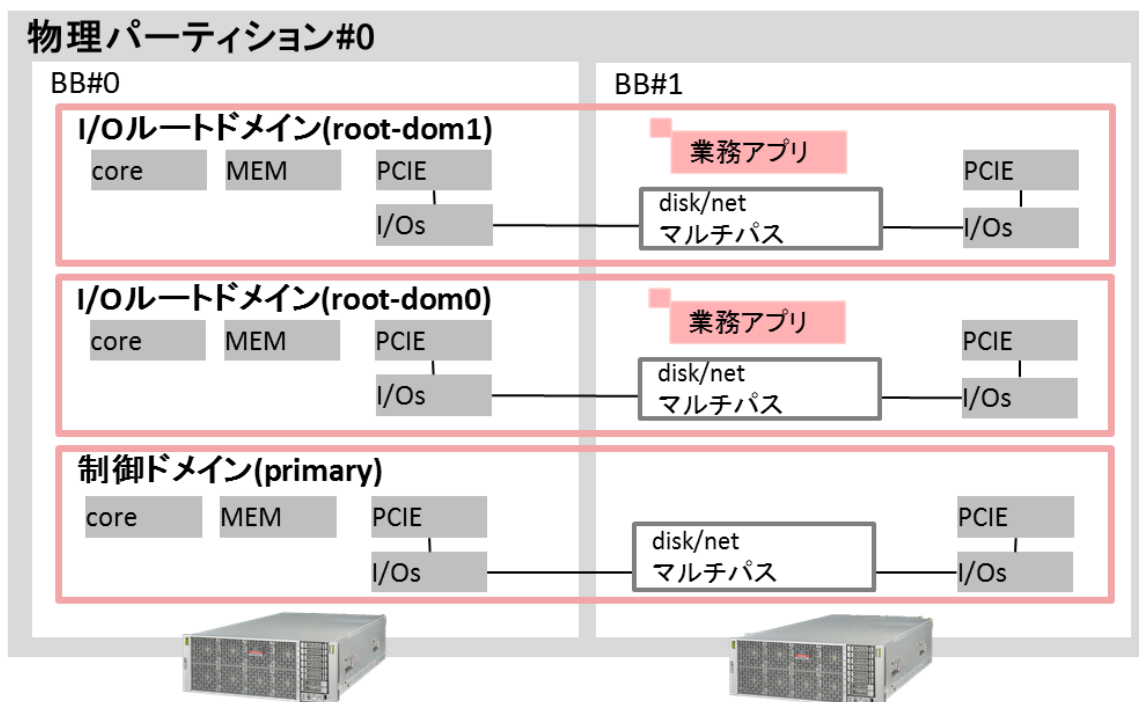
表 1. 従来型構成における制御ドメイン(グローバルゾーン)のリソース割り当て

論理ドメイン名	CPU コア	メモリ	物理 I/O(BB#0 側)	物理 I/O(BB#1 側)
制御ドメイン(グローバルゾーン)	24	124GB	PCIE1(内蔵ディスク, 空き SLOT x1) PCIE0(FC カード, 内蔵 LAN)	PCIE9(内蔵ディスク, 空き SLOT x1) PCIE8(FC カード, 内蔵 LAN)
(空きリソース)	24	128GB	PCIE2(空き SLOT x1) PCIE3(空き SLOT x1) PCIE4(内蔵 LAN, 空き SLOT x1) PCIE5(内蔵ディスク, 空き SLOT x1) PCIE6(空き SLOT x1) PCIE7(空き SLOT x1)	PCIE10(空き SLOT x1) PCIE11(空き SLOT x1) PCIE12(内蔵 LAN, 空き SLOT x1) PCIE13(内蔵ディスク, 空き SLOT x1) PCIE14(空き SLOT x1) PCIE15(空き SLOT x1)

1.3.2. 制御ドメインと複数の I/O ルートドメインによる構成(集約型)

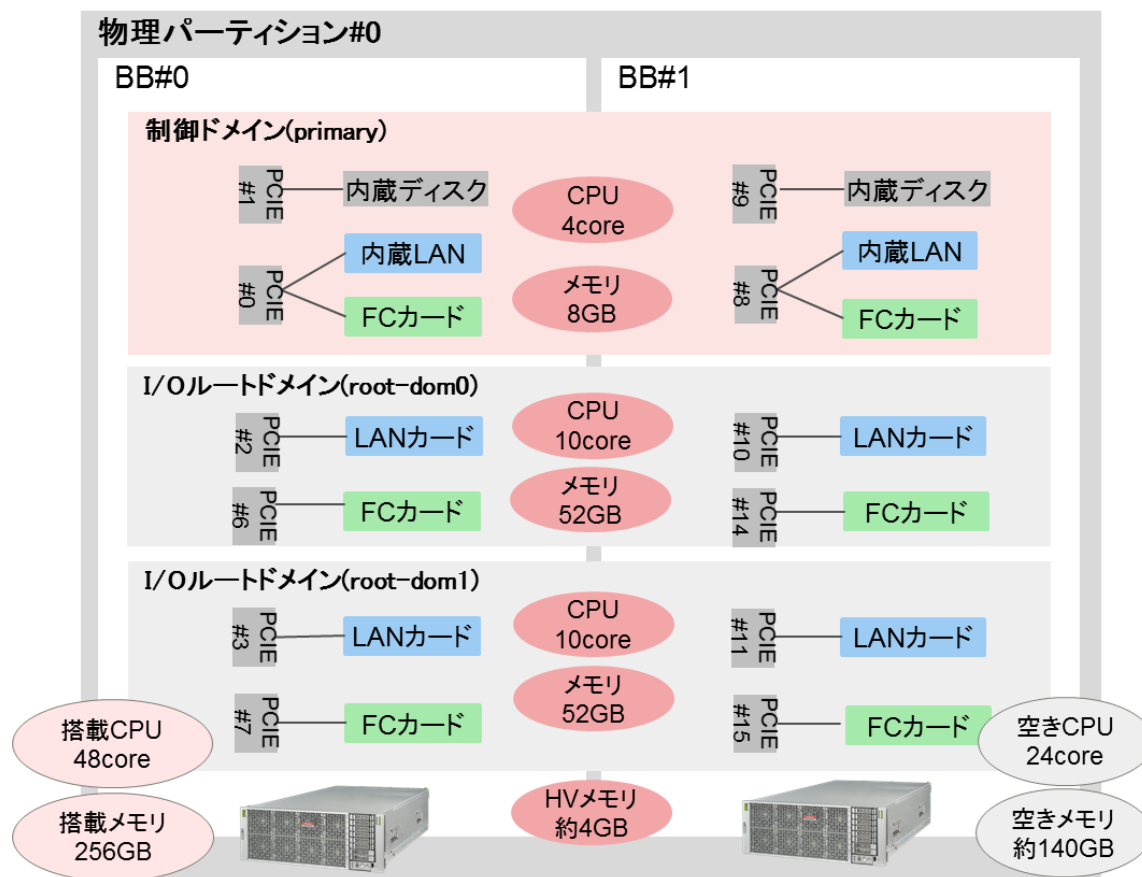
本構成手順では図 5 のように制御ドメインと 2 つの I/O ルートドメインを構築します。また、業務アプリケーションは 2 つの I/O ルートドメイン上に構築します。

図 5. BB HA の集約型の要件を満たすシステム構成の概要例



各ドメインに割り当てる CPU コア、メモリ、物理 I/O を図 6 に示します。

図 6. 各ドメインの CPU/メモリ/物理 I/O 割り当て

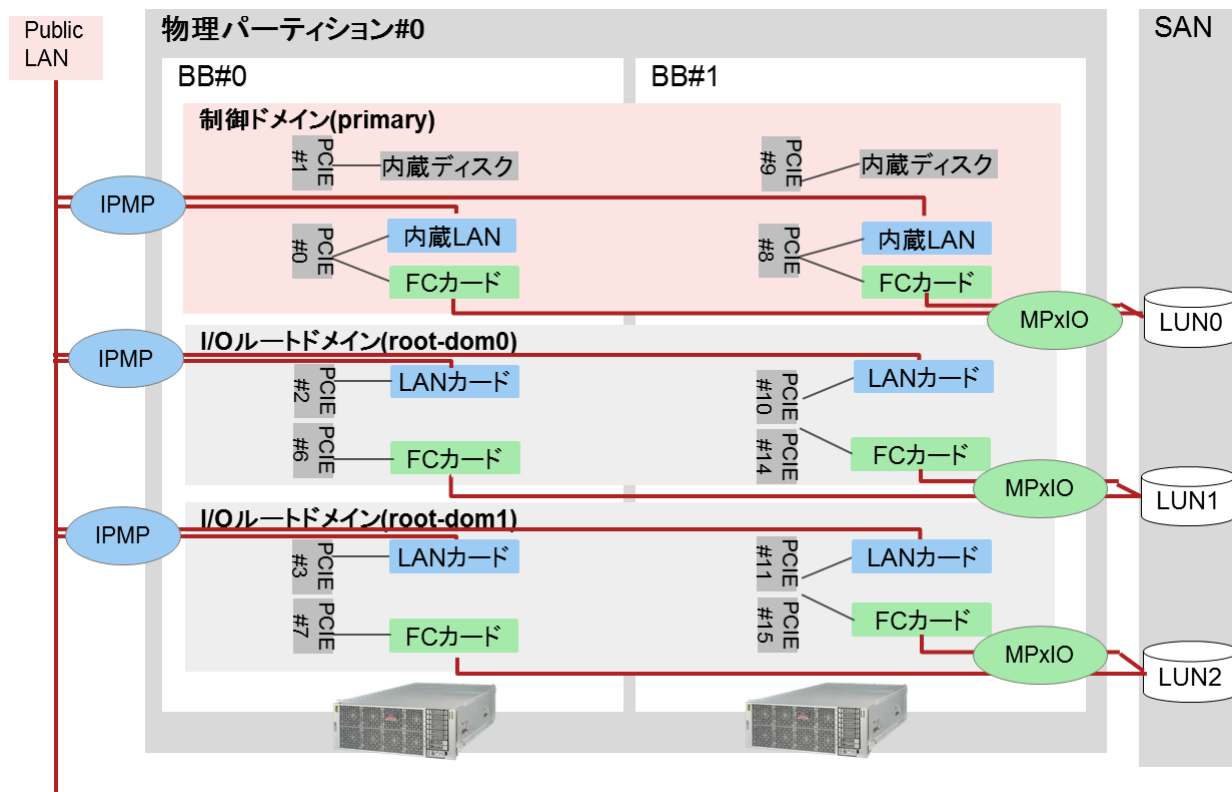


BB HA の要件に示したように、各ドメインの CPU コア/メモリの合計サイズに 1BB 分以上の余剰を持たせる必要があります。

本構成例では搭載メモリ量 256GB のうち、2.5GB + 1.5GB のメモリがハイパーバイザに割り当てられるため、ドメインに割り当て可能なメモリ量は最大で 124GB となります。本構成例では合計 112GB を各ドメインに割り当て、1BB 分以上(140GB)の余剰を持たせています。また、CPU についても搭載 CPU の半分 (24core)を使用し、1BB 分の余剰を持たせています。

各ドメインで使用する I/O 構成および冗長構成を図 7 に示します。

図 7. 各ドメインの I/O 構成



BB HA の要件に示したように、各ドメインは、各 BB のディスクボリュームやネットワークインターフェースを接続し冗長構成にする必要があります。

本構成例では各 BB のネットワークインターフェースで IP ネットワークマルチパス(IPMP)を構成し、各ドメインのネットワーク接続を冗長化しています。また、ディスクボリュームはストレージエリアネットワーク(SAN)上の LUN を使用し、各 BB のファイバーチャネルカードでマルチパスを構成し、ディスクボリュームの冗長化を実現しています。

