SPARC M12/M10

ドメイン構築ガイド

マニュアル番号： C120-E680-21
2024 年 7 月

Copyright © 2007, 2024, 富士通株式会社 All rights reserved.

本書には、オラクル社および/またはその関連会社により提供および修正された技術情報が含まれています。

オラクル社および/またはその関連会社、および富士通株式会社は、それぞれ本書に記述されている製品および技術に関する知的所有権を所有または管理しています。これらの製品、技術、および本書は、著作権法、特許権などの知的所有権に関する法律および国際条約により保護されています。

本書およびそれに付属する製品および技術は、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。オラクル社および/またはその関連会社、および富士通株式会社およびそのライセンサーの書面による事前の許可なく、このような製品または技術および本書のいかなる部分も、いかなる方法によっても複製することが禁じられます。本書の提供は、明示的であるか黙示的であるかを問わず、本製品またはそれに付随する技術に関するいかなる権利またはライセンスを付与するものでもありません。本書は、オラクル社および富士通株式会社の一部、あるいはそのいずれかの関連会社のいかなる種類の義務を含むものでも示すものでもありません。

本書および本書に記述されている製品および技術には、ソフトウェアおよびフォント技術を含む第三者の知的財産が含まれている場合があります。これらの知的財産は、著作権法により保護されているか、または提供者からオラクル社および/またはその関連会社、および富士通株式会社へライセンスが付与されているか、あるいはその両方です。

GPLまたはLGPLが適用されたソースコードの複製は、GPLまたはLGPLの規約に従い、該当する場合に、お客様からのお申し込みに応じて入手可能です。オラクル社および/またはその関連会社、および富士通株式会社にお問い合わせください。この配布には、第三者が開発した構成要素が含まれている可能性があります。本製品の一部は、カリフォルニア大学からライセンスされているBerkeley BSDシステムに由来しています。

UNIXはThe Open Groupの登録商標です。

OracleとJavaはOracle Corporationおよびその関連企業の登録商標です。

富士通および富士通のロゴマークは、富士通株式会社の登録商標です。

SPARC Enterprise、SPARC64、SPARC64ロゴ、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。

その他の名称は、それぞれの所有者の商標または登録商標です。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

免責条項: 本書または本書に記述されている製品や技術に関してオラクル社、富士通株式会社および/またはそのいずれかの関連会社が行う保証は、製品または技術の提供に適用されるライセンス契約で明示的に規定されている保証に限ります。このような契約で明示的に規定された保証を除き、オラクル社、富士通株式会社および/またはそのいずれかの関連会社は、製品、技術、または本書に関して、明示、黙示を問わず、いかなる種類の保証も行いません。これらの製品、技術、または本書は、現状のまま提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も、かかる免責が法的に無効とされた場合を除き、行われないものとします。このような契約で明示的に規定されていないかぎり、オラクル社、富士通株式会社および/またはそのいずれかの関連会社は、いかなる法理論のもとの第三者に対しても、その収益の損失、有用性またはデータに関する損失、あるいは業務の中断について、あるいは間接的損害、特別損害、付随的損害、または結果的損害について、そのような損害の可能性が示唆されていた場合であっても、適用される法律が許容する範囲内で、いかなる責任も負いません。

本書は、「現状のまま」提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も、かかる免責が法的に無効とされた場合を除き、行われないものとします。

目次

はじめに　xi

第1章　ドメイン構築の概要を理解する　1

1.1　SPARC M12/M10のドメイン構成　1

1.1.1　1台構成／ビルディングブロック構成の概要　1

1.1.2　論理ドメイン構成の概要　3

1.1.3　ドメイン構成の特長　5

1.2　SPARC M12/M10のハードウェアリソース　9

1.3　物理パーティションとは　12

1.3.1　物理パーティションの構成要素を理解する　12

1.3.2　物理パーティション構成を操作する　14

1.4　論理ドメインとは　17

1.4.1　論理ドメインの構成要素を理解する　17

1.4.2　論理ドメインの構成を操作する　19

1.4.3　物理I/Oデバイスの再構成　21

1.5　物理パーティションの動的再構成とは　22

1.5.1　物理パーティションの動的再構成の概要　22

1.5.2　物理パーティションの動的再構成の用途　22

1.5.3　CPUとメモリの動的再構成　23

1.5.4　ソフトウェア条件および確認方法　24

1.6　SPARC64 X+プロセッサとSPARC64 Xプロセッサの混在　27

1.6.1　CPU動作の種類とCPU動作モード　28

第2章　システム運用のための条件と設定　29

2.1　XSCFのステータス管理　29

2.1.1　物理パーティションのステータス　29

2.1.2　システムボードのステータス　30

2.1.3　ステータスの変化　32

2.2　XSCFの条件と設定　35

2.2.1　XSCFが必要とする構築条件　36

2.2.2　XSCFによる設定　36

2.2.3　メモリミラーモードの留意点　38

2.2.4　CPUモジュールを増設する場合の留意点　38

2.3　Oracle Solarisの条件と設定　38

2.3.1　I/Oおよびソフトウェアの要件　39

2.3.2　スワップ領域に関する留意点　39

2.3.3　リアルタイムプロセスの留意点　40

2.4　論理ドメインの条件と設定　40

2.4.1　論理ドメインの構築に関する留意点　41

2.4.2　論理ドメインの再構成に関する留意点　52

2.5　動的再構成の条件と設定　53

2.5.1　構築時の留意点　53

2.5.2　運用時の留意点　55

2.5.3　PPAR DRモードの確認／設定方法　65

2.6　SPARC64 X+プロセッサを使用する場合の留意点　67

2.6.1　XCPファームウェアのアップデート　67

2.6.2　CPU動作モードの設定　67

2.6.3　SPARC64 X+プロセッサとSPARC64 Xプロセッサを PPAR内で混在させるための条件　71

2.6.4　CPU動作モードと物理パーティションの動的再構成の関係　72

第3章　ドメイン構築のための操作　75

3.1　物理パーティション構築に関する操作とコマンド　75

3.1.1　物理パーティション構成情報を確認する　76

3.1.2　物理パーティションのステータスを確認する　78

3.1.3　システムボードのステータスを確認する　79
3.1.4　デバイスの設定情報を確認する　82
3.1.5　論理ドメインのステータスを確認する　83
3.1.6　CPUコア アクティベーションキー情報を確認する　84
3.1.7　CPUコアリソースの使用状況を確認する　85
3.1.8　CPUコア アクティベーションの登録および設定情報を確認する　87
3.1.9　物理パーティションの動作モードを確認する　88
3.1.10　物理パーティションの論理ドメイン構成情報を表示する　89
3.1.11　メモリミラーを設定する　90
3.1.12　物理パーティション構成情報を設定する　91
3.1.13　CPUコア アクティベーションキーを追加する　93
3.1.14　物理パーティションで使用するCPUコアリソースを割り当てる　93
3.1.15　ビルディングブロック構成においてシステムボードを追加する　96
3.1.16　ビルディングブロック構成においてシステムボードを削除する　98
3.1.17　物理パーティションの動作モードを設定する　100
3.1.18　物理パーティションの論理ドメイン構成情報を指定する　102
3.1.19　物理パーティションを起動する　103
3.1.20　物理パーティションを停止する　103
3.1.21　制御ドメインコンソールに接続する　103
3.2　論理ドメイン構築に関する操作とコマンド　104
3.2.1　Logical Domains Managerの起動を確認する　105
3.2.2　サービスを確認する　106
3.2.3　CPUコア アクティベーションに基づく割り当て可能な仮想CPU数を確認する　106
3.2.4　リソースの割り当て状況を確認する　107
3.2.5　リソースの使用状況を確認する　107
3.2.6　リソースグループごとに使用状況を確認する　109

3.2.7　CPUソケットごとにリソースの使用状況を確認する　109
3.2.8　論理ドメインの状態を確認する　111
3.2.9　論理ドメイン構成情報を表示する　112
3.2.10　I/Oデバイスの使用状況を確認する　112
3.2.11　遅延再構成モードを開始する　114
3.2.12　デフォルトのサービスを設定する　115
3.2.13　仮想CPUを設定する　116
3.2.14　仮想メモリを設定する　121
3.2.15　CPUソケット制約を設定する　124
3.2.16　論理ドメイン構成情報を設定する　124
3.2.17　論理ドメインを作成する　125
3.2.18　I/Oデバイスを設定する　126
3.2.19　SR-IOVの仮想機能を作成・破棄する　128
3.2.20　仮想ネットワークデバイスを設定する　132
3.2.21　仮想ディスクサーバを設定する　133
3.2.22　仮想ディスクを設定する　134
3.2.23　仮想コンソールを設定する　135
3.2.24　起動デバイスを設定する　135
3.2.25　リソースをバインドする　136
3.2.26　ゲストドメインを起動する　136
3.2.27　シャットダウングループを指定する　136
3.2.28　デバイスを再構成する　137
3.2.29　復旧モードを有効にする　137
第4章　物理パーティションの構築　139
4.1　物理パーティション構築のながれ　139
4.2　物理パーティションの構築と操作　140
第5章　論理ドメインの構築例　149
5.1　論理ドメイン構築のながれ　149
5.2　論理ドメイン構築の操作例　151
5.2.1　制御ドメインにログインする　152
5.2.2　デフォルトサービスを設定する　152

5.2.3　制御ドメインを初期設定する　153
5.2.4　ゲストドメインを構築する　154
5.2.5　復旧モード（recovery mode）を設定する　156
5.2.6　論理ドメインの構成情報を保存する　156
第6章　物理パーティションの再構築　157
6.1　物理パーティション再構築のながれ　157
6.1.1　システムボードを追加する操作のながれ　157
6.1.2　システムボードを削除する操作のながれ　159
6.1.3　システムボードを移動する操作のながれ　160
6.1.4　システムボードを交換する操作のながれ　162
6.2　システムボードを追加する操作例　163
6.2.1　システムボードを割り当てる操作例　164
6.2.2　システムボードを組み込む操作例　167
6.2.3　システムボードの組み込みを予約する操作例　170
6.3　システムボードを削除する操作例　173
6.3.1　システムボードの割り当てを解除する操作例　174
6.3.2　システムボードを切り離す操作例　175
6.3.3　システムボードの割り当て解除を予約する操作例　179
6.4　システムボードを移動する操作例　181
6.5　システムボードを交換する操作例　188
6.6　物理パーティション番号を変更する場合の留意点　193
第7章　ゲストドメインを移行する　195
7.1　概要　195
7.1.1　ライブマイグレーションの要件　197
7.2　ゲストドメインを移行する　198
7.3　ゲストドメインの移行例　198
付録 A　物理パーティションの動的再構成を使用した環境構築・手順例　201
A.1　物理パーティションの動的再構成が未サポートのシステムからサポートされるシステムへアップデートする場合　202
A.1.1　構築例　202
A.1.2　構築手順　203

A.2　XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがない論理ドメインの構成の場合　223
A.2.1　構成例　223
A.2.2　物理パーティションの構築手順例　225
A.2.3　活性交換手順例（Oracle VM Server for SPARC 3.1.xの場合）　249
A.2.4　活性交換手順例（PCIeバスを動的に割り当て可能な場合）　273
A.3　XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがある論理ドメインの構成の場合　296
A.3.1　構成例　296
A.3.2　物理パーティションの構築手順例　298
A.3.3　活性交換手順例（Oracle VM Server for SPARC 3.1.xの場合）　325
A.3.4　活性交換手順例（PCIeバスを動的に割り当て可能な場合）　340
A.4　XCP 2220以降の版数で新規に導入したシステムを1BBから2BB構成に増設する場合　355
A.4.1　構成例　356
A.4.2　増設手順（Oracle VM Server for SPARC 3.1.xの場合）　358
A.4.3　増設手順（PCIeバスを動的に割り当て可能な場合）　375
A.5　SPARC64 Xプロセッサだけで構成された物理パーティションにSPARC64 X+プロセッサで構成されたシステムボードを増設する場合　392
A.5.1　構成例　393
A.5.2　構築手順　395
A.6　制御ドメインのみの構成でシステムボードを活性交換する場合（Oracle VM Server for SPARC 3.2以降）　430
A.6.1　構成例　430
A.6.2　物理パーティションの構築手順例　431
A.6.3　活性交換手順例　440
付録 B　動的再構成使用時の補足情報　447
B.1　XSCF再起動またはフェイルオーバー時の留意点　447

追加の完了確認　447
削除の完了確認　448
B.2　CPU動作モードに関する補足情報　449
B.3　その他の留意事項　449
付録 C　メッセージの説明と対処　453
C.1　コマンドメッセージ　453
C.1.1　addboard　453
C.1.2　deleteboard　457
索引　463

はじめに

本書は、オラクルまたは富士通のSPARC M12/M10 システムが提供するドメイン構築の機能について説明しています。本書は、コンピュータネットワークおよびOracle Solarisに関する高度な知識を有するシステム管理者を対象としています。

なお、SPARC M12は、Fujitsu SPARC M12という製品名でも販売されています。
SPARC M12とFujitsu SPARC M12は同一製品です。
SPARC M10は、Fujitsu M10という製品名でも販売されています。
SPARC M10とFujitsu M10は同一製品です。

対象読者

本書は、コンピュータネットワークおよびOracle Solarisの高度な知識を有するシステム管理者を対象にして書かれています。

関連マニュアル

お使いのサーバに関連するすべてのマニュアルはオンラインで提供されています。

- Oracle Solarisなどのオラクル社製ソフトウェア関連マニュアル
  https://docs.oracle.com/en/
- 富士通マニュアル
  グローバルサイト

  https://www.fujitsu.com/global/products/computing/servers/unix/sparc/downloads/manuals/

  日本語サイト

  https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/downloads/manual/

SPARC M12をお使いの場合は、「SPARC M12 関連マニュアル」に記載されているマニュアルを参照してください。

SPARC M10をお使いの場合は、「SPARC M10 関連マニュアル」に記載されているマニュアルを参照してください。

SPARC M12 関連マニュアル

マニュアルタイトル（*1）
SPARC M12 プロダクトノート
SPARC M12 早わかりガイド
Fujitsu SPARC M12 Getting Started Guide/SPARC M12 はじめにお読みください（*2）
Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Important Legal and Safety Information（*2）
Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Safety and Compliance Guide SPARC M12/M10 安全に使用していただくために
Software License Conditions for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 SPARC M12/M10 ソフトウェアライセンス使用許諾条件
Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Security Guide
SPARC Servers/SPARC Enterprise/PRIMEQUEST 共通設置計画マニュアル
SPARC M12-1 インストレーションガイド
SPARC M12-2 インストレーションガイド
SPARC M12-2S インストレーションガイド
SPARC M12 PCIカード搭載ガイド
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 RCILユーザーズガイド（*3）
SPARC M12/M10 XSCFリファレンスマニュアル
SPARC M12/M10 XSCF MIB・Trap一覧
SPARC M12-1 サービスマニュアル
SPARC M12-2/M12-2S サービスマニュアル
SPARC M12/M10 クロスバーボックス サービスマニュアル
SPARC M12/M10 PCIボックス サービスマニュアル
SPARC M12/M10 用語集
外付けUSB-DVD ドライブ使用手順書

*1: 掲載されるマニュアルは、予告なく変更される場合があります。
*2: 印刷されたマニュアルが製品に同梱されます。
*3: 特にSPARC M12/M10とFUJITSU ETERNUSディスクストレージシステムを対象にしています。

SPARC M10 関連マニュアル

マニュアルタイトル（*1）
SPARC M10 システム プロダクトノート
SPARC M10 システム 早わかりガイド
Fujitsu M10/SPARC M10 Systems Getting Started Guide/SPARC M10 システム はじめにお読みください（*2）
Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Important Legal and Safety Information（*2）
Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Safety and Compliance Guide SPARC M12/M10 安全に使用していただくために
Software License Conditions for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 SPARC M12/M10 ソフトウェアライセンス使用許諾条件
Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Security Guide
SPARC Servers/SPARC Enterprise/PRIMEQUEST共通設置計画マニュアル
SPARC M10-1 インストレーションガイド
SPARC M10-4 インストレーションガイド
SPARC M10-4S インストレーションガイド
SPARC M10 システム PCIカード搭載ガイド
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 RCILユーザーズガイド（*3）
SPARC M12/M10 XSCFリファレンスマニュアル
SPARC M12/M10 XSCF MIB・Trap一覧
SPARC M10-1 サービスマニュアル
SPARC M10-4/M10-4S サービスマニュアル
SPARC M12/M10 クロスバーボックス サービスマニュアル
SPARC M12/M10 PCIボックス サービスマニュアル
SPARC M12/M10 用語集
外付けUSB-DVD ドライブ使用手順書

*1: 掲載されるマニュアルは、予告なく変更される場合があります。
*2: 印刷されたマニュアルが製品に同梱されます。
*3: 特にSPARC M12/M10とFUJITSU ETERNUSディスクストレージシステムを対象にしています。

安全上の注意事項

SPARC M12/M10をご使用または取り扱う前に、次のドキュメントを熟読してください。

- Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Important Legal and Safety

Information

- Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Safety and Compliance Guide
  SPARC M12/M10 安全に使用していただくために

表記上の規則

本書では、以下のような字体や記号を、特別な意味を持つものとして使用しています。

字体または記号	意味	記述例
AaBbCc123	ユーザーが入力し、画面上に表示される内容を示します。 この字体は、コマンドの入力例を示す場合に使用されます。	XSCF> **adduser jsmith**
AaBbCc123	コンピュータが出力し、画面上に表示されるコマンドやファイル、ディレクトリの名称を示します。 この字体は、枠内でコマンドの出力例を示す場合に使用されます。	XSCF> **showuser -P** User Name: jsmith Privileges: useradm auditadm
『』	参照するマニュアルのタイトルを示します。	『SPARC M10-1 インストレーションガイド』を参照してください。
「」	参照する章、節、項、ボタンやメニュー名を示します。	「第2章 ネットワーク接続」を参照してください。

本文中のコマンド表記について

XSCFコマンドには(8)または(1)のセクション番号が付きますが、本文中では(8)や(1)を省略しています。
コマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

CLI（コマンドライン・インターフェース）の表記について

コマンドの記載形式は以下のとおりです。

- 値を入力する変数は斜体で記載
- 省略可能な要素は[ ]で囲んで記載
- 省略可能なキーワードの選択肢は、まとめて[ ]で囲み、|で区切り記載

マニュアルへのフィードバック

本書に関するご意見、ご要望がございましたら、マニュアル番号、マニュアル名称、ページおよび具体的な内容を、次のURLからお知らせください。

- グローバルサイト
  https://www.fujitsu.com/global/contact/
- 日本語サイト
  https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/contact/

第1章

ドメイン構築の概要を理解する

ここでは、ドメイン構築の概要を説明します。

- SPARC M12/M10のドメイン構成
- SPARC M12/M10のハードウェアリソース
- 物理パーティションとは
- 論理ドメインとは
- 物理パーティションの動的再構成とは
- SPARC64 X+プロセッサとSPARC64 Xプロセッサの混在

1.1 SPARC M12/M10のドメイン構成

ここでは、SPARC M12/M10で構築できるドメイン構成の概要、および特長を説明します。

1.1.1 1台構成／ビルディングブロック構成の概要

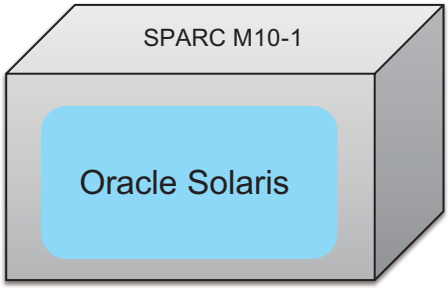
ここでは、SPARC M12/M10が1台で構成されるシステム、および複数のSPARC M12またはSPARC M10で構成されるシステム（ビルディングブロック構成のシステム）の概要を説明します。

- 1台構成のシステム
- ビルディングブロック構成のシステム（SPARC M12-2S/M10-4S）

1台構成のシステム

図 1-1は、SPARC M10-1のシステム構成例です。この例では、1台で1つのOracle Solarisを稼働させています。

図 1-1　　1台構成のシステム（SPARC M10-1の例）



ビルディングブロック構成のシステム

SPARC M12-2SまたはSPARC M10-4Sでは、ビルディングブロック(BB)構成で、複数のSPARC M12-2Sを接続、または複数のSPARC M10-4Sを接続できます。1つの独立したシステムとして構築したり、BB単位でシステムを複数の物理パーティションに分割したりできます。
ビルディングブロック構成では、複数のBBで、1つの物理パーティションとすることができます。

注－ビルディングブロック方式の場合、SPARC M12-2SとSPARC M10-4Sを混在して構成することはできません。

図 1-2は、複数のSPARC M12-2Sのシステム構成例です。この例では、複数のSPARC M12-2Sで、1つのOracle Solarisを稼働させています。

図 1-2　　ビルディングブロック構成のシステム（物理パーティション分割なし）

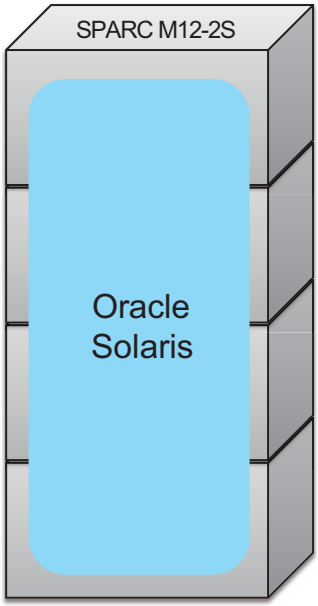
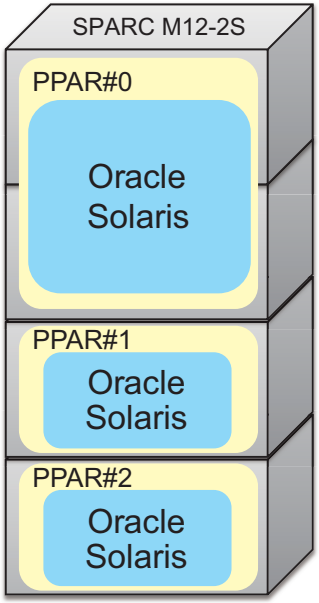


図 1-3は、4台のSPARC M12-2Sを接続し、3つの物理パーティションに分割した例です。この例では、複数の物理パーティションに分割して、それぞれに1つのOracle Solarisを稼働させています。

図 1-3　ビルディングブロック構成のシステム（物理パーティション分割有）



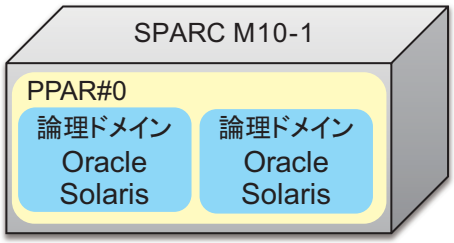
1.1.2　論理ドメイン構成の概要

複数の論理ドメイン構成

1つの物理パーティションに、複数の論理ドメインを構築できます。

1つの物理パーティションに複数の論理ドメインを構築して、それぞれ1つのOracle Solarisを稼働させることができます。図 1-4は、SPARC M10-1のシステム構成例です。この例では、1台で1つの物理パーティション、2つの論理ドメインを構築しています。

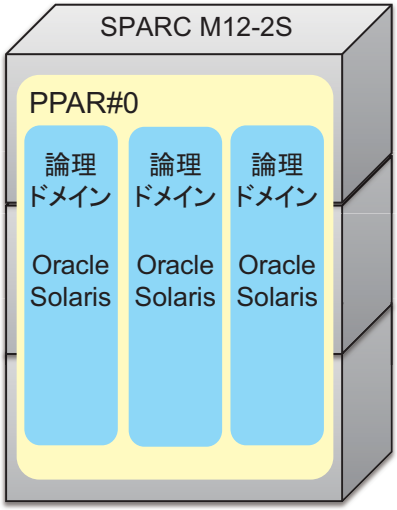
図 1-4　複数の論理ドメイン構成（1台構成の場合）



ビルディングブロック構成のシステムで複数の論理ドメインを構築して、それぞれに

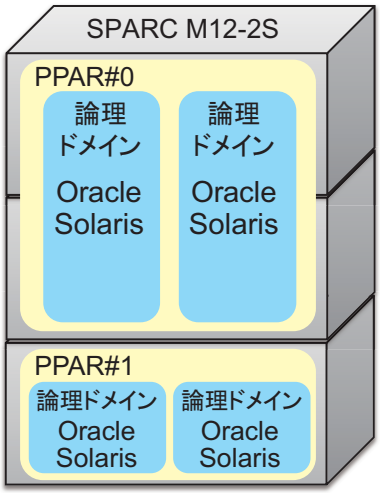
Oracle Solarisを稼働させることができます。図 1-5は、3台のSPARC M12-2Sを接続したビルディングブロック構成のシステムにおいて、1つの物理パーティションに3つの論理ドメインを構築した例です。

図 1-5 複数の論理ドメイン構成 （ビルディングブロック構成の場合-1）



ビルディングブロック構成のシステムを複数の物理パーティションに分割して、それぞれの物理パーティション上に複数の論理ドメインを構築できます。また、それぞれの論理ドメインにOracle Solarisを稼働させることができます。図 1-6は、3台のSPARC M12-2Sを接続したビルディングブロック構成のシステムにおいて、2つの物理パーティションに、それぞれ2つの論理ドメインを構築した例です。

図 1-6 複数の論理ドメイン構成 （ビルディングブロック構成の場合-2）



1.1.3 ドメイン構成の特長

ここでは、SPARC M12/M10システムの特長であるドメイン構成について説明します。

論理ドメインで構成されたシステムの特長

SPARC M12/M10システムでは、Oracle VM Server for SPARCを使用して複数の論理ドメインを構築し、それぞれの論理ドメイン上で独立したOSを動作させることができます。論理ドメインごとに異なる版数のOSを動作させることができます。それぞれの論理ドメインには、CPU、メモリ、I/Oを柔軟に割り当てることができ、ハードウェアリソースを有効に活用できます。

図 1-7は、バージョンの異なるOracle Solaris間において、2つのCPUコアと、一部のメモリリソースを移動した例です。

図 1-7 論理ドメイン間のハードウェアリソースの活用（1台構成 SPARC M10-1の例）

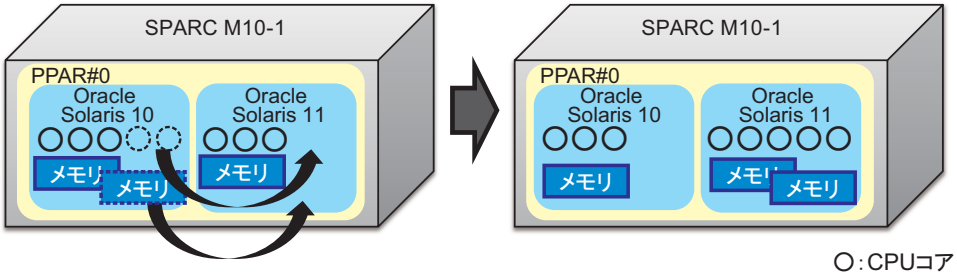
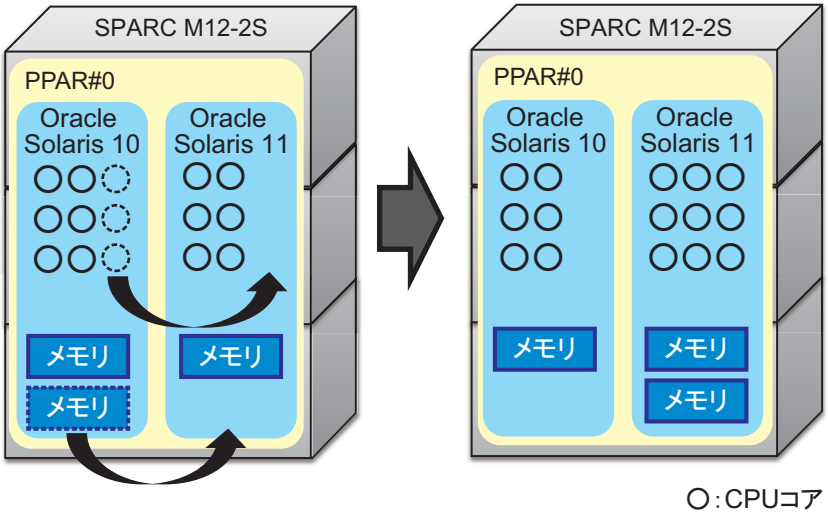


図 1-8は、バージョンの異なるOracle Solaris間において、3つのCPUコアと、1つのメモリリソースを移動した例です。

図 1-8　論理ドメイン間のハードウェアリソースの活用（ビルディングブロック構成）

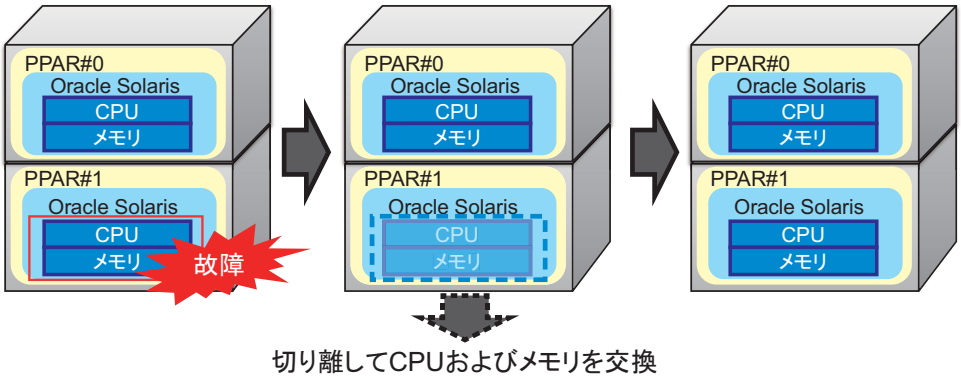


物理パーティションで構成されたシステムの特長

SPARC M12-2S/M10-4Sのシステムでは、1つまたは複数の筐体で物理パーティションを構成し、それぞれの物理パーティション上で独立したOSを動作させることができます。
物理パーティションで構成されたシステムは、Oracle VM Sever for SPARCによる論理ドメイン構成のシステムよりも高いハードウェアの独立性を実現しています。