表2では、各ドメインに割り当てる CPU コア、メモリ、物理 I/O をまとめています。

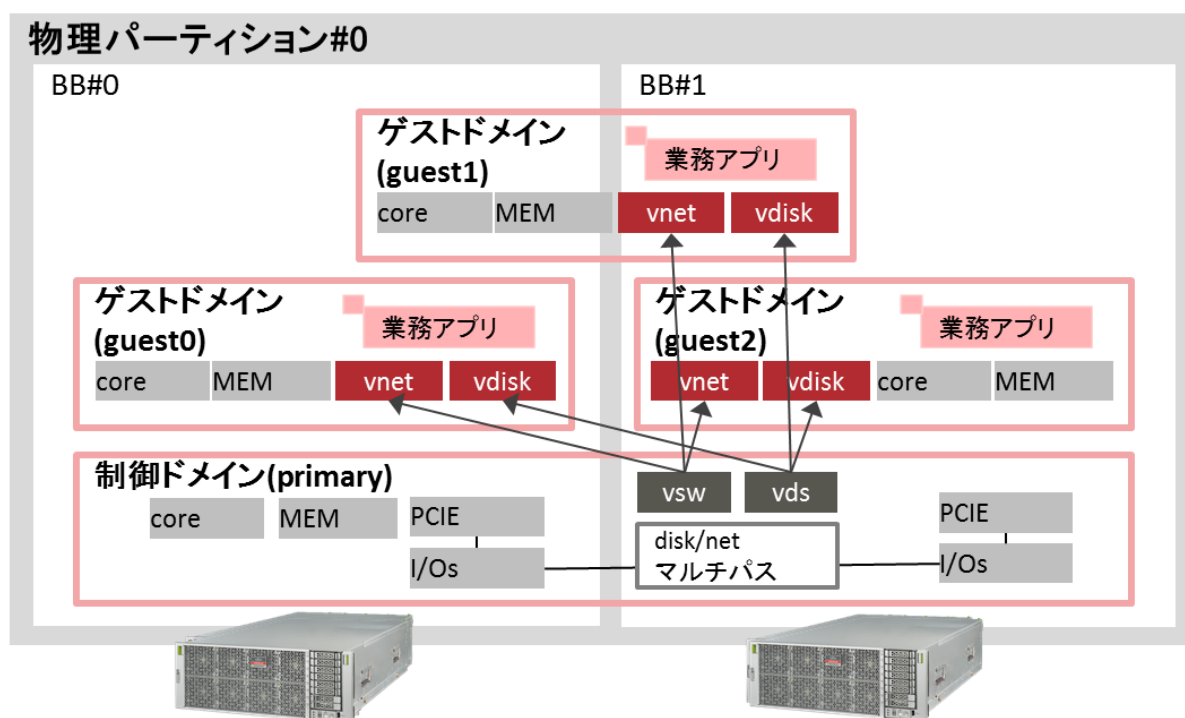
表 2. 集約型構成における各ドメインのリソース割り当て

論理ドメイン名	CPU コア	メモリ	物理 I/O(BB#0 側)	物理 I/O(BB#1 側)
primary	4	8GB	PCIE1(内蔵ディスク, 空き SLOT x1) PCIE0(FC カード, 内蔵 LAN)	PCIE9(内蔵ディスク, 空き SLOT x1) PCIE8(FC カード, 内蔵 LAN)
root-dom0	10	52GB	PCIE2(LAN カード) PCIE6(FC カード)	PCIE10(LAN カード) PCIE14(FC カード)
root-dom1	10	52GB	PCIE3(LAN カード) PCIE7(FC カード)	PCIE11(LAN カード) PCIE15(FC カード)
(空きリソース)	24	140GB	PCIE4(内蔵 LAN, 空き SLOT x1) PCIE5(内蔵ディスク, 空き SLOT x1)	PCIE12(内蔵 LAN, 空き SLOT x1) PCIE13(内蔵ディスク, 空き SLOT x1)

1.3.3. 制御ドメインと複数のゲストドメインによる構成(高集約型)

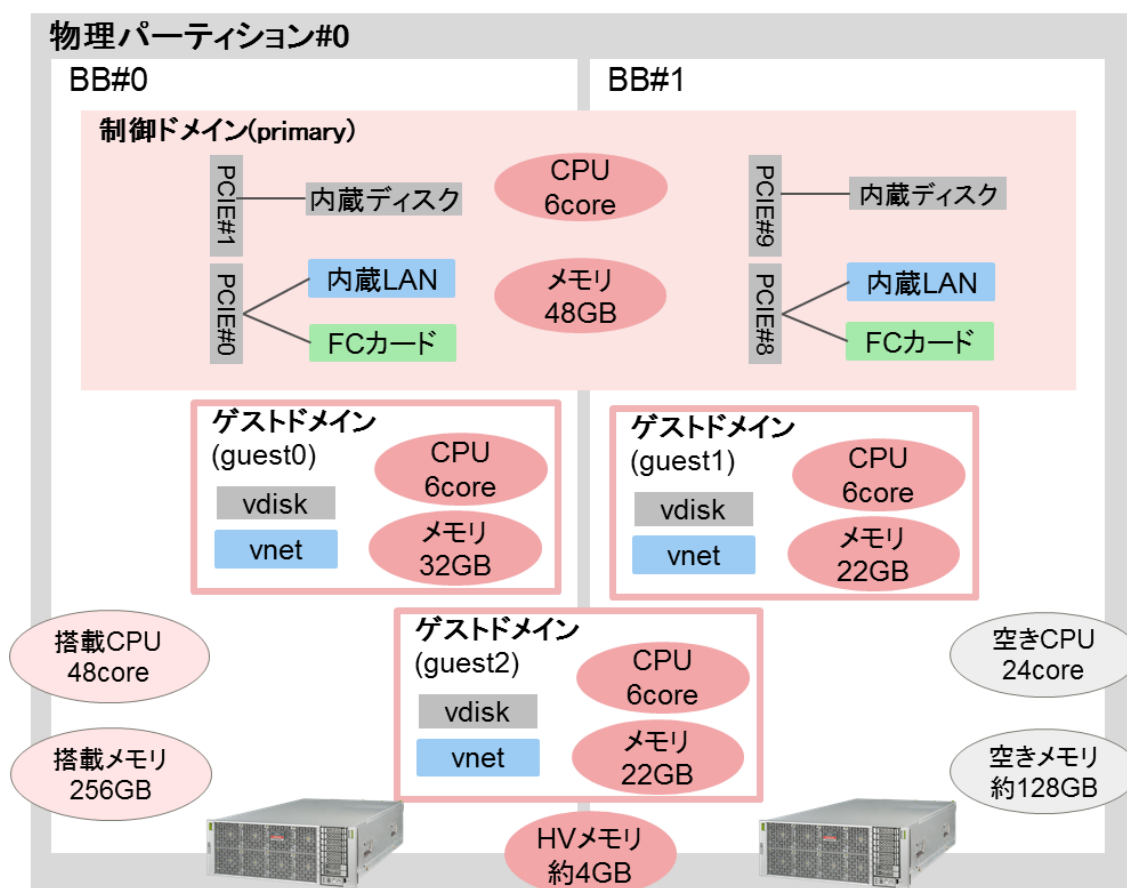
本構成手順では図 8 のように制御ドメインと 3 つのゲストドメインを構築します。また、業務アプリケーションは 3 つのゲストドメイン上に構築します。

図 8. BB HA の高集約型の要件を満たすシステム構成の概要例



各ドメインに割り当てる CPU コア、メモリ、仮想 I/O を図 9 に示します。

図 9. 各ドメインの CPU/メモリ/仮想 I/O 割り当て

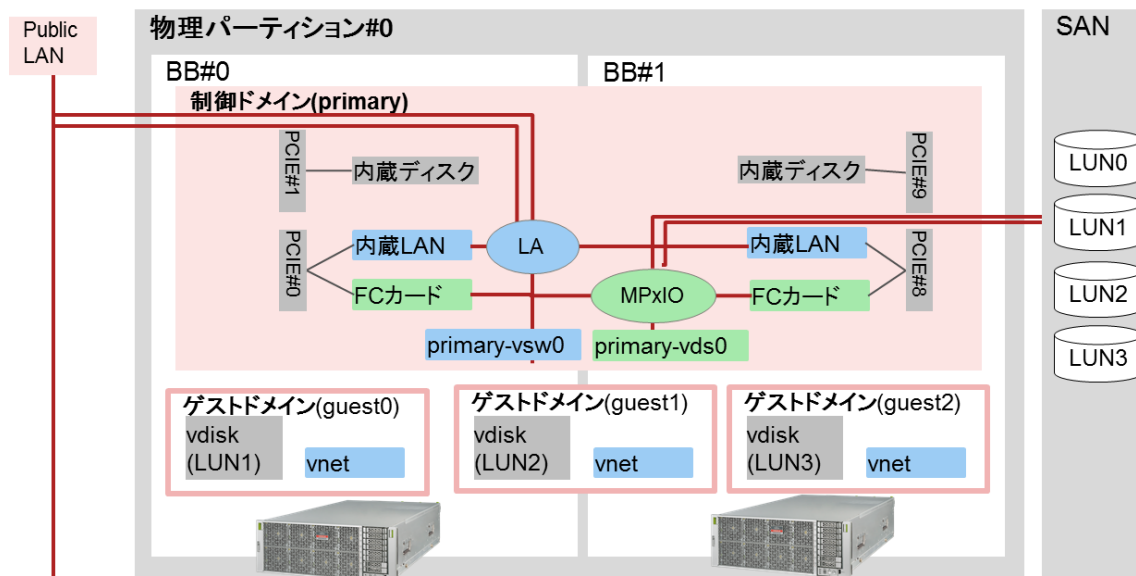


BB HA の要件に示したように、各ドメインの CPU コア/メモリの合計サイズに 1BB 分以上の余剰を持たせる必要があります。

本構成例では搭載メモリ量 256GB のうち、2.5GB + 1.5GB のメモリがハイパーバイザに割り当てられるため、ドメインに割り当て可能なメモリ量は最大で 124GB となります。本構成例では合計 124GB を各ドメインに割り当て、1BB 分の余剰(128GB)を持たせています。また、CPU についても搭載 CPU の半分(24core)を使用し、1BB 分の余剰(24core)を持たせています。

各ドメインで使用する I/O 構成および冗長構成を図 10 に示します。

図 10. 各ドメインの I/O 構成



BB HA の要件に示したように、各ドメインは、各 BB のディスクボリュームやネットワークインターフェースを接続し冗長構成にする必要があります。

本構成例では各 BB のネットワークインターフェースでリンクアグリゲーション(LA)を構成し、制御ドメインのネットワーク接続を冗長化しています。また、ディスクボリュームはストレージエリアネットワーク(SAN)上の LUN を使用し、各 BB のファイバーチャネルカードでマルチパスを構成し、ディスクボリュームの冗長化を実現しています。

表3では、各ドメインに割り当てる CPU コア、メモリをまとめています。

表 3. 高集約型構成における各ドメインのリソース割り当て

論理ドメイン名	CPU コア	メモリ	物理 I/O(BB#0 側)	物理 I/O(BB#1 側)
primary	6	48GB	PCIE1(内蔵ディスク, 空き SLOT x1) PCIE0(FC カード、内蔵 LAN)	PCIE9(内蔵ディスク, 空き SLOT x1) PCIE8(FC カード、内蔵 LAN)
guest0	6	32GB	-	-
guest1	6	22GB	-	-
guest2	6	22GB	-	-
(空きリソース)	24	128GB	PCIE2(空き SLOT x1) PCIE3(空き SLOT x1) PCIE4(内蔵 LAN, 空き SLOT x1) PCIE5(内蔵ディスク, 空き SLOT x1) PCIE6(空き SLOT x1) PCIE7(空き SLOT x1)	PCIE10(空き SLOT x1) PCIE11(空き SLOT x1) PCIE12(内蔵 LAN, 空き SLOT x1) PCIE13(内蔵ディスク, 空き SLOT x1) PCIE14(空き SLOT x1) PCIE15(空き SLOT x1)

1.4. 構築の流れ

ここでは、BB HA の 3 つのタイプの構築手順の流れを、それぞれ以下の項にて解説します。

- 1.4.1. 従来型の構築の流れ
- 1.4.2. 集約型の構築の流れ
- 1.4.3. 高集約型の構築の流れ

1.4.1. 従来型の構築の流れ

従来型は、図 11 に示す流れで物理パーティションの構築を行います。

図 11. 従来型の物理パーティション構築の流れ

■ 物理パーティションの設定



物理パーティションの構成情報作成
システムボードの割り当て
CPUコア アクティベーションキーの登録

■ 制御ドメイン(グローバルゾーン)の構築



Oracle Solaris/Oracle VM Server for SPARC(OVM)のインストール
OVMによる制御ドメインへのリソース割り当て
I/Oリソースの冗長構成設定

■ Oracle Solarisゾーンの構築



Oracle Solarisのインストール

■ OVM/Oracle Solarisゾーン設定変更・構成情報保存

Oracle Solarisゾーン設定の保存
復旧モードの有効化
論理ドメイン構成情報の保存

1.4.2. 集約型の構築の流れ

集約型は、図 12 に示す流れで物理パーティションの構築を行います。

図 12. 集約型の物理パーティション構築の流れ

■ 物理パーティションの設定



物理パーティションの構成情報作成
システムボードの割り当て
CPUコア アクティベーションキーの登録

■ 制御ドメインの構築



Oracle Solaris/Oracle VM Server for SPARC(OVM)のインストール
各論理ドメインへのリソース分配・仮想サービス設定
I/Oリソースの冗長構成設定

■ ルートドメインの構築



Oracle Solarisのインストール
I/Oリソースの冗長構成設計

■ OVMの設定変更・構成情報保存

復旧モードの有効化
論理ドメイン構成情報の保存

1.4.3. 高集約型の構築の流れ

高集約型は、図 13 に示す流れで物理パーティションの構築を行います。

図 13. 高集約型の物理パーティション構築の流れ

■ 物理パーティションの設定



物理パーティションの構成情報作成
システムボードの割り当て
CPUコア アクティベーションキーの登録

■ 制御ドメインの構築



Oracle Solaris/Oracle VM Server for SPARC(OVM)のインストール
各論理ドメインへのリソース分配・仮想サービス設定
I/Oリソースの冗長構成設定

■ ゲストドメインの構築



Oracle Solarisのインストール

■ OVMの設定変更・構成情報保存

復旧モード有効化
論理ドメイン構成情報の保存

2. 物理パーティションの設定

2.1. マスタ XSCF にログイン

showbbstatus コマンドを実行し、ログインした XSCF がマスタ XSCF であることを確認します。スタンバイ XSCF であった場合には、マスタ XSCF へログインし直してください。

```
XSCF> showbbstatus
BB#00 (Master)
```

2.2. 物理パーティション構成情報の作成

- a. showpcl コマンドを実行し、物理パーティション構成情報が存在しないことを確認します。

```
XSCF> showpcl -p 0
```

PPAR-ID	LSB	PSB	Status
---------	-----	-----	--------

- b. setpcl コマンドを実行し、物理パーティション構成情報に BB を登録します。

setpcl コマンドを実行し、BB を組み込み先の物理パーティション構成情報に登録します。

次の例では、物理システムボード(PSB)の 00-0 と 01-0 を、物理パーティション 0 の論理システムボード(LSB)の 00 と 01 に対応付けています。

物理システムボード(PSB)および論理システムボード(LSB)の詳細は、『SPARC M12/M10 ドメイン構築ガイド』の「1.3.1 物理パーティションの構成要素を理解する」を参照してください。

```
XSCF> setpcl -p 0 -a 00=00-0 01=01-0
```

- c. showpcl コマンドを実行し、物理パーティション構成情報を確認します。

```
XSCF> showpcl -p 0
```

PPAR-ID	LSB	PSB	Status
00			Powered Off
	00	00-0	
	01	01-0	

2.3. 物理パーティションへシステムボードを割り当て

- a. showboards -a コマンドを実行し、システムボードの状態を確認します。
showboards -a コマンドを実行し、それぞれのシステムボードの状態が、「SP」(システムボードプール)になっていることを確認します。

```
XSCF> showboards -a
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
----	-----	-----	----	----	----	-----	-----
00-0	SP	Available	n	n	n	Passed	Normal
01-0	SP	Available	n	n	n	Passed	Normal

- b. addboard -c assign コマンドを実行し、物理パーティション 0 に物理システムボード"00-0"と"01-0"を割り当てます。

```
XSCF> addboard -c assign -p 0 00-0 01-0
```

- c. showboards -p コマンドを実行し、システムボードの状態を確認します。
showboards -p コマンドを実行し、物理パーティションに割り当てた、それぞれのシステムボードの状態を確認します。
この例では、それぞれのシステムボードが物理パーティション 0 に正常に割り当てられたことで、[Assignment]欄が「Assigned」になることを確認します。

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
----	-----	-----	----	----	----	-----	-----
00-0	00(00)	Assigned	n	n	n	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

2.4. CPU コア アクティベーションキーの登録・割り当て

- a. showcodactivation コマンドを実行し、CPU コア アクティベーションキーの情報を確認します。
showcodactivation コマンドを実行し、物理パーティションに割り当て可能な CPU コア アクティベーションキーがあるかを確認します。以下のようにヘッダのみが表示された場合は、XSCF に CPU コア アクティベーションキーが登録されていないため、addcodactivation コマンドで CPU コア アクティベーションキーを登録する必要があります。

```
XSCF> showcodactivation
```

Index	Description	Count
-----	-----	-----

- b. addcodactivation コマンドを実行し、CPU コア アクティベーションキーを追加します。すでに CPU コア アクティベーションキーが登録されている場合は、本手順は不要です。

```
XSCF> addcodactivation "Product: SPARC M12-2S
SequenceNumber:10005
Cpu: noExpiration 1
Text:Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtgvyZPsrtYguqiNUieB+mTDC
:
:
b1GCKFx1RH27FdVHiB2H0A=="
AboveKeywillbeadded,Continue?[y|n]:y
```

- c. showcodactivation コマンドを実行し、CPU コア アクティベーションキーの情報を確認します。
showcodactivation コマンドを実行し、物理パーティションに割り当て可能な CPU コア アクティベーションキーがあるかを確認します。

```
XSCF> showcodactivation
```

Index	Description	Count
0	PROC	1
:		
23	PROC	1

- d. setcod コマンドを実行し、CPU コアリソースを割り当てます。
setcod コマンドを実行し、CPU リソースを物理パーティションに割り当てます。
この例では、24 個の CPU コアを物理パーティション 0 に割り当てています。

```
XSCF> setcod -p 0 -s cpu 24
```

showcod コマンドを実行し、物理パーティションに割り当てられている CPU リソース情報を確認します。

この例では、直前に実行した setcod コマンドで割り当てた 24 個の CPU コアが、物理パーティション 0 に割り当てられていることを確認します。

```
XSCF> showcod -p 0
```

```
PROC Permits assigned for PPAR 0: 24
```

2.5. XSCF の時刻補正

resetdateoffset コマンドを実行し、XSCF が管理する時刻と物理パーティションが管理する時刻の差分をリセットします。

```
XSCF> resetdateoffset -p 0
```

2.6. 物理パーティションの設定確認

showpparmode コマンドを実行し、診断メッセージの詳細レベル、PPAR DR モードの設定を確認します。
showpparmode コマンドを実行して、診断メッセージの詳細レベル(Message Level)が「normal」(標準)、PPAR DR モードの Next が「on」(有効)であることを確認します。

```
XSCF> showpparmode -p 0
```

```
Host-ID                :9007002b
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal           :on
Autoboot(Guest Domain) :on
Power Aware Dispatcher  :off
Power Management Policy :disabled
IOreconfigure           :false
CPU Mode                :-
PPAR DR(Current)        :on
PPAR DR(Next)           :on
```

参考) 診断メッセージの詳細レベルが「normal」でない場合は、setpparmode コマンドを実行し、「normal」にします。

```
XSCF> setpparmode -p 0 -m message=normal
```

PPAR DR モードが「off」(無効)の場合は、setpparmode コマンドを実行し、「on」にします。

```
XSCF> setpparmode -p 0 -m ppar_dr=on
```

2.7. 物理パーティションの電源投入

poweron コマンドを実行し、物理パーティションを電源投入します。

```
XSCF> setpparparam -y -p 0 -s bootscript "setenv auto-boot? false"  
XSCF> poweron -p 0
```

2.8. 物理パーティションのコンソールに接続

console コマンドを実行し、物理パーティションにコンソール接続します。

```
XSCF> console -p 0
```


3. 従来型の構築

本章では、従来型の構築手順を解説します。

3.1. Oracle Solaris と Oracle VM Server for SPARC のインストール

制御ドメイン(グローバルゾーン)に Oracle Solaris をインストールします。

BB HA に必要な Oracle Solaris の版数や条件などについては、『SPARC M12/M10 Building Block を活用した高可用性システム(概要編)』を参照してください。

インストールの詳細は、Oracle 社のホームページ(<https://docs.oracle.com/>)で公開されている、以下のドキュメントを参照してください。

- Oracle Solaris 11
『Oracle Solaris 11.3 システムのインストール』
お使いのバージョンの『Oracle VM Server for SPARC インストールガイド』

3.2. 制御ドメイン(グローバルゾーン)の構築

ここでは、「表 1. 従来型構成における制御ドメイン(グローバルゾーン)のリソース割り当て」で定義した論理ドメインの構築方法について説明します。

- a. 制御ドメイン(グローバルゾーン)のリソースを減らします。

factory-default の構成の場合、すべての CPU コア、メモリが制御ドメイン(primary)に割り当てられています。これらのリソースを他の論理ドメインに割り当てられるようにするために、最初に、制御ドメインからこれらのリソースの一部を解放します。以下にコマンドの実行例を示します。

ldm start-reconf コマンドで遅延再構成に移行します。

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

ldm set-core コマンド、ldm set-memory コマンドを実行して、元のサイズより小さいサイズを指定することで、制御ドメインに割り当てている CPU コア数とメモリサイズを減らします。

次の例では、構成例にしたがって制御ドメインの CPU コアを 24 個、メモリサイズを 124GB に設定します。

```
primary# ldm set-core 24 primary
```

```
-----
```

Notice: The primary domain is in the process of a delayed reconfiguration.

Any changes made to the primary domain will only take effect after it reboots.

```
-----
```

```
primary# ldm set-memory 124G primary
```

```
-----
```

Notice: The primary domain is in the process of a delayed reconfiguration.

Any changes made to the primary domain will only take effect after it reboots.

```
-----
```

注一CPU コアとメモリの設定は、ldm set-core、ldm set-memory の順で設定を行うことを推奨します。これにより、まとまったメモリの連続領域が割り当てられやすくなります。CPU コアとメモリの配置の条件の詳細は『SPARC M12/M10 ドメイン構築ガイド』の「2.5.2 運用時の留意点」の「CPU コアとメモリの配置」を参照してください。

- b. 制御ドメイン（グローバルゾーン）の最大ページサイズを 256MB に設定します。

SPARC M12 かつ Oracle VM Server for SPARC 3.5 以降の場合、ドメインの最大ページサイズを 256MB に設定します。ドメインの最大ページサイズを 256MB に設定しない場合、物理パーティションの動的再構成を実施した際にメモリが必要以上に削除される場合があります。

SPARC M10 の場合は、設定不要です。Oracle VM Server for SPARC 3.4 の場合は設定できないため、省略します。

fj_dr_sw_limit_pagesize の値を確認します。false の場合はドメインの最大ページサイズを 256MB に設定します。true の場合は fj_dr_sw_limit_pagesize の値を false に設定した後にドメインの最大ページサイズを 256MB に設定します。

```
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
```

```
ldmd/fj_dr_sw_limit_pagesize boolean true
```

値が true の場合は以下を実行し、false に変更します。

```
primary# svcadm disable ldmd
```

```
primary# svccfg -s ldmd setprop ldmd/fj_dr_sw_limit_pagesize=false
```

```
primary# svcadm refresh ldmd
```

```
primary# svcadm enable ldmd
```

```
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
```

```
ldmd/fj_dr_sw_limit_pagesize boolean false
```

ドメインの最大ページサイズを 256MB に設定します。

```
primary# ldm set-domain fj-software-limit-pagesize=256MB primary
```

```
-----
```

Notice: The primary domain is in the process of a delayed reconfiguration.

Any changes made to the primary domain will only take effect after it reboots.