図 1-9は、システムから筐体を切り離して、CPUおよびメモリを交換する例です。

図 1-9　物理パーティションの特長（ハードウェアの独立性）

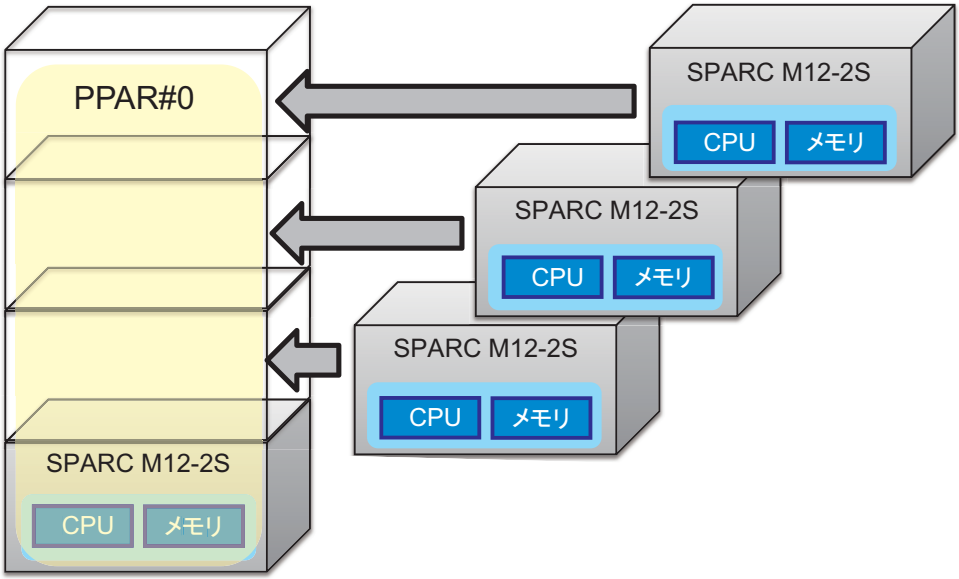


SPARC M12-2S/M10-4Sのシステムでは、筐体を組み上げることでハードウェアリソースを増やすことができます。これにより、小さいシステムから環境に応じて必要な分だけシステムを段階的に拡張できます。

図 1-10は、3台のSPARC M12-2Sを追加して、4台を1つの物理パーティションとして

使用する例です。

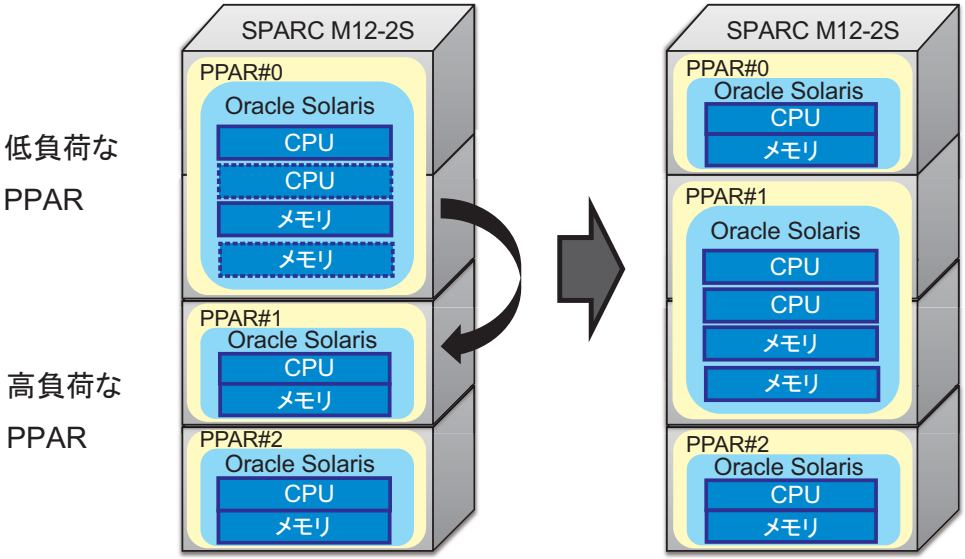
図 1-10　物理パーティションの特長（ハードウェアの拡張性）



また、物理パーティションには、ビルディングブロック(PSB)単位でCPU、メモリ、I/Oを追加／削除することができます。
さらに、複数の物理パーティションで構成されたシステムでは、これらの物理パーティションを一元管理できるため、負荷の低い物理パーティションから高い物理パーティションへ、ハードウェアリソースを移動することができます。

図 1-11は、PPAR#0からPPAR#1に筐体1台分のリソースを移動した例です。

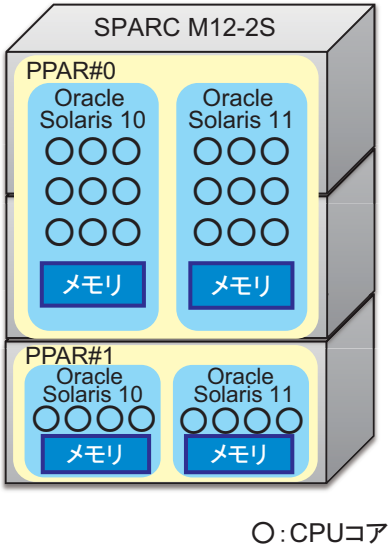
図 1-11　物理パーティションの特長（ハードウェアの柔軟性）



さらに、各物理パーティション上に論理ドメインを構築できます。

図 1-12は、3台のSPARC M12-2Sを接続したシステムにおいて、2つの物理パーティションにそれぞれ2つの論理ドメインを構築した例です。

図 1-12　物理パーティションに論理ドメインを構築した例



1.2 SPARC M12/M10のハードウェアリソース

ここでは、SPARC M12/M10システムのハードウェアリソースを説明します。

SPARC M12/M10システムでは、CPU、メモリ、I/Oなどのハードウェアリソースを柔軟に配分し、リソースを有効活用できます。

図 1-13は、SPARC M12-1/M10-1のハードウェアリソースの活用例です。

図 1-13 ハードウェアリソースの活用（SPARC M12-1/M10-1）

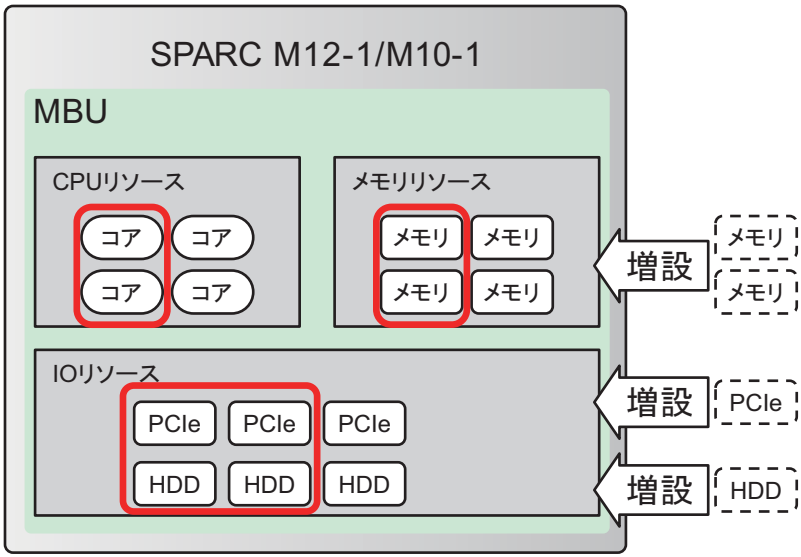
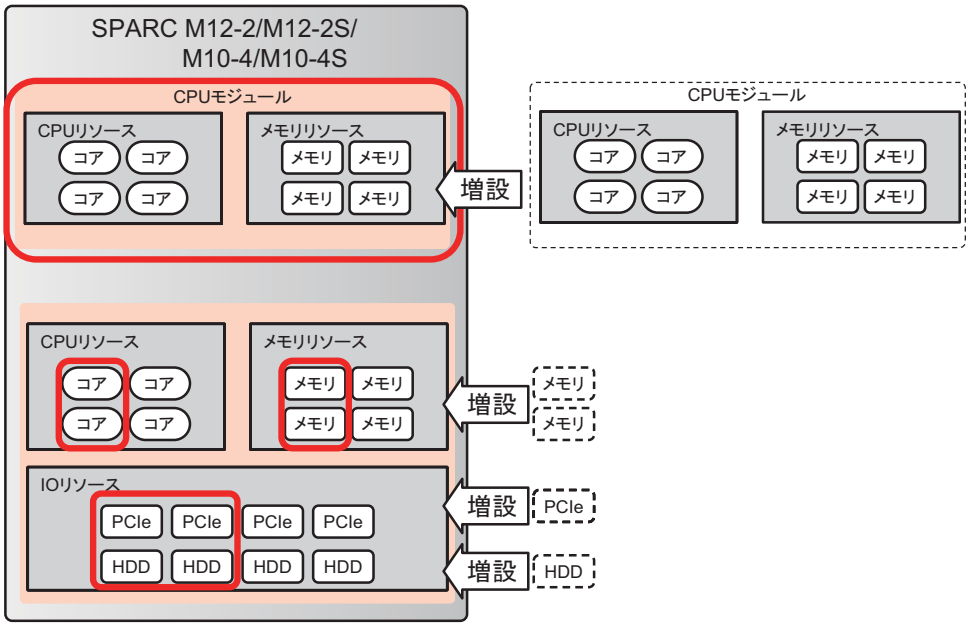


図 1-14は、SPARC M12-2/M12-2S/M10-4/M10-4Sのハードウェアリソースの活用例です。

図 1-14 ハードウェアリソースの活用（SPARC M12-2/M12-2S/M10-4/M10-4S）



- **CPUリソース**

  【SPARC M12-1】

  CPUコア アクティベーションにより最大6 CPUコアのリソースが使用できます。

  【SPARC M12-2/M12-2S】

  CPUコア アクティベーションにより、最大（2 CPU構成時）24 CPUコアのリソースが使用できます。

  【SPARC M10-1】

  CPUコア アクティベーションにより最大16 CPUコアのリソースが使用できます。

  【SPARC M10-4/M10-4S】

  CPUコア アクティベーションにより最大（4 CPU構成時）32 CPUコアのリソースが使用できます。

- **メモリリソース**

  すべてのモデルで、メモリをミラー構成にすることができます。その場合は、使用できるメモリ容量は半分となります。

  【SPARC M12-1】

  最大1 TB搭載できます。

  【SPARC M12-2/M12-2S】

  最大（2CPU構成時）で2 TB搭載できます。

  【SPARC M10-1】

最大1 TB搭載できます。

【SPARC M10-4/M10-4S】

最大（4CPU構成時）で4 TB搭載できます。

- **I/Oリソース**

【SPARC M12-1】

PCIeカードを最大3枚まで搭載できます。PCIボックスを接続することで、PCIeカードを最大33枚まで搭載できます。内蔵ストレージは最大8台まで搭載できます。

【SPARC M12-2/M10-4】

PCIeカードを最大11枚まで搭載できます。PCIボックスを接続することで、PCIeカードを最大71枚まで搭載できます。内蔵ストレージは最大8台まで搭載できます。

PHPを使ってPCIeカードを動的に増やすことができます。

【SPARC M12-2S/M10-4S】

PCIeカードを最大8枚まで搭載できます。さらにPCIボックスを接続することで、PCIeカードを最大58枚まで搭載できます。内蔵ストレージは最大8台まで搭載できます。

PHPを使ってPCIeカードを動的に増やすことができます。

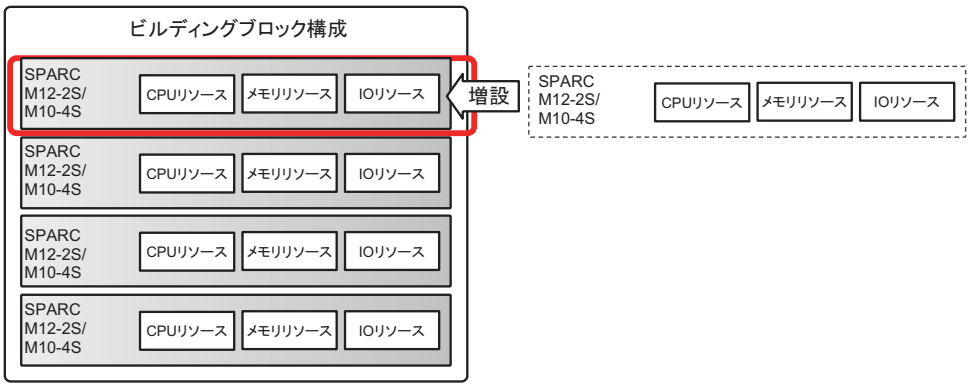
【SPARC M10-1】

PCIeカードを最大3枚まで搭載できます。PCIボックスを接続することで、PCIeカードを最大23枚まで搭載できます。内蔵ストレージは最大8台まで搭載できます。

- **ビルディングブロック構成（SPARC M12-2S/M10-4S）**

複数のSPARC M12-2S/M10-4Sからなるビルディングブロック構成のシステムでは、1つの筐体を1ブロックとし、ブロックを積み上げるようなイメージで、1台から最大16台まで段階的にシステムを拡張できます。

図 1-15　ビルディングブロック構成のハードウェアリソースの活用



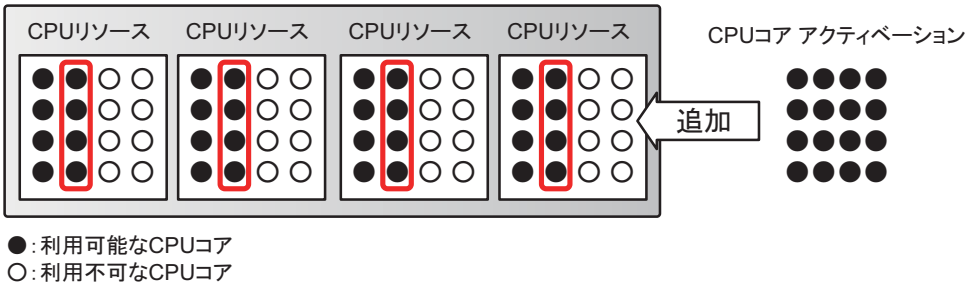
- **CPUコア アクティベーション**

CPUコア アクティベーション機能により、CPUチップよりも粒度の小さいCPUコア単位でCPUリソースを追加できます。

必要最小限のCPUリソースで利用開始して、CPUコア アクティベーションによって、段階的にCPUリソースを増やすことができます。

CPUコア アクティベーションの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「第5章　CPUコア アクティベーション」を参照してください。

図 1-16　CPUコア アクティベーション（SPARC M10-4/M10-4Sの例）



1.3　物理パーティションとは

ここでは、物理パーティションの構成要素と、物理パーティションに関する操作を説明します。

1.3.1　物理パーティションの構成要素を理解する

物理パーティションは、複数のビルディングブロックで構成されます。ビルディングブロック構成でない場合は、1台のサーバが物理パーティションとなります。XSCFファームウェアでは、物理パーティションを構築する際、1つのビルディングブロックを1つの物理システムボード（PSB）として扱います。その物理システムボードは、CPU、メモリ、I/Oデバイスの3種類のリソースから構成されます。また、物理パーティションでは、物理システムボードのハードウェアリソースを論理システムボード（LSB）という論理的なシステムボードに割り当てます。

- 物理システムボード（PSB）
  SPARC M12/M10の1筐体上の物理的なコンポーネント（CPU、メモリ、I/O）から構成されます。SPARC M12-1/M10-1では、マザーボードユニット、SPARC M12-2/M12-2S/M10-4/M10-4Sでは、CPUメモリユニット（下段（CMUL）と上段（CMUU）を含む）が物理システムボードです。物理システムボードは、SPARC M12/M10筐体の増設／減設／交換のための保守の際、筐体を指すユニットとして使用される場合があります。また、ビルディングブロック構成のシステムの場合、物理システムボードは1つのビルディングブロック(BB)を指します。
- 論理システムボード（LSB）
  物理システムボードに割り当てる論理ユニット名です。各物理パーティションは、

自身に割り当てられた論理システムボードのセットを持ちます。物理パーティションのある論理システムボードは、1つのPSB番号を割り当てられることで、システムに認識されます。

LSB番号は、メモリなどのリソースをどのように各物理パーティションに割り当てるかを制御するために使用されます。

- 物理パーティション番号（PPAR-ID）
  物理パーティションを識別する番号です。0〜15の数字を割り当てることが可能です。ただし、物理パーティション番号には以下の制約があります。

  - SPARC M12-1/M12-2/M10-1/M10-4の場合

    物理パーティション番号は0で固定です。変更することはできません。

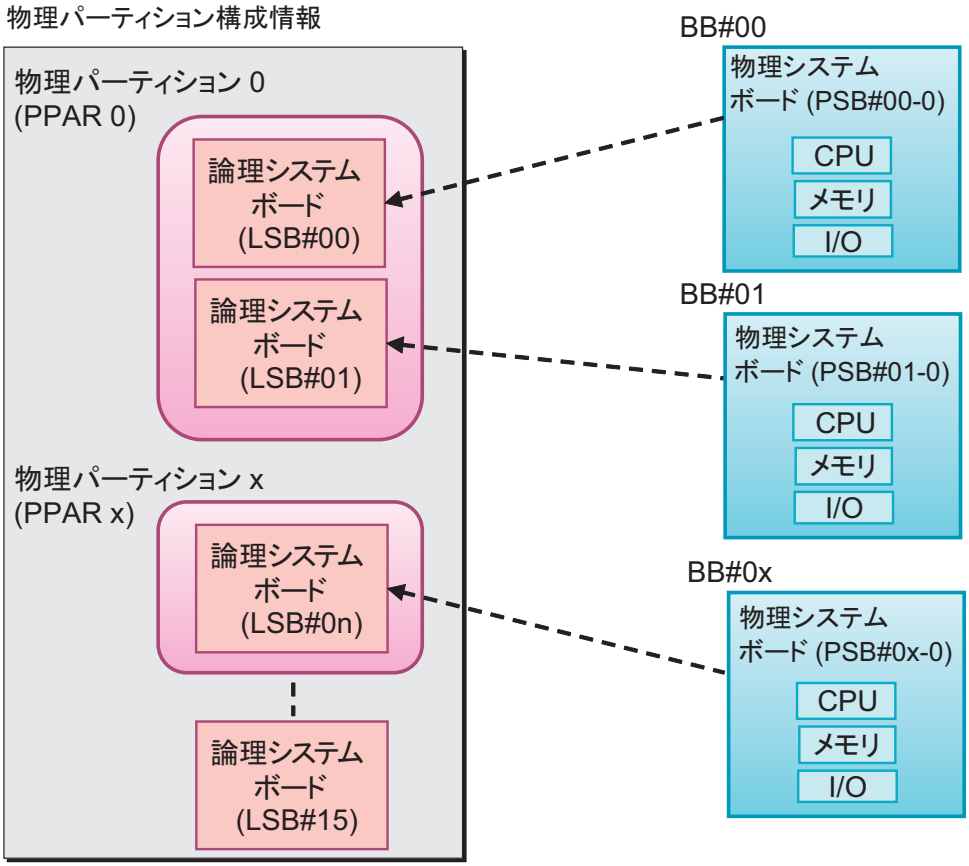
  - SPARC M12-2S/M10-4Sの場合

    システムに搭載されている筐体のBB-IDと同じ番号の物理パーティション番号を割り当てることができます。搭載されていないBB-IDを物理パーティション番号に指定した場合、その物理パーティションは電源投入に失敗します。
    SPARC M12-2S/M10-4Sの減設を考慮した推奨される物理パーティションの構築については、「4.2　物理パーティションの構築と操作」を参照してください。

    [例] BB#0, BB#1が搭載されたシステムの場合、割り当て可能な物理パーティション番号は0または1です。物理パーティション番号に2を指定した場合、その物理パーティションは電源投入に失敗します。

物理パーティション上の論理システムボードと物理システムボードの対応付けは、物理パーティション構成情報（PPAR構成情報）に設定します。図 1-17は、その対応付けのイメージを示しています。

図 1-17　論理システムボードと物理システムボードの対応付けのイメージ



動的再構成の操作対象であるすべてのシステムボードは、該当物理パーティションの物理パーティション構成情報（PPAR構成情報）に登録されていなければなりません。PPAR構成情報への登録には、setpclコマンドを使用します。

PPAR構成情報は、物理パーティションを構成する論理システムボード（LSB） とハードウェアリソース情報である物理システムボード（PSB）を対応付け、XSCFにより管理されています。各PPAR構成情報には、物理パーティションおよびLSBへ対応させたPSB のハードウェアリソースに関して設定情報も格納されます。

1.3.2　物理パーティション構成を操作する

ここでは、複数台のSPARC M12-2S、または複数台のM10-4Sにおける物理パーティション構成の操作を説明します。

物理パーティションを構成するための設定、および物理パーティションの操作は、XSCFで提供されるXSCFシェルとXSCF Webを使用します。これらの操作は、XSCFにより一括して管理されます。さらにXSCFのセキュリティ管理によって、特定のアクセス権限がある管理者だけが物理パーティションの構成を操作できます。物理パー

ティションの構成を操作するためのXSCFシェルコマンドについては、「3.1　物理パーティション構築に関する操作とコマンド」を参照してください。
XSCF Webによる操作は、『SPARC M12/M10 システム運用・管理ガイド』の「第3章　システムを設定する」を参照してください。

システムボードの登録／登録解除

1つのビルディングブロック（BB）を示す物理システムボード（PSB）に、論理システムボード番号（LSB番号）を付与して、物理パーティション構成情報に設定することを「システムボードの登録」といいます。
また、PSB（BB）を物理パーティション構成情報から削除することを「システムボードの登録解除」といいます。物理パーティション構成情報に物理システムボード（PSB）を登録することで、物理パーティションへのPSB（BB）の割り当てや組み込みなどPSB（BB）の操作が可能になります。

システムボードの追加

増設したビルディングブロック(PSB)、またはどの物理パーティションにも属さず使用していないPSB（BB）を物理パーティションに組み込むことを「システムボードの追加」といいます。PSB（BB）の追加は、接続（connect）、組み込み（configure）というように段階的に処理が行われます。PSB（BB）の追加を行うと、指示されたPSB（BB）は、まず該当する物理パーティションに接続します。その後PSB（BB）の組み込み処理が行われることで、PSB（BB）の追加が完了します。

システムボードの削除

構成する物理パーティションから不要となったビルディングブロック(PSB)を切り離すことを「システムボードの削除」といいます。PSB（BB）の削除は、切り離し（unconfigure）、切断（disconnect）というように段階的に処理が行われます。

PSB（BB）を別の物理パーティションに割り当てる場合は、削除の操作に割り当て解除（unassign）の処理を含める必要があります。PSB（BB）の削除を行うと、まず指定されたPSB（BB）の切り離し処理が行われます。その後、属していた物理パーティションから切断されることで、PSB（BB）の削除が完了します。

システムボードの割り当て／割り当て解除

増設したビルディングブロック(PSB)、またはどの物理パーティションにも属さず使用していないPSB（BB）を物理パーティションに属すようにすることを「システムボードの割り当て」といい、物理パーティションに属している状態を解除することを「システムボードの割り当て解除」といいます。

PSB（BB）を該当の物理パーティションへ割り当てる（assign）ことで、別の物理パーティションからは、そのPSB（BB）への操作ができなくなります。PSB（BB）を物理パーティションへ割り当て、物理パーティションを電源投入すると、PSB（BB）の追加が行われます。物理パーティションを電源切断すると、PSB（BB）の削除が行われ、PSB（BB）は物理パーティションに割り当てられた状態（assign）になります。

物理パーティションに割り当てられた状態（assign）から割り当て解除（unassign）すると、どの物理パーティションにも属さず、別の物理パーティションに割り当てることができます。

システムボードの交換

保守のためにビルディングブロック(PSB)を交換することを「システムボードの交換」といいます。PSB（BB）の交換は、CPU、メモリ、I/Oデバイスのハードウェアリソースを交換する場合に利用することができます。PSB（BB）の交換は段階的に処理が行われます。PSB（BB）の交換を行うことで、指示されたPSB（BB）に対して、PSB（BB）の削除が行われます。PSB（BB）の削除が完了することで、PSB（BB）を減設できる状態となり、該当するPSB（BB）を減設します。そして、部品交換などの作業を経たあと、交換する新たなPSB（BB）を再度増設します。その後、増設が完了し、PSB（BB）の追加を行うことで交換が完了します。

システムボードプール機能

システムボードプール機能とは、特定のシステムボード（PSB<BB>）を、どの物理パーティションにも属さない状態に配置する機能です。複数の物理パーティション間で必要に応じてPSB（BB）を移動するのに有効な機能です。例えば、CPUやメモリの負荷が高い物理パーティションに対して、システムボードプールからPSB（BB）を獲得して、その物理パーティションへ追加することができます。また、不要になった時点で、システムボードプールへ返却することもできます。

プール状態のPSB（BB）は、物理パーティション構成情報（PPAR構成情報）に登録されている場合にのみ、物理パーティションに割り当てることが可能です。つまり、複数のPPAR構成情報に同一のPSB（BB）を登録しておくことで、そのPSB（BB）を物理パーティションの稼働状況に応じて組み込んだり切り離したりして、柔軟にシステムを運用することができます。ただし、このような運用では、プール状態のPSB（BB）の使用状況は、常にきちんと管理される必要があります。

また、「2.2　XSCFの条件と設定」で説明しているメモリ無効化オプションおよびI/O無効化オプションと組み合わせることで、PSB（BB）の追加および削除を容易に行うことができます。

物理パーティションの構成変更の予約

物理パーティションでは、システムボード（PSB<BB>）の追加、または削除を動的に行えるだけでなく、該当物理パーティションの電源が次に投入／切断されるとき、あるいは再起動されるときに再構成が行われるように予約することができます。

物理パーティションの構成変更の予約は、次のような場合に使用することができます。

- 業務や運用の都合から、動的再構成によってハードウェアリソースを再構成できない場合
- 物理パーティションの構成をただちに変更しない場合
- 動的再構成に対応していないドライバやPCIカードが搭載されているPSB（BB）を削除する場合など、動的再構成による設定変更操作を回避し、物理パーティションの再起動によりただちに構成変更を行いたい場合

表 1-1は、物理パーティションを構成するシステムボード（PSB<BB>）に対して可能な操作を示しています。

表 1-1　システムボードに対して可能な操作

用語	説明
登録（Register）	物理パーティション構成情報にシステムボードを登録すること
登録解除（Release）	物理パーティション構成情報からシステムボードの登録を解除すること
追加（Add）	システムボードを物理パーティションに追加すること
削除（Delete）	システムボードを物理パーティションから削除すること
割り当て（Assign）	物理パーティションにシステムボードを割り当てること
割り当て解除（Unassign）	物理パーティションからシステムボードの割り当てを解除すること
接続（Connect）	システムボードを物理パーティションに接続すること
切断（Disconnect）	システムボードを物理パーティションから切断すること
組み込み（Configure）	システムボードを物理パーティション構成に組み込むこと
切り離し（Unconfigure）	システムボードを物理パーティション構成から切り離すこと
予約（Reserve）	物理パーティションを電源投入または再起動したときに、物理パーティションにシステムボードを組み込むために予約すること。さらに、物理パーティションへのシステムボードの割り当て解除を予約すること
増設（Install）	SPARC M12/M10の筐体をシステムに挿入すること
減設（Remove）	SPARC M12/M10の筐体をシステムから抜き取ること
交換（Replace）	保守点検などでSPARC M12/M10の筐体を減設し、新たに、またはあらためて増設すること

1.4　論理ドメインとは

ここでは、論理ドメインの構成要素と、論理ドメインに関する操作を説明します。

1.4.1　論理ドメインの構成要素を理解する

論理ドメインは、仮想CPU、仮想メモリ、仮想I/Oで構成されます。

- 仮想CPU
  論理ドメインには、CPUを仮想CPU（スレッド）単位で割り当てることができます。SPARC M12/M10システムでは、1つの物理的なCPU（＝1ソケット）が複数のコアを持ち、それぞれのコアがスレッドを持ちます。そのため、1つの物理CPUにはスレッド数分の仮想CPUが存在します。これらの仮想CPUを論理ドメインに割り当てることができます。通常、論理ドメインのパフォーマンスを考慮して、コア単位で仮想CPUを割り当てるようにします。
- 仮想メモリ
  論理ドメインには、メモリを256 MB単位で割り当てることができます。

- 仮想I/O
  論理ドメインには、I/Oを仮想I/Oの単位で割り当てることができます。例えば、仮想I/Oの1つである仮想ディスクには、次の実体が使用できます。