```
-----
```

- c. Oracle Solaris を再起動します。

制御ドメイン(グローバルゾーン)の再起動によって構成情報を反映します。

```
primary# shutdown -i6 -g0 -y
```

- d. 構成情報を保存します。

ldm add-spconfig コマンドで現在の構成情報を XSCF に保存します。

次の例では、ldm list-spconfig コマンドで保存されている構成情報を確認し、ldm add-spconfig コマンドで、ldm-set1 という名前で構成を保存し、再度 ldm list-spconfig コマンドで保存されたことを確認する例を示しています。

```
primary# ldm list-spconfig
factory-default [next poweron]

primary# ldm add-spconfig ldm-set1

primary# ldm list-spconfig
factory-default
ldm-set1 [current]
```

- e. グローバルゾーンのシステムボリュームを冗長構成にします。

ここでは、グローバルゾーンに割り当てた 2 つの FibreChannel ポートで SAN 上に作成したシステムボリュームのパスを冗長構成にする手順の例を説明します。その他の冗長構成の手順については、各冗長構成ソフトウェアのドキュメントを参照してください。

グローバルゾーンの起動時間の短縮、および物理パーティションの動的再構成を実施した際の一時停止時間の短縮のため、/etc/system に以下を追加します。また、FibreChannel の空きポートに光ケーブルを接続し、リンクアップ状態にします。

```
forceload: drv/qlc
forceload: drv/emlxs
forceload: drv/ssd
forceload: drv/fp
```

stmsboot コマンドを実行して、現在のマルチパスの設定を確認します。次の例では、マルチパスの設定が無効となっていることが確認できます。

```
primary# stmsboot -D fp -L
stmsboot: MPXIO disabled
```

※または、以下のような出力結果となります。

```
primary# stmsboot -D fp -L
non-STMS device name          STMS device name
-----
```

stmsboot コマンドを実行してマルチパス設定を有効にします。この際、グローバルゾーンの再起動が必要となります。

```
primary# stmsboot -D fp -e
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y <-- y を押下して OS 再起動
Jul 29 23:43:57 primary reboot: initiated by root on /dev/console
syncing file systems... done
rebooting...
Resetting...
```

グローバルゾーンの再起動後、stmsboot コマンドを実行してマルチパス設定が有効となったかを確認します。次の例では、2つのディスクパスが1つのマルチパスディスクとして認識されています。

```
primary# stmsboot -D fp -L
non-STMS device name          STMS device name
-----
/dev/rdisk/c10t500000E0D0000087d0    /dev/rdisk/c0t600000E0D00000000000000000000000d0
/dev/rdisk/c9t500000E0D0000086d0    /dev/rdisk/c0t600000E0D00000000000000000000000d0
```

ldm コマンドを実行して、2つのディスクパスを boot-device として指定します。ディスクパスとマルチパスディスクの対応関係については、各冗長構成ソフトウェアのドキュメント(今回の例では、[『SAN Boot 環境構築ガイド\(Leadville ドライバ編\) for SPARC Servers/SPARC Enterprise』](#)) を参照してください。

```
primary# ldm set-variable boot-device=¥
"/pci@8100/pci@4/pci@0/pci@0/SUNW,qlc@0/fp@0,0/disk@w500000e0d0000086,0¥
/pci@8900/pci@4/pci@0/pci@0/SUNW,qlc@0,1/fp@0,0/disk@w500000e0d0000087,0 disk net" primary
```

- f. 制御ドメイン(グローバルゾーン)の構成情報を確認します。

ldm コマンドを実行して、制御ドメイン(グローバルゾーン)の構成情報を確認します。次の例では、SPARC M12 かつ Oracle VM Server for SPARC 3.5 の場合の制御ドメイン(グローバルゾーン)の CPU、メモリ、物理 I/O が「表 1. 従来型構成における制御ドメイン(グローバルゾーン)のリソース割り当て」の定義と一致していることと、設定内容が正しく反映されていること確認しています。

```
primary# ldm list-domain -l
NAME      STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary   active   -n-c-    UART    192     124G      0.0%    0.0%    27m
:
CONTROL
:
fj-software-limit-pagesize=256MB
VARIABLES
boot-device=/pci@8100/pci@4/pci@0/pci@0/SUNW,qlc@0/fp@0,0/disk@w500000e0d0000086,0
          /pci@8900/pci@4/pci@0/pci@0/SUNW,qlc@0,1/fp@0,0/disk@w500000e0d0000087,0
          disk net
pm_boot_policy=disabled=1;ttfc=0;ttmr=0;

IO
DEVICE                                PSEUDONYM    OPTIONS
pci@8000                              PCIE0
pci@8100                              PCIE1
pci@8200                              PCIE2
pci@8300                              PCIE3
pci@8400                              PCIE4
pci@8500                              PCIE5
pci@8600                              PCIE6
pci@8700                              PCIE7
pci@8800                              PCIE8
pci@8900                              PCIE9
pci@8a00                              PCIE10
pci@8b00                              PCIE11
pci@8c00                              PCIE12
pci@8d00                              PCIE13
```

pci@8e00	PCIE14
pci@8f00	PCIE15

- g. 制御ドメイン(グローバルゾーン)のネットワークを冗長構成にします。

ここでは、制御ドメイン primary に割り当てた 2 つのネットワークインターフェースをリンクアグリゲーション(LA)で冗長構成にする手順の例を説明します。その他の冗長構成の手順については、各冗長構成のソフトウェアのドキュメントを参照してください。

dladm コマンドを実行して、ネットワークデバイスが見えることを確認します。

次の例では、ネットワークインターフェース net0, net4 としてネットワークデバイスが参照できています。

また、net0 はシステムボード 00-0(BB#0) 配下、net4 はシステムボード 01-0(BB#1) 配下にあることがわかります。

```
primary# dladm show-link
LINK          CLASS      MTU    STATE    OVER
net0          phys       1500   up       --
net4          phys       1500   up       --

primary# dladm show-phys -L
LINK          DEVICE        LOC
net0          ixgbe0        /BB0/CMUL
:
net4          ixgbe4        /BB1/CMUL
```

ipadm show-if コマンドを実行して、net0, net4 が表示されていないことを確認します。

```
primary# ipadm show-if
IFNAME CLASS  STATE  ACTIVE OVER
lo0      loopback ok      yes    --
```

LA に構成しようとしているネットワークデバイスが IP インターフェースとしてここに表示される場合は、その IP インターフェースを削除してください。

```
primary# ipadm show-if
IFNAME CLASS  STATE  ACTIVE OVER
lo0      loopback ok      yes    --
net0     ip        ok      yes    --

primary# ipadm delete-ip net0
primary# ipadm show-if
```

IFNAME	CLASS	STATE	ACTIVE	OVER
lo0	loopback	ok	yes	--

dladm create-aggr コマンドを実行して、アグリゲーション aggr0 を作成し、dladm show-aggr コマンドで、作成できたことを確認します。

```
primary# dladm create-aggr -m dlm -l net0 -l net4 aggr0
primary# dladm show-aggr
```

LINK	MODE	POLICY	ADDRPOLICY	LACPACTIVITY	LACPTIMER
aggr0	dlmp	--	--	--	--

ipadm create-addr コマンドを実行して、LA インターフェース aggr0 に IP アドレスを割り当て、ipadm show-addr コマンド、dladm コマンドで設定値を確認します。次の例では、固定 IP アドレスを割り当てる例を示しています。

```
primary# ipadm create-ip aggr0
primary# ipadm create-addr -T static -a local=192.168.1.101/24 aggr0/v4
primary# ipadm show-addr
```

ADDR	OBJ	TYPE	STATE	ADDR
lo0/v4	static	ok		127.0.0.1/8
aggr0/v4	static	ok		192.168.1.101/24
lo0/v6	static	ok		::1/128

```
primary# dladm show-link
```

LINK	CLASS	MTU	STATE	OVER
net0	phys	1500	up	--
net4	phys	1500	up	--
:				
aggr0	aggr	1500	up	net0 net4

3.3. Oracle Solaris ゾーンの構築

ここでは、Oracle Solaris ゾーンの構築方法について説明します。

Oracle Solaris ゾーンを使用しない場合は、本手順の実施は不要です。6. 構成情報の保存へ進んでください。

構築方法の詳細は、Oracle 社のホームページ(<https://docs.oracle.com/>)で公開されている、以下のドキュメントを参照してください。

- Oracle Solaris 11
『Oracle Solaris 11.3 Information Library』

→「Oracle Solaris 仮想化環境の作成と使用」

3.3.1. ノングローバルゾーンの作成

ここでは、ノングローバルゾーンの作成手順について説明します。

制御ドメイン(グローバルゾーン)上で、zonecfg コマンドを実行し、ノングローバルゾーンを作成します。

以下、作成例(LA で冗長化した aggr0 をノングローバルゾーンに割り当てます)。

```
primary# zonecfg -z zone0
Use 'create' to begin configuring a new zone.
zonecfg:zone0> create
create: Using system default template 'SYSdefault'
zonecfg:zone0> set zonepath=/zones/zone0
zonecfg:zone0> set autoboot=true
zonecfg:zone0> set bootargs="-m verbose"    ※
zonecfg:zone0> select anet linkname=net0
zonecfg:zone0:anet> set lower-link=aggr0
zonecfg:zone0:anet> end
zonecfg:zone0> verify
zonecfg:zone0> commit
zonecfg:zone0> exit
primary# zonecfg -z zone0 info
zonename: zone0
zonepath: /zones/zone0
:
primary#
```

※ zonecfg:zone0> set bootargs="-m verbose"は必要に応じて実行してください。

3.3.2. Oracle Solaris のインストール

ここでは、ノングローバルゾーンに Oracle Solaris をインストールする方法を説明します。

以下の例では、zoneadm コマンドによりノングローバルゾーン zone0 に Oracle Solaris をインストールし、zone0 の状態および、zonepath が作成されたことを確認しています。


```
primary# zoneadm -z zone0 install
```

The following ZFS file system(s) have been created:

```
    rpool/zones
```

```
    rpool/zones/zone0
```

Progress being logged to /var/log/zones/zoneadm.20150805T144314Z.zone0.install

Image: Preparing at /zones/zone0/root.

Install Log: /system/volatile/install.2106/install_log

AI Manifest: /tmp/manifest.xml.0LaWfe

SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml

Zonename: zone0

Installation: Starting ...

```
:
```

```
:
```

Log saved in non-global zone as /zones/zone0/root/var/log/zones/zoneadm.20150805T144314Z.zone0.install

```
primary# zoneadm list -iv
```

ID	NAME	STATUS	PATH	BRAND	IP
0	global	running	/	solaris	shared
-	zone0	installed	/zones/zone0	solaris	excl

```
primary# zfs list | grep zones
```

rpool/zones	720M	4.02G	32K	/zones
rpool/zones/zone0	720M	4.02G	32K	/zones/zone0
rpool/zones/zone0/rpool	720M	4.02G	31K	/zones/zone0/root/rpool
rpool/zones/zone0/rpool/ROOT	719M	4.02G	31K	legacy
rpool/zones/zone0/rpool/ROOT/solaris	719M	4.02G	674M	/zones/zone0/root
rpool/zones/zone0/rpool/ROOT/solaris/var	45.1M	4.02G	44.1M	/zones/zone0/root/var
rpool/zones/zone0/rpool/export	63K	4.02G	32K	/zones/zone0/root/export
rpool/zones/zone0/rpool/export/home	31K	4.02G	31K	/zones/zone0/root/export/home

3.3.3. ノングローバルゾーンのシステム構成設定

ここでは、ノングローバルゾーンにおける、システム構成の設定および起動の方法を説明します。

次のコマンドを実行してノングローバルゾーンを起動し、コンソールにアクセスします。

```
primary# zoneadm -z zone0 boot; zlogin -C zone0
```

メニュー画面にしたがい、ノングローバルゾーンのシステム構成の設定が完了すると、ノングローバルゾーンが起動し、コンソールログインのプロンプトが表示されます。

```
[ system/console-login:default starting (Console login) ]
```

```
zone0 console login:
```

ノングローバルゾーンにログインし、IP アドレスを確認します。

```
root@zone0:~# ipadm show-addr
```

ADDROBJ	TYPE	STATE	ADDR
lo0/v4	static	ok	127.0.0.1/8
net0/v4	static	ok	192.168.1.102/24
lo0/v6	static	ok	::1/128
net0/v6	addrconf	ok	fe80::8:20ff:fe85:423d/10

```
root@zone0:~#
```

“~.”を入力し、コンソールを終了します。

```
root@zone0:~# exit
```

```
logout
```

```
zone0 console login: ~.
```

```
[Connection to zone 'zone0' console closed]
```

```
primary#
```

3.3.4. ノングローバルゾーンの状態確認

ここでは、ノングローバルゾーン起動後の状態確認について説明します。

ノングローバルゾーンが正しく起動している場合、zoneadm list コマンドの[STATUS]欄に「running」と表示されます。

以下の例では、ノングローバルゾーン zone0 が正しく起動していること、LA インターフェース aggr0 に IP アドレスが正しく割り当てられていることを確認しています。

```
primary# zoneadm list -v
```

ID	NAME	STATUS	PATH	BRAND	IP
0	global	running	/	solaris	shared
1	zone0	running	/zones/zone0	solaris	excl