  - 物理ディスク

  - 物理ディスクスライス

  - ZFS、UFSなどのファイルシステムのファイル

  - ZFSなどのボリュームマネージャーからの論理ボリューム

また、論理ドメインは、その役割によって次のものに分かれます。

- 制御ドメイン
  制御ドメインは、ほかの論理ドメインを作成、管理したり、リソースを割り当てたりする論理ドメインです。制御ドメインは、物理パーティションに1つだけ存在します。制御ドメインには、Oracle VM Server for SPARCがインストールされ、管理ソフトウェアであるLogical Domains Managerが稼働しています。

- I/Oドメイン
  I/Oドメインは、仮想デバイスサービスを提供する論理ドメインです。仮想デバイスサービスには、ディスク、ネットワーク、およびコンソールがあります。

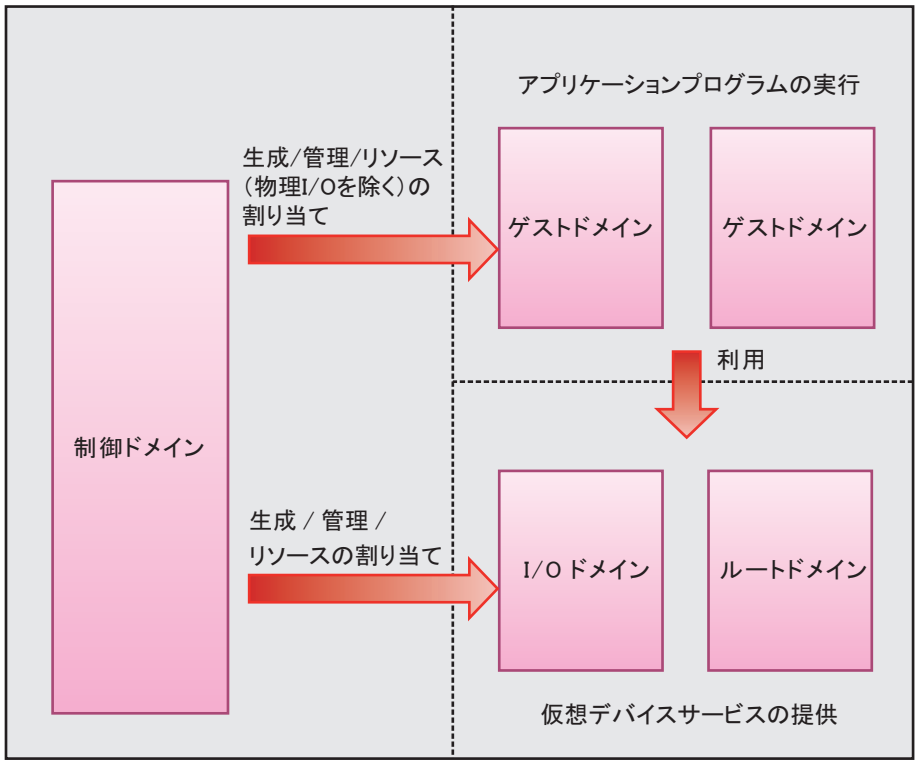
- ルートドメイン
  ルートドメインは、PCIeルートコンプレックスが割り当てられたI/Oドメインです。PCIeルートコンプレックスとはPCIeバス全体を指し、PCIeバス、すべてのPCIスイッチ、およびデバイスで構成されます。ルートドメインが物理的なI/Oデバイスを所有し、それらに直接アクセスします。

- ゲストドメイン
  ゲストドメインは、制御ドメインによって管理され、I/Oドメインの仮想デバイスサービスを利用する論理ドメインです。通常、ゲストドメインでは、ミドルウェアやアプリケーションプログラムが動作します。ゲストドメインでは、独立したOracle Solarisが動作し、ほかのゲストドメインに影響を与えることなく、起動や停止が可能です。また、ゲストドメインには、仮想CPU、仮想メモリ、仮想I/Oを動的に追加したり削除したりできます。

- サービスドメイン
  サービスドメインは、ゲストドメインにサービスを提供するドメインの総称です。具体的には、I/Oドメインやルートドメインなどがあります。

図 1-18　論理ドメイン同士の関係の概念

物理パーティション



1.4.2　論理ドメインの構成を操作する

論理ドメインに対する操作は、Oracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerを使用して行います。Logical Domains Managerを使用することで、論理ドメインを構築したり再構成したりできます。これらの作業は、制御ドメインにログインして、各種オプションとともにldmコマンドを使用して行います。

ここでは、論理ドメインに対して可能な操作を説明します。

論理ドメインの構築

論理ドメインを構築する場合、Oracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用して、次の操作を行います。

- 論理ドメインを作成する
- 論理ドメインに対してハードウェアリソースを仮想CPU、仮想メモリ、仮想I/Oとして割り当てる
- 論理ドメインの構成情報を保存する
- 論理ドメインを起動／停止する

また、制御ドメイン以外の論理ドメインでは、仮想CPU、仮想メモリ、仮想I/Oのハードウェアリソースの追加また削除をオペレーティングシステムやアプリケーションの動作中に行うこともできます。この再構成を論理ドメインの動的再構成といいます。ただし、論理ドメインの動的再構成が可能かどうかは、論理ドメインで動作しているオペレーティングシステムやアプリケーションに依存します。

再構成できるハードウェアリソース

論理ドメイン内で再構成できるハードウェアリソースは次のとおりです。

- 仮想CPU
  CPUは仮想CPU（スレッド）単位で再構成できます。ただし、通常は、パフォーマンスを考慮して、コア単位で再構成するようにします。
- 仮想メモリ
  メモリは256 MB単位で再構成できます。
- 仮想I/O
  仮想ディスク、仮想ネットワークの単位で再構成できます。ゲストドメインに割り当てる仮想I/Oを再構成するときには、ゲストドメインを再起動する必要はありません。

  仮想ディスク、仮想ネットワークを削除する前に、次の作業が必要です。

  - 仮想ディスクの削除
    仮想ディスクをアンマウントしてから削除します。
  - 仮想ネットワークの削除
    仮想ネットワークのインターフェースをunplumbしてから削除します。
- 物理I/O
  物理I/Oは、以下の単位で論理ドメインへの割り当てが可能です。

表 1-2 物理I/Oが再構成可能なOracle VM Server for SPARCのバージョン

物理I/Oの種類	静的再構成	動的再構成
PCIe SR-IOV（シングルルートI/O仮想機能）の仮想機能（Virtual Function）	3.0以降	3.1以降
PCIeエンドポイント	3.0以降	3.1.1.1以降
PCIeルートコンプレックス	3.1以降	3.2以降(*1)

*1: PCIeルートコンプレックスを追加または削除できる論理ドメインは、Oracle Solaris 11.2 SRU11.2.8.4.0以降です。

SR-IOV、PCIe エンドポイントによる各デバイス（オンボードデバイスを含む）の割り当て条件、ルートコンプレックス（PCIeバス）の割り当て条件については、以下のマニュアルを参照してください。

- お使いのサーバの最新の『プロダクトノート』
- お使いのサーバの『PCIカード搭載ガイド』
- お使いのバージョンの『Oracle VM Server for SPARC 管理ガイド』
- お使いのバージョンの『Oracle VM Server for SPARC リリースノート』

制御ドメインおよびルートドメインに割り当てている物理I/Oを静的に再構成するときには、遅延再構成モードに変更し、制御ドメインおよびルートドメインをそ

れぞれ再起動して再構成後の設定内容を適用します。

物理I/Oを削除する前に、次の作業が必要です。

- 物理I/Oの削除
  ディスクであればアンマウント、ネットワークであればunplumbした状態にして、物理I/Oのデバイスを未使用の状態にしてから削除します。
  また、ゲストドメインのハードウェアリソースを動的に移動するときには、対象のゲストドメインでLogical Domains DRデーモン（drd）が動作している必要があります。

1.4.3 物理I/Oデバイスの再構成

物理I/Oデバイスの追加または削除

各物理I/Oデバイスの動的再構成の可否は、Oracle VM Server for SPARCのバージョンによって異なります。詳細は、表 1-2を参照してください。

- 制御ドメインに対して物理I/Oデバイスを追加または削除する場合は、遅延再構成モードに移行したあとに物理I/Oデバイスを追加または削除する必要があります。
- ルートドメインに物理I/Oデバイスを追加または削除する場合は、遅延再構成モードに移行したあとに物理I/Oデバイスを追加または削除するか、ルートドメインをシャットダウン（ldm stop-domain）したあとに物理I/Oデバイスを追加または削除する必要があります。
- I/Oドメインに物理I/Oデバイスを追加または削除する場合は、I/Oドメインをシャットダウン（ldm stop-domain）したあとに物理I/Oデバイスを追加または削除する必要があります。

デバイスドライバ

動的再構成を行う物理パーティション上で物理I/Oデバイスを制御するドライバは、Oracle Solarisのsuspendおよびresume機能をサポートしている必要があります。

SR-IOV対応

SPARC M12/M10は、Oracle VM Server for SPARCとの組み合わせにより、PCIe SR-IOV機能（シングルルートI/O仮想化機能）がサポートされています。
SR-IOV機能は、SR-IOV対応のPCI Expressカードを使うことで、1つの物理機能（Physical Function）に対して、複数の仮想機能（Virtual Function）が作成できます。作成された仮想機能をI/Oドメインに割り当てることで、今まで以上に効率よくPCIeカードの帯域を論理ドメインで共有しつつ、サービスドメインのオーバーヘッドをなくすことができます。SR-IOV機能の詳細は、『Oracle VM Server for SPARC 管理ガイド』、およびお使いのサーバの『PCIカード搭載ガイド』を参照してください。また、利用可能なOracle Solarisは、『Oracle VM Server for SPARC リリースノート』、および『Oracle VM Server for SPARC 管理ガイド』を参照してください。

1.5 物理パーティションの動的再構成とは

ここでは、物理パーティションの動的再構成（PPAR DR）の概要を説明します。

1.5.1 物理パーティションの動的再構成の概要

SPARC M12-2S、またはSPARC M10-4Sをビルディングブロック方式で接続することで物理パーティションを構成できます。物理パーティションの動的再構成（PPAR DR）機能は、物理パーティション上の業務を停止することなく、ハードウェアリソース（CPU、メモリ、I/O）を物理パーティションへ追加／削除を行うための技術です。追加／削除を行う単位は、1つのSPARC M12-2S、またはSPARC M10-4S（ビルディングブロック<BB>）です。ファームウェアは、この単位をシステムボード（PSB）として扱います。PSB（BB）に搭載されるハードウェアリソースは、論理的に、かつ、動的に再構成することができます。

注－物理パーティションの動的再構成機能が使用できるハードウェアリソース（CPU、メモリ、I/O）のサポート情報の詳細は、お使いのサーバの最新の『プロダクトノート』を参照してください。

1.5.2 物理パーティションの動的再構成の用途

物理パーティションの動的再構成機能（PPAR DR機能）は、次の用途に使うことができます。

- ハードウェアリソースの増強
  業務の拡張やシステム負荷の増大に対処するため、物理パーティション上の業務を停止させずに、物理パーティションへシステムボード（PSB<BB>）を追加できます。必要に応じて、CPUコア アクティベーションの追加手配、設定が必要です。

- ハードウェアリソースの縮小
  業務量の変化によりハードウェアリソースを有効活用するため、運用中の業務を停止させることなくPSB（BB）を物理パーティションから削除することができます。

- ハードウェアリソースの移動
  一時的な業務の拡張や負荷の増大に対処するため、ほかの物理パーティションのPSB(BB)を一時的に削除します。削除したPSB(BB)は、ハードウェアリソースの増強が必要な物理パーティションに追加することにより、2つの物理パーティション上の業務を稼働させたまま、ハードウェアリソースを移動することができます。負荷変動に対して柔軟に対応するシステムを構築できます。

- 部品故障時の活性交換
  何らかの故障が発生したPSB（BB）のハードウェアリソースが縮退している場合に、物理パーティション上の業務を停止させずに故障が発生しているシステムボードを一時的に物理パーティションから削除し、故障部品を交換できます。部品交換したあとに、業務を停止させずに物理パーティションにPSB（BB）を追加する

ことで、元の構成に戻すことができます。

1.5.3 CPUとメモリの動的再構成

CPUの追加または削除

物理パーティションの動的再構成によってCPUを追加した場合、それらのCPUは自動的にOracle VM Server for SPARCに認識され、使用可能となります。

物理パーティションの動的再構成によってシステムボード（PSB<BB>）を組み込み、CPUを追加した場合、次のいずれかの動作になります。

- 論理ドメイン構成情報が出荷時構成のfactory-defaultである物理パーティションにPSB（BB）を追加した場合、追加したPSB（BB）に搭載されているCPUは、すべて制御ドメインに自動的に追加されます。
- 論理ドメインが構成されている状態、または論理ドメイン構成情報が保存された状態の物理パーティションに新たにPSB（BB）を追加した場合、PSB（BB）に搭載されているCPUは、どの論理ドメインにも割り当てられていない状態となります。このCPUを使用するためには、ldmコマンドで論理ドメインにCPUを追加する必要があります。
- 論理ドメインが構成されている物理パーティションのPSB（BB）を交換する際に、PSB（BB）の切り離しによって自動的に論理ドメインのCPUの割り当てが削減された場合、PSB（BB）を追加すると、削減された数のCPUが自動的に論理ドメインに追加されます。

動的再構成によってCPUを削除する場合、次の条件を満たす必要があります。条件を満たしていない場合、物理パーティションの動的再構成処理が停止し、メッセージが表示されます。

- 削除されるCPUには、稼働中のプロセスがバインドされていない。
  プロセスがバインドされている場合は、プロセスのバインドを解除するか、プロセスを停止させる必要があります。
- 削除されるCPUは、どのプロセッサセットにも属していない。
  プロセッサセットに属しているCPUが存在する場合、psrsetコマンドにより、プロセッサセットから削除しておく必要があります。
- ldm add-coreコマンドまたはldm set-coreコマンドでcore ID（CID）を指定せずに、論理ドメインにCPUコアが割り当てられている。

物理パーティションの動的再構成において、論理ドメインのリソースを削減してシステムボードの切り離しを行う際に上記の条件を満たしていない場合、物理パーティションの動的再構成処理が停止し、メッセージが表示されます。

メモリの追加または削除

物理パーティションの動的再構成によってシステムボード（PSB<BB>）を組み込み、メモリを追加した場合、次のいずれかの動作になります。

- 論理ドメイン構成情報が出荷時構成のfactory-defaultである物理パーティションにPSB（BB）を追加した場合、追加したPSB（BB）に搭載されているメモリは、すべて制御ドメインに自動的に追加されます。
- 論理ドメインが構成されている状態、または論理ドメイン構成情報が保存された

状態の物理パーティションに新たにPSB（BB）を追加した場合、PSB（BB）に搭載されているメモリは、どの論理ドメインにも割り当てられていない状態となります。このメモリを使用するためには、ldmコマンドで論理ドメインにメモリを追加する必要があります。

- 論理ドメインが構成されている物理パーティションのPSB（BB）を交換する際に、PSB（BB）の切り離しによって自動的に論理ドメインのメモリの割り当てが削減された場合、PSB（BB）を追加すると、削減されたサイズのメモリが自動的に論理ドメインに追加されます。
  物理パーティションの動的再構成によってPSB（BB）を削除する場合、以下の点に注意してください。
  - ldm add-memoryコマンドまたはldm set-memoryコマンドのmblockオプションによって、論理ドメインに物理アドレスを指定したメモリが割り当てられている場合、物理パーティションの動的再構成によってPSB（BB）を削除しないでください。

1.5.4 ソフトウェア条件および確認方法

物理パーティションの動的再構成を使用するためのソフトウェア条件

物理パーティションの動的再構成を実施するために必要なXCP／Oracle Solarisおよび必須SRU／パッチは、表 1-3、表 1-4のとおりです。物理パーティションの動的再構成の機能を使用するためには、XCPファームウェアおよびOracle VM Server for SPARCソフトウェアの版数の組み合わせを正しく設定する必要があります。最新のソフトウェア条件は、お使いのサーバの最新の『プロダクトノート』を参照してください。

表 1-3 SPARC M12-2Sでの物理パーティションの動的再構成に必要なOracle Solarisおよび必須SRU／パッチ一覧

OS版数	ドメイン種			
	制御ドメイン 非仮想化環境	ルートドメイン	I/Oドメイン	ゲストドメイン
Oracle Solaris 11	Oracle Solaris 11.4 (*1)	Oracle Solairs 11.4以降 (*2)	Oracle Solairs 11.4以降 (*2)	Oracle Solairs 11.4以降 (*2)
	Oracle Solaris 11.3 (*1) SRU11.3.17.5.0以降	Oracle Solaris 11.3以降 (*2)	Oracle Solaris 11.3以降 (*2)	Oracle Solaris 11.3以降 (*2)
		Oracle Solaris 11.2以降 (*2)	Oracle Solaris 11.2以降 (*2)	Oracle Solaris 11.2以降 (*2)
			Oracle Solaris 11.1 (*2) SRU1.4以降	Oracle Solaris 11.1以降 (*2)

表 1-3　SPARC M12-2Sでの物理パーティションの動的再構成に必要なOracle Solarisおよび必須SRU／パッチ一覧 (続き)

OS版数	ドメイン種			
	制御ドメイン 非仮想化環境	ルートドメイン	I/Oドメイン	ゲストドメイン
Oracle Solaris 10	-	-	-	Oracle Solaris 10 1/13 (*3) 150310-03以降 Oracle Solaris 10 8/11 (*3)(*4) Oracle Solaris 10 1/13 SPARCバンドルまたはPTF R15051 150310-03以降 Oracle Solaris 10 9/10 (*3)(*4) Oracle Solaris 10 1/13 SPARCバンドルまたはPTF R15051 150310-03以降

*1: system/ldomsおよびsystem/ldoms/ldomsmanagerパッケージが必要です。これらのパッケージはgroup/system/solaris-large-serverおよびgroup/system/solaris-small-serverに含まれます。
*2: system/ldomsパッケージが必要です。このパッケージはgroup/system/solaris-large-serverおよびgroup/system/solaris-small-serverに含まれます。
*3: ゲストドメインに割り当て可能なCPUはLSB番号の制限はありませんが、ゲストドメインでOracle Solaris 10を動作させた場合、1つのゲストドメインに割り当て可能なCPU（vcpu）の数は1024までです。
*4: Oracle Solaris 10 9/10、またはOracle Solaris 10 8/11をインストールする前に、追加の手順を実施する必要があります。詳細は、最新の『プロダクトノート』の「ゲストドメインにOracle Solaris 10をインストールする場合の留意事項」の手順を参照してください。

表 1-4　SPARC M10-4Sでの物理パーティションの動的再構成に必要なXCP／Oracle Solarisおよび必須SRU／パッチ一覧

サーバ	XCP	Oracle Solaris	必須パッケージ 必須製品	必須SRU 必須パッチ
SPARC M10-4S	2220以降	Oracle Solaris 11.2以降	system/ldoms(*1) system/ldoms/ldomsmanager(*2)	なし
		Oracle Solaris 11.1	system/ldoms(*1) system/ldoms/ldomsmanager(*2)	SRU11.1.14.5.0以降(*3)
		Oracle Solaris 10 1/13	Oracle VM for SPARC 3.1(*4)	150817-02以降(*4)(*5)

*1: 制御ドメインおよびゲストドメインに必須です。group/system/solaris-large-serverおよびgroup/system/solaris-small-serverに含まれます。
*2: 制御ドメインのみに必須です。group/system/solaris-large-serverおよびgroup/system/solaris-small-serverに含まれます。
*3: 制御ドメインおよびゲストドメインに必須です。SRU11.1.14.5.0にOracle VM Server for SPARC 3.1.0.1が含まれていますが、物理パーティションの動的再構成機能を安定稼働させるためにはOracle Solaris 11.1の修正（CR:17709858）が必要になります。なお、この問題は、SRU11.1.15.4.0以降で修正されています。
*4: 制御ドメインのみに必須です。
*5: 150400-01から150400-06までのパッチを適用しないでください。

ソフトウェア版数の確認方法

- **XCPファームウェア版数の確認方法**

1. **XSCFシェルにログインします。**
2. **version -c xcpコマンドを実行し、ファームウェアバージョン情報を確認します。**

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
BB#01-XSCF#0 (Standby)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
```

- **Oracle VM Server for SPARCソフトウェア版数の確認方法**

1. **物理パーティションの制御ドメインコンソールにログインします。**
   制御ドメインコンソールへのログイン方法は、『SPARC M12/M10 システム運用・管理ガイド』の「8.3　XSCFシェルから制御ドメインコンソールに切り替える」を参照してください。
2. **Oracle Solaris 11の場合は、pkg info entireコマンドを実行し、Oracle SolarisのSRUの版数を確認します。**

```
# pkg info entire
Name: entire
Summary: entire incorporation including Support Repository Update (Oracle
Solaris 11.1.14.5.0). <-- SRUの版数
```

3. **Oracle Solaris 10の場合は、ldm -Vコマンドを実行し、Oracle VM Server for SPARCの版数を確認します。**

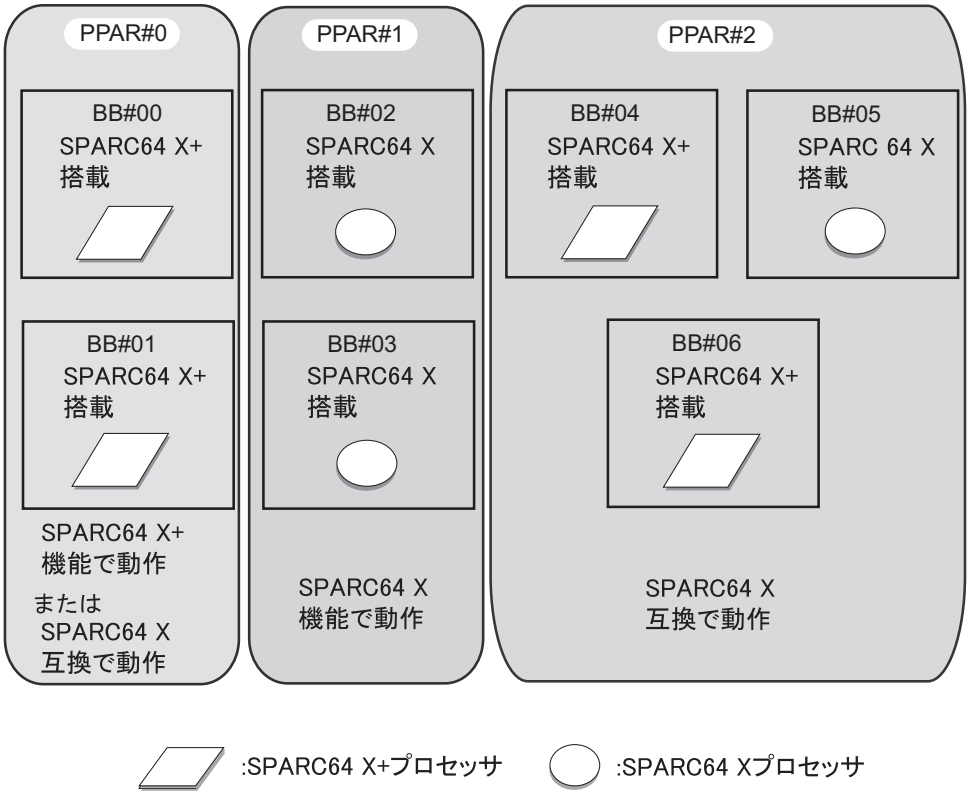
```
# ldm -V
Logical Domains Manager (v 3.1.0.1) <-- Oracle VM Server for SPARCの版数

Hypervisor control protocol v 1.9
Using Hypervisor MD v 1.3
```

1.6 SPARC64 X+プロセッサとSPARC64 Xプロセッサの混在

SPARC M10-4Sでは、SPARC64 X+プロセッサが搭載されたSPARC M10-4Sと、SPARC64 Xプロセッサが搭載されたSPARC M10-4Sを混在させて、1つの物理パーティションを構成できます。
SPARC64 X+プロセッサとSPARC64 Xプロセッサを混在させた物理パーティションを起動すると、すべてのプロセッサは、SPARC64 X互換で動作します。
SPARC64 X+プロセッサとSPARC64 Xプロセッサの混在に関する詳細は、「2.6.3 SPARC64 X+プロセッサとSPARC64 XプロセッサをPPAR内で混在させるための条件」を参照してください。

図 1-19　SPARC M10システムに搭載されるプロセッサと物理パーティションの構成例



1.6.1 CPU動作の種類とCPU動作モード

SPARC M10-4Sでは、XSCFファームウェアのsetpparmodeコマンドで設定されるCPU動作モードによって、物理パーティションごとにCPU動作の種類を指定すると、SPARC64 X+プロセッサの機能を利用して動作するか、SPARC64 Xプロセッサの機能を利用して動作するかが、Oracle Solarisが次回起動するときに自動的に判別されます。

CPU動作には、以下の種類があります。

- SPARC64 X+機能を利用して動作する
  物理パーティションに搭載されているすべてのCPUが、SPARC64 X+プロセッサの持つ拡張された機能を使用して動作します。

- SPARC64 X機能を利用して動作する
  物理パーティションに搭載されているすべてのCPUが、SPARC64 Xプロセッサの機能で動作します。物理パーティションを構成するシステムボードに搭載されているCPUの種類にかかわらず、すべてのCPUをSPARC64 Xプロセッサの機能で動作させるときに使用します。SPARC64 Xプロセッサは、常に、SPARC64 Xプロセッサの機能を利用して動作します。

注－CPU動作種類とサポートされるXCPファームウェアの版数については、最新のXCP版数の『SPARC M10 システム プロダクトノート』を参照してください。

注－CPU動作モードの設定については、「2.6.2　CPU動作モードの設定」を参照してください。

注－CPU動作モードと物理パーティションの動的再構成については「2.6.4　CPU動作モードと物理パーティションの動的再構成の関係」を参照してください。

第2章

システム運用のための条件と設定

ここでは、ドメイン構築を開始する前に知っておくべき内容を説明します。

- XSCFのステータス管理
- XSCFの条件と設定
- Oracle Solarisの条件と設定
- 論理ドメインの条件と設定
- 動的再構成の条件と設定
- SPARC64 X+プロセッサを使用する場合の留意点

2.1 XSCFのステータス管理

物理パーティションの再構成を正しく行うためには、物理パーティションおよび1つのビルディングブロックを表すシステムボード（PSB<BB>）のステータスに応じて操作を行う必要があります。ここでは、物理パーティションの再構成のための機能が動作する条件を理解するために、XSCFが管理する物理パーティションおよびPSB（BB）のステータス情報、およびそれぞれのステータスによる注意すべき点について説明します。

2.1.1 物理パーティションのステータス

物理パーティションは、XSCFによってステータスが管理されています。物理パーティションのステータスは、XSCFが提供するユーザーインターフェースにより表示、参照することができます。次の例では、コマンドライン形式のXSCFシェルで物理パーティションのステータスを確認するために、showpparstatusコマンドを使用しています。

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Initialization Complete
```

ユーザーインターフェースの詳細については、「第3章　ドメイン構築のための操作」を参照してください。

XSCFでは、物理パーティションに関する次のステータスを管理しています。

表 2-1　物理パーティションのステータス

ステータス	意味
Powered Off	電源が切断されている状態。
Initialization Phase	Power-On Self-Test（POST）動作中の状態。
Initialization Complete	POST完了の状態。
Running	POST処理が完了し稼働中の状態。
Hypervisor Aborted	Hypervisor Abortが発生し、リセットされるまでの状態。
－	上記以外（物理パーティションが定義されていない状態）。

物理パーティションの動的再構成を行うためには、該当する物理パーティションのステータスに従って、システムボード（PSB<BB>）の操作方法を判断する必要があります。

2.1.2　システムボードのステータス

XSCFは、1つのビルディングブロックのステータス管理をシステムボード（PSB）単位で行います。PSBのステータスは、XSCFが提供するユーザーインターフェースにより表示、参照することができます。次の例では、コマンドライン形式のXSCFシェルでPSBのステータスを確認するために、showboardsコマンドを使用しています。

```
XSCF> showboards -av
PSB   R PPAR-ID(LSB) Assignment   Pwr   Conn Conf Test    Fault
---- - ------------ ----------- ---- ---- ---- ------- --------
00-0 * 00(00)        Assigned    n    n    n    Unmount Normal
01-0   00(01)        Assigned    y    y    n    Passed  Normal
02-0   00(02)        Assigned    y    y    n    Passed  Normal
04-0   00(04)        Assigned    y    y    n    Passed  Normal
05-0   01(00)        Assigned    y    y    n    Passed  Normal
06-0   01(01)        Assigned    y    y    n    Passed  Normal
09-0   00(09)        Assigned    y    y    n    Passed  Normal
13-0   13(00)        Assigned    y    y    n    Passed  Normal
15-0   13(01)        Assigned    y    y    n    Passed  Normal
```

ユーザーインターフェースの詳細については、「第3章　ドメイン構築のための操作」を参照してください。

XSCFでは、システムボードに関する次のステータスを管理しています。

表 2-2　システムボードのステータス

表示項目	表示内容	意味
PSB	xx-y（xx:00から15までの整数、y:0固定）	1つのビルディングブロック（BB）を表す物理システムボード（PSB）番号（xxは、BB-ID）
R(*1)	*	物理パーティションが再起動されると、PSB（BB）が現行の物理パーティション構成に組み込まれる、または切り離されて構成が変更される状態
PPAR-ID	0から15までの整数	PSB（BB）が割り当てられている物理パーティションの番号
	SP	PSB（BB）がシステムボードプール状態
	Other	ユーザー権限を持つ物理パーティションに物理パーティション構成情報が設定され、かつ、ユーザー権限を持たない物理パーティションに属している状態
LSB	00から15までの整数	物理パーティションで使用される論理システムボードの番号
Assignment（PPARへの割り当て状態）	Unavailable	PSB（BB）が物理パーティションに割り当てられていない状態であり、システムボードプール、未診断、診断中、診断異常のいずれかの状態（PSBが認識できない場合も含まれる）
	Available	PSB（BB）がシステムボードプールであり、診断が正常に終了している状態
	Assigned	PSB（BB）が割り当てられている状態
Pwr（システムボードの電源投入状態）	n	電源が切断されている状態
	y	電源が投入されている状態
Conn（PPAR構成への組み込み状態）	n	PSB（BB）が物理パーティション構成から切り離されている状態
	y	PSB（BB）が物理パーティション構成に組み込まれている状態
Conf（Oracle Solarisでの稼働状態）	n	PSB（BB）でOracle Solarisが稼働していない状態
	y	PSB（BB）でOracle Solarisが稼働している状態
Test（診断状態）	Unmount	未実装または未定義の状態
	Unknown	診断されていない状態
	Testing	診断中
	Passed	診断が正常に終了した状態
	Failed	診断で異常が検出され、PSB（BB）が動作しない状態
Fault（縮退状態）	Normal	正常な状態
	Degraded	縮退部品がある状態（PSB（BB）は動作可能）
	Faulted	異常によりPSB（BB）が動作できない状態、または通信異常によりPSB（BB）が制御できない状態

*1:　-v オプションを指定した場合だけ表示される項目です。

XSCFは、PSB（BB）の増設／減設、物理パーティション構成情報への登録、物理パーティションの起動や停止など、そのときの状況に応じてPSB（BB）のステータスを変化／設定します。物理パーティションの動的再構成においても、同様にこれらのステータスは、PSB（BB）の追加および削除によって変化します。

物理パーティションの動的再構成を行うためには、該当するPSB（BB）のステータスに従って、PSB（BB）の操作方法を判断する必要があります。

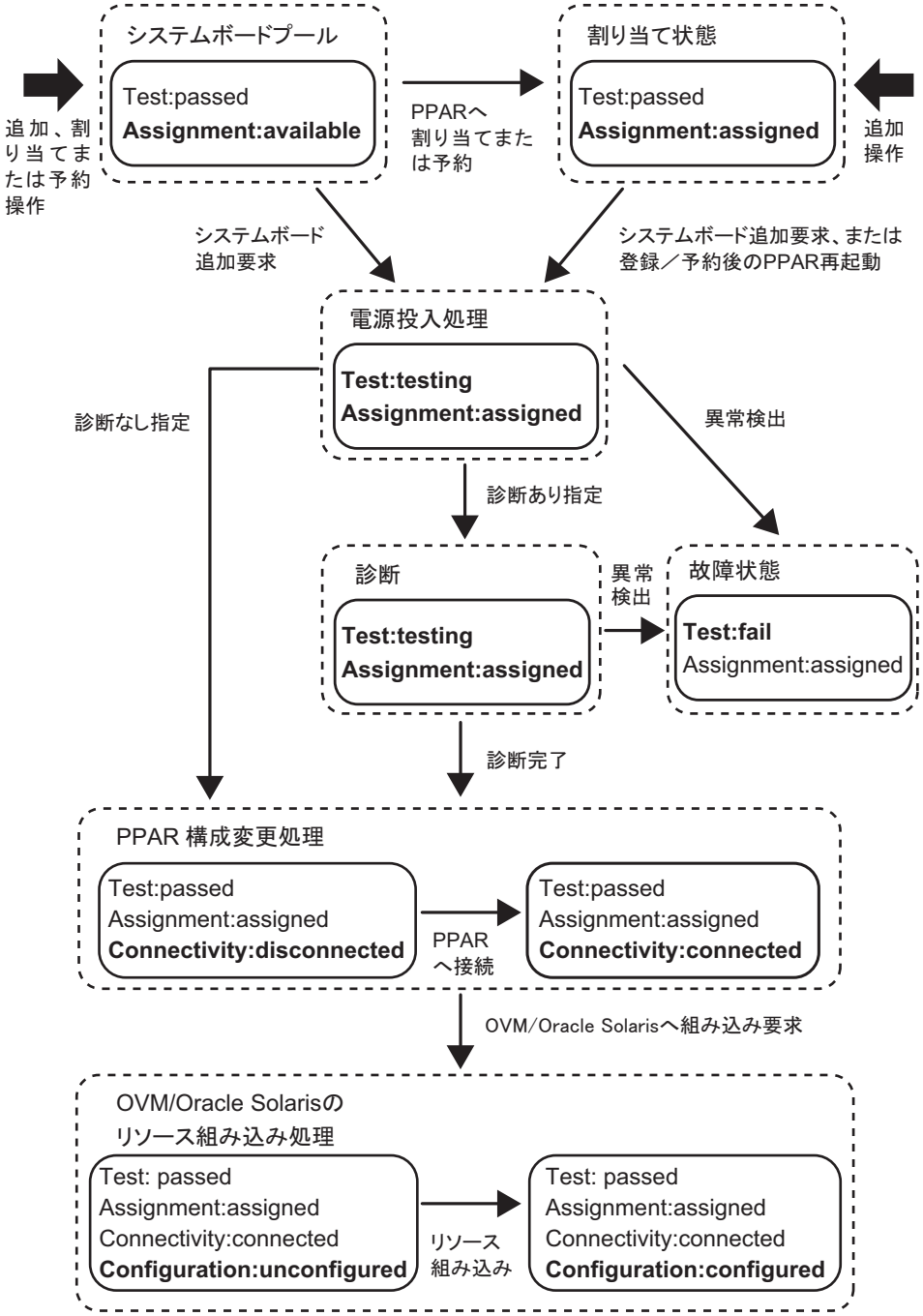
2.1.3 ステータスの変化

ここでは、それぞれの物理パーティションの構成変更に沿って、処理のながれ、およびシステムボード（PSB<BB>）のステータスの変化を説明します。

システムボードの追加処理

システムボード（PSB<BB>）の追加または追加予約を操作したときの処理のながれ、およびPSB（BB）のステータスの変化を概略図により説明します。図 2-1に記載されるPSB（BB）のステータスは、変化する主なステータスを記載しています。

図 2-1　システムボードの追加処理

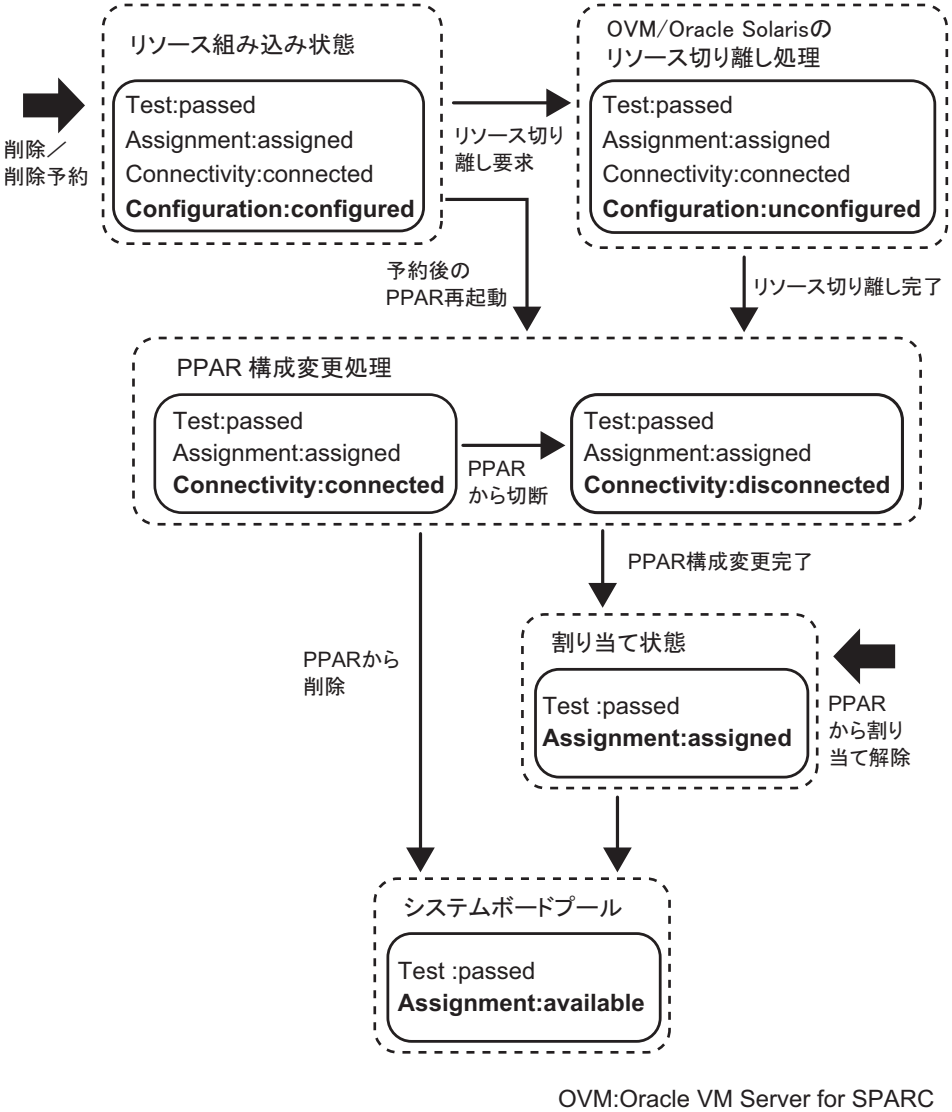


OVM:Oracle VM Server for SPARC

システムボードの削除処理

システムボード（PSB<BB>）の削除または削除予約を操作したときの処理のながれ、およびPSB（BB）のステータスの変化を概略図により説明します。図 2-2に記載されるPSB（BB）のステータスは、変化する主なステータスを記載しています。

図 2-2　システムボードの削除処理

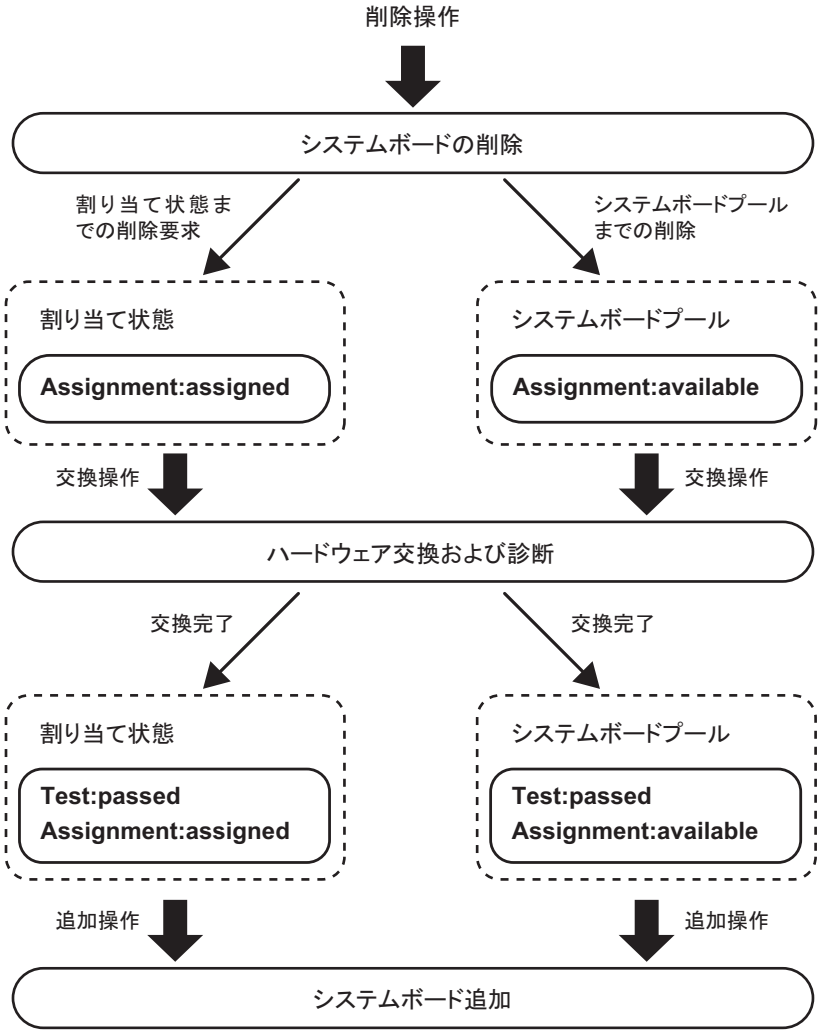


システムボードの交換処理

システムボード（PSB<BB>）の交換を操作したときの処理のながれ、およびPSB（BB）のステータスの変化を概略図により説明します。図 2-3に記載されるPSB（BB）のス

テータスは、変化する主なステータスを記載しています。記載されるステータスは、ハードウェアの交換前と交換後の例として記載しています。ハードウェアを交換するにあたり、記載されているステータスに一致していない場合があります。PSB（BB）の追加処理と削除処理のながれやステータスについては、「システムボードの追加処理」、および「システムボードの削除処理」を参照してください。ハードウェアの交換操作については、お使いのサーバの『サービスマニュアル』を参照してください。

図 2-3　システムボードの交換処理



2.2　XSCFの条件と設定

ここでは、XSCFが必要とする条件とXSCFによる設定について説明します。

2.2.1 XSCFが必要とする構築条件

システムボード（PSB<BB>）を追加する場合、SPARC M12-2S/M10-4Sを設置および接続しただけでは物理パーティションに組み込むことはできません。XSCFシェルまたはXSCF Webを使用し、PSB（BB）を物理パーティション構成情報（PPAR構成情報）に登録することで、物理パーティションに対する操作が可能となります。物理パーティションを操作する場合、PSB（BB）を組み込む前に、追加するPSB（BB）がPPAR構成情報に登録されていることを確認しなければなりません。

PSB（BB）の削除および交換の場合、PSB（BB）はすでにPPAR構成情報に登録されている状態です。あえてPPAR構成情報に登録されていることを確認する必要はありません。

2.2.2 XSCFによる設定

物理パーティションの構成に関する機能には、Oracle Solarisの再構成やメモリアロケーションなどに関する複雑さを回避し、より円滑に物理パーティションを再構成して運用を続けるために、いくつかのオプション機能が提供されています。これらのオプション機能は、XSCFシェルまたはXSCF Webを使用し設定します。ここでは、以下のオプション機能について説明します。

- コンフィグレーションポリシーオプション
- メモリ無効化オプション
- I/O無効化オプション

これらのオプションは、setpclコマンドを使用して設定します。

コンフィグレーションポリシーオプションの設定

物理パーティションの電源投入や、物理パーティションの動的再構成によるシステムボード（PSB<BB>）の追加では、PSB（BB）を安全に組み込むために、ハードウェアの自動診断を行います。コンフィグレーションポリシーは、これらの自動診断で異常が検出された場合の縮退範囲を指定するオプションです。

縮退の単位は、ハードウェア異常を検出した部品単位、その部品が実装されているシステムボード（PSB）単位（BB単位）、および物理パーティション単位で設定できます。このオプションで設定できる値と縮退の単位を表 2-3に記載します。本オプションのデフォルトは「fru」です。

表 2-3 縮退の単位

設定値	縮退の単位
fru	CPUやメモリといった部品単位で縮退する。
psb	システムボード（PSB）単位（BB単位）で縮退する。
system	物理パーティション単位であり、縮退せずに物理パーティションを停止する。

注－コンフィグレーションポリシーオプションは、該当の物理パーティションを停止させて設定してください。

メモリ無効化オプション

物理パーティションに組み込むシステムボード（PSB<BB>）上のメモリを論理ドメインで使用できないようにするためのオプションです。対象の物理パーティションに追加するシステムボードのメモリを論理ドメインに組み込まず、I/O無効化オプションとの組み合わせによって、CPUのみを組み込む場合に利用できます。このオプションでは、メモリ無効化を有効とする「true」、および無効とする「false」のいずれかを選択できます。デフォルトは「false」です。

注－メモリ無効化オプションを有効にしても、PSB（BB）の診断や管理するために実際にメモリを搭載する必要があります。また、メモリ無効化オプションを有効とした場合、物理パーティション内で使用可能なメモリが減少し、性能が低下することが懸念されます。このオプションは、業務への影響を考慮したうえで設定する必要があります。

注－メモリ無効化オプションは、該当PSB（BB）がシステムボードプール状態、または物理パーティションが電源切断されている状態で設定してください。

I/O無効化オプション

システムボード（PSB<BB>）やPCIボックス上のPCIカード、ディスクドライブや基本ローカルエリアネットワーク（LAN）ポートを対象の物理パーティションに組み込ませないためのオプションです。物理パーティションにPSB（BB）のCPUとメモリだけを使用する必要がある場合は、I/O無効化オプションを有効にする「true」に設定してください。物理パーティションにPSB（BB）のPCIカードとI/Oデバイスを使用する必要がある場合は、このオプションを「false」に設定してください。この場合、これらのI/Oデバイスを使用するうえでの制約を十分に理解しておく必要があります。また、動的再構成によるPSB（BB）の削除を行う前に、I/Oデバイスを使用しているソフトウェア（例：アプリケーションプログラムまたはデーモン）やI/Oドメインを停止しなければなりません。

論理ドメイン構成情報をfactory-defaultに設定して物理パーティションを起動すると、I/O無効化オプションの設定に従って、物理パーティションは起動されます。論理ドメイン構成情報をfactory-default以外で物理パーティションを起動したとき、いずれかの論理ドメインに該当システムボードのルートコンプレックスやI/Oデバイスが属していると、物理パーティションの構成変更が行われたと判断して、factory-defaultに切り替えて物理パーティションを起動します。属していない場合は、factory-defaultへ切り替えは行われません。動的再構成によるPSB（BB）の追加を行った場合は、I/O無効化オプションの設定に従って、物理パーティションに組み込み処理が行われます。

I/O無効化オプションのデフォルトは「false」です。

注－I/O無効化オプションは、該当PSB（BB）がシステムボードプール状態、または物理パーティションが電源切断されている状態で設定してください。

2.2.3 メモリミラーモードの留意点

メモリミラーモードとは、メモリのハードウェア信頼性確保のため、メモリを二重化する機能です。メモリミラーモードを有効することにより、メモリの一部に故障が発生しても、回復可能な故障であれば、物理パーティションの運用を継続することができます。物理パーティションの動的再構成を使用する場合、メモリミラーモードを有効にすることで、物理パーティションの動的再構成の機能が制限されることはありません。

しかし、物理パーティションの構成および運用上で注意が必要です。例えば、メモリミラーモードを有効にしたシステムボード（PSB<BB>）を動的再構成で操作する場合、PSB（BB）の削除を行うことで、Oracle VM Server for SPARCやOracle Solarisが使用しているメモリリソースはほかのシステムボードに移動します。このとき、移動先となるPSB（BB）のメモリミラーモードが無効であった場合、Oracle VM Server for SPARCやOracle Solarisが使用しているメモリリソースは正しく移動されますが、結果的にメモリの信頼性が低下した状態で運用することになります。メモリミラーモードは、ドメインの構成や運用など要件に考慮し、事前の計画により設定するようにしてください。

メモリミラーリングの設定の詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「第14章　信頼性の高いシステムを構築する」を参照してください。

2.2.4 CPUモジュールを増設する場合の留意点

I/Oバス再構成を有効に 設定しているとき、CPUモジュール(CMUU)を増設すると、それまで使用していたI/Oデバイスのデバイスパスが変化します。
setpparmodeコマンドの-m functionオプションで設定できるI/Oバス再構成(ioreconfigure)機能は、 通常、無効（false）にしてください。
I/Oバス再構成機能を有効（true）にすると、Oracle Solarisの再インストールが必要になることがあります。
setpparmode (8)コマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

2.3 Oracle Solarisの条件と設定

ここでは、物理パーティションの再構成による運用を開始するために必要な条件と設定について説明します。

2.3.1 I/Oおよびソフトウェアの要件

I/Oデバイスの冗長構成

複数の筐体で物理パーティションを構築する場合や、I/Oデバイスの故障で業務プロセスに支障をきたす場合は、I/Oデバイスの冗長構成の構築とI/Oデバイスの切り替えができるソフトウェアを使用することを推奨します。

2.3.2 スワップ領域に関する留意点

使用可能な仮想メモリ容量は、システムに搭載されたメモリ容量とディスク上のスワップ領域の総和です。使用可能なメモリ容量は、すべての必要な操作のために十分な容量が搭載されているように留意する必要があります。

BB(PSB)追加時の留意点

Oracle Solarisではデフォルトで、スワップ領域がシステムのクラッシュダンプを保持する目的でも使用されます。これの代わりにダンプ専用のデバイスを使用すべきです。詳細は、Oracle Solarisのマニュアルページのdumpadmを参照してください。クラッシュダンプの格納に使用されるデフォルトのスワップ領域は、搭載されているメモリ容量の大きさに応じて変わります。

クラッシュダンプを保存するダンプデバイスのサイズは、実装メモリより大きくする必要があります。システムボードが追加され実装メモリが増える場合、必要に応じてダンプデバイスを再設定してください。詳細については、Oracle Solarisのマニュアルページのdumpadmを参照してください。

BB(PSB)削除時の留意点

BB(PSB)の削除によってドメインのメモリが削除される場合、BB(PSB)のメモリはディスクのスワップ領域にスワップされます。そして、削除されるメモリ容量だけ減少します。BB(PSB)を削除するコマンドを実行する前に、スワップ領域全体を確認し、BB(PSB)の物理メモリの内容を維持できる十分なスワップ領域が確保されていることを確認してください。さらに、削除するBB(PSB)に接続されているディスクがスワップ領域全体の一部として使用されている場合があるので、注意してください。つまり、BB(PSB)の削除によって、そのスワップ領域分も減少する場合は、失われるスワップ領域も必ず計算する必要があります。

- （使用可能なスワップ領域：1.5 GB）＞（削除されるメモリ容量：1 GB）の場合、削除後の使用可能なメモリ容量は0.5 GBとなります。
- （使用可能なスワップ領域：1.5 GB）＜（削除される容量：2 GB）の場合、BB(PSB)の削除処理は失敗します。

現在の使用可能なスワップ領域のサイズは、Oracle Solaris上でswap -sコマンドを実行して、「available」となっているサイズを見ることでわかります。詳細については、Oracle Solarisのマニュアルページのswapを参照してください。

2.3.3 リアルタイムプロセスの留意点

物理パーティションの動的再構成でシステムボードを削除する場合、削除するシステムボード上のメモリが割り当てられている論理ドメインにおいて、Oracle Solarisは一時停止（suspend）されます。システムにリアルタイムの要件（リアルタイムプロセスの存在により表示されるもの）がある場合、このような処理がこれらのプロセスにかなりの影響を与えることがありますので注意してください。

2.4 論理ドメインの条件と設定

注ー論理ドメインは、Oracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerを使用して構築します。このため、論理ドメインの構築時には、お使いのバージョンのOracle VM Server for SPARC のマニュアルで必須ソフトウェアを確認してください。

注ーここでは、論理ドメインの構築の際に必要最低限の情報を記載しています。Oracle VM Server for SPARC で使用可能な機能の詳細については、お使いのバージョンのOracle VM Server for SPARC のマニュアルで必須ソフトウェアを確認してください。

論理ドメインの構築時には、次の内容を設定します。

制御ドメインの初期構築

初期状態では、すべてのハードウェアリソースが制御ドメインに割り当てられています（factory-default構成）。そのため、制御ドメインに対するハードウェアリソースの割り当てを見直し、ゲストドメインのためのハードウェアリソースを用意します。このときには、次の作業を行います。

- デフォルトサービスの設定
- 遅延再構成モードの設定
- 仮想CPUのリソースの解放
- 仮想メモリのリソースの解放
- I/Oまたは仮想I/Oのリソースの解放
- 制御ドメインの設定内容の保存

ゲストドメインの構築

制御ドメインに対するハードウェアリソースの割り当てを見直したあとに、ゲストドメインに対してハードウェアリソースを割り当てます。このときには、次の作業を行います。

- 仮想CPUの割り当て
- 仮想メモリの割り当て
- 仮想ネットワークの設定

- 仮想ディスクの設定
- 起動デバイスの設定
- 自動的な再起動の設定
- ゲストドメインの設定の保存

2.4.1 論理ドメインの構築に関する留意点

ここでは、論理ドメインを構築するときに留意すべき点を説明します。

論理ドメイン数に関する留意点

構築できる論理ドメインの最大数は、論理ドメインに割り当て可能な仮想CPU（スレッド）の数になります。また、物理パーティションあたりの最大数は、ソフトウェアによって制限されます。表 2-4は、構築できる論理ドメインの最大数をモデル別に示しています。

表 2-4　モデルごとの最大論理ドメイン数

モデル	最大論理ドメイン数
SPARC M12-1	48
SPARC M12-2	192
SPARC M12-2S	物理パーティションあたりのスレッド数または256の小さい方
SPARC M10-1	32
SPARC M10-1（3.7 GHz）	16
SPARC M10-4	128
SPARC M10-4（3.7 GHz）	64
SPARC M10-4S	物理パーティションあたりのスレッド数または256の小さい方

ただし、通常、論理ドメインにはコア単位で仮想CPUを割り当てるようにします。

物理パーティションのコア数を超える論理ドメインを作成する場合は、事前に動作テストなどの検証を実施してください。

CPUの自動交替機能利用時の留意点

Oracle VM Server for SPARC 3.0以降でサポートされる、CPUの自動交替機能を使うときは、CPUの構成について留意する必要があります。詳細は、『SPARC M12/M10システム運用・管理ガイド』の「10.7　故障CPUの自動交替を設定する」を参照してください。

Logical Domains DRデーモンの稼働

論理ドメインにハードウェアリソースを割り当てるときには、その論理ドメインを管理する制御ドメインで、Logical Domains DRデーモン（drd）が稼働している必要があります。drdデーモンの詳細は、『Oracle VM Server for SPARC リファレンスマニュアル』を参照してください。

CPUコア アクティベーションキーの登録

論理ドメインに割り当てるCPUコアは、XSCFに登録したCPUコア アクティベーションキーを物理パーティションに割り当てることで使用可能な状態にしておく必要があります。

CPUコア アクティベーションキーは、サーバ導入時に同梱されている「<モデル名> CPUコア アクティベーション」とラベルの印刷されたCD-ROMに含まれています。CPUコア アクティベーションキーを登録する前にご用意ください。

CPUコア アクティベーションキーは、CD-ROM内の「ACTIVATION_KEY」フォルダにテキストファイルで含まれています。キーをまとめて登録するためのファイル（XXXXX_XX.TXT）と、1つずつ登録するためのファイル（XXXXX_XX_001.TXTなど）が用意されています。必要に応じていずれかのファイルをご使用ください。

注－CPUコア アクティベーションキーは、システム導入時にすでにXSCFに登録されている場合があります。登録されているキーは、システムに同梱されているCD-ROMと同じ内容です。この場合はキーを登録する必要はありません。

注－CPUコア アクティベーションキーによって使用可能なCPUコアが減るときには、物理パーティションに割り当てているCPUコアが使用可能数以下となるよう、事前に割り当てなおしてください。

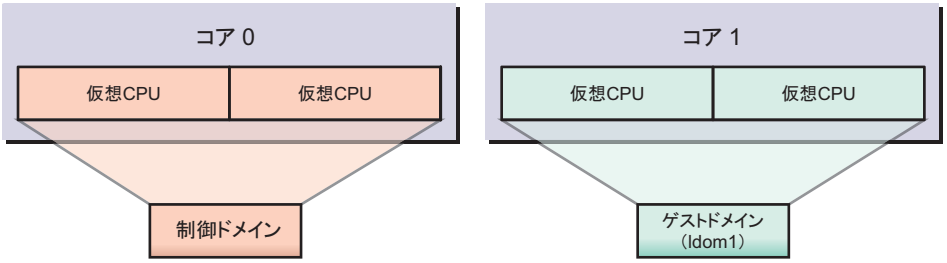
制御ドメインの初期設定時の注意

制御ドメインを初期設定するときには、ldmコマンドによるメモリの動的再構成（Dynamic Reconfiguration: DR）を使用しないでください。制御ドメインの初期設定では、論理ドメインの再起動によって構成情報が反映される遅延再構成モードを使用してください。

仮想CPUの割当単位

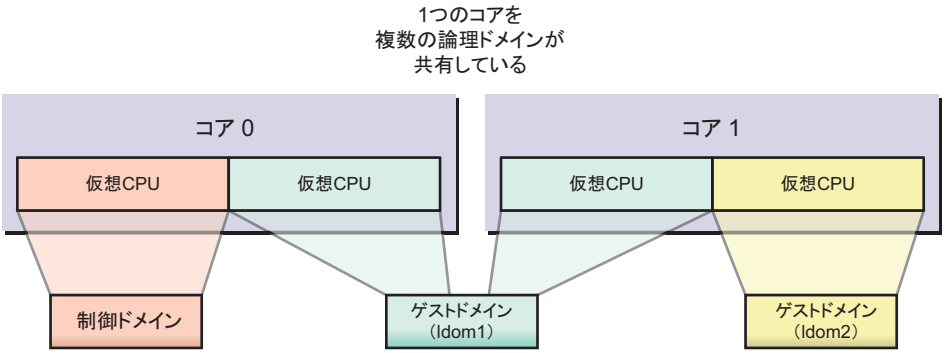
仮想CPUはコア単位、またはスレッド単位で割り当て可能ですが、通常はコア単位で割り当てるようにします。図 2-4は、通常の割り当ての例を示しています。