```
primary# dladm show-link
```

LINK	CLASS	MTU	STATE	OVER
net4	phys	1500	up	--
net5	phys	1500	unknown	--
net6	phys	1500	unknown	--
net7	phys	1500	unknown	--
net0	phys	1500	up	--
net1	phys	1500	unknown	--
net2	phys	1500	unknown	--
net3	phys	1500	unknown	--
aggr0	aggr	1500	up	net0 net4
zone0/net0	vnic	1500	up	aggr0

```
primary#
```

4. 集約型の構築

本章では、集約型の構築手順を解説します。

4.1. Oracle Solaris と Oracle VM Server for SPARC のインストール

制御ドメインに Oracle Solaris と Oracle VM Server for SPARC (以降 OVM) をインストールします。

BB HA に必要な Oracle Solaris と OVM の版数や条件などについては、『SPARC M12/M10 Building Block を活用した高可用性システム(概要編)』を参照してください。

インストールの詳細は、Oracle 社のホームページ(<https://docs.oracle.com/>)で公開されている、以下のドキュメントを参照してください。

- Oracle Solaris 11
『Oracle Solaris 11.3 システムのインストール』
お使いのバージョンの『Oracle VM Server for SPARC インストールガイド』

4.2. 論理ドメインの構築

ここでは、「表 2. 集約型構成における各ドメインのリソース割り当て」で定義した論理ドメインの構築方法について説明します。

- a. 制御ドメインのリソースを減らします。

factory-default の構成の場合、すべての CPU コア、メモリ、PCIe ルートコンプレックスが制御ドメイン(primary)に割り当てられています。これらのリソースを他の論理ドメインに割り当てられるようにするために、最初に、制御ドメインからこれらのリソースの一部を削減します。以下にコマンドの実行例を示します。

ldm start-reconf コマンドで遅延再構成に移行します。

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

ldm remove-io コマンドでルートコンプレックスを削除します。

次の例では、構成例にしたがって、PCIE2, PCIE3, PCIE4, PCIE5, PCIE6, PCIE7, PCIE10, PCIE11, PCIE12, PCIE13, PCIE14, PCIE15 を削除するコマンドの例の一部を示します。

```
primary# ldm remove-io PCIE2 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.
```

```
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
....
```

```
primary# ldm remove-io PCIE15 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.
```

```
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

ldm set-core コマンド、ldm set-memory コマンドを実行して、元のサイズより小さいサイズを指定することで、制御ドメインに割り当てている CPU コア数とメモリサイズを減らします。

次の例では、構成例にしたがって制御ドメインの CPU コアを 4 個、メモリサイズを 8GB に設定します。

```
primary# ldm set-core 4 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.
```

```
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
primary# ldm set-memory 8G primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.
```

```
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

注一CPU コアとメモリの設定は、ldm set-core、ldm set-memory の順で設定を行うことを推奨します。これにより、まとまったメモリの連続領域が割り当てられやすくなります。CPU コアとメモリの配置の条件の詳細は『SPARC M12/ M10 ドメイン構築ガイド』の「2.5.2 運用時の留意点」の「CPU コアとメモリの配置」を参照してください。

- b. 制御ドメインの最大ページサイズを 256MB に設定します。

SPARC M12 かつ Oracle VM Server for SPARC 3.5 以降の場合、ドメインの最大ページサイズを 256MB に設定します。ドメインの最大ページサイズを 256MB に設定しない場合、物理パーティション

の動的再構成を実施した際にメモリが必要以上に削除される場合があります。

SPARC M10 の場合は、設定不要です。Oracle VM Server for SPARC 3.4 の場合は設定できないため、省略します。

fj_dr_sw_limit_pagesize の値を確認します。false の場合はドメインの最大ページサイズを 256MB に設定します。true の場合は fj_dr_sw_limit_pagesize の値を false に設定した後にドメインの最大ページサイズを 256MB に設定します。

```
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean true
```

値が true の場合は以下を実行し、false に変更します。

```
primary# svcadm disable ldmd
primary# svccfg -s ldmd setprop ldmd/fj_dr_sw_limit_pagesize=false
primary# svcadm refresh ldmd
primary# svcadm enable ldmd
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean false
```

ドメインの最大ページサイズを 256MB に設定します。

```
primary# ldm set-domain fj-software-limit-pagesize=256MB primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

- c. Oracle Solaris を再起動します。

制御ドメインの再起動によって構成情報を反映します。

```
primary# shutdown -i6 -g0 -y
```

- d. 仮想コンソール端末集配信装置サービスを作成します。

ldm add-vconscon コマンドで vcc0 という名前の仮想コンソール端末集配信装置サービスを作成し、svcadm コマンドで仮想ネットワーク端末サーバデーモン(vntsd)を起動します。各論理ドメインへのコンソール接続は、この vcc0 を経由するように構成します。

```
primary# ldm add-vconscon port-range=5000-5200 vcc0 primary
primary# svcadm enable vntsd
```

- e. 構成情報を保存します。

ldm add-spconfig コマンドで現在の構成情報を XSCF に保存します。

次の例では、ldm list-spconfig コマンドで保存されている構成情報を確認し、ldm add-spconfig コマンドで、ldm-set1 という名前で構成を保存し、再度 ldm list-spconfig コマンドで保存されたことを確認する例を示しています。

```
primary# ldm list-spconfig
factory-default [next poweron]

primary# ldm add-spconfig ldm-set1

primary# ldm list-spconfig
factory-default
ldm-set1 [current]
```

f. 制御ドメインのシステムボリュームを冗長構成にします。

ここでは、制御ドメインに割り当てた 2 つの FibreChannel ポートで SAN 上に作成したシステムボリュームを冗長構成にする手順の例を説明します。その他の冗長構成の手順については、各冗長構成ソフトウェアのドキュメントを参照してください。

制御ドメインの起動時間の短縮、および物理パーティションの動的再構成を実施した際の一時停止時間の短縮のため、/etc/system に以下を追加します。また、FibreChannel の空きポートに光ケーブルを接続し、リンクアップ状態にします。

```
forceload: drv/qlc
forceload: drv/emlxs
forceload: drv/ssd
forceload: drv/fp
```

stmsboot コマンドを実行して、現在のマルチパスの設定を確認します。次の例では、マルチパスの設定が無効となっていることが確認できます。

```
primary# stmsboot -D fp -L
stmsboot: MPXIO disabled
```

※または、以下のような出力結果となります。

```
primary# stmsboot -D fp -L
non-STMS device name          STMS device name
-----
```

stmsboot コマンドを実行してマルチパス設定を有効にします。この際、制御ドメインの再起動が必要となります。

```
primary# stmsboot -D fp -e
WARNING: This operation will require a reboot.
```

```

Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y <-- y を押下して OS 再起動
Jul 29 23:43:57 primary reboot: initiated by root on /dev/console
syncing file systems... done
rebooting...
Resetting...

```

制御ドメインの再起動後、stmsboot コマンドを実行してマルチパス設定が有効となったかを確認します。次の例では、2つのディスクパスが1つのマルチパスディスクとして認識されています。

```

primary# stmsboot -D fp -L
non-STMS device name          STMS device name
-----
/dev/rdisk/c10t500000E0D0000087d0    /dev/rdisk/c0t600000E00D0000000000000000000000d0
/dev/rdisk/c9t500000E0D0000086d0    /dev/rdisk/c0t600000E00D0000000000000000000000d0

```

ldm コマンドを実行して、2つのディスクパスを boot-device として指定します。ディスクパスとマルチパスディスクの対応関係については、各冗長構成ソフトウェアのドキュメント(今回の例では、[『SAN Boot 環境構築ガイド\(Leadville ドライバ編\) for SPARC Servers/SPARC Enterprise』](#))を参照してください。

```

primary# ldm set-variable boot-device=Y
"/pci@8100/pci@4/pci@0/pci@0/SUNW,qlc@0/fp@0,0/disk@w500000e0d0000086,0¥
/pci@8900/pci@4/pci@0/pci@0/SUNW,qlc@0,1/fp@0,0/disk@w500000e0d0000087,0 disk net" primary

```

g. 制御ドメインの構成情報を確認します。

ldm コマンドを実行して、制御ドメインの構成情報を確認します。次の例では、SPARC M12 かつ Oracle VM Server for SPARC 3.5 の場合の制御ドメインの CPU、メモリ、物理 I/O が「表 2. 集約型構成における各ドメインのリソース割り当て」の定義と一致していることと、設定内容が正しく反映されていることを確認しています。

```

primary# ldm list-domain -l
NAME    STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary active    -n-cv-   UART    32      8G        0.0%    0.0%    27m
:

```



```

CONTROL
:
  fj-software-limit-pagesize=256MB
VARIABLES
  auto-boot?=false
  boot-device=/pci@8100/pci@4/pci@0/pci@0/SUNW,qlc@0/fp@0,0/disk@w500000e0d0000086,0
              /pci@8900/pci@4/pci@0/pci@0/SUNW,qlc@0,1/fp@0,0/disk@w500000e0d0000087,0
  disk net
  pm_boot_policy=disabled=1;ttfc=0;ttmr=0;

IO
  DEVICE                                PSEUDONYM      OPTIONS
  pci@8000                              PCIE0
  pci@8100                              PCIE1
  pci@8800                              PCIE8
  pci@8900                              PCIE9
  pci@8000/pci@4/pci@0/pci@9            /BB0/CMUL/NET0
  pci@8100/pci@4/pci@0/pci@0            /BB0/CMUL/SASHBA0
  pci@8100/pci@4/pci@0/pci@0            /BB0/PCI0
  pci@8800/pci@4/pci@0/pci@9            /BB1/CMUL/NET0
  pci@8900/pci@4/pci@0/pci@0            /BB1/CMUL/SASHBA0
  pci@8900/pci@4/pci@0/pci@0            /BB1/PCI0
VCC
  NAME      PORT-RANGE
  vcc0      5000-5200
VCONS
  NAME      SERVICE      PORT  LOGGING

```

h. 制御ドメインのネットワークを冗長構成にします。

ここでは、制御ドメイン primary に割り当てた 2 つのネットワークインターフェースを IPMP で冗長構成にする手順の例を説明します。その他の冗長構成の手順については、各冗長構成のソフトウェアのドキュメントを参照してください。(LA にて冗長化することも可能です。)

dladm コマンドを実行して、ネットワークデバイスが見えることを確認します。

次の例では、ネットワークインターフェース net0, net4 としてネットワークデバイスが参照できていま

す。

また、net0 はシステムボード 00-0(BB#0) 配下、net4 はシステムボード 01-0(BB#1) 配下にあることがわかります。

```
primary# dladm show-link
LINK          CLASS      MTU    STATE   OVER
net0          phys       1500   up      --
net4          phys       1500   up      --

primary# dladm show-phys -L
LINK          DEVICE        LOC
net0          ixgbe0        /BB0/CMUL
:
net4          ixgbe4        /BB1/CMUL
```

ipadm show-if コマンドを実行して、net0, net4 が表示されていないことを確認します。

```
primary# ipadm show-if
IFNAME CLASS  STATE  ACTIVE OVER
lo0      loopback ok      yes    --
```

ipadm create-ip コマンドを実行して、IP インターフェース net0, net4 を作成し、ipadm show-if コマンドで、作成できたことを確認します。

```
primary# ipadm create-ip net0
primary# ipadm create-ip net4
primary# ipadm show-if
IFNAME CLASS          STATE  ACTIVE OVER
lo0      loopback          ok      yes    --
net0     ip                down    no     --
net4     ip                down    no     --
```

ipadm create-ipmp コマンドを実行して、IPMP インターフェース ipmp0 を作成し、ipadm add-ipmp コマンドを実行して、IP インターフェース net0, net4 を IPMP グループに追加します。

```
primary# ipadm create-ipmp ipmp0
primary# ipadm add-ipmp -i net0 -i net4 ipmp0
```

ipadm create-addr コマンドを実行して、IPMP インターフェース ipmp0 に IP アドレスを割り当てて、

ipadm show-addr コマンドで設定値を確認します。次の例では、固定 IP アドレスを割り当てる例を示しています。

```
primary# ipadm create-addr -T static -a local=192.168.1.101/24 ipmp0/v4
primary# ipadm show-addr
```