図 2-4　通常の割り当ての例



仮想CPUをコア単位で割り当てない場合、パフォーマンスが低下するおそれがあります。図 2-5は、パフォーマンスが低下する例を示しています。

図 2-5　望ましくない割り当ての例



CPUのコアIDの指定について

ldmコマンドを使用して、論理ドメインにCPUコアID（CID）を指定して、名前付きコアを割り当てたり削除したりする場合、CPUコアの物理的な位置との関係を把握してCIDを設定する必要があります。

注－CIDを指定して論理ドメインにCPUコアを割り当てる場合は、以下の機能が使用できなくなります。内容をよく理解したうえで設定してください。

- CPUの動的再構成
- CPUの自動交替機能
- CPU動的リソース管理（DRM）

Oracle VM Server for SPARC によるコアID(CID)は、物理CPU番号(PID)とスレッド数によって決定されます。

SPARC M12/M10の、各モデルのCPUの物理的な配置のイメージは以下のとおりです。

図 2-6　SPARC M12-1のCPU配置イメージ

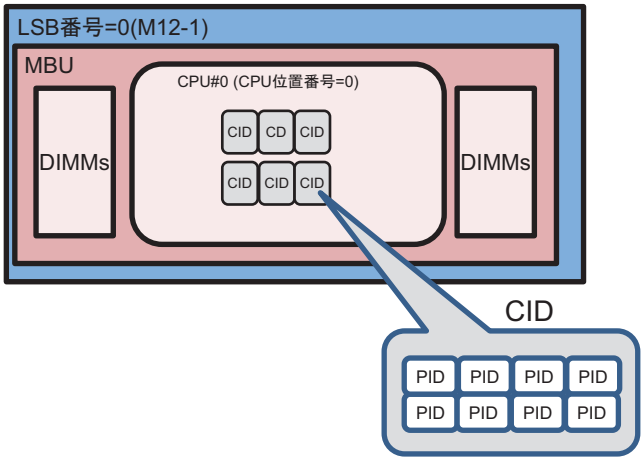


図 2-7　SPARC M12-2またはSPARC M12-2S（1CPU搭載）のCPU配置イメージ

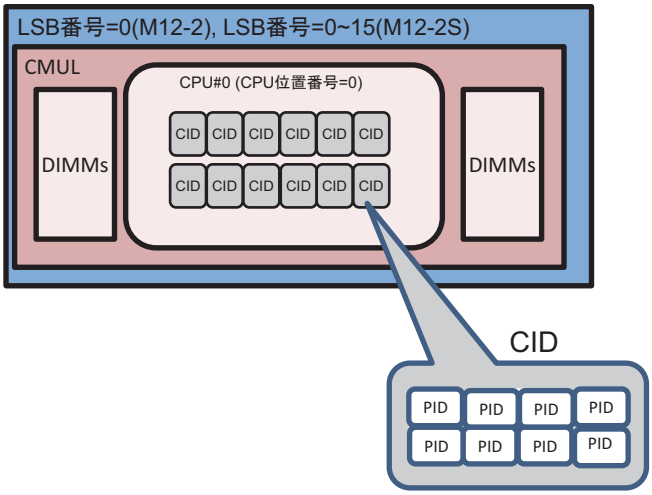


図 2-8　SPARC M12-2またはSPARC M12-2S（2CPU搭載）のCPU配置イメージ

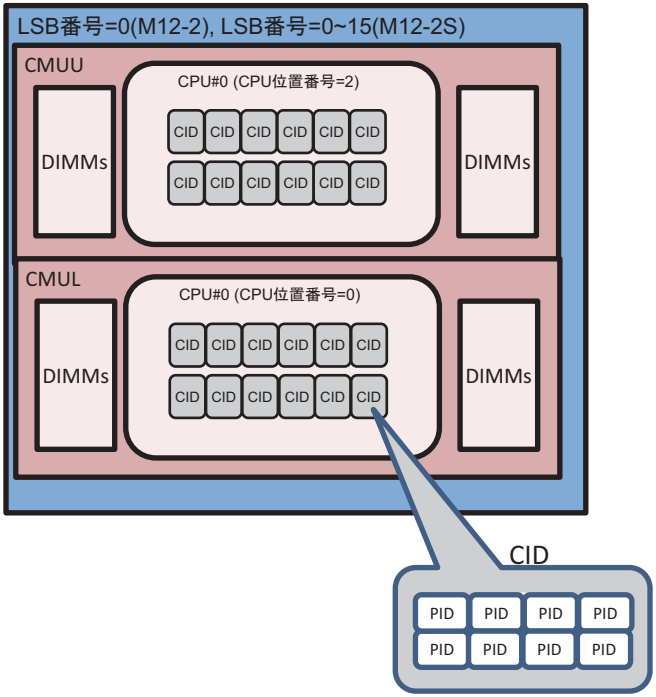


図 2-9　SPARC M10-1のCPU配置イメージ

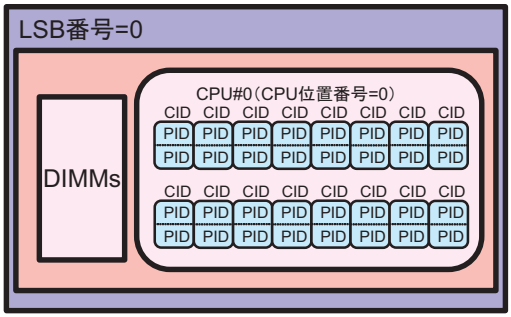


図 2-10　SPARC M10-4またはSPARC M10-4S（2CPU搭載）のCPU配置イメージ

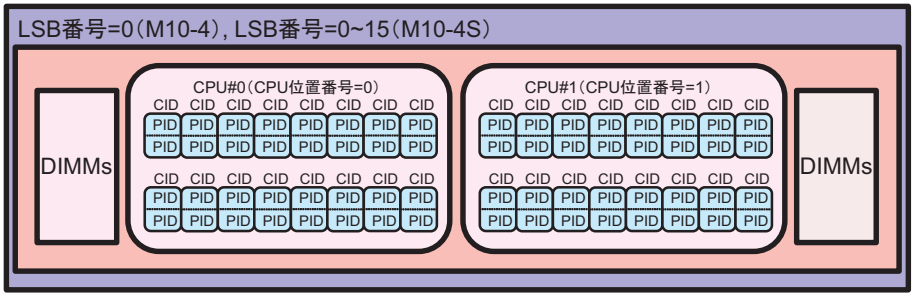
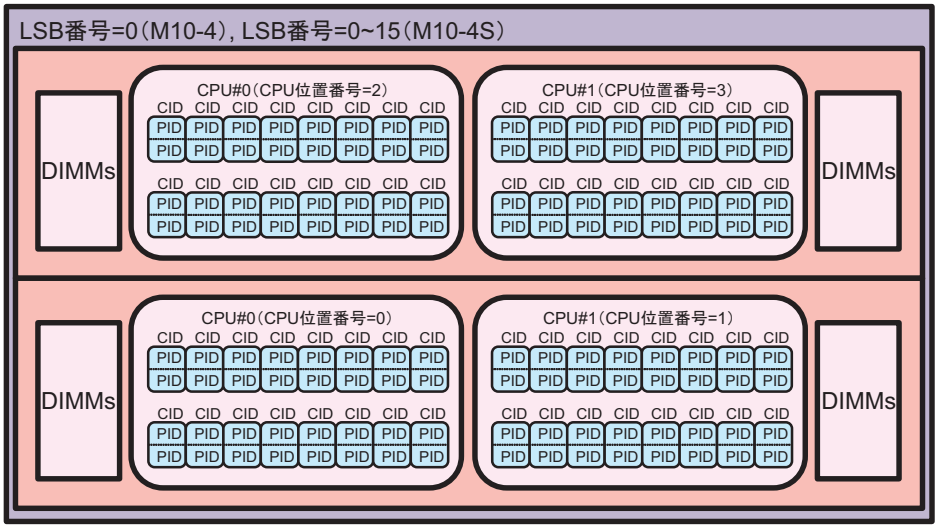


図 2-11　SPARC M10-4またはSPARC M10-4S（4CPU搭載）のCPU配置イメージ



SPARC M12/M10システムの場合、CPUの物理的な位置とPIDおよびCIDは以下の関係があります。

ここでは、CPUの場所を示すパラメーターとして、論理システムボード（LSB）上の

CPU通し番号をCPU位置番号と呼んでいます。

```
PID = LSB番号×1024＋CPU位置番号×256＋CPUコア番号×8＋スレッド番号
CID = PID ÷ スレッド数
```

上記の計算式で使用される各番号の、値の範囲は以下のとおりです。

表 2-5 CPUの物理位置を示す番号の範囲（SPARC M12）

番号の種類	範囲
LSB番号	0～15(*1)
CPU位置番号	0, 2(*2)
CPUコア番号	0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13, 14 (0, 1, 2, 4, 5, 6)(*3)
スレッド番号	0～7
スレッド数	8

*1: SPARC M12-1/M12-2の場合、LSB番号は0で固定です。SPARC M12-2Sの場合、XSCFファームウェアのsetpclコマンドでLSB番号を設定できます。
*2: SPARC M12-1/M12-2およびSPARC M12-2Sで1 CPU搭載時のCPU位置番号は0です。2 CPU搭載時のCPU位置番号は、CMULのCPU#0が0、CMUUのCPU#0が2になります。
*3: カッコ内の数値は、SPARC M12-1のCPUコア番号です。

表 2-6 CPUの物理位置を示す番号の範囲（SPARC M10）

番号の種類	範囲
LSB番号	0～15(*1)
CPU位置番号	0～3(*2)
CPUコア番号	0～15(0～7)(*3)
スレッド番号	0～1
スレッド数	2

*1: SPARC M10-1およびM10-4の場合、LSB番号は0で固定です。SPARC M10-4Sの場合、XSCFファームウェアのsetpclコマンドでLSB番号を設定できます。
*2: SPARC M10-4およびM10-4Sで2 CPU搭載時のCPU位置番号は0～1です。4 CPU搭載時のCPU位置番号は、CMULのCPU#0、#1がそれぞれ0と1、CMUUのCPU#0、#1がそれぞれ2と3になります。
*3: カッコ内の数値は、SPARC64 X+プロセッサ 3.7 GHz（8コア）が搭載されたSPARC M10-1/M10-4の場合のCPUコア番号です。

したがって、SPARC M12システムの各モデルのCPUコアIDと物理位置の関係は、表 2-7のとおりです。
SPARC M10システムの各モデルのCPUコアIDと物理位置の関係は、表 2-8のとおりです。

表 2-7　各モデルのCPUコアID一覧（SPARC M12）

モデル	LSB番号	CPU位置番号	PID	CID
SPARC M12-1	0	0	0～55	0～6
SPARC M12-2	0	0	0～119	0～14
		2	512～631	64～78
SPARC M12-2S	0	0	0～119	0～14
		2	512～631	64～78
	1	0	1024～1143	128～142
		2	1536～1655	192～206
	2	0	2048～2167	256～270
		2	2560～2679	320～334
	3	0	3072～3191	384～398
		2	3584～3703	448～462
	4	0	4096～4215	512～526
		2	4608～4727	576～590
	5	0	5120～5239	640～654
		2	5632～5751	704～718
	6	0	6144～6263	768～782
		2	6656～6775	832～846
	7	0	7168～7287	896～910
		2	7680～7799	960～974
	8	0	8192～8311	1024～1038
		2	8704～8823	1088～1102
	9	0	9216～9335	1152～1166
		2	9728～9847	1216～1230
	10	0	10240～10359	1280～1294
		2	10752～10871	1344～1358
	11	0	11264～11383	1408～1422
		2	11776～11895	1472～1486
	12	0	12288～12407	1536～1550
		2	12800～12919	1600～1614
	13	0	13312～13431	1664～1678
		2	13824～13943	1728～1742
	14	0	14336～14455	1792～1806
		2	14848～14967	1856～1870
	15	0	15360～15479	1920～1934

表 2-7　各モデルのCPUコアID一覧（SPARC M12）(続き)

モデル	LSB番号	CPU位置番号	PID	CID
		2	15872～15991	1984～1998

表 2-8　各モデルのCPUコアID一覧（SPARC M10）

モデル	LSB番号	CPU位置番号	PID	CID
SPARC M10-1	0	0	0～121(0～57)(*1)	0～60(0～28)(*1)
SPARC M10-4	0	0	0～121(0～57)(*1)	0～60(0～28)(*1)
		1	256～377(256～313)(*1)	128～188(128～156)(*1)
		2	512～633(512～569)(*1)	256～316(256～284)(*1)
		3	768～889(768～825)(*1)	384～444(384～412)(*1)
SPARC M10-4S	0	0	0～121	0～60
		1	256～377	128～188
		2	512～633	256～316
		3	768～889	384～444
SPARC M10-4S	1	0	1024～1145	512～572
		1	1280～1401	640～700
		2	1536～1657	768～828
		3	1792～1913	896～956
SPARC M10-4S	2	0	2048～2169	1024～1084
		1	2304～2425	1152～1212
		2	2560～2681	1280～1340
		3	2816～2937	1408～1468
SPARC M10-4S	3	0	3072～3193	1536～1596
		1	3328～3449	1664～1724
		2	3584～3705	1792～1852
		3	3840～3961	1920～1980
SPARC M10-4S	4	0	4096～4217	2048～2108
		1	4352～4473	2176～2236
		2	4608～4729	2304～2364
		3	4864～4985	2432～2492
SPARC M10-4S	5	0	5120～5241	2560～2620
		1	5376～5497	2688～2748
		2	5632～5753	2816～2876
		3	5888～6009	2944～3004
SPARC M10-4S	6	0	6144～6265	3072～3132
		1	6400～6521	3200～3260
		2	6656～6777	3328～3388
		3	6912～7033	3456～3516
SPARC M10-4S	7	0	7168～7289	3584～3644
		1	7424～7545	3712～3772
		2	7680～7801	3840～3900
		3	7936～8057	3968～4028

表 2-8　各モデルのCPUコアID一覧（SPARC M10）(続き)

モデル	LSB番号	CPU位置番号	PID	CID
SPARC M10-4S	8	0	8192～8313	4096～4156
		1	8448～8569	4224～4284
		2	8704～8825	4352～4412
		3	8960～9081	4480～4540
SPARC M10-4S	9	0	9216～9337	4608～4668
		1	9472～9593	4736～4796
		2	9728～9849	4864～4924
		3	9984～10105	4992～5052
SPARC M10-4S	10	0	10240～10361	5120～5180
		1	10496～10617	5248～5308
		2	10752～10873	5376～5436
		3	11008～11129	5504～5564
SPARC M10-4S	11	0	11264～11385	5632～5692
		1	11520～11641	5760～5820
		2	11776～11897	5888～5948
		3	12032～12153	6016～6076
SPARC M10-4S	12	0	12288～12409	6144～6204
		1	12544～12665	6272～6332
		2	12800～12921	6400～6460
		3	13056～13177	6528～6588
SPARC M10-4S	13	0	13312～13433	6656～6716
		1	13568～13689	6784～6844
		2	13824～13945	6912～6972
		3	14080～14201	7040～7100
SPARC M10-4S	14	0	14336～14457	7168～7228
		1	14592～14713	7296～7356
		2	14848～14969	7424～7484
		3	15104～15225	7552～7612
SPARC M10-4S	15	0	15360～15481	7680～7740
		1	15616～15737	7808～7868
		2	15872～15993	7936～7996
		3	16128～16249	8064～8124

*1:　カッコ内の数値は、SPARC64 X+プロセッサ 3.7 GHz（8コア）が搭載されたSPARC M10-1/M10-4の場合のPIDおよびCIDです。

論理ドメインに割り当てられていないCPUのCIDおよびPIDを確認するには、ldm list-devices coreコマンドを使用します。
次の例ではldm list-devices coreコマンドを実行しています。
[ID]はCPUコアID（CID）、[CPUSET]はCPUコアIDに対応する物理CPU番号(PID)を表します。

```
# ldm list-devices core
CORE
    ID      %FREE   CPUSET
    920     100    (1840, 1841)
    924     100    (1848, 1849)
```

```
    936     100    (1872, 1873)
    940     100    (1880, 1881)
    944     100    (1888, 1889)
    948     100    (1896, 1897)
    CID                PID
```

論理ドメインに割り当てられているCPUのCIDとPIDを確認するには、ldm list-domain -o coreコマンドを使用します。
以下の例では、 制御ドメインに割り当てられているCPUコアのCIDとPIDを表示しています。

```
# ldm list-domain -o core primary
NAME
primary

CORE
    CID    CPUSET
    0      (0, 1)
    4      (8, 9)
    8      (16, 17)
--- 中略 ---
    896    (1792, 1793)
    900    (1800, 1801)
    904    (1808, 1809)
    908    (1816, 1817)
    912    (1824, 1825)
    916    (1832, 1833)
    CID         PID
```

論理ドメインに対するメモリ割り当ての目安

論理ドメインには、4 GB以上の仮想メモリを割り当ててください。

論理ドメインのメモリサイズ

論理ドメインのメモリを動的再構成で変更しない場合、論理ドメインに割り当てるメモリサイズは、4 MB単位で設定できます。
論理ドメインのメモリを動的再構成で変更する必要がある場合、論理ドメインに割り当てるメモリサイズは、必ず256 MB単位にしてください。

論理ドメイン構成情報の保存

論理ドメインを構築したあとには、論理ドメイン構成情報をXSCFに保存します。また、論理ドメインの構築に関する作業の前にも、論理ドメインの構成情報を保存します。作業前に構成情報を保存しておくことで、構築前の状態に確実に戻せるようにしておきます。

論理ドメインの構成情報は、ハードウェア障害などによって失われる可能性があります。そのため、ゲストドメインの構成情報を変更したときには、Oracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用してゲストドメインの構成情報を取得し、安全な場所に保存するようにしてください。取得した構成情報は、テープ装置やファイルサーバに複製し、ディスクからの消

失にも備えてください。また、ldmコマンドを使用することで、取得した構成情報を元にゲストドメインを再構築することもできます。詳細は、お使いのバージョンの『Oracle VM Server for SPARC 管理ガイド』および『Oracle VM Server for SPARC リファレンスマニュアル』を参照してください。

XSCFに保存した構成情報はバックアップできます。構成情報をファイルへバックアップするときには、XSCFファームウェアのdumpconfigコマンドを使用します。構成情報をバックアップファイルから復元するときには、restoreconfigコマンドを使用します。

論理ドメイン構築時の指定項目

論理ドメインを構築するときには、各種名称を定義したり、各種リソースやその個数を指定したりする必要があります。次に、論理ドメインを構築時に定義したり指定したりする項目を示します。これらの項目は、作業を開始する前に決定しておいてください。

- 仮想コンソール端末集配信装置名
- 仮想ディスクサーバ名
- 仮想スイッチサービス名
- 仮想ネットワークデバイスのインターフェース名
- 仮想デバイス名
- 仮想ディスク名
- ゲストドメインのドメイン名
- 論理ドメイン構成情報名
- 仮想コンソール端末のポート番号の範囲
- 仮想スイッチサービスで使用するデバイス
- 仮想ディスクサービスで使用するデバイス
- ゲストドメインに割り当てるCPUの数量
- ゲストドメインに割り当てるメモリのサイズ
- ゲストドメインに割り当てる仮想コンソール端末のポート番号

論理ドメインのシャットダウン順序

優先度シャットダウン機能を使用してXSCFのpoweroffコマンドによる各論理ドメインのシャットダウンの順序を指定するときには、作成したゲストドメインに対してシャットダウングループを設定します。シャットダウングループとは、論理ドメインのシャットダウンの順番を指定するためのものです。

シャットダウングループは0から15まであり、数値が大きいものから順にシャットダウンされます。ゲストドメインはデフォルトでグループ15に属します。制御ドメインは常にグループ0に属し、変更できません。論理ドメインの依存関係を考えて設定してください。

2.4.2 論理ドメインの再構成に関する留意点

論理ドメインに割り当てるハードウェアリソースを再構成するときには、次の点に留意する必要があります。

デバイス側の準備

仮想I/Oデバイスのうち仮想ディスクと仮想ネットワークを再構成するときには、あらかじめ次の作業を行っておきます。

- 仮想ディスクをアンマウントする
- 仮想ネットワークのインターフェースをunplumbする

制御ドメインおよびルートドメインの遅延再構成

制御ドメインおよびルートドメインに、物理I/Oを静的に追加または削除する場合、遅延再構成によって操作します。遅延再構成によって制御ドメインおよびルートドメインに割り当てる物理I/Oを再構成しているときには、対象の制御ドメインまたはルートドメインを再起動するまで、その制御ドメインまたはルートドメインに対するその他の再構成要求は延期されます。

遅延再構成の制限

遅延再構成モードを設定して制御ドメインまたはルートドメインに対するハードウェアリソースを再構成しているときには、ゲストドメインに対するハードウェアリソースは再構成できません。制御ドメインまたはルートドメインに対する再構成の作業が完了したあとに、ゲストドメインに対する再構成の作業を行ってください。

遅延再構成の取り消し

遅延再構成による制御ドメインまたはルートドメインへの操作を取り消すときには、ldm cancel-operation reconfコマンドを使用します。ldmコマンドの詳細は、『Oracle VM Server for SPARC リファレンスマニュアル』を参照してください。

論理ドメインの要件

- 物理パーティションの動的再構成を行う場合、必ず制御ドメインのOracle Solarisが起動されている状態にしてください。
- 制御ドメインのシステムボリュームを構成するディスクは、動的再構成によって削除されるシステムボードと削除されないシステムボード上のディスクで、冗長構成にしておく必要があります。冗長構成としなかった場合は、ディスクを搭載したシステムボードを削除するために事前にI/Oデバイスを切り離した時点で、制御ドメインが起動できなくなります。

ハイパーバイザに割り当てられるメモリサイズ

論理ドメインで使用できるメモリは、実際に筐体に搭載されているメモリサイズよりも少なくなります。これは、ハイパーバイザが動作するために使用するメモリが、各筐体上にリザーブされているためです。システム構築時は、このハイパーバイザが使用するメモリを差し引いてドメインの構成を設計してください。ハイパーバイザに割り当てられるメモリは、モデルやビルディングブロック構成、PPAR DRモードの設

定によって異なります。PPAR DRモードについては、「2.5.3　PPAR DRモードの確認／設定方法」を参照してください。

各物理パーティションに割り当てられるハイパーバイザのメモリは以下のとおりです。

表 2-9　ハイパーバイザに割り当てられるメモリサイズ

モデル	PPAR DRモード	
	有効	無効
SPARC M12-1/M12-2	2.5 GB	
SPARC M12-2S	2.5 GB + (1.5 GB x (物理パーティションを構成する筐体数-1))	
SPARC M10-1/M10-4	－	2 GB
SPARC M10-4S	2 GB＋(1.25 GB × (筐体数 - 1))	2 GB＋(1.0 GB × (筐体数 - 1))

CPUコアとメモリ配置

物理パーティションの動的再構成で、システムボードを切り離せるようにするためには、CPUコア数の割り当てと、メモリの配置について、注意が必要です。詳細は、「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。

2.5　動的再構成の条件と設定

2.5.1　構築時の留意点

ここでは、物理パーティションの動的再構成（PPAR DR）機能を利用して、システムを構築する場合の留意点を説明します。動的再構成機能は、SPARC M12-2S/M10-4Sでサポートされます。

XCPファームウェアのアップデート

物理パーティションの動的再構成（PPAR DR）機能は、XCP 2220以降でサポートされます。XCPファームウェアのアップデート方法の詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「第16章　XCPのファームウェアをアップデートする」を参照してください。

PPAR DRモードの確認

物理パーティションの動的再構成（PPAR DR）機能を使用するためには、XSCFファームウェアのsetpparmodeコマンドで設定できるPPAR DRモードが有効でなければなりません。PPAR DRモードの確認方法の詳細は、「2.5.3　PPAR DRモードの確認／設定方法」を参照してください。

デバイスドライバとI/Oデバイスの要件

物理パーティションの動的再構成（PPAR DR）機能を使用する場合は、物理パーティションの動的再構成の対象になるシステムボードに搭載されているすべてのPCIeカードとI/Oデバイスのインターフェースが PPAR DRをサポートしている必要があります。また、インストールされている I/Oデバイスのドライバや二重化ソフトウェアすべてが物理パーティションの動的再構成に対応している必要があります。PPAR DRに対応したPCIeカード、ドライバ、二重化ソフトウェアの詳細については、お使いのサーバの『PCIカード搭載ガイド』の「PCI Hot PlugおよびDynamic Reconfiguration対応カード」を参照してください。

PPAR DRの対象になるシステムボードに搭載されているすべてのPCIeカード、I/OデバイスがPPAR DRをサポートしていない場合は、そのシステムボードについて物理パーティションの動的再構成を実行できません。

ソフトウェアの要件

物理パーティションの動的再構成（PPAR DR）機能を使用すると、一時的に論理ドメインが停止状態（suspend）になることがあります。そのため、クラスタソフトウェアや、システムを監視するソフトウェアなど、タイマーなどを使って異常を検出するようなソフトウェアは、タイムアウト検出時間を変更する、またはタイムアウト監視を一時的に停止するなどの対応が必要となります。

ハイパーバイザに割り当てられるメモリサイズ

論理ドメインで使用できるメモリは、実際に筐体に搭載されているメモリサイズよりも少なくなります。これは、ハイパーバイザが動作するために使用するメモリが、各筐体上にリザーブされているためです。システム構築時は、このハイパーバイザが使用するメモリを差し引いてドメインの構成を設計してください。ハイパーバイザに割り当てられるメモリは、モデルやビルディングブロック構成、PPAR DRモードの設定によって異なります。PPAR DRモードについては、「2.5.3　PPAR DRモードの確認／設定方法」を参照してください。

各物理パーティションに割り当てられるハイパーバイザのメモリは表 2-9を参照してください。

メモリサイズの設定単位

Oracle VM Server for SPARCのバージョンが3.2よりも古い場合、物理パーティションの動的再構成が行えるようにするために、論理ドメインに割り当てるメモリサイズを、「論理ドメインに割り当てるコア数×256 MB」の倍数に設定してください。設定方法の例は、「付録 A　物理パーティションの動的再構成を使用した環境構築・手順例」を参照してください。

CPUコアとメモリの配置

物理パーティションの動的再構成で、SPARC M12-2S/M10-4Sを切り離せるようにするためには、CPUコア数の割り当てと、メモリの配置について、注意が必要です。詳細は、「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。

時刻同期

物理パーティションの動的再構成を実施した場合、論理ドメインのシステム時刻が遅れることがあります。物理パーティションの動的再構成（PPAR DR）機能を使用する場合は、事前に、NTP（Network Time Protocol）による時刻同期の設定を行ってください。
Oracle SolarisのNTPサーバの設定は、『Oracle Solarisのシステム管理（ネットワークサービス）』（Oracle Solaris 10）または『Oracle Solaris 11ネットワークサービスの紹介』（Oracle Solaris 11）を参照してください。

PCIeルートコンプレックスの動的追加／削除について

SPARC M10の場合は、XCP 2240以降で、かつOracle VM Server for SPARC 3.2以降では、ルートドメインのOracle Solarisを停止せずにPCIeルートコンプレックスを追加／削除できます。これは、Oracle Solaris 11.2 SRU11.2.8.4.0以降がインストールされたルートドメインのみ追加／削除できます。
したがって、Oracle Solaris 10の制御ドメインだけで構成されている場合にPCIeルートコンプレックスを追加／削除するときは、遅延再構成を行ってからOracle Solarisを再起動する必要があるため、物理パーティションの動的再構成はできません。
本機能の詳細については、Oracle VM Server for SPARC 3.2以降の『Oracle VM Server for SPARC 管理ガイド』の「動的PCIeバスの割り当て」を参照してください。

物理パーティションの動的再構成を使用したシステムを構築するために、より具体的な考慮すべき事項、およびベストプラクティスを記載したドキュメントを富士通Technical Parkで公開しています。
https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/technical/document/index.html
『SPARC M12/M10 Building Block を活用した高可用性システム (運用編)』の「付録. A PPAR DRの概要」および「付録. B PPAR DR deleteboardのベストプラクティス」を参照してください。

2.5.2 運用時の留意点

ここでは、動的再構成のためのシステムを運用する際の留意点を説明します。

Oracle Solaris 11.3以降がインストールされている論理ドメインおよびカーネルゾーンについて

- SPARC M10-4Sシステムでは、制御ドメインに Oracle Solaris 11.3がインストールされている場合は、あらかじめ、制御ドメインの/etc/systemファイルに以下の行を追加したあと再起動してください。ただし、XCPファームウェアがXCP 2260以降の場合は不要です。

```
set enable_user_tick_stick_emulation = 0
```

- SPARC M12-2S/M10-4Sで構成された物理パーティション内で、Oracle Solaris 11.3以降がインストールされている論理ドメイン（制御ドメインを除く）およびOracle Solarisカーネルゾーンがある場合は、あらかじめ、該当する論理ドメインおよびOracle Solarisカーネルゾーンの/etc/systemファイルに以下の行を追加した

あと再起動してください。

```
set uhrt_enable = 0x0
```

deleteboardコマンド実行時のOracle Solarisの停止時間

deleteboardコマンドを実行すると、論理ドメインのOracle Solarisが一時的に停止状態になる（サスペンドする）ことがあります。この間、物理デバイスの通信やI/Oが停止したり、Oracle Solaris上のアプリケーションの動作も停止したりするため、リモート装置とのネットワーク切断など業務プロセスに影響を与える可能性があります。したがって、各論理ドメインが停止状態になる時間（サスペンド時間）について、あらかじめ認識して運用などを決定する必要があります。
サスペンド時間の最大値は、メモリの移動によるサスペンド／レジューム処理時間と各I/Oデバイスのサスペンド／レジュームの処理時間を合計した時間になります。
そのため、deleteboardコマンドを実行する筐体に搭載されているメモリと、物理パーティションに搭載されている物理I/Oデバイスの種類／数を元に、概算値を計算できます。
計算式は以下のとおりです。

サスペンド時間＝メモリ移動時間＋オンボードデバイスのサスペンド／レジューム時間の合計＋PCIカードのサスペンド／レジューム時間の合計

メモリの移動によるサスペンド時間は、1 TBあたり 168秒です。deleteboardコマンドの対象となる筐体の搭載メモリをshowpparinfoコマンドで確認して、計算してください。
以下の出力例では、システムボード(BB#01)を切り離す場合、Memory: に表示されるPSBが01-0のinstall sizeの値が128 GBになるため、サスペンド時間は21秒になります。

128（GB）×168（秒）／1024（GB）＝21（秒）

```
XSCF> showpparinfo -p 0
PPAR#00 Information:
--------------------
....
Memory:
-------
install
PID PSB size GB
00 00-0 128
00 01-0 128
IO Devices:
-----------
PID PSB device
....
```

物理I/Oのサスペンド時間は、パーティションに搭載されているすべての物理I/Oデバイスの種類と数から計算します。オンボードデバイスのサスペンド／レジューム処理時間は、以下のとおりです。

- SPARC M12の場合
  1筐体あたり、5.2秒
  例）2BB構成の場合は、10.4秒

- SPARC M10の場合
  1筐体あたり、21.4秒
  例）2BB構成の場合は、42.8秒

PCIカードのサスペンド／レジューム処理時間は、お使いのサーバの『PCIカード搭載ガイド』の「PCI Hot PlugおよびDynamic Reconfiguration対応カード」を参照してください。

ライブマイグレーションとの組み合わせ

ライブマイグレーションと物理パーティションの動的再構成を同時に実施しないでください。

CPUコアとメモリの配置

CPUまたはメモリの構成が下記に示す条件を満たしていない場合、deleteboardコマンドを実行すると、以下のメッセージが表示されることがあります。各条件を参照して、対処してください。

```
Some domain will be reduced with DR operation. But reducing
resource is not allowed.
```

- 制御ドメインのみの構成の場合
  論理ドメイン構成情報がfactory-defaultの場合など、制御ドメインだけで構成されていて、かつ複数のシステムボードのCPUコアやメモリ領域が制御ドメインに割り当てられている場合、deleteboardコマンドで物理パーティションの動的再構成によるシステムボードの削除を実行すると、上記のメッセージが出力されてエラー終了します。

  これを防ぐための対処は以下のとおりです。

  - deleteboardコマンドで-m unbind=resourceオプションを指定して、動的にCPUコアやメモリ領域が削除されるようにする。
  - 事前にldm remove-coreコマンドやldm remove-memoryコマンド でCPUやメモリを削除して、削除するBB(PSB)の分だけ、あらかじめリソースを空けておく。

  制御ドメインが使用しているCPUコア、メモリを減らすことなく PPAR DRを実行するためには、ドメインに割り当てられているCPUコア数やメモリ領域のサイズをあらかじめ空けておく構成にする必要があります。

  なお、deleteboardコマンドの-m unbind=resourceオプションはOracle VM Server for SPARC 3.2からサポートされます。また、PPAR DRポリシーはOracle VM Server for SPARC 3.4からサポートされます。ポリシーの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「8.15　物理パーティションの動的再構成ポリシーを設定する」を参照してください。

- 制御ドメイン以外に論理ドメインを構成している場合

  ［CPUコア］

  論理ドメインに割り当てられているCPUコア数が、deleteboardコマンドで削除されたあとに残るCPUコア数よりも多い場合、割り当てられたCPUコアの移動先となる未割り当てのCPUコア数が足りなくなるため、deleteboardコマンドが失敗し

ます。

これを防ぐための対処は以下のとおりです。

- deleteboardコマンドの実行により削除されるCPUコア数分の未割り当てのCPUコアを残した状態で、あらかじめ論理ドメインを構成しておく。
- deleteboardコマンドで-m unbind=resourceオプションを指定して、動的にCPUコアやメモリ領域が削除されるようにする。
- 事前に ldm remove-coreコマンドやldm remove-memoryコマンド でCPUコアやメモリを削除して、削除するシステムボードの分だけ、あらかじめリソースを空けておく。

なお、deleteboardコマンドの-m unbind=resourceオプションはOracle VM Server for SPARC 3.2からサポートされます。また、PPAR DRポリシーはOracle VM Server for SPARC 3.4からサポートされます。ポリシーの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「8.15　物理パーティションの動的再構成ポリシーを設定する」を参照してください。

CPUコア数の割り当て状況の確認方法は以下のとおりです。

1. **各論理ドメインに割り当てられているCPUコアの総数を確認します。**
   制御ドメインでldm list-devices -a coreを実行して、「%FREE」の欄が「100」以外になっているCPUコアの数の合計です。次の例では、ldm list-devices -a -p coreコマンドを使って、論理ドメインに割り当てられているコア数の総数を表示するコマンドの実行例を示しています。

```
# ldm list-devices -a core
CORE
    ID       %FREE   CPUSET
    0        0       (0, 1)
    4        0       (8, 9)
    8        0       (16, 17)
    12       0       (24, 25)
（以下省略）
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
     112
```

2. **削除されないシステムボードのCPUコアの総数を確認します。**
   XSCFでshowpparinfoコマンドを実行して、削除されないPSB番号の"Cores"の数の合計です。次はSPARC M10-4Sの場合の、showpparinfoコマンドの実行例です。

```
XSCF>  showpparinfo  -p  0
PPAR#00  Information:
--------------------
CPU(s)                                :          8
CPU  Cores                            :        128
CPU  Threads                          :        256
Memory  size  (GB)                    :        256
CoD  Assigned  (Cores)                :        256
CPU(s):
```

```
-------
PID    PSB     CPU#    Cores    Threads
00      00-0    0              16            32
00      00-0    1              16            32
00      00-0    2              16            32
00      00-0    3              16            32
00      01-0    0              16            32
00      01-0    1              16            32
00      01-0    2              16            32
00      01-0    3              16            32
（以下省略）
```

以下の計算式から、SPARC M12-2S/M10-4S筐体切り離し後に不足するCPUコア数を計算します。

不足するCPUコア数＝論理ドメインで使用するコア数(1)－切り離し後の物理コア数(2)

コア数が不足する場合は、ldm remove-coreコマンドで論理ドメインに割り当てられたCPUコアを削除して、コア数を減らす必要があります。

［メモリ］

物理パーティションの動的再構成でdeleteboardコマンドを実行した場合、削除されるシステムボード上のメモリ領域が論理ドメインに割り当てられていると、その論理ドメインに割り当てられたメモリ領域の内容を、削除されないシステムボード上のメモリ領域に割り当てなおして内容を移動する動作をします。

したがって、移動するメモリのサイズよりも大きな空き領域が、移動先（削除されないシステムボード上）の未使用のメモリ領域として存在しない場合、deleteboardコマンドがエラー終了します。

これを防ぐための対処は以下のとおりです。

- deleteboardコマンドで-m unbind=resourceオプションを指定して、動的にCPUコアやメモリ領域が削除されるようにする。
- 事前にldm remove-memoryコマンドでメモリを削除して、削除するシステムボードの分だけ、あらかじめリソースを空けておく。

なお、deleteboardコマンドの-m unbind=resourceオプションはOracle VM Server for SPARC 3.2からサポートされます。また、PPAR DRポリシーはOracle VM Server for SPARC 3.4からサポートされます。ポリシーの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「8.15　物理パーティションの動的再構成ポリシーを設定する」を参照してください。

メモリ領域の使用状況の確認方法は以下のとおりです。

1. **メモリの連続領域（メモリブロック）の使用状況を確認します。**

prtdiagコマンドを実行し、メモリの物理アドレスとビルディングブロック構成のSPARC M10-4Sの対応を確認します。

```
# prtdiag
（略）
======================== Physical Memory Configuration ========================
Segment Table:
--------------------------------------------------------------
Base Segment Interleave Bank Contains
Address Size Factor Size Modules
--------------------------------------------------------------
0x7e0000000000 32 GB 4 8 GB /BB0/CMUL/CMP0/MEM00A
（略）
0x7c0000000000 32 GB 4 8 GB /BB0/CMUL/CMP1/MEM10A
（略）
0x7a0000000000 32 GB 4 8 GB /BB0/CMUU/CMP0/MEM00A
（略）
0x780000000000 32 GB 4 8 GB /BB0/CMUU/CMP1/MEM10A
（略）
0x760000000000 32 GB 4 8 GB /BB1/CMUL/CMP0/MEM00A
（略）
0x740000000000 32 GB 4 8 GB /BB1/CMUL/CMP1/MEM10A
（略）
0x720000000000 32 GB 4 8 GB /BB1/CMUU/CMP0/MEM00A
（略）
0x700000000000 32 GB 4 8 GB /BB1/CMUU/CMP1/MEM10A
（以下省略）
```

この例の結果を、メモリの物理アドレスを小さい順に並べ直して、物理アドレスとSPARC M10-4Sの対応を表にすると以下のようになります。

表 2-10 物理アドレスとSPARC M10-4Sの対応例

Base Address（物理アドレス）	SPARC M10-4S
0x700000000000～	ビルディングブロックBB-ID#01
0x720000000000～	ビルディングブロックBB-ID#01
0x740000000000～	ビルディングブロックBB-ID#01
0x760000000000～	ビルディングブロックBB-ID#01
0x780000000000～	ビルディングブロックBB-ID#00
0x7a0000000000～	ビルディングブロックBB-ID#00
0x7c0000000000～	ビルディングブロックBB-ID#00
0x7e0000000000～	ビルディングブロックBB-ID#00

次に、制御ドメインでldm list-devices -a memoryコマンドを実行し、各論理ドメインに割り当てられているメモリ領域と未使用のメモリ領域を表示します。

```
# ldm list-devices -a memory
MEMORY
PA SIZE BOUND
0x700000000000 24G root-dom1
0x700600000000 8G
0x720000000000 32G guest0
0x740000000000 32G guest1
0x760000800000 1272M _sys_
0x760050000000 24G root-dom0
0x760650000000 6912M
0x780000000000 32G
0x7a0000000000 32G
0x7c0000000000 32G
0x7e0000800000 1272M _sys_
0x7e0050000000 512M _sys_
0x7e0070000000 256M _sys_
0x7e0080000000 14G primary
0x7e0400000000 16G
```

上記の結果と、「表 2-10　物理アドレスとSPARC M10-4Sの対応例」の物理位置を合わせると、以下のようなメモリブロックの使用状況であることがわかります。

表 2-11　メモリブロックの使用状況の例

SPARC M10-4S	物理アドレス	サイズ	論理ドメイン
ビルディングブロックBB-ID#01（交換対象）	0x700000000000	24 GB	root-dom1
	0x700600000000	8 GB	未割り当て
	0x720000000000	32 GB	guest0
	0x740000000000	32 GB	guest1
	0x760050000000	24 GB	root-dom0
	0x760650000000	6912 MB	未割り当て
ビルディングブロックBB-ID#00	0x780000000000	32 GB	未割り当て
	0x7a0000000000	32 GB	未割り当て
	0x7c0000000000	32 GB	未割り当て
	0x7e0080000000	14 GB	primary
	0x7e0400000000	16 GB	未割り当て

2. **移動元のメモリブロックのサイズと個数を確認します。**

   メモリブロックの使用状況の確認結果を参照し、交換対象のSPARC M10-4S上で割り当てられているメモリブロック（以降、移動元のメモリブロックと呼びます）を確認します。

   「表 2-11　メモリブロックの使用状況の例」の場合、ビルディングブロックBB-ID#01側で論理ドメインに割り当てられているメモリブロックとして、32 GB×2個（guest0、guest1に割り当て）、24 GB×1個（root-dom0）であることがわかります。

注－ ビルディングブロックBB-ID#01を切り離す前にビルディングブロックBB-ID#01のI/Oを割り当てているルートドメインroot-dom1を、アンバインドしてinactive状態にする場合は、root-dom1を移動対象から除外できます。

3. **空きメモリブロックを確認します。**
   次に、手順1で確認した結果から、切り離されないSPARC M10-4S上の、論理ドメインに割り当てられていないメモリブロック（以降、空きメモリブロックと呼びます）を確認します。
   「表 2-11　メモリブロックの使用状況の例」の例の場合、32 GB×3個、16 GB×1個が空きメモリブロックであることがわかります。
4. **メモリブロックが移動可能かどうかを確認します。**
   手順2と手順3で確認した結果より、移動元のメモリブロックが空きメモリブロックに移動できるかどうかを確認します。
   移動元のメモリブロックと同じかそれ以上のサイズの空きメモリブロックがあれば、移動可能です。

移動先に空きリソースがある場合

例えば、「表 2-11　メモリブロックの使用状況の例」の場合、guest0（32 GB）、guest1（32 GB）、root-dom0（24 GB）の移動先として、32 GB×3個の空きメモリブロックがあるため、ビルディングブロックBB-ID#01を切り離せるメモリ配置であることがわかります。表にすると、「表 2-12　メモリブロックの移動先候補」のようになります。

表 2-12　メモリブロックの移動先候補

SPARC M10-4S	サイズ	論理ドメイン	移動先候補
ビルディングブロックBB-ID#01（交換対象）	24 GB	root-dom1	－
	8 GB	未割り当て	－
	32 GB	guest0	ビルディングブロックBB-ID#00側の32 GB
	32 GB	guest1	ビルディングブロックBB-ID#00側の32 GB
	24 GB	root-dom0	ビルディングブロックBB-ID#00側の32 GB
	6912 MB	未割り当て	－
ビルディングブロックBB-ID#00	32 GB	未割り当て	ここに移動
	32 GB	未割り当て	ここに移動
	32 GB	未割り当て	ここに移動
	14 GB	primary	－
	16 GB	未割り当て	サイズ不足で移動先の候補にならない

移動先に空きリソースがない場合

例えば、次の「表 2-13　移動先がないメモリブロックの配置例」の構成の場合、

移動元のメモリブロックは、32 GB×2個と、24 GB×2個になります。いっぽう、移動先の空きメモリブロックは、32 GB×3個と、16 GB×1個です。

その結果、32 GBのメモリブロック1個（guest0）と、24 GBのメモリブロック2個（guest1、guest2、root-dom0のいずれか2つ）は移動可能です。

しかし、残りの移動先の空きメモリブロックは、16 GB×1個と、32 GBのメモリブロックに24 GBのメモリブロックが移動したあとに残る8 GB×2個になるため、24 GBのメモリブロックを割り当てられているguest1、guest2、root-dom0のどれか1つが移動できなくなります。 この場合、移動できない可能性のある論理ドメインのメモリブロックを、移動先のメモリブロックのサイズに合わせて減らしておく必要があります。

上記の例の場合、ldm remove-memoryコマンドを実行し、論理ドメインguest1、guest2、root-dom0のうちどれか1つのメモリを24 GBから16 GB以下に変更する必要があります。

表 2-13 移動先がないメモリブロックの配置例

SPARC M10-4S	サイズ	論理ドメイン	移動先候補
ビルディングブロックBB-ID#01（交換対象）	24 GB	guest2	移動できない可能性あり
	8 GB	root-dom1	―
	32 GB	guest0	ビルディングブロックBB-ID#00側の32 GB
	32 GB	guest1	移動できない可能性あり
	24 GB	root-dom0	移動できない可能性あり
	6912 MB	未割り当て	―
ビルディングブロックBB-ID#00	32 GB	未割り当て	ここに移動
	32 GB	未割り当て	guest1、guest2、root-dom0のどれか（24 GB）がここに移動して、8 GB残る
	32 GB	未割り当て	guest1、guest2、root-dom0のどれか（24 GB）がここに移動して、8 GB残る
	14 GB	primary	―
	16 GB	未割り当て	サイズ不足で移動先の候補にならない

復旧モード（Recovery Mode）有効時の動的再構成の操作について

- 縮退構成でドメイン構成が復旧された状態で、物理パーティションの動的再構成を使用してシステムボードを追加した場合、追加されたリソースはどの論理ドメインにも自動的には割り当てられません。追加されたリソースを手動で割り当て

るか、ldm set-spconfigコマンドで元のドメイン構成を選択したあと、XSCFファームウェアのpoweronコマンドおよびpoweroffコマンドを使用して物理パーティションを再起動してください。

- Oracle VM Server for SPARCのバージョンが3.2よりも古い場合、縮退構成でドメイン構成が復旧されたあと、物理パーティション（PPAR）の電源が投入されている状態で、deleteboardコマンドを使用してシステムボード（PSB）を削除すると、deleteboardコマンドが失敗することがあります。縮退構成でドメイン構成が復旧されたあとは、物理パーティションの動的再構成を使用してシステムボードを削除しないでください。

Oracle Solarisカーネルゾーンとの組み合わせ

物理パーティション（PPAR）内のいずれかの論理ドメインでOracle Solarisカーネルゾーンが動作している場合は、物理パーティションの動的再構成は実行できません。Oracle Solarisカーネルゾーンを停止してから、物理パーティションの動的再構成を実行してください。

仮想サービスを割り当てている論理ドメインについて

仮想サービスを割り当てている論理ドメインが以下のいずれかに該当する場合、事前に物理I/Oに割り当てた仮想ディスクサーバデバイス（vdsdev）、仮想ネットワークスイッチ（vsw）、および、それらに割り当てた仮想ディスク（vdisk）、仮想ネットワーク（vnet）を、それぞれldm remove-vdsdev、ldm remove-vsw、ldm remove-vdisk、ldm remove-vnetコマンドを使用して削除しておく必要があります。

- ldm remove-ioコマンドで動的にPCIeエンドポイントを削除する場合
- 物理パーティションの動的再構成で論理ドメインの物理I/Oが動的に削除される場合

PPAR DR操作 に伴うUSBデバイスパスの変更

物理パーティションの動的再構成機能によりシステムボード（PSB<BB>）を削除したあと、物理パーティションの動的再構成機能によりPSB（BB）を追加した場合、削除／追加されたPSB（BB）のUSBデバイスパスが、以下の例のとおり「usb@4,1」の部分が「usb@4」に変更されます。

- 外付けDVDドライブ（front）の場合の例
  削除前: /pci@8000/pci@4/pci@0/pci@1/pci@0/usb@4,1/hub@2/cdrom@1/disk@0,0:a
  追加後: /pci@8000/pci@4/pci@0/pci@1/pci@0/*usb*@4/hub@2/cdrom@1/disk@0,0:a
- リモートストレージ用DVDドライブの場合の例
  削除前: /pci@8000/pci@4/pci@0/pci@1/pci@0/usb@4,1/storage@3/disk@0,0:a
  追加後: /pci@8000/pci@4/pci@0/pci@1/pci@0/usb@4/storage@3/disk@0,0:a

ただし、デバイスパスが変更されたUSBデバイスが割り当てられている論理ドメインを再起動すれば、削除前のデバイスパスに戻ります。
この現象は、物理パーティションの動的再構成機能で削除／追加を行わないPSB（BB）にUSBデバイスを接続している場合は発生しません。

物理パーティションの動的再構成を使用したシステムを構築するために、より具体的な考慮すべき事項、およびベストプラクティスを記載したドキュメントを 富士通Technical Parkで公開しています。
https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/technical/document/index.html
『SPARC M12/M10 Building Block を活用した高可用性システム (運用編)』の「付録. A PPAR DRの概要」および「付録. B PPAR DR deleteboardのベストプラクティ

ス」を参照してください。

2.5.3 PPAR DRモードの確認／設定方法

ここでは、PPAR DRモードの確認方法、および設定方法について説明します。

重要－PPAR DRモードの設定を変更する前に、必ず、論理ドメイン構成情報を退避してください。
setpparmodeコマンドを実行して、PPAR DRモードの設定を無効から有効、または有効から無効に変更する場合、設定を反映するには、物理パーティションをリセットする必要があります。このリセットにより、論理ドメインの構成情報はfactory-defaultに戻ります。論理ドメインの再構築には、事前に退避した論理ドメインの構成情報が必要となるため、PPAR DRモードの設定を変更する前に、必ずデータを退避してください。論理ドメインの構成情報の退避および再構築方法については、「A.1　物理パーティションの動的再構成が未サポートのシステムからサポートされるシステムへアップデートする場合」の構築の例を参照してください。

PPAR DRモードの設定の手順は次のとおりです。

1. **論理ドメインの構成情報を退避します。**
   次の例では、論理ドメインの構成情報をfile.xmlというXMLファイルに退避しています。

注－論理ドメインが未構成の場合は、この手順は不要です

```
# ldm list-constraints -x > file.xml
```

2. **showpparmodeコマンドを実行し、PPAR DRモードの設定を確認します。**
   表示されるPPAR DR（Current）は、現在のPPAR DRモードの値、PPAR DR（Next）は、次回、物理パーティションが起動されたあとに設定されるPPAR DRモードの値を示します。

   次の例では、物理パーティションの電源が切断されていて、次回、物理パーティションが起動されたときには、PPAR DR機能が無効に設定されます。

```
XSCF> showpparmode -p PPAR_ID
Host-ID                 :00000001
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal            :off
Autoboot(Guest Domain)  :on
Elastic Mode            :off
IOreconfigure           :false
PPAR DR(Current)        :-
PPAR DR(Next)           :off
```

3. **setpparmodeコマンドを実行し、PPAR DRモードを設定します。**

次の例では、PPAR DR機能を有効に設定しています。

```
XSCF> setpparmode -p PPAR_ID -m ppar_dr=on
Diagnostic Level        :max        -> -
Message Level           :normal     -> -
Alive Check             :on         -> -
Watchdog Reaction       :reset      -> -
Break Signal            :on         -> -
Autoboot(Guest Domain)  :on         -> -
Elastic Mode            :off        -> -
IOreconfigure           :true       -> -
PPAR DR                 :off        -> on
The specified modes will be changed.
Continue? [y|n] :y
configured.
Diagnostic Level        :max
Message Level           :normal
Alive Check             :on (alive check:available)
Watchdog Reaction       :reset (watchdog reaction:reset)
Break Signal            :on (break signal:non-send)
Autoboot(Guest Domain)  :on
Elastic Mode            :on
IOreconfigure           :false
PPAR DR                 :on
```

4. **poweronコマンドを実行し、物理パーティションの電源を投入します。**

```
XSCF> poweron -p PPAR_ID
```

注ー論理ドメインの構成情報は、factory-defaultに戻ります。物理パーティションの再起動後、手順1で退避した論理ドメインの構成情報を使用して、論理ドメインを再構築してください。論理ドメインの構成情報の再構築方法については、「A.1　物理パーティションの動的再構成が未サポートのシステムからサポートされるシステムへアップデートする場合」の構築の例を参照してください。

---

5. **showpparmodeコマンドを実行し、現在および次回の設定値を確認します。**
   次の例では、PPAR DR機能が現在、有効に設定されていて、次回も変更がないことを示しています。

```
XSCF> showpparmode -p PPAR_ID
Host-ID                 :00000001
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal            :off
Autoboot(Guest Domain)  :on
Elastic Mode            :off
IOreconfigure           :false
PPAR DR(Current)        :on
```

```
PPAR DR(Next)            :on
```

2.6 SPARC64 X+プロセッサを使用する場合の留意点

ここでは、SPARC64 X+プロセッサが含まれたシステムを構築する場合の留意点を説明します。

注ーSPARC M10の場合の留意点です。SPARC M12は対象ではありません。

2.6.1 XCPファームウェアのアップデート

SPARC M10では、SPARC64 X+プロセッサが搭載されたシステムをサポートするXCPファームウェアとOracle Solarisの版数は、「1.5.4　ソフトウェア条件および確認方法」、および最新の『SPARC M10 システム プロダクトノート』を参照してください。

2.6.2 CPU動作モードの設定

SPARC M10では、SPARC64 X+プロセッサには、SPARC64 Xプロセッサと同等の機能で動作する機能があります。ここでは、CPU動作モードの設定方法や留意点について説明します。

CPU動作モードの設定方法

CPU動作モードの詳細は、「1.6.1　CPU動作の種類とCPU動作モード」を参照してください。
CPU動作モードは、XSCFファームウェアのsetpparmodeコマンドを使用し、物理パーティションごとに設定できます。

setpparmodeコマンドで設定できるCPU動作モード（cpumode）には、「auto」モードと「compatible」モードの2種類があります。CPU動作モードの初期値は、「auto」モードです。
各モードは次のように動作します。

- autoモード
  SPARC64 X+プロセッサで拡張された機能である、データベース高速化の命令を有効にします。ただし、SPARC64 X+プロセッサとSPARC64 Xプロセッサが混在した構成の場合は、SPARC64 X+プロセッサをSPARC64 X互換で動作させます。
  SPARC64 X+プロセッサとSPARC64 Xプロセッサが混在する構成で、動的再構成による保守作業を実施する場合には、compatibleモードに設定してください。
- compatibleモード

SPARC64 X+プロセッサをSPARC64 X互換で動作させます。SPARC64 X+プロセッサとSPARC64 Xプロセッサが混在する物理パーティションを構成する場合は、このモードに設定します。

表 2-14に、物理パーティションの構成、CPU動作モードの設定値、およびCPU動作の関係を示します。

表 2-14 物理パーティションのCPU構成とCPU動作モード

物理パーティションの構成	CPU動作モードの設定値	CPU動作
SPARC64 X+	auto	SPARC64 X+機能で動作
SPARC64 X+	compatible	SPARC64 X互換で動作
SPARC64 X+/X	autoまたはcompatible	SPARC64 X+はSPARC64 X互換で動作
SPARC64 X	autoまたはcompatible	SPARC64 X機能で動作

CPU動作モードの設定を変更するには、設定を変更する物理パーティションの電源が切断されている状態で行います。

重要－CPU動作モードを変更する前に、必ず、論理ドメイン構成情報を保存（バックアップ）してください。
Oracle VM Server for SPARCの論理ドメイン構成情報には、論理ドメインが使用しているCPU動作モードの情報が含まれています。SPARC64 X+機能で動作している論理ドメインの論理ドメイン構成情報を、SPARC64 X機能を利用して動作する物理パーティションに適用した場合、構成情報の不一致が検出され、XSCFにより論理ドメイン構成情報はfactory-defaultに切り替えられて、物理パーティションが起動されます。
factory-defaultに切り替えられた論理ドメイン構成情報を元の構成に再構築するには、事前に保存（バックアップ）した論理ドメイン構成情報が必要となります。これらの理由により、setpparmodeコマンドでCPU動作モードを変更する前には、必ず論理ドメイン構成情報を保存（バックアップ）してください。論理ドメイン構成情報の保存（バックアップ）および再構築方法については、「A.1　物理パーティションの動的再構成が未サポートのシステムからサポートされるシステムへアップデートする場合」の構築の例を参照してください。論理ドメインの再構築が必要なるケースについては、「B.2　CPU動作モードに関する補足情報」を参照してください。

CPU動作モードを変更するには、setpparmodeコマンドを使用します。CPU動作モードを「compatible」に変更する場合を例にした手順を、以下に示します。

1. **物理パーティションの電源を切断します。**

```
XSCF> poweroff -p PPAR_ID
```

2. **showpparmodeコマンドを実行し、CPU動作モード（CPU mode）の現在の設定を確認します。**

```
XSCF> showpparmode -p PPAR_ID
Host-ID                     :00000002
Diagnostic Level            :min
Message Level               :normal
Alive Check                 :on
Watchdog Reaction           :reset
Break Signal                :on
```

```
Autoboot(Guest Domain)       :on
Elastic Mode                 :off
IOreconfigure                :false
CPU Mode                     :auto
PPAR DR(Current)             :-
PPAR DR(Next)                :on
```

3. **setpparmode**コマンドを実行し、**CPU**動作モード（**CPU mode**）を「**auto**」から「**compatible**」に変更します。

```
XSCF> setpparmode -p PPAR_ID -m cpumode=compatible
```

4. **showpparmode**コマンドを実行し、**CPU**動作モード（**CPU mode**）が「**compatible**」に設定されていることを確認します。

```
XSCF> showpparmode -p PPAR_ID
Host-ID                      :00000002
Diagnostic Level             :min
Message Level                :normal
Alive Check                  :on
Watchdog Reaction            :reset
Break Signal                 :on
Autoboot(Guest Domain)       :on
Elastic Mode                 :off
IOreconfigure                :false
CPU Mode                     :compatible
PPAR DR(Current)             :-
PPAR DR(Next)                :on
```