ADDR	OBJ	TYPE	STATE	ADDR
lo0/v4	static	ok		127.0.0.1/8
ipmp0/v4	static	ok		192.168.1.101/24
lo0/v6	static	ok		::1/128

ipadm set-ifprop コマンドを実行して、スタンバイインターフェースを設定し、ipmpstat -i コマンドで、IPMP 構成を確認します。

```
primary# ipadm set-ifprop -p standby=on -m ip net4
primary# ipmpstat -i
```

INTERFACE	ACTIVE	GROUP	FLAGS	LINK	PROBE	STATE
net4	no	ipmp0	is-----	up	disabled	ok
net0	yes	ipmp0	--mbM--	up	disabled	ok

- i. ldmd/fj_dr_sw_limit_pagesize プロパティを設定します。

SPARC M12 かつ Oracle VM Server for SPARC 3.5 以降の場合、新しいドメインを追加する前に ldmd/fj_dr_sw_limit_pagesize プロパティを true に設定します。true に変更することにより、新規に作成する論理ドメインの最大ページサイズが 256 MB に設定されます。

ldmd/fj_dr_sw_limit_pagesize プロパティを true に設定しない場合、物理パーティションの動的再構成を実施した際にドメインからメモリが必要以上に削除される場合があります。

SPARC M10 の場合は、設定不要です。Oracle VM Server for SPARC 3.4 の場合は設定できないため、省略します。

```
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean false
primary# svcadm disable ldmd
primary# svccfg -s ldmd setprop ldmd/fj_dr_sw_limit_pagesize=true
primary# svcadm refresh ldmd
primary# svcadm enable ldmd
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean true
```

- j. I/O ルートドメインを構成します。

ここでは I/O ルートドメインを構成する手順を説明します。

ldm add-domain コマンドを実行し、root-dom0 という名前の論理ドメインを追加します。

```
primary# ldm add-domain root-dom0
```

ldm set-variable コマンドを実行し、OS を自動起動するための OpenBoot PROM 環境変数「auto-boot?」を「false」(無効)に変更します。デフォルトではこの設定が「true」(有効)になっているため、Oracle Solaris がインストールされていない状態で OpenBoot PROM が自動的に OS を起動しようとしてしまいます。これを無効に変更することで、Oracle Solaris のインストールを開始する前の作業がしやすくなります。

```
primary# ldm set-variable auto-boot?=false root-dom0
```

ldm set-core、ldm set-memory コマンドの順序で実行し、CPU コアとメモリを割り当てます。

次の例では、構成例にしたがって、ldm set-core コマンドで CPU コアを 10 個、ldm set-memory コマンドでメモリを 52GB 割り当てています。

```
primary# ldm set-core 10 root-dom0
primary# ldm set-memory 52G root-dom0
```

注—CPU コアとメモリの設定は、ldm set-core、ldm set-memory の順で設定を行うことを推奨します。これにより、まとまったメモリの連続領域が割り当てられやすくなります。CPU コアとメモリの配置の条件の詳細は『SPARC M12/M10 ドメイン構築ガイド』の「2.5.2 運用時の留意点」の「CPU コアとメモリの配置」を参照してください。

ldm set-vconsole コマンドを実行して、仮想コンソール(vconsole)を割り当てます。

次の例では、ldm set-vconsole コマンドを実行して、制御ドメインの仮想コンソール端末集配信装置サービス(vcc0)のポート番号 5000 を仮想コンソールに割り当てています。

```
primary# ldm set-vconsole service=vcc0 port=5000 root-dom0
```

以下の例では、ldm list-io -l コマンドを実行して、PCI の割り当て状況を表示しています。

[NAME]が「/BB0」で始まって、[TYPE]欄が「PCIE」の行が、システムボード 00-0 上の PCIe エンドポイントを意味しています。[DOMAIN]欄が空欄になっている行が、未割り当ての PCIe エンドポイントであり、そのルートコンプレックスは[BUS]欄に表示されます。

したがって、PCIE2, PCIE6, PCIE10, PCIE14 が、未割り当てのルートコンプレックスであることがわかります。

```
primary# ldm list-io -l
```

NAME	TYPE	BUS	DOMAIN	STATUS
/BB0/PCIE0	PCIE	00-0	root-dom0	Allocated
/BB0/PCIE2	PCIE	00-0		Unallocated
/BB0/PCIE6	PCIE	00-0		Unallocated
/BB0/PCIE10	PCIE	00-0		Unallocated
/BB0/PCIE14	PCIE	00-0		Unallocated

----	----	---	-----	-----
(略)				
/BB0/CMUL/NET0	PCIE	PCIE0	primary	OCC
[pci@8000/pci@4/pci@0/pci@9]				
network@0				
network@0,1				
/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
[pci@8000/pci@4/pci@0/pci@0]				
scsi@0/iport@f/disk@w50000393e802cce2,0				
scsi@0/iport@f/disk@w50000393d8285226,0				
scsi@0/iport@f/smp@w500000e0e06d027f				
scsi@0/iport@f/enclosure@w500000e0e06d027d,0				
scsi@0/iport@v0				
(略)				
/BB0/PCI7	PCIE	PCIE2	UNK	
[pci@8200/pci@4/pci@0/pci@0]				
/BB0/PCI3	PCIE	PCIE6	UNK	
[pci@8200/pci@4/pci@0/pci@8]				
(略)				
/BB1/PCI7	PCIE	PCIE10	UNK	
[pci@8a00/pci@4/pci@0/pci@0]				
/BB1/PCI3	PCIE	PCIE14	UNK	
[pci@8a00/pci@4/pci@0/pci@8]				
(以下省略)				

上記の結果で表示されるデバイスパス([pci@....]と表示される文字列)と、『SPARC M12/M10 システム運用・管理ガイド』の「A.3 SPARC M12-2S のデバイスパス」を参照して、どのルートコンプレックスを I/O ルートドメインに割り当ててするかを決めます。

構成例の場合、各ルートドメインにシステムボード 00-0 とシステムボード 01-0 の未割り当てのルートコンプレックスを追加し、各システムボード上の PCIe カードを 2 枚使用できる必要があります。ldm add-io コマンドを実行して root-dom0 に PCIE2 と PCIE6 と PCIE10 と PCIE14 を割り当てます。

```
primary# ldm add-io PCIE2 root-dom0
primary# ldm add-io PCIE6 root-dom0
primary# ldm add-io PCIE10 root-dom0
```

```
primary# ldm add-io PCIE14 root-dom0
```

ldm bind-domain コマンドで I/O ルートドメインを bound 状態にしてから、ldm list-io コマンドを実行して、ルートコンプレックスが割り当てられたことを確認します。

次の例では、SPARC M12 かつ Oracle VM Server for SPARC 3.5 の場合の ldm bind-domain コマンドで root-dom0 をバインドして、ldm list-io でルートコンプレックスが割り当てられたことを確認しています。

[TYPE] 欄が「BUS」になっていて、[DOMAIN] 欄が「root-dom0」になっている行が、root-dom0 に割り当てられたルートコンプレックスを示しています。その行の BUS が、割り当てたルートコンプレックス名です。

下記の例では、PCIE2,PCIE6,PCIE10,PCIE14 が root-dom0 に割り当てられていることが確認できます。

```
primary# ldm bind-domain root-dom0
primary# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
/BB0/CMUL/CMP0/TDM0	BUS	PCIE0	primary	IOV
/BB0/CMUL/CMP0/TDM1	BUS	PCIE1	primary	IOV
/BB0/CMUL/CMP0/TDM2	BUS	PCIE2	root-dom0	IOV
/BB0/CMUL/CMP0/TDM3	BUS	PCIE3		
/BB0/CMUU/CMP0/TDM0	BUS	PCIE4		
/BB0/CMUU/CMP0/TDM1	BUS	PCIE5		
/BB0/CMUU/CMP0/TDM2	BUS	PCIE6	root-dom0	IOV
/BB0/CMUU/CMP0/TDM3	BUS	PCIE7		
/BB1/CMUL/CMP0/TDM0	BUS	PCIE8	primary	IOV
/BB1/CMUL/CMP0/TDM1	BUS	PCIE9	primary	IOV
/BB1/CMUL/CMP0/TDM2	BUS	PCIE10	root-dom0	IOV
/BB1/CMUL/CMP0/TDM3	BUS	PCIE11		
/BB1/CMUU/CMP0/TDM0	BUS	PCIE12		
/BB1/CMUU/CMP0/TDM1	BUS	PCIE13		
/BB1/CMUU/CMP0/TDM2	BUS	PCIE14	root-dom0	IOV
/BB1/CMUU/CMP0/TDM3	BUS	PCIE15		

(以下省略)

手順 j.にしたがって、他の I/O ルートドメイン(例の場合、root-dom1)も同様に構成してください。以下は SPARC M12 かつ Oracle VM Server for SPARC 3.5 以降の場合の例です。

```
primary# ldm add-domain root-dom1
primary# ldm set-variable auto-boot?=false root-dom1
primary# ldm set-core 10 root-dom1
primary# ldm set-memory 52G root-dom1
primary# ldm set-vconsole service=vcc0 port=5001 root-dom1
primary# ldm add-io PCIE3 root-dom1
primary# ldm add-io PCIE7 root-dom1
primary# ldm add-io PCIE11 root-dom1
primary# ldm add-io PCIE15 root-dom1
primary# ldm bind-domain root-dom1
```

4.3. I/O ルートドメインの構築

ここでは、I/O ルートドメインの構築手順を解説します。

4.3.1. Oracle Solaris のインストール

I/O ルートドメインのシステムボリュームに Oracle Solaris をインストールします。

ここでは、I/O ルートドメインのシステムボリュームを SAN 上に作成し、Oracle Solaris をインストールする手順を説明します。SAN 上にシステムボリュームを作成する手順は、各 SAN のドキュメントを参照してください。

ldm start-domain コマンドを実行して I/O ルートドメイン root-dom0 を起動します。

```
primary# ldm start-domain root-dom0
LDom root-dom0 started
```

telnet コマンドを実行して、I/O ルートドメインのコンソールに接続します。次の例では、ldm list-domain コマンドを実行して root-dom0 のコンソールのポート番号が 5000 であることを確認し、telnet コマンドで localhost のポート番号 5000 に接続し、root-dom0 が OpenBoot PROM (OBP) の状態で止まっていることが確認できます。

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	8G	0.0%	7h 7m
root-dom0	active	-t----	5000	80	52G	0.0%	20s
root-dom1	bound	-----	5001	80	52G		

```
primary# telnet localhost 5000
....
{0} ok
```

I/O ルートドメインに Oracle Solaris をインストールします。

次の例は、ネットワーク経由で Oracle Solaris11 のインストールを開始するコマンドの実行例です。

```
{0} ok boot net:dhcp
....
```

インストールの詳細は、Oracle 社のホームページ(<https://docs.oracle.com/>)で公開されている、以下のドキュメントを参照してください。

- Oracle Solaris 11
『Oracle Solaris 11.3 システムのインストール』

ldm set-variable コマンドを実行し、OS を自動起動するための OprnBoot PROM 環境変数「auto-boot?」を「true」(有効)に変更します。Oracle Solaris をインストール後は、ldm start-domain コマンドを実行すると Oracle Solaris まで起動します。

```
primary# ldm set-variable auto-boot?="true" root-dom0
```

「4.3.1. Oracle Solaris のインストール」にしたがって、他の I/O ルートドメイン(例の場合、root-dom1)も同様にインストールします。

4.3.2. I/O ルートドメインの冗長構成設定

- 各ドメインのシステムボリュームを冗長構成にします。

ここでは、I/O ルートドメイン root-dom0 に割り当てた 2 つの FibreChannel ポートで SAN 上に作成したシステムボリュームを冗長構成にする手順の例を説明します。その他の冗長構成の手順については、各冗長構成ソフトウェアのドキュメントを参照してください。

I/O ルートドメイン root-dom0 にログインします。

次の例では、ldm list-domain コマンドで root-dom0 のコンソールのポート番号を確認し、telnet コマンドでポート番号「5000」に接続する実行例を示しています。

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	8G	0.0%	8h 7m
root-dom0	active	-n----	5000	80	52G	0.0%	43s
root-dom1	active	-n----	5001	80	52G	0.0%	20s

```
primary# telnet localhost 5000
....
```



```
root-dom0 console login: root
Password:
...
root-dom0#
```

I/O ルートドメインの起動時間の短縮、および物理パーティションの動的再構成を実施した際の一時停止時間の短縮のため、/etc/system に以下を設定します。また、FibreChannel の空きポートに光ケーブルを接続し、リンクアップ状態にします。

```
forceload: drv/qlc
forceload: drv/emlxs
forceload: drv/ssd
forceload: drv/fp
```

stmsboot コマンドを実行して、現在のマルチパスの設定を確認します。次の例では、マルチパスの設定が無効となっていることが確認できます。

```
root-dom0# stmsboot -D fp -L
stmsboot: MPXIO disabled
```

stmsboot コマンドを実行してマルチパス設定を有効にします。この際、I/O ルートドメインの再起動が必要となります。

```
root-dom0# stmsboot -D fp -e
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y <- y を押下して OS 再起動
Jul 29 23:43:57 root-dom0 reboot: initiated by root on /dev/console
syncing file systems... done
rebooting...
Resetting...
```

I/O ルートドメインの再起動後、stmsboot コマンドを実行してマルチパス設定が有効になったか確認します。次の例では、2 つのディスクパスが 1 つのマルチパスディスクとして認識されています。

```
root-dom0# stmsboot -D fp -L
non-STMS device name          STMS device name
```

```
/dev/rdsd/c4t500000E0D0000087d0 /dev/rdsd/c0t600000E0D00000000000000000000030000d0  
/dev/rdsd/c3t500000E0D0000086d0 /dev/rdsd/c0t600000E0D00000000000000000000030000d0
```

制御ドメイン上で ldm コマンドを実行し、2 つのディスクパスを boot-device として指定します。ディスクパスとマルチパスディスクの対応関係については、各冗長構成ソフトウェアのドキュメント(今回の例では、『[SAN Boot 環境構築ガイド\(Leadville ドライバ編\) for SPARC Servers/SPARC Enterprise](#)』)を参照してください。

```
primary# ldm set-variable boot-device=¥  
"/pci@8200/pci@4/pci@0/pci@0/emlx@0/fp@0,0/disk@w500000e0d0000086,0:a ¥  
/pci@8a00/pci@4/pci@0/pci@0/emlx@0,1/fp@0,0/disk@w500000e0d0000087,0:a disk net" root-dom0
```

手順 a にしたがって、他の I/O ルートドメイン(例の場合、root-dom1)も同様に設定を行います。

- b. 各ドメインのネットワークを冗長構成にします。

4.2. 手順 h. にしたがって、root-dom0, root-dom1 のネットワークインターフェースを冗長構成にします。

5. 高集約型の構築

本章では、高集約型の構築手順を解説します。

5.1. Oracle Solaris と Oracle VM Server for SPARC のインストール

制御ドメインに Oracle Solaris と OVM をインストールします。

BB HA に必要な Oracle Solaris と OVM の版数や条件などについては、『SPARC M12/M10 Building Block を活用した高可用性システム(概要編)』を参照してください。

インストールの詳細は、Oracle 社のホームページ(<https://docs.oracle.com/>)で公開されている、以下のドキュメントを参照してください。

- Oracle Solaris 11
『Oracle Solaris 11.3 システムのインストール』
お使いのバージョンの『Oracle VM Server for SPARC インストールガイド』

5.2. 論理ドメインの構築

ここでは、「表 3. 高集約型構成における各ドメインのリソース割り当て」で定義した論理ドメインの構築方法について説明します。

- a. 制御ドメインのリソースを減らします。

factory-default の構成の場合、すべての CPU コア、メモリが制御ドメイン(primary)に割り当てられています。これらのリソースを他の論理ドメインに割り当てられるようにするために、最初に、制御ドメインからこれらのリソースの一部を削減します。以下にコマンドの実行例を示します。

ldm start-reconf コマンドで遅延再構成に移行します。

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

ldm set-core コマンド、ldm set-memory コマンドを実行して、元のサイズより小さいサイズを指定することで、制御ドメインに割り当てている CPU コア数とメモリサイズを減らします。

次の例では、構成例にしたがって制御ドメインの CPU コアを 6 個、メモリサイズを 48GB に設定します。

```
primary# ldm set-core 6 primary
```

```
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

```
primary# ldm set-memory 48G primary
```

```
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

注—CPU コアとメモリの設定は、ldm set-core、ldm set-memory の順で設定を行うことを推奨します。これにより、まとまったメモリの連続領域が割り当てられやすくなります。CPU コアとメモリの配置の条件の詳細は『SPARC M12/M10 ドメイン構築ガイド』の「2.5.2 運用時の留意点」の「CPU コアとメモリの配置」を参照してください。

- b. 制御ドメインの最大ページサイズを 256MB に設定します。

SPARC M12 かつ Oracle VM Server for SPARC 3.5 以降の場合、ドメインの最大ページサイズを 256MB に設定します。ドメインの最大ページサイズを 256MB に設定しない場合、物理パーティションの動的再構成を実施した際にメモリが必要以上に削除される場合があります。

SPARC M10 の場合は、設定不要です。Oracle VM Server for SPARC 3.4 の場合は設定できないため、省略します。

fj_dr_sw_limit_pagesize の値を確認します。false の場合はドメインの最大ページサイズを 256MB に設定します。true の場合は fj_dr_sw_limit_pagesize の値を false に設定した後にドメインの最大ページサイズを 256MB に設定します。

```
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean true
```

値が true の場合は以下を実行し、false に変更します。

```
primary# svcadm disable ldmd
primary# svccfg -s ldmd setprop ldmd/fj_dr_sw_limit_pagesize=false
primary# svcadm refresh ldmd
primary# svcadm enable ldmd
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean false
```

ドメインの最大ページサイズを 256MB に設定します。

```
primary# ldm set-domain fj-software-limit-pagesize=256MB primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

- c. Oracle Solaris を再起動します。

制御ドメインの再起動によって構成情報を反映します。

```
primary# shutdown -i6 -g0 -y
```

- d. 仮想コンソール端末集配信装置サービスを作成します。

ldm add-vconscon コマンドで vcc0 という名前の仮想コンソール端末集配信装置サービスを作成し、svcadm コマンドで仮想ネットワーク端末サーバデーモン(vntsd)を起動します。各論理ドメインへのコンソール接続は、この vcc0 を経由するように構成します。

```
primary# ldm add-vconscon port-range=5000-5200 vcc0 primary
primary# svcadm enable vntsd
```

- e. 構成情報を保存します。

ldm add-spconfig コマンドで現在の構成情報を XSCF に保存します。

次の例では、ldm list-spconfig コマンドで保存されている構成情報を確認し、ldm add-spconfig コマンドで、ldm-set1 という名前で構成を保存し、再度 ldm list-spconfig コマンドで保存されたことを確認する例を示しています。

```
primary# ldm list-spconfig
factory-default [next poweron]

primary# ldm add-spconfig ldm-set1
primary# ldm list-spconfig
factory-default
ldm-set1 [current]
```

- f. 制御ドメインのシステムボリュームを冗長構成にします。

ここでは、制御ドメインに割り当てた 2 つの FibreChannel ポートで SAN 上に作成したシステムボリュームを冗長構成にする手順の例を説明します。その他の冗長構成の手順については、各冗長構成ソフトウェアのドキュメントを参照してください。

制御ドメインの起動時間の短縮、および物理パーティションの動的再構成を実施した際の一時停止時

間の短縮のため、/etc/system に以下を追加します。また、FibreChannel の空きポートに光ケーブルを接続し、リンクアップ状態にします。

```
forceload: drv/qlc
forceload: drv/emlxs
forceload: drv/ssd
forceload: drv/fp
```

stmsboot コマンドを実行して、現在のマルチパスの設定を確認します。次の例では、マルチパスの設定が無効となっていることが確認できます。

```
primary# stmsboot -D fp -L
stmsboot: MPXIO disabled
```

※または、以下のような出力結果となります。

```
primary# stmsboot -D fp -L
non-STMS device name          STMS device name
-----
```

stmsboot コマンドを実行してマルチパス設定を有効にします。この際、制御ドメインの再起動が必要となります。

```
primary# stmsboot -D fp -e
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y <-- y を押下して OS 再起動
Jul 29 23:43:57 primary reboot: initiated by root on /dev/console
syncing file systems... done
rebooting...
Resetting...
```

制御ドメインの再起動後、stmsboot コマンドを実行してマルチパス設定が有効となったかを確認します。次の例では、2 つのディスクパスが 1 つのマルチパスディスクとして認識されています。

```
primary# stmsboot -D fp -L
non-STMS device name          STMS device name
-----
/dev/rdisk/c10t500000E0D0000087d0    /dev/rdisk/c0t600000E00D0000000000000000000000d0
/dev/rdisk/c9t500000E0D0000086d0    /dev/rdisk/c0t600000E00D0000000000000000000000d0
```

ldm コマンドを実行して、2 つのディスクパスを boot-device として指定します。ディスクパスとマルチパスディスクの対応関係については、各冗長構成ソフトウェアのドキュメント(今回の例では、[『SAN Boot 環境構築ガイド\(Leadville ドライバ編\) for SPARC Servers/SPARC Enterprise』](#)) を参照してください。

```
primary# ldm set-variable boot-device=¥
"/pci@8100/pci@4/pci@0/pci@0/SUNW,qlc@0/fp@0,0/disk@w500000e0d0000086,0¥
/pci@8900/pci@4/pci@0/pci@0/SUNW,qlc@0,1/fp@0,0/disk@w500000e0d0000087,0 disk net" primary
```

g. 制御ドメインの構成情報を確認します。

ldm コマンドを実行して、制御ドメインの構成情報を確認します。次の例では、SPARC M12 かつ Oracle VM Server for SPARC 3.5 の場合の制御ドメインの CPU、メモリ、仮想 I/O が「表 3. 高集約型構成における各ドメインのリソース割り当て」の定義と一致していることと、設定内容が正しく反映されていること確認しています。

```
primary# ldm list-domain -l
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	48	48G	0.0%	0.0%	27m

```

:
CONTROL
:
fj-software-limit-pagesize=256MB
VARIABLES
auto-boot?=false
boot-device=/pci@8100/pci@4/pci@0/pci@0/SUNW,qlc@0/fp@0,0/disk@w500000e0d0000086,0
/pci@8900/pci@4/pci@0/pci@0/SUNW,qlc@0,1/fp@0,0/disk@w500000e0d0000087,0
disk net
pm_boot_policy=disabled=1;ttfc=0;ttmr=0;
IO
DEVICE                                PSEUDONYM          OPTIONS
pci@8000                              PCIE0
pci@8100                              PCIE1
pci@8200                              PCIE2
pci@8300                              PCIE3
pci@8400                              PCIE4
pci@8500                              PCIE5
```

pci@8600	PCIE6		
pci@8700	PCIE7		
pci@8800	PCIE8		
pci@8900	PCIE9		
pci@8a00	PCIE10		
pci@8b00	PCIE11		
pci@8c00	PCIE12		
pci@8d00	PCIE13		
pci@8e00	PCIE14		
pci@8f00	PCIE15		
:			
VCC			
NAME	PORT-RANGE		
vcc0	5000-5200		
VCONS			
NAME	SERVICE	PORT	LOGGING
	UART		

h. 制御ドメインのネットワークを冗長構成にします。

ここでは、制御ドメイン primary に割り当てた 2 つのネットワークインターフェースをリンクアグリゲーション(LA)で冗長構成にする手順の例を説明します。その他の冗長構成の手順については、各冗長構成のソフトウェアのドキュメントを参照してください。

補足 高集約型の構成では、仮想ネットワークスイッチ(vsw)を作成する必要がありますが、IPMP にて冗長化したネットワークインターフェースからは仮想ネットワークスイッチ(vsw)を作成することができません。そのため、ここでは LA にて作成する手順を記載しています。制御ドメインのネットワークを、LA にて冗長化することで、業務が動作しているゲストドメインに冗長化構成を意識させない構成にできます。

dladm コマンドを実行して、ネットワークデバイスが見えることを確認します。

次の例では、ネットワークインターフェース net0, net4 としてネットワークデバイスが参照できています。

また、net0 はシステムボード 00-0(BB#0) 配下、net4 はシステムボード 01-0(BB#1) 配下にあることがわかります。

primary# dladm show-link				
LINK	CLASS	MTU	STATE	OVER
net0	phys	1500	up	--


```

net4                phys        1500    up      --
primary# dladm show-phys -L
LINK                DEVICE      LOC
net0                ixgbe0       /BB0/CMUL
:
net4                ixgbe4       /BB1/CMUL

```

ipadm show-if コマンドを実行して、net0, net4 が表示されていないことを確認します。