5. 物理パーティションの電源を投入します。

```
XSCF> poweron -p PPAR_ID
```

CPU動作の種類の確認

CPU動作の種類を確認するには、Oracle Solaris上でprtdiagコマンド、またはpsrinfoコマンドを実行します。

SPARC64 X+機能を利用して動作している場合、prtdiagコマンドであれば、[Virtual CPUs]に「SPARC64-X+」と出力されます。SPARC64 X機能を利用して動作している場合は、[Virtual CPUs]に「SPARC64-X」と出力されます。
psrinfoコマンドの場合は、-pvオプションを指定し実行します。SPARC64 X+機能を利用して動作している場合は、出力された物理プロセッサ情報に「SPARC64-X+」と出力されます。いっぽう、SPARC64 X機能を利用して動作している場合は、「SPARC64-X」と出力されます。

以下に、prtdiagコマンドとpsrinfoコマンドの表示例を示します。

- **SPARC64 X+機能を利用して動作している場合**

[prtdiagコマンドの表示例]

```
# prtdiag
System Configuration:    Oracle Corporation    sun4v  SPARC  M10-4S
Memory size:  391168 Megabytes

================================ Virtual CPUs ================================

CPU ID Frequency Implementation         Status
------ --------- --------------------- -------
0       3700 MHz SPARC64-X+             on-line
:
```

[psrinfoコマンドの表示例]

```
# psrinfo -pv
The physical processor has 16 cores and 32 virtual processors
(0-31)
   The core has 2 virtual processors (0 1)
   The core has 2 virtual processors (2 3)
   The core has 2 virtual processors (4 5)
   The core has 2 virtual processors (6 7)
   The core has 2 virtual processors (8 9)
   The core has 2 virtual processors (10 11)
   The core has 2 virtual processors (12 13)
   The core has 2 virtual processors (14 15)
   The core has 2 virtual processors (16 17)
   The core has 2 virtual processors (18 19)
   The core has 2 virtual processors (20 21)
   The core has 2 virtual processors (22 23)
   The core has 2 virtual processors (24 25)
   The core has 2 virtual processors (26 27)
   The core has 2 virtual processors (28 29)
   The core has 2 virtual processors (30 31)
      SPARC64-X+ (chipid 0, clock 3700 MHz)
   :
   :
```

- **SPARC64 X機能を利用して動作している場合**
  [prtdiagコマンドの表示例]

```
# prtdiag
System Configuration:    Oracle Corporation    sun4v  SPARC  M10-4S
Memory size:  391168 Megabytes

================================ Virtual CPUs ================================

CPU ID Frequency Implementation         Status
------ --------- --------------------- -------
0       3700 MHz SPARC64-X              on-line
:
```

[psrinfoコマンドの表示例]

```
# psrinfo -pv
The physical processor has 16 cores and 32 virtual processors
(0-31)
   The core has 2 virtual processors (0 1)
   The core has 2 virtual processors (2 3)
   The core has 2 virtual processors (4 5)
   The core has 2 virtual processors (6 7)
   The core has 2 virtual processors (8 9)
   The core has 2 virtual processors (10 11)
   The core has 2 virtual processors (12 13)
   The core has 2 virtual processors (14 15)
   The core has 2 virtual processors (16 17)
   The core has 2 virtual processors (18 19)
   The core has 2 virtual processors (20 21)
   The core has 2 virtual processors (22 23)
   The core has 2 virtual processors (24 25)
   The core has 2 virtual processors (26 27)
   The core has 2 virtual processors (28 29)
   The core has 2 virtual processors (30 31)
      SPARC64-X (chipid 0, clock 3700 MHz)
The physical processor has 16 cores and 32 virtual processors
(32-63)
   The core has 2 virtual processors (32 33)
```

2.6.3 SPARC64 X+プロセッサとSPARC64 XプロセッサをPPAR内で混在させるための条件

SPARC M10-4Sでは、1つの物理パーティション内で、SPARC64 X+プロセッサを搭載したSPARC M10-4SとSPARC64 Xプロセッサを搭載したSPARC M10-4Sを混在させることができます。混在させる場合には、ビルディングブロックを構成する、すべてのSPARC M10-4Sに対して、SPARC64 X+プロセッサに対応したXCPファームウェアを適用する必要があります。お使いのシステム構成によっては、SPARC64 X+プロセッサに対応したXCPファームウェアが自動的に適用されます。

SPARC64 Xプロセッサ搭載のSPARC M10-4Sで構成された物理パーティションに、SPARC64 X+プロセッサ搭載のSPARC M10-4Sを増設する場合は、SPARC64 Xプロセッサ搭載のSPARC M10-4SのXCPファームウェアを、事前に、アップデートする必要があります。
SPARC64 X+プロセッサ搭載のSPARC M10-4Sで構成された物理パーティションに、SPARC64 Xプロセッサ搭載のSPARC M10-4Sを増設する場合は、SPARC64 Xプロセッサ搭載のSPARC M10-4SのXCPファームウェアは、マスタXSCFに適用されているXCPファームウェアの版数に自動的に更新されます。

注－SPARC64 X+プロセッサが搭載されたシステムをサポートするXCPファームウェアの版数は、最新のXCP版数の『SPARC M10システム プロダクトノート』を参照してください。

2.6.4 CPU動作モードと物理パーティションの動的再構成の関係

ここでは、CPU動作モードと、物理パーティションの動的再構成によりSPARC M10-4Sを追加、削除する場合の関係について説明します。

CPU動作モードと物理パーティションの動的再構成による追加の関係

SPARC M10では、SPARC64 X+機能を利用して動作している物理パーティションには、SPARC64 X+プロセッサ搭載のシステムボードだけを動的再構成で追加することができます。
SPARC64 X+機能をサポートしていないSPARC64 Xプロセッサを動的に組み込むことはできません。
SPARC64 Xプロセッサ搭載のシステムボードを動的再構成で追加しようとした場合、以下のメッセージが出力され、追加処理が失敗します。

```
The current configuration does not support this operation.
```

SPARC64 X機能を利用して動作している物理パーティションの場合は、SPARC64 XまたはSPARC64 X+のどちらのプロセッサが搭載されたシステムボードでも、物理パーティションに、動的再構成で追加することができます。

表 2-15に、物理パーティション構成、CPU動作の種類、および動的再構成で追加できるシステムボードのCPU構成との関係を示します。

表 2-15　CPU動作モードと物理パーティションの動的再構成による追加処理との関係（SPARC M10）

物理パーティション構成	CPU動作の種類	SPARC64 X+プロセッサ搭載システムボードの動的再構成	SPARC64 Xプロセッサ搭載システムボードの動的再構成
SPARC64 X+	SPARC64 X+機能で動作	可能	不可
SPARC64 X+	SPARC64 X互換で動作	可能	可能
SPARC64 X+/SPARC64 X混在	SPARC64 X+はSPARC64 X互換で動作	可能	可能
SPARC64 X	SPARC64 X機能で動作	可能	可能

CPU動作モードと物理パーティションの動的再構成による削除の関係

SPARC M10では、SPARC64 X+機能で動作している場合、またはSPARC64 X機能で動作している場合の、いずれでの場合でも動的再構成によりシステムボードを削除できます。

表 2-16に、物理パーティション構成、CPU動作の種類、および動的再構成で削除できるシステムボードのCPU構成との関係を示します。

表 2-16　CPU動作モードと物理パーティションの動的再構成による削除処理との関係（SPARC M10）

物理パーティション構成	CPU動作の種類	SPARC64 X+プロセッサ搭載システムボードの動的再構成	SPARC64 Xプロセッサ搭載システムボードの動的再構成
SPARC64 X+	SPARC64 X+機能で動作	可能	－(*1)
SPARC64 X+	SPARC64 X互換で動作	可能	－(*1)
SPARC64 X+/SPARC64 X混在	SPARC64 X互換で動作	可能	可能(*2)
SPARC64 X	SPARC64 X機能で動作	－(*1)	可能

*1:　システム構成上、この操作はありません。

*2:　SPARC64 X+プロセッサとSPARC64 Xプロセッサが混在する物理パーティションの場合は、setpparmodeコマンドでCPU動作モードをcompatibleモードに設定してください。CPU動作モードがautoモードに設定されている状態で、SPARC64 Xプロセッサ搭載のシステムボードを動的再構成で削除し、SPARC64 X+プロセッサ搭載のシステムボードだけで構成されるシステムになった場合に、物理パーティションの電源を切断して再投入したり、リセットしたりすると、SPARC64 X+機能でCPUは起動されます。
その後、SPARC64 Xプロセッサ搭載のシステムボードを動的再構成で追加しようとしても、SPARC64 X+機能を利用して動作しているため、動的再構成で追加することはできなくなります。そのため、SPARC64 X+プロセッサとSPARC64 Xプロセッサが混在された物理パーティションに対して、システムボードの活性交換を実施する運用を検討する場合は、CPU動作モードをcompatibleモードに設定してください。

第3章

ドメイン構築のための操作

ここでは、ドメイン構築のための操作とコマンドを説明します。

- 物理パーティション構築に関する操作とコマンド
- 論理ドメイン構築に関する操作とコマンド

3.1 物理パーティション構築に関する操作とコマンド

物理パーティションを構築するためにXSCFが提供するユーザーインターフェースには、コマンドライン形式のXSCFシェルとウェブブラウザ形式のXSCF Webがあります。ここでは、物理パーティション構築に関する操作で主に使用するXSCFシェルのコマンドについて説明します。

物理パーティションの構築で使用するXSCFシェルコマンドには、表示系コマンドと操作系コマンドがあります。

表 3-1 表示系コマンド

コマンド名	機能概要
showpcl	物理パーティションの構成情報を表示します。
showpparstatus	物理パーティションのステータスを表示します。
showboards	システムボードのステータスを表示します。
showfru	デバイスの設定情報を表示します。
showdomainstatus	論理ドメインのステータスを表示します。
showcodactivation	CPUコア アクティベーションキーの情報を表示します。
showcodusage	CPUコアリソースの使用状況を表示します。
showcod	CPUコア アクティベーションの登録および設定情報を表示します。
showpparmode	物理パーティションの動作モードを表示します。

表 3-1　表示系コマンド (続き)

コマンド名	機能概要
showdomainconfig	物理パーティションの論理ドメイン構成情報を表示します。

表 3-2　操作系コマンド

コマンド名	機能概要
setupfru	メモリミラーを設定します。
setpcl	物理パーティション構成情報を設定します。
addcodactivation	CPUコア アクティベーションキーを追加します。
setcod	物理パーティション（PPAR）で使用されるCPUコアリソースを割り当てます。
addboard	物理パーティションにシステムボード（PSB）を追加します。
deleteboard	物理パーティションからPSBを削除します。
setpparmode	物理パーティションの動作モードを設定します。
setdomainconfig	物理パーティションの論理ドメイン構成情報を指定します。
poweron	物理パーティションを起動します。
poweroff	物理パーティションを停止します。
console	制御ドメインコンソールに接続します。

ここでは、物理パーティションの構築で使用する表示系コマンドと操作系コマンドについて、使用例を交えて詳細を説明します。なお、各コマンドのオプション、オペランド、および使用方法などの詳細な情報については、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

注－XSCFシェルおよびXSCF Webによるユーザーインターフェースは、一部の管理者に提供される機能であり、物理パーティションを構築するための管理者権限が必要になります。また、複数の物理パーティションの管理者によってシステムボードを共有するような場合は、より安全なシステム運用のために事前の準備と慎重な運用計画を図る必要があります。

3.1.1　物理パーティション構成情報を確認する

物理パーティション構成情報（PPAR構成情報）を表示するには、showpclコマンドを使用します。PPAR-ID、PPAR構成情報に登録されているPSB番号、物理パーティションのステータスなど、PPAR構成情報を一覧形式で表示します。

showpclコマンドは、物理パーティションの動的再構成に関する操作を行う前に、物理パーティションが動的再構成を行える状態であることを判断するためや、動的再構成するシステムボードの物理パーティション構成情報への登録状態を確認するために使用します。また、動的再構成の操作後に物理パーティションの状態や構成を確認するためにも使用します。

注－showpclコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showpcl [-v] -a
XSCF> showpcl [-v] -p ppar_id
```

- -v オプション
  デフォルト表示に加えて、 [Cfg-policy] 欄、 [No-Mem] 欄、 [No-IO] 欄を表示します。
- -a オプション
  すべての物理パーティションのPPAR構成情報を表示します。
- -p ppar_id オプション
  物理パーティション番号を指定します。指定した物理パーティションのPPAR構成情報を表示します。

表 3-3は、showpclコマンドの表示項目を示しています。

表 3-3 showpclの表示項目

表示項目	表示内容	意味
PPAR-ID	0から15までの整数	物理パーティション番号
LSB	0から15までの整数	論理システムボード番号
PSB	xx-y（xx:00から15までの整数、y:0固定）	物理システムボード番号。1つのビルディングブロック（BB）を表し、LSBに対応する。
Status（PPARの稼働状態）	Powered Off	電源が切断されている状態
	Initialization Phase	POST動作中の状態
	Initialization Complete	POST完了の状態
	Running	POST処理が完了し稼働中の状態
	Hypervisor Aborted	ハイパーバイザがアボートし、リセットされるまでの状態
Cfg-policy(*1)（PPAR構成情報のコンフィグレーションポリシー）	FRU	CPUやメモリといった部品単位で縮退
	PSB	PSB単位で縮退
	System	PPAR単位で縮退
No-Mem(*1)（PPAR構成情報のno-memフラグ）	True	LSBに搭載されているメモリを論理ドメインで使用させない
	False	LSBに搭載されているメモリを論理ドメインで使用させる
No-IO(*1)（PPAR構成情報のno-ioフラグ）	True	LSBに搭載されている入出力デバイスを論理ドメインで使用させない
	False	LSBに搭載されている入出力デバイスを論理ドメインで使用させる

*1: -v オプションを指定した場合だけ表示される項目です。

次の例では、物理パーティション0のPPAR構成情報を表示しています。

```
XSCF> showpcl -p 0
PPAR-ID   LSB   PSB   Status
00                    Running
          00    00-0
          04    01-0
          08    02-0
          12    03-0
```

コンフィグレーションポリシー、no-mem、no-ioなどの詳細を表示するには、-vオプションを指定します。次の例では、物理パーティション0のPPAR構成情報の詳細を表示しています。

```
XSCF> showpcl -v -p 0
PPAR-ID   LSB   PSB   Status   No-Mem   No-IO    Cfg-policy
00                    Running
                                                System
          00    -
          01    -
          02    -
          03    -
          04    01-0           False    False
          05    -
          06    -
          07    -
          08    02-0           True     False
          09    -
          10    -
          11    -
          12    03-0           False    True
          13    -
          14    -
          15    -
```

3.1.2 物理パーティションのステータスを確認する

物理パーティションのステータスを表示するには、showpparstatusコマンドを使用します。

注－showpparstatusコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
useradm, platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showpparstatus -a
XSCF> showpparstatus -p ppar_id
```

- -a オプション

すべての物理パーティションのステータスを表示します。

- -p ppar_id オプション
  物理パーティション番号を指定します。指定した物理パーティションのステータスを表示します。

表 3-4は、showpparstatusコマンドの表示項目を示しています。

表 3-4 物理パーティションのステータス

表示項目	表示内容	意味
PPAR-ID	0から15までの整数	物理パーティション番号
PPAR Status（PPARの状態）	Powered Off	電源が切断されている状態
	Initialization Phase	POST動作中の状態
	Initialization Complete	POST完了の状態
	Running	POST処理が完了し、稼働中の状態
	Hypervisor Aborted	ハイパーバイザがアボートし、リセットされるまでの状態
	－	上記以外（PPARにPSBが割り当てられていない状態）

次の例では、稼働中の物理パーティション0のステータスを表示しています。

```
XSCF> showpparstatus -p 0
PPAR-ID         PPAR Status
00              Running
```

3.1.3 システムボードのステータスを確認する

システムボード（PSB<BB>）の物理パーティションへの割り当て状況を表示するには、showboardsコマンドを使用します。

showboardsコマンドは、物理パーティションの動的再構成の操作を行う前に、PSB（BB）が動的再構成の操作を行える状態であることを確認するためや、PSB（BB）が属している物理パーティション番号などを確認するために使用します。また、動的再構成の操作後に、PSB（BB）の状態を確認するためにも使用します。

注－showboardsコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showboards [-v] -a [-c sp]
XSCF> showboards [-v] -p ppar_id [-c sp]
XSCF> showboards [-v] psb
```

注ーshowboardsコマンドは、BB構成の場合にのみ有効です。

- -v オプション
  詳細情報を表示します。PSB（BB）の追加予約／削除予約の状態を確認するときに指定します。
- -a オプション
  すべての物理パーティションの情報を表示します。
- -c sp オプション
  システムボードプール状態のPSB（BB）に限定して表示します。システムボードプール状態とは、どの物理パーティションにも属さない状態のことです。
- -p ppar_id オプション
  物理パーティション番号を指定します。指定した物理パーティションについて情報を表示します。
- psb
  PSB（BB）番号を指定します。

表 3-5は、showboardsコマンドの表示項目を示しています。

表 3-5 システムボードのステータス

表示項目	表示内容	意味
PSB	xx-y（xx:00から15までの整数、y:0固定）	1つのビルディングブロック（BB）を表す物理システムボード（PSB）番号（xxは、BB-ID）
R(*1)	*	物理パーティションが再起動されると、PSB（BB）が現行の物理パーティション構成に組み込まれる、または切り離されて構成が変更される状態
PPAR-ID	0から15までの整数	PSB（BB）が割り当てられている物理パーティションの番号
	SP	PSB（BB）がシステムボードプール状態
	Other	ユーザー権限を持つ物理パーティションに物理パーティション構成情報が設定され、かつ、ユーザー権限を持たない物理パーティションに属している状態
LSB	00から15までの整数	物理パーティションで使用される論理システムボードの番号
Assignment（PPARへの割り当て状態）	Unavailable	PSB（BB）が物理パーティションに割り当てられていない状態であり、システムボードプール、未診断、診断中、診断異常のいずれかの状態（PSB（BB）が未実装の状態も含まれる）
	Available	PSB（BB）がシステムボードプールであり、診断が正常に終了している状態
	Assigned	PSB（BB）が割り当てられている状態
Pwr（システムボードの電源投入状態）	n	電源が切断されている状態
	y	電源が投入されている状態

表 3-5　システムボードのステータス (続き)

表示項目	表示内容	意味
Conn（PPAR構成への組み込み状態）	n	PSB（BB）が物理パーティション構成から切り離されている状態
	y	PSB（BB）が物理パーティション構成に組み込まれている状態
Conf（Oracle Solarisでの稼働状態）	n	PSB（BB）がOracle Solarisで稼働していない状態
	y	PSB（BB）がOracle Solarisで稼働している状態
Test（診断状態）	Unmount	未実装または未定義の状態
	Unknown	診断されていない状態
	Testing	診断中
	Passed	診断が正常に終了した状態
	Failed	診断で異常が検出され、PSB（BB）が動作しない状態
Fault（縮退状態）	Normal	正常な状態
	Degraded	縮退部品がある状態（PSB（BB）は稼働可能）
	Faulted	異常によりPSB（BB）が稼働できない状態、または通信異常によりPSB（BB）が制御できない状態

*1:　-v オプションを指定した場合だけ表示される項目です。

次の例では、showboardsコマンドを実行して、すべての物理パーティションでのPSB（BB）の割り当て状態を表示しています。[PPAR-ID]欄に「SP」と表示されているPSBは、システムボードプール状態です。システムボードプール状態とは、PSBがどの物理パーティションにも属さない状態です。

```
XSCF> showboards -a
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
02-0 SP           Available   n    n    n    Passed  Normal
03-0 SP           Available   n    n    n    Passed  Normal
```

PSBに対する追加予約／削除予約の状態を表示するには、-vオプションを指定します。次の例では、搭載されているすべてのPSBの詳細情報を表示しています。

```
XSCF> showboards -v -a
PSB  R PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- - ------------ ----------- ---- ---- ---- ------- --------
00-0 * 00(00)       Assigned    y    y    y    Passed  Normal
01-0   SP           Unavailable n    n    n    Testing Normal
02-0   Other        Assigned    y    y    n    Passed  Degraded
03-0   SP           Unavailable n    n    n    Failed  Faulted
```

[R] 欄の下に表示されている「*」が予約状態を表します。上の例のPSB 00-0は、割り当て中なので、削除予約であるとわかります。

3.1.4 デバイスの設定情報を確認する

デバイスの設定情報を表示するには、showfruコマンドを使用します。

注－showfruコマンドはplatadmまたはfieldeng権限を持つユーザーアカウントで実行します。

```
XSCF> showfru device location
```

- device
  デバイス名として、sb（システムボード（PSB<BB>））またはcpu（PSB（BB）内のCPU）を指定します。
- location
  デバイス名がsbの場合はBB-ID-0、cpuの場合はBB-ID-0-CPUを指定します。ただしBB-IDの範囲は0から15まで、CPUの範囲は0から3までです。

表 3-6は、showfruコマンドの表示項目を示しています。

表 3-6 showfruコマンドの表示項目

表示項目	表示内容	意味
Device（デバイス）	sb	システムボード（PSB）
	cpu	CPU
Location（デバイスが搭載されている位置）	Deviceがsbの場合 BB-ID-0（BB-ID: 0から15までの整数）	PSB番号
	Deviceがcpuの場合 BB-ID-0-x（BB-ID: 0から15までの整数、x:0から3までの整数）	CPU番号
Memory Mirror Mode（メモリミラーモード）	yes	メモリミラーモード
	no	非メモリミラーモード

次の例では、ビルディングブロックBB-ID#01内のPSBのデバイス設定情報を表示しています。

```
XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb     01-0
   cpu 01-0-0   yes
   cpu 01-0-1   yes
   cpu 01-0-2   yes
   cpu 01-0-3   yes
```

次の例では、ビルディングブロックBB-ID#01内のPSB中のCPU#03の設定情報を表示しています。

```
XSCF> showfru cpu 01-0-3
Device Location Memory Mirror Mode
sb     01-0
   cpu 01-0-3   yes
```

3.1.5 論理ドメインのステータスを確認する

論理ドメインのステータスを表示するには、showdomainstatusコマンドを使用します。

注ーshowdomainstatusコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
useradm, platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showdomainstatus -p ppar_id [-g domainname]
```

- -p ppar_id オプション
  表示対象の物理パーティションの番号を指定します。
- -g domainname オプション
  表示対象の論理ドメイン名を指定します。

表示される項目は表 3-7のとおりです。

表 3-7 論理ドメインのステータス

表示項目	表示内容	意味
Logical Domain Name	ー	論理ドメイン名
Status（現在の論理ドメインの稼働状態）	Host stopped	論理ドメインが停止している状態
	Solaris booting	論理ドメインのOracle Solarisが起動中の状態
	Solaris running	論理ドメインのOracle Solarisが稼働している状態
	Solaris halting	論理ドメインのOracle Solarisが停止処理をしている状態
	Solaris suspended	論理ドメインのOracle Solarisがサスペンドしている状態
	Solaris powering down	論理ドメインのOracle Solarisが電源切断処理をしている状態
	Solaris rebooting	論理ドメインのOracle Solarisが再起動中の状態
	Solaris panicking	論理ドメインのOracle Solarisでパニックが発生している状態
	Solaris debugging	論理ドメインのkmdbプロンプトで停止している状態（カーネルデバッグ中）

表 3-7　論理ドメインのステータス (続き)

表示項目	表示内容	意味
	OpenBoot initializing	論理ドメインのOpenBoot PROMが初期化処理中の状態
	OpenBoot Running	論理ドメインのOpenBoot PROMによる初期化が完了した状態、またはokプロンプトで停止している状態
	OpenBoot Primary Boot Loader	論理ドメインがOracle Solaris load中の状態
	OpenBoot Running OS Boot	論理ドメインがOracle Solarisに遷移中の状態
	OS Started. No state support.	SUNW,soft-state-supported CIF未実施状態で、SUNW,set-trap-table CIF実行中の状態
	OpenBoot Running Host Halted	論理ドメインのOracle Solarisからinit 0実行中の状態
	OpenBoot Exited	論理ドメインのokプロンプトから reset-all実行中の状態
	OpenBoot Host Received Break	論理ドメインのOracle Solarisからenterサービスが呼ばれた状態
	OpenBoot Failed	論理ドメインのOpenBoot PROMによる初期化でエラーが発生
	Unknown	ユーザーがオプションで指定した論理ドメイン名と一致する論理ドメインが見つからず不明な状態（Logical Domains Managerでadd-spconfigコマンドを未実行の状態も含む）
	－	上記以外（PPARにPSBが割り当てられていない状態）

次の例では、物理パーティション0上にあるすべての論理ドメインの状態を表示しています。

```
XSCF> showdomainstatus -p 0
Logical Domain Name     Status
primary                 Solaris running
guest00                 Solaris running
guest01                 Solaris booting
guest02                 Solaris powering down
guest03                 Solaris panicking
guest04                 Shutdown Started
guest05                 OpenBoot initializing
guest06                 OpenBoot Primary Boot Loader
```

3.1.6　CPUコア アクティベーションキー情報を確認する

XSCFに登録されているCPUコア アクティベーションキーの情報はshowcodactivationコマンドで参照します。

注－showcodactivationコマンドはplatadmまたはplatop権限を持つユーザーアカウントで実行します。

```
XSCF> showcodactivation
```

次の例では、SPARC M10で4つのCPUコアリソース分のCPUコア アクティベーションが表示されています（2セット分）。

```
XSCF> showcodactivation
Index   Description Count
------- ----------- ------
      0 PROC            2
      1 PROC            2
```

注－SPARC M10では、1セットで2つのCPUコア アクティベーションをXSCFに登録でき、2つのCPUコアリソースを有効にできます。SPARC M12では、1セットで1つのCPUコア アクティベーションをXSCFに登録でき、1つのCPUコアリソースを有効にできます。

3.1.7 CPUコアリソースの使用状況を確認する

CPUコアリソースの使用状況を表示するには、showcodusageコマンドを使用します。

注－showcodusageコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showcodusage [-v] [-p {resource|ppar|all}]
```

- -v オプション
  詳細な情報を表示します。
- -p {resource|ppar|all} オプション
  allはすべてのCPUコアリソースの使用状況を表示します。resourceはリソース別に、pparは物理パーティション別に表示します。

次の例では、showcodusageコマンドで-p resourceオプションを指定して、CPUコアリソースの使用状況をリソース別に表示しています。
ここでは、システムには16個のCPUコアリソースが実装されており、4つのCPUコアアクティベーションが登録され、4つのCPUコアリソースが使用中であり、現在未使用のCPUコア アクティベーションの数が0個であることを示しています。

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-------- ------ --------- ------------- ------
PROC          4        16             4 OK: 0 cores available
```

```
Note:
  Please confirm the value of the "In Use" by the ldm command of
 Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
logical domains.
```

表 3-8は、このコマンドの表示項目の意味を示しています。

表 3-8 showcodusage -p resourceコマンドでの表示内容

表示項目	意味
Resource	使用可能なCPUコアリソース（PROC: CPU、現在はCPUのみ）
In Use	現在、システムで使用されているCPUコアリソースの数（ハイパーバイザとの通信が確立できない場合は0になる）
Installed	システムに実装されているCPUコアリソースの数
CoD Permitted	システムに登録されているCPUコア アクティベーションの数
Status	使用中のCPUコアリソースに対するCPUコア アクティベーションの有無（OK: あり、VIOLATION: なし）

次の例では、SPARC M10においてshowcodusageコマンドで-p pparオプションを指定して、物理パーティションでのCPUコアリソースの使用状況を表示しています。この例では、物理パーティション0に32個のCPUコアリソースが実装され、4つのCPUコア アクティベーション数が割り当てられ、3つのCPUコアリソースが現在使用されています。また、物理パーティション1に32個のCPUコアリソースが実装され、4つのCPUコア アクティベーション数が割り当てられ、4つのCPUコアリソースが現在使用されています。さらに、システムで現在未使用（XSCFに登録しているがPPARに割り当てられていない）のCPUコア アクティベーション数が12であることがわかります。

```
XSCF> showcodusage -p ppar
PPAR-ID/Resource In Use Installed Assigned
---------------- ------ --------- --------------
0  - PROC             3        32         4 cores
1  - PROC             4        32         4 cores
2  - PROC             0         0         0 cores
3  - PROC             0         0         0 cores
4  - PROC             0         0         0 cores
5  - PROC             0         0         0 cores
6  - PROC             0         0         0 cores
7  - PROC             0         0         0 cores
8  - PROC             0         0         0 cores
9  - PROC             0         0         0 cores
10 - PROC             0         0         0 cores
11 - PROC             0         0         0 cores
12 - PROC             0         0         0 cores
13 - PROC             0         0         0 cores
14 - PROC             0         0         0 cores
15 - PROC             0         0         0 cores
```

```
Unused - PROC        0        0      12 cores
Note:
  Please confirm the value of the "In Use" by the ldm command of
 Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
logical domains.
```

showcodusageコマンドで-p pparオプションを指定した場合、表 3-9に示す内容が確認できます。

表 3-9　showcodusage -p pparコマンドでの表示項目

表示項目	意味
PPAR-ID/Resource	物理パーティション番号とCPUコアリソースの種類。Unusedと表示されているCPUコアリソースは、物理パーティションで使用されていないCPUコアリソース
In Use	現在、物理パーティションで使用されているCPUコアリソースの数
Installed	物理パーティションに実装されているCPUコアリソースの数
Assigned	物理パーティションに割り当てられているCPUコア アクティベーションの数