```

primary# ipadm show-if
IFNAME CLASS  STATE  ACTIVE OVER
lo0      loopback ok      yes    --

```

LA に構成しようとしているネットワークデバイスが IP インターフェースとしてここに表示される場合は、その IP インターフェースを削除してください。

```

primary# ipadm show-if
IFNAME CLASS  STATE  ACTIVE OVER
lo0      loopback ok      yes    --
net0     ip      ok      yes    --
primary# ipadm delete-ip net0
primary# ipadm show-if
IFNAME CLASS  STATE  ACTIVE OVER
lo0      loopback ok      yes    --

```

dladm create-aggr コマンドを実行して、アグリゲーション aggr0 を作成し、dladm show-aggr コマンドで、作成できたことを確認します。

```

primary# dladm create-aggr -m dlmp -l net0 -l net4 aggr0
primary# dladm show-aggr
LINK  MODE  POLICY  ADDRPOLICY  LACPACTIVITY  LACPTIMER
aggr0  dlmp  --      --          --            --

```

ipadm create-addr コマンドを実行して、LA インターフェース aggr0 に IP アドレスを割り当て、ipadm show-addr コマンド、dladm コマンドで設定値を確認します。次の例では、固定 IP アドレスを割り当てる例を示しています。

```

primary# ipadm create-ip aggr0
primary# ipadm create-addr -T static -a local=192.168.1.101/24 aggr0/v4

```

```
primary# ipadm show-addr
ADDR    OBJ      TYPE    STATE   ADDR
lo0/v4   static   ok      127.0.0.1/8
aggr0/v4 static   ok      192.168.1.101/24
lo0/v6   static   ok      ::1/128

primary# dladm show-link
LINK           CLASS    MTU    STATE   OVER
net0           phys     1500   up      --
net4           phys     1500   up      --
:
aggr0          aggr     1500   up      net0 net4
```

- i. ldmd/fj_dr_sw_limit_pagesize プロパティを設定します。

SPARC M12 かつ Oracle VM Server for SPARC 3.5 以降の場合、新しいドメインを追加する前に ldmd/fj_dr_sw_limit_pagesize プロパティを true に設定します。true に変更することにより、新規に作成する論理ドメインの最大ページサイズが 256 MB に設定されます。

ldmd/fj_dr_sw_limit_pagesize プロパティを true に設定しない場合、物理パーティションの動的再構成を実施した際にドメインからメモリが必要以上に削除される場合があります。

SPARC M10 の場合は、設定不要です。Oracle VM Server for SPARC 3.4 の場合は設定できないため、省略します。

```
primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean false

primary# svcadm disable ldmd

primary# svccfg -s ldmd setprop ldmd/fj_dr_sw_limit_pagesize=true

primary# svcadm refresh ldmd

primary# svcadm enable ldmd

primary# svccfg -s ldmd listprop ldmd/fj_dr_sw_limit_pagesize
ldmd/fj_dr_sw_limit_pagesize boolean true
```

- j. ゲストドメインを構成します。

ここではゲストドメインを構成する手順を説明します。

ldm add-domain コマンドを実行し、guest0 という名前の論理ドメインを追加します。

```
primary# ldm add-domain guest0
```

ldm set-variable コマンドを実行し、OS を自動起動するための OpenBoot PROM 環境変数「auto-boot?」を「false」(無効)に変更します。デフォルトではこの設定が「true」(有効)になっているため、

Oracle Solaris がインストールされていない状態で OpenBoot PROM が自動的に OS を起動しようとしてしまいます。これを無効に変更することで、Oracle Solaris のインストールを開始する前の作業がしやすくなります。

```
primary# ldm set-variable auto-boot?=false guest0
```

ldm set-core、ldm set-memory コマンドの順序で実行し、CPU コアとメモリを割り当てます。

次の例では、構成例にしたがって、ldm set-core コマンドで CPU コアを 6 個、ldm set-memory コマンドでメモリを 32GB 割り当てています。

```
primary# ldm set-core 6 guest0
primary# ldm set-memory 32G guest0
```

注—CPU コアとメモリの設定は、ldm set-core、ldm set-memory の順で設定を行うことを推奨します。これにより、まとまったメモリの連続領域が割り当てられやすくなります。CPU コアとメモリの配置の条件の詳細は『SPARC M12/M10 ドメイン構築ガイド』の「2.5.2 運用時の留意点」の「CPU コアとメモリの配置」を参照してください。

ldm set-vconsole コマンドを実行して、仮想コンソール(vconsole)を割り当てます。

次の例では、ldm set-vconsole コマンドを実行して、制御ドメインの仮想コンソール端末集配信装置サービス(vcc0)のポート番号 5000 を仮想コンソールに割り当てています。

```
primary# ldm set-vconsole service=vcc0 port=5000 guest0
```

5.3. ゲストドメインの構築

ここでは、ゲストドメインの構築手順を解説します。

5.3.1. Oracle Solaris のインストール

ゲストドメインに Oracle Solaris をインストールします。

ここでは、ゲストドメインを SAN 上に作成し、Oracle Solaris をインストールする手順を説明します。SAN 上にシステムボリュームを作成する手順は、各 SAN のドキュメントを参照してください。

ldm start-domain コマンドを実行して、ゲストドメイン guest0 を起動します。

```
primary# ldm bind guest0
primary# ldm start-domain guest0
LDom guest0 started
```

telnet コマンドを実行して、ゲストドメインのコンソールに接続します。次の例では、ldm list-domain コマンドを実行して guest0 のコンソールのポート番号が 5000 であることを確認し、telnet コマンドで localhost のポート番号 5000 に接続し、guest0 が OpenBoot PROM (OBP) の状態で止まっていることが確認できます。

```
primary# ldm list-domain
NAME          STATE   FLAGS   CONS  VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active  -n-cv-  UART   48    48G     0.0%  0.0%  1h 45m
guest0        active  -t-----  5000   48    32G     3.1%  3.1%  2m
primary# telnet localhost 5000
....
{0} ok
```

ゲストドメインに Oracle Solaris をインストールします。

次の例は、ネットワーク経由で Oracle Solaris 11 のインストールを開始するコマンドの実行例です。

```
{0} ok boot net:dhcp
....
```

インストールの詳細は、Oracle 社のホームページ (<https://docs.oracle.com/>) で公開されている、以下のドキュメントを参照してください。

- Oracle Solaris 10
『Oracle Solaris 10 1/13 インストールガイド』
- Oracle Solaris 11
『Oracle Solaris 11.3 システムのインストール』

ldm set-variable コマンドを実行し、OS を自動起動するための OpenBoot PROM 環境変数「auto-boot?」を「true」(有効)に変更します。Oracle Solaris をインストール後は、ldm start-domain コマンドを実行すると Oracle Solaris まで起動します。

```
primary# ldm set-variable auto-boot?=Ytrue guest0
```

「5.3.1. Oracle Solaris のインストール」にしたがって、他のゲストドメイン(例の場合、guest1, guest2)も同様にインストールします。

6. 構成情報の保存

本章では、構成情報の保存について解説します。

6.1. Oracle Solaris ゾーン設定の保存

ここでは、従来型構成における、Oracle Solaris ゾーン設定の保存方法を解説します。Oracle Solaris ゾーンを使用していない場合は、実施不要です。

a. ノングローバルゾーンの停止

ノングローバルゾーン上で、shutdown コマンドを実行します。

```
root@zone0:~# shutdown -i5 -g0 -y

Shutdown started.    2015 年 08 月 06 日 (木) 01 時 38 分 34 秒 JST

Changing to init state 5 - please wait
Broadcast Message from root (console) on zone0 木  8 月  6 01:38:34...
THE SYSTEM zone0 IS BEING SHUT DOWN NOW !!!
Log off now or risk your files being damaged

root@zone0:~# svc.startd: The system is coming down.  Please wait.
svc.startd: 94 system services are now being stopped.

[NOTICE: Zone halted]
~.
[Connection to zone 'zone0' console closed]
primary# zoneadm list -cv
```

ID	NAME	STATUS	PATH	BRAND	IP
0	global	running	/	solaris	shared
-	zone0	installed	/zones/zone0	solaris	excl

```
primary#
```

b. Oracle Solaris ゾーン設定の保存

zonecfg コマンドにより、任意の領域に Oracle Solaris ゾーンの設定を export します。

```
primary# zonecfg -z zone0 export -f /export/home/zone0_cfg
primary# ls /export/home/
zone0_cfg
primary#
```

6.2. Oracle VM Server for SPARC の設定と構成情報の保存

ここでは、OVM の設定および、構成情報の保存方法を解説します。※従来型も同様に実施してください。

6.2.1. 復旧モードの有効化

制御ドメイン上で svccfg コマンドを実行して、ldmd の復旧モードを有効にします。復旧モードの詳細は、『Oracle VM Server for SPARC 管理ガイド』の「ハードウェアエラーの処理」を参照してください。

```
primary# svccfg -s ldmd setprop ldmd/recovery_mode = astring: auto
primary# svcadm refresh ldmd
primary# svcadm restart ldmd
```

ハードウェア故障発生後、制御ドメインを自動起動させるために、制御ドメインが自動起動するための OpenBoot PROM 環境変数「auto-boot?」を「true」(有効)に変更します。

```
primary# ldm set-variable auto-boot?=true primary
```

6.2.2. ハイパーバイザダンプ採取後のシステム自動再起動の有効化

制御ドメイン上で ldm コマンドを実行して、ハイパーバイザダンプ採取後のシステム自動再起動を有効にします。ハイパーバイザダンプの詳細は『SPARC M12/M10 システム運用・管理ガイド』の「ハイパーバイザのダンプファイルを採取する」を参照してください。

制御ドメイン上で ldm list-hvdump コマンドを実行し、現在の設定を確認します。次の例ではハイパーバイザダンプ採取が有効、ハイパーバイザダンプ採取後のシステム自動再起動が無効になっています。

```
primary# ldm list-hvdump
hvdump=on
hvdump-reboot=off
```

制御ドメイン上で `ldm set-hvdump` コマンドを実行し、ハイパーバイザダンプ採取後のシステム自動再起動を有効にします。

```
primary# ldm set-hvdump hvdump-reboot=on
primary# ldm list-hvdump
hvdump=on
hvdump-reboot=on
```

6.2.3. 論理ドメイン構成情報を XSCF へ保存

制御ドメイン上で `ldm add-spconfig` コマンドを実行して、構成した情報を保存します。

次の例は、すでに保存されている構成情報を確認して、同じ名前で再度保存しています。

制御ドメイン上で `ldm list-spconfig` コマンドを実行して、現在の構成情報を確認します。

```
primary# ldm list-spconfig
factory-default
ldm-set1 [next poweron]
```

`ldm remove-spconfig` コマンドを実行して、上書きしたい構成情報を削除します。

```
primary # ldm remove-spconfig ldm-set1
```

`ldm add-spconfig` コマンドを実行して、構成情報を保存し直します。

```
primary # ldm add-spconfig ldm-set1
```

`ldm list-spconfig` コマンドを実行して、保存した構成情報が `[current]` になっていることを確認します。

```
primary# ldm list-spconfig
factory-default
ldm-set1 [current]
```

6.2.4. 論理ドメイン構成情報のバックアップ

構築した論理ドメイン構成情報を XML ファイルに保存します。

XSCF に保存した構成情報が使えなくなった場合に備えて、XML ファイルに構成情報を保存しておきます。

XML ファイルは別媒体に保存しておくことをお勧めします。

以下に手順の例を説明します。

制御ドメイン上で `ldm list-domain` コマンドを実行して、すべての論理ドメインが `active` 状態になっていることを確認します。

以下、集約型の例です。

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	8G	0.0%	6h 9m
root-dom0	active	-n----	5000	80	52G	0.0%	15m
root-dom1	active	-n----	5001	80	52G	0.0%	15m

ldm list-constraints コマンドを実行して、構成情報を XML ファイルに保存します。

```
primary # ldm list-constraints -x > /var/tmp/ldm-set1.xml
```

改版履歴

改版日時	版数	改版内容
2015 年 6 月	初版	新規作成
2015 年 9 月	2 版	従来型、高集約型を追加
2017 年 4 月	3 版	SPARC M12-2S サポートによる記事修正
2017 年 10 月	4 版	Oracle VM Server for SPARC 3.5 サポートによる記事修正
2021 年 4 月	5 版	各ドキュメントリンクアドレス最新化