3.1.8　CPUコア アクティベーションの登録および設定情報を確認する

CPUコア アクティベーションの登録および設定情報を確認するには、showcodコマンドを使用します。

注－showcodコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
platadm, platop, pparadm, pparmgr, pparop

```
XSCF> showcod [-v]
XSCF> showcod [-v] -s cpu
XSCF> showcod [-v] -p ppar_id
```

- -v オプション
  詳細情報を表示します。CPUコア アクティベーションの内訳を表示します。
- -s cpu オプション
  CPU番号を指定します。
- -p ppar_id オプション
  物理パーティション番号を指定します。

次の例では、showcodコマンドですべてのCPUコア アクティベーションの登録およ

び設定情報を表示しています。

```
XSCF> showcod
PROC Permits installed: 6 cores
PROC Permits assigned for PPAR 0: 4
```

3.1.9 物理パーティションの動作モードを確認する

物理パーティションの動作モードを表示するには、showpparmodeコマンドを使用します。

注ーshowpparmodeコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
platadm, fieldeng, pparadm

```
XSCF> showpparmode -p ppar_id [-v]
```

- -p ppar_id オプション
  物理パーティションを指定します。
- -v オプション
  詳細情報を表示します。物理パーティションのイーサネット（MAC）アドレスも表示されます。

表 3-10は、showpparmodeコマンドの表示項目を示しています。

表 3-10 showpparmodeコマンドの表示項目

表示項目	意味
HOST-ID	ホストID（ホストIDが未割り当ての場合はハイフン「ー」）(*1)
Diagnostic Level	POSTの診断レベル（なし／標準／最大）
Message Level	POST診断のコンソールメッセージの詳細レベル（なし／抜粋／標準／最大／デバッグ）
Alive Check	Alive Checkの有効／無効
Watchdog Reaction	HOST Watchdogタイムアウト時のPPARのリアクション（リセット／Panic処理／無効(none)）
Break Signal	Break信号（STOP-A）の有効／無効
Autoboot(Guest Domain)	PPAR起動時のゲストドメインAuto boot機能の有効／無効
Elastic Mode	CPU、メモリに対する低電力動作の有効／無効
IOreconfigure	PPARの電源ON時や、リブート時にバス構成に応じてIOバス再構成を行う機能の有効／無効（有効／無効／次回起動時のみ有効で以後自動的に無効）
PPAR DR(Current)	現在のPPAR DR機能の設定内容（PPARの電源が投入されていない場合はハイフン「ー」）(*2)

表 3-10　showpparmodeコマンドの表示項目 (続き)

表示項目	意味
PPAR DR(Next)	次回の起動後のPPAR DR機能の設定内容(*2)
Ethernet Address	物理パーティションのイーサネット（MAC）アドレス(*1)

*1:　HOST-IDおよびEthernet Addressは、物理パーティションごとに異なる値が自動的に割り当てられます。
*2:　PPAR DR機能はXCP 2210以降から、デフォルトで有効になっています。物理パーティションの動的再構成を実施する場合は、showpparmodeコマンドでPPAR DR機能が有効になっているか確認してください。無効のときは、-m ppar_dr=onを指定してsetpparmodeコマンドを実行し、動的再構成を行う物理パーティションのPPAR DR機能を有効にしてください。

3.1.10　物理パーティションの論理ドメイン構成情報を表示する

論理ドメイン構成情報を表示するには、showdomainconfigコマンドを使用します。

```
XSCF> showdomainconfig -p ppar_id
```

- -p ppar_id オプション
  物理パーティションを指定します。

表 3-11は、showdomainconfigコマンドの表示項目を示しています。

表 3-11　showdomainconfigコマンドの表示項目

表示項目	意味
Index	構成情報のXSCFでの管理番号
PPAR-ID	物理パーティション番号
Booting config(Current)	現在稼働中の物理パーティションで使用している構成名
Booting config(Next)	次回物理パーティション起動時に使用する構成名
config_name	構成名
domains	論理ドメイン構成に含まれる論理ドメイン数(*1)
date_created	構成情報作成日時

*1:　ldm add-spconfigコマンドで論理ドメイン構成情報をXSCFに保存したときに、boundまたはactive状態であった論理ドメインの数が表示されます。

次の例では、物理パーティション0に設定された構成情報を表示しています。

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) :ldm-set1
(Next) :ldm-set2
```

```
-----------------------------------------------------------------
Index :1
config_name :factory-default
domains :1
date_created:-
-----------------------------------------------------------------
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-08 11:34:56'
-----------------------------------------------------------------
Index :3
config_name :ldm-set2
domains :20
date_created:'2012-08-09 12:43:56'
```

3.1.11 メモリミラーを設定する

メモリミラーを設定するには、setupfruコマンドを使用します。

注－setupfruコマンドはplatadmまたはfieldeng権限を持つユーザーアカウントで実行します。

- SPARC M12の場合

SPARC M12でメモリをミラー構成にする場合は、-c mirror=yesを指定します。

```
XSCF> setupfru [[-q] -{y|n}] -c function=mode device location
```

- SPARC M10の場合

SPARC M10でメモリをミラー構成にする場合は、-m yを指定してください。-c mirrorオプションは使用できません。

```
XSCF> setupfru [-m {y|n}] device location
```

- -c function=mode
  メモリをミラーモードに設定するかどうかを指定します。ミラーモードに設定する場合は、-c mirror=yes、設定しない場合は-c mirror=noを指定します。SPARC M12でのみ使用できます。
- -m {y|n}
  メモリをミラーモードに設定するかどうかを指定します。ミラーモードに設定する場合はy、設定しない場合はnを指定します。-mオプションを省略した場合は、前回の設定が引き継がれます。
- device
  ミラーモードに設定する範囲を指定します。sbを指定した場合、指定したPSBのす

べてのCPUに対して設定が反映されます。cpuを指定した場合、指定したCPUのみに設定が反映されます。

- location
  デバイス名がsbの場合はPSB番号を指定します。形式はBB-ID-0です。cpuの場合はCPUを指定します。形式はBB-ID-0-CPUです。ただしBB-IDは0から15までの整数、CPUは0から3までの整数です。

次の例では、SPARC M12でBB#01(PSB#01)配下のすべてのCPUをメモリミラーモードに設定しています。

```
XSCF> setupfru -c mirror=yes sb 01-0

Notice:
- Logical domain config_name will be set to "factory-default".

Memory mirror mode setting will be changed, Continue? [y|n] :y
```

次の例では、SPARC M12でBB#02のCPU1をメモリミラーモードに設定しています。

```
XSCF> setupfru -c mirror=yes cpu 02-0-1


Notice:
- Logical domain config_name will be set to "factory-default".

Memory mirror mode setting will be changed, Continue? [y|n] :y
```

注-SPARC M10をお使いの場合、以下を指定します。
XSCF> **setupfru -m y** *device location*

3.1.12 物理パーティション構成情報を設定する

物理パーティション構成情報（PPAR構成情報）にシステムボードを登録したり、コンフィグレーションポリシー、メモリ無効化オプション、I/O無効化オプションを設定したりするときには、setpclコマンドを使用します。

注－setpclコマンドはplatadm権限を持つユーザーアカウントで実行します。

```
XSCF> setpcl -p ppar_id -a lsb=psb [lsb=psb...]
XSCF> setpcl -p ppar_id -s policy=value
XSCF> setpcl -p ppar_id -s no-mem=value lsb [lsb...]
XSCF> setpcl -p ppar_id -s no-io=value lsb [lsb...]
```

- -p ppar_idオプション
  物理パーティションに割り当てる番号を指定します。ここで指定した番号が、以

降、物理パーティション番号として参照されます。

注ー物理パーティション番号は、システムに搭載されている筐体のBB-IDと同じ番号のいずれかを割り当てます。搭載されていないBB-IDを物理パーティション番号に指定した場合、その物理パーティションは電源投入に失敗します。

- -a lsb=psb [lsb=psb...] オプション
  論理システムボート（LSB）に物理システムボード（PSB）を対応させます。lsbに論理システムボード（LSB）番号、psbに物理システムボード（PSB）番号を指定します。
- -s policy=value オプション
  物理パーティション全体に対するコンフィグレーションポリシーを設定します。valueには、縮退の単位としてfru（部品単位）、psb（システムボード単位（BB単位））、system（物理パーティション全体）のいずれかを指定します。

注ーコンフィグレーションポリシーオプションは、該当の物理パーティションの電源が切断されているときに設定してください。

- -s {no-io|no-mem}=value
  trueかfalseのいずれかを指定します。trueを指定すると、無効化となり、メモリやI/Oが使用できなくなります。
- lsb
  論理システムボード番号（LSB番号）を指定します。

注ーメモリ無効化オプションとI/O無効化オプションは、該当PSB（BB）がシステムボードプール状態、または物理パーティションが電源切断されている状態で設定してください。システムボードプール状態とは、どの物理パーティションにも属さない状態のことです。

注ーメモリ無効化／I/O無効化は、SPARC M12-2S/M10-4Sで設定できます。

次の例では、物理パーティション0の論理システムボード（LSB）0と1に、それぞれ物理システムボード（PSB）00-0と01-0を対応付けています。

```
XSCF> setpcl -p 0 -a 0=00-0 1=01-0
```

次の例では、物理パーティション0に対して、コンフィグレーションポリシーを「物理パーティション全体」に設定しています。

```
XSCF> setpcl -p 0 -s policy=system
```

次の例では、物理パーティション0の中の論理システムボード（LSB）1について、メモリもI/Oも使用しないと設定しています。

```
XSCF> setpcl -p 0 -s no-mem=true 1
XSCF> setpcl -p 0 -s no-io=true 1
```

3.1.13 CPUコア アクティベーションキーを追加する

CPUコア アクティベーションキーを追加するには、addcodactivationコマンドを使用します。

注－addcodactivationコマンドはplatadm権限を持つユーザーアカウントで実行します。

```
XSCF> addcodactivation {key-signature | -F url}
```

- key-signature
  CPUコア アクティベーションキーを二重引用符（"）で囲って指定します。CPUコア アクティベーションキーのテキストファイル（XXXXX_XX_XXX.TXT）の内容をコピーして貼り付けます。
- -F url
  CPUコア アクティベーションキーの保存先となるURLを指定します。

次の例では、2 CPUコア分のCPUコア アクティベーションキーを追加しています。

```
XSCF> addcodactivation "Product: SPARC M10-1
SequenceNumber: 10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtxgvZPsrtYguqiNUieB+mTDC
 :
 :
b1GCkFx1RH27FdVHiB2H0A=="
Above Key will be added, Continue?[y|n]:y
```

システムにCPUコア アクティベーションキーを追加した時点では、まだCPUコアリソースがOracle Solaris上で使用できる状態にありません。
CPUコアリソースを使用可能な状態にするには、「3.1.14　物理パーティションで使用するCPUコアリソースを割り当てる」を参照し、CPUコアリソースを物理パーティションに割り当てる操作を実施してください。

3.1.14 物理パーティションで使用するCPUコアリソースを割り当てる

システムに登録したCPUコア アクティベーションを元にCPUコアリソースを物理パーティションに割り当てるには、setcodコマンドを対話形式で実行します。

注－setcodコマンドはplatadm権限を持つユーザーアカウントで実行します。

```
XSCF> setcod [-p ppar_id] -s cpu
PROC Permits installed: XX cores
PROC Permits assigned for PPAR 0 (X MAX)
[Permanent Xcores]
Permanent [X]: permits
PROC Permits assigned for PPAR 1 (X MAX)
[Permanent Xcores]
Permanent [X]: permits
:略
```

また、XCP 2260以降のXSCFファームウェアが適用されている場合は、以下のコマンドでも実行できます。

```
XSCF> setcod [[-q] -{y|n}] -p ppar_id -s cpu -c {set|add|del} permits
```

- -p ppar_id オプション
  CPUコアリソースを割り当てる物理パーティションIDを指定します。permitsオペランドを指定しない場合は、CPUコアリソースを割り当てる対話形式のセションが開始されます。
- -s cpu
  CPUを指定します。
- permits
  CPUコア アクティベーション数を指定します。割り当てるCPUコア アクティベーションの単位は1コア単位です。使用可能なCPUコア アクティベーションの数を超えて設定することはできません。permitsオペランドを指定しない場合は、CPUコアリソースを割り当てる対話形式のセションが開始されます。
- -c {set|add|del}
  物理パーティションに対してCPUコア アクティベーションを割り当てる場合は-c set、割り当てられているCPUコア アクティベーションの数を追加する場合は-c add、割り当てられているCPUコア アクティベーションを削除する場合は-c delを指定します。

注ー以下の指定方法でsetcodコマンドを使用することは推奨しません。
XSCF> setcod -p ppar_id -s cpu permits
XCP 2260以降のXCPファームウェアでsetcodコマンドを実行する場合は、-cオプションを指定するか、対話形式を使用してください。また、XCP 2250以前の場合は、対話形式を使用してください。
理由は次のとおりです。

- コマンド実行時に、設定した内容で変更してよいかを確認するメッセージ（[y|n]）が出力されない
- 運用中の物理パーティションに対してCPUコア アクティベーションの割り当て数を削減するときに、警告メッセージが出力されない
  permitsの指定を誤るなどして、CPUコア アクティベーションが不足した場合、システム停止を引き起こすおそれがあります。

---

addcodactivationコマンドを使用して登録したCPUコア アクティベーションの数が上限となります。

注－-c setを使う、または-cオプションを使わないで、CPUコア アクティベーション数を指定する場合、オペランドpermitsには、追加および削除する数ではなく、現在設定されている数に追加分を足した数、または現在設定されている数から削除分を引いた数を、指定してください。誤って追加／削除する数だけを指定すると、CPUコア アクティベーション数を減らしてしまいシステム停止を引き起こすことがあります。

次の例では、4つのCPUコアリソースを物理パーティション1に割り当てています。

```
XSCF> setcod -p 1 -s cpu -c set 4
PROC Permits assigned for PPAR 1 : 0 -> 4
PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y
Completed.
```

次の例では、CPUコアリソースを対話形式で物理パーティションに割り当てています。

```
XSCF> setcod -s cpu
PROC Permits installed: 5 cores
PROC Permits assigned for PPAR 0 (5 MAX) [Permanent 2cores]
Permanent [2]:1
PROC Permits assigned for PPAR 1 (4 MAX) [Permanent 0cores]
Permanent [0]:4
PROC Permits assigned for PPAR 2 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 3 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 4 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 5 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 6 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 7 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 8 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 9 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 10 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 11 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 12 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 13 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 14 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 15 (0 MAX) [Permanent 0cores]
Permanent [0]:
```

次の例では、2つのCPUコアリソースを物理パーティション0に追加しています。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 10
XSCF> setcod -p 0 -s cpu -c add 2
PROC Permits assigned for PPAR 0 : 10 -> 12
PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y
Completed.
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 12
```

注－setcod -p 0 -s cpu -c set 12と指定しても同じです。

注－XCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

3.1.15 ビルディングブロック構成においてシステムボードを追加する

ビルディングブロック（BB）を表すシステムボード（PSB<BB>）を物理パーティションに追加または追加予約するには、addboardコマンドを使用します。本コマンドを実行する前に、各種コマンドを使って、物理パーティションの状態（showpparstatusコマンド）、PSB（BB）の状態（showboardsコマンド）を確認してください。

注－addboardコマンドはplatadmまたはpparadm権限を持つユーザーアカウントで実行します。

注－追加するPSB（BB）は、該当物理パーティションのPPAR構成情報にあらかじめ登録されている必要があります。

```
XSCF> addboard [-c configure] [-m bind={resource|none}]
                 [-m diag={off|min}] -p ppar-id psb [psb...]
XSCF> addboard -c assign -p ppar_id psb [psb...]
XSCF> addboard -c reserve -p ppar_id psb [psb...]
```

- -c configure オプション
  指定した物理パーティションにPSB（BB）を組み込みます。物理パーティションの電源が切断されているか、制御ドメインのOracle Solarisが稼働していない場合、組み込みは行われずエラーとなります。-c assignまたは-c reserveオプションを省略した場合、-c configureオプションが指定されたものとみなされます。
- -c assign オプション
  指定した物理パーティションにPSB（BB）を割り当てます。割り当てられたPSB（BB）は、指定した物理パーティションに予約され、ほかの物理パーティションからは

割り当てができなくなります。割り当てられたあとは、物理パーティションを再起動するか、-c configureオプションを指定してaddboardコマンドを実行することで、PSB（BB）が物理パーティションに組み込まれます。

- -c reserve オプション
  指定した物理パーティションへのPSB（BB）の割り当てを予約します。動作はassignと同じです。

- -m bind={resource|none} オプション
  組み込み対象のPSB（BB）による追加リソースの自動割り当て機能を設定します。resourceを指定した場合、追加リソースの自動割り当て機能が有効になります。noneを指定した場合は、追加リソースの自動割り当て機能が無効になり、追加したPSB（BB）のリソースは、物理パーティションでの空きリソースになります。デフォルトはresourceです。

  addboardコマンドの実行前にdeleteboardコマンドを実行してリソースを削除していた場合、自動割り当て機能を有効にすることで、deleteboardコマンドの実行前のリソース割り当て状態に戻すことができます。ただし、addboardコマンドの実行前に論理ドメイン構成情報を変更していた場合には、論理ドメイン構成情報に従ったリソースの割り当て状態になります。

  また、物理パーティションの論理ドメイン構成情報がfactory-defaultであった場合、本オプションの設定にかかわらず、追加リソースは制御ドメインに割り当てられます。

- -m diag={off|min} オプション
  物理パーティションにPSB（BB）を組み込む際に実行するハードウェア診断のレベルを設定します。offを指定した場合、ハードウェア診断は行われません。minを指定した場合、ハードウェア診断レベルとして標準が設定されます。デフォルトはminです。

- -p ppar_id　オプション
  物理パーティション番号を指定します。

- psb
  PSB（BB）番号を指定します。

注－-c configureオプションによるPSB（BB）の組み込みでは、対象となるPSB（BB）のハードウェア診断が行われたあとに、物理パーティションにPSBが組み込まれます。このため、addboardコマンドの実行終了までに時間がかかることがあります。

注－addboardコマンドで-c configureオプションを指定してPSB（BB）を組み込む場合、あらかじめ組み込み対象の物理パーティションに対してsetpparmodeコマンドを使用してPPAR DR機能を有効にし、物理パーティションを再起動しておく必要があります。PPAR DR機能が無効となっている物理パーティションには、-c configureオプションを指定してPSB（BB）を組み込むことはできません。

注－物理パーティションでCPUコア アクティベーション違反が発生している状態では、addboardコマンドによる動的再構成は実行できません。CPUコア アクティベーション違反の状態を解消したあとに、addboardコマンドによる動的再構成を実行してください。

注－addboardコマンドによる動的再構成の操作に失敗した場合、対象のPSB（BB）は操作前の状態に戻りません。addboardコマンドが出力したエラーメッセージおよびOracle Solarisのメッセージから原因を特定し、適切な対処を行ってください。ただし、発生したエラー内容によっては、物理パーティションの電源切断または再起動が必要になることがあります。

注－PSB（BB）の交換で、addboardコマンドの実行時に-mオプションを省略した場合、または-mオプションでbind=resourceを指定した場合、deleteboardコマンドを実行する前のリソース割り当て状態に戻らないことがあります。
CPUやメモリなど、交換前のPSB（BB）のリソース量が交換後のPSB（BB）のリソース量と異なっている場合、リソース割り当て状態を元の状態に戻すことができません。
元のリソース割り当て状態に戻らなかった場合、リソースは空きリソース状態になります。この場合、ldmコマンドを使用して、論理ドメインへのリソース割り当てを再度行ってください。

次の例では、物理パーティション0に、物理システムボード（PSB）00-0、01-0、02-0、03-0を割り当てています。-yオプションはコマンドのすべての確認にyを答えるためのものです。

```
XSCF> addboard -y -c assign -p 0 00-0 01-0 02-0 03-0
```

3.1.16 ビルディングブロック構成においてシステムボードを削除する

ビルディングブロック（BB）を表すシステムボード（PSB<BB>）を物理パーティションから削除または削除予約するには、deleteboardコマンドを使用します。削除されたPSB（BB）は、どの物理パーティションにも属さないシステムボードプールに割り当てられます。

本コマンドを実行する前に、各種コマンドを使って、物理パーティションの状態（showpparstatusコマンド）、PSB（BB）の状態（showboardsコマンド）、PSB（BB）上のデバイスの使用状況（showfruコマンド）を確認してください。

注－deleteboardコマンドはplatadmまたはpparadm権限を持つユーザーアカウントで実行します。

```
XSCF> deleteboard [-f] [-c disconnect]
                     [-m unbind={none|resource|shutdown}]
                    psb [psb...]
XSCF> deleteboard [-f] -c unassign psb [psb...]
XSCF> deleteboard [-f] -c reserve psb [psb...]
```

- -c disconnect オプション
  物理パーティションで稼働しているPSB（BB）を物理パーティション構成から切り離して、割り当て状態にします。PSB（BB）は、物理パーティション構成に割

り当てられた状態のままで、物理パーティションを再起動するか、addboardコマンドを実行することで、再び物理パーティションに組み込まれます。-c unassignまたは-c reserveオプションを省略した場合、-c disconnectオプションが指定されたものとみなされます。

- -c unassign オプション
  PSB（BB）の物理パーティションへの割り当てを解除して、システムボードプール状態にします。システムボードプール状態のPSB（BB）は、ほかの物理パーティションに組み込んだり割り当てたりすることができます。

- -c reserve オプション
  PSB（BB）の物理パーティションへの割り当て解除を予約します。割り当て解除を予約すると、指定した物理パーティションの電源が切断されたときに、PPAR構成情報を元にPSB（BB）の割り当てが解除され、システムボードプール状態になります。

- -m unbind={none|resource|shutdown} オプション
  切り離し対象のPSB（BB）のリソースを使用している論理ドメインを、切り離さないPSB（BB）に移動する際に、移動先のリソースが不足したときの動作を設定します。

設定内容	説明
none	移動先のリソースを確保しません。移動先のリソースが不足した場合、deleteboardコマンドの実行は失敗します。 物理パーティションの論理ドメイン構成情報がfactory-defaultである場合、-m unbind=noneオプションは指定できません。 エラーとなる場合は、Oracle VM Server for SPARCの仮想DR機能によって、論理ドメインからCPUコアやメモリを削除しておく必要があります。
resource	リソース移動が必要な論理ドメイン自体、または、ほかの論理ドメインのリソースを削減して、移動先のリソースを確保します。ただし、移動先のリソースを確保するために論理ドメインをシャットダウンすることはありません。また、XCP 2240以降、かつ、Oracle VM Server for SPARC 3.2以降の場合、物理I/Oを自動的に削除します。
shutdown	リソース移動が必要な論理ドメイン自体、または、ほかの論理ドメインのリソースを削減して、移動先のリソースを確保します。また、移動先のリソースを確保するために、論理ドメインをシャットダウンします。

- psb
  システムボード番号を指定します。

- -fオプション
  PSB（BB）を強制的に切り離します。

注－deleteboardコマンドで-fオプションを指定して、物理パーティションから強制的にPSB（BB）を削除する場合、CPUバインドしているプロセスや、デバイスにアクセスしているプロセスで、重要な問題が発生することが懸念されます。このため、-fオプションは、通常の運用では使用しないことを推奨します。-fオプションを指定した場合は、deleteboardコマンドの実行前後に必ず物理パーティションや業務プロセスの状態を確認してください。

注－物理パーティションを稼働させたままdeleteboardコマンドで-c disconnectオプションか-c unassignオプションを指定してPSB（BB）を削除する場合、あらかじめ削除対象の物理パーティションに対してsetpparmodeコマンドを使用してPPAR DR機能を有効にし、物理パーティションを再起動しておいてください。PPAR DR機能が無効となっている物理パーティションでは、物理パーティションの稼働中に-c disconnectオプションや-c unassignオプションを指定してPSB（BB）を削除することはできません。

注－物理パーティションでCPUコア アクティベーション違反が発生している状態では、deleteboardコマンドによる動的再構成は実行できません。CPUコア アクティベーション違反の状態を解消したあとに、deleteboardコマンドによる動的再構成を実行してください。

注－deleteboardコマンドによる動的再構成の処理が失敗した場合、対象のPSB（BB）は処理前の状態に戻りません。deleteboardコマンドが出力したエラーメッセージおよびOracle Solarisのメッセージから原因を特定し、適切な対処を行ってください。ただし、発生したエラー内容によっては、物理パーティションの電源切断または再起動が必要になることがあります。

次の例では、物理システムボード（PSB）00-0、01-0、02-0、03-0をシステムボードプール状態にしています。

```
XSCF> deleteboard -c unassign 00-0 01-0 02-0 03-0
```

次の例では、PSB 00-0、01-0、02-0、03-0を切り離し予約しています。

```
XSCF> deleteboard -c reserve 00-0 01-0 02-0 03-0
```

3.1.17 物理パーティションの動作モードを設定する

物理パーティションの動作モードを設定するには、setpparmodeコマンドを使用します。

注－setpparmodeコマンドの動作モードのうち、診断レベル、メッセージレベル、ゲストドメインAuto boot機能の設定については、fieldengの権限を持つユーザーアカウントで実行します。
Alive Check、Break信号、HOST Watchdogタイムアウト時のリアクション、ゲストドメインAuto boot機能、省電力機能、IOバス再構成機能、PPAR DR機能については、platadmまたはpparadm権限を持つユーザーアカウントで実行します。

```
XSCF> setpparmode -p ppar_id -m function=mode
```

- -p ppar_id オプション
  物理パーティションを指定します。
- -m function=mode オプション
  機能と動作モードを指定します。具体的な値については『SPARC M12/M10 XSCF リファレンスマニュアル』を参照してください。

表 3-12は、設定可能なモードを示しています。

表 3-12 PPARに設定可能なモード

機能名	意味	設定可能なモード	反映
診断レベル	電源投入時の自己診断テスト（POST）の診断レベルを設定	なし／標準（初期設定値）／最大	即時(*1)
メッセージレベル	POST診断のコンソールメッセージの詳細レベルを設定	なし／抜粋／標準（初期設定値）／最大／デバッグ	即時(*1)
Alive Check	Alive Checkの有無を設定	有効（初期設定値）／無効	次回起動時(*2)
HOST Watchdogタイムアウト時のリアクション	Watchdogタイムアウト時の各論理ドメインのリアクションを設定	なし／Panic処理／PPARリセット（初期設定値）	即時(*1)
Break信号（STOP-A）抑止	Break信号抑止の有無を設定	有効（初期設定値）／無効	即時(*1)
ゲストドメインAuto boot機能	物理パーティションの電源投入時にゲストドメインの自動的な起動の有無を設定	有効（初期設定値）／無効	次回起動時(*2)
省電力機能	CPU、メモリに対する低電力動作の有無を設定	有効／無効（初期設定値）	即時(*1)
I/Oバス再構成（ioreconfigure）機能	物理パーティションの電源の投入時や、電源の再投入時（リセット時）にバス構成に応じてI/Oバス再構成を行う機能の動作モードを選択	有効／無効（初期設定値）／次回起動時のみ有効で以後自動的に無効	即時(*1)
PPAR DR機能	稼働中の物理パーティション構成に対するシステムボード（PSB<BB>）の組み込み、および切り離しを行う機能を設定	有効／無効（*3）	次回起動時(*2)

*1: 設定は即時に反映されます。
*2: 設定を反映させるには、物理パーティションの電源を投入、または電源を切断したあとに再投入する必要があります。
*3: PPAR DR機能の初期値はXCP 2210より前の版数では無効、XCP 2210以降からは有効に設定されています。

注－I/Oバス再構成機能は、SPARC M12-1/M10-1では未サポートです。

注－PPAR DR機能は、SPARC M12-1/M12-2/M10-1/M10-4では未サポートです。

注－ゲストドメインとともにOracle Solarisを自動的に起動するかどうかは、ldm set-variable コマンドを使用して設定します。デフォルトでは、ゲストドメインとともにOracle Solarisが自動的に起動されます。なお、ドメインを自動的に起動させるには、ドメインをactive状態にして論理ドメイン構成情報を保存する必要があります。詳細は、お使いのバージョンの『Oracle VM Server for SPARC 管理ガイド』を参照してください。

モードスイッチの動作

オペレーションパネルのモードスイッチを設定することによって、物理パーティションは次のように動作します。

- Locked
  setpparmodeコマンドで指定した動作モードで動作します。
- Service
  表 3-13は、モードスイッチがService時の動作を示しています。

表 3-13 モードスイッチがService時の動作

機能名	動作モード
診断レベル	setpparmodeコマンドで指定した動作モードで動作
メッセージレベル	setpparmodeコマンドで指定した動作モードで動作
Alive Check	動作モードの状態にかかわらず無効
HOST Watchdogタイムアウト時のリアクション	setpparmodeコマンドで指定した動作モードで動作
Break信号（STOP-A）抑止	動作モードの状態にかかわらずBreak信号を送信
ゲストドメインAuto boot機能	setpparmodeコマンドで指定した動作モードで動作
省電力機能	setpparmodeコマンドで指定した動作モードで動作
I/Oバス再構成機能	setpparmodeコマンドで指定した動作モードで動作
PPAR DR機能	setpparmodeコマンドで指定した動作モードで動作

3.1.18 物理パーティションの論理ドメイン構成情報を指定する

物理パーティションの構成情報を指定するには、setdomainconfigコマンドを使用します。

```
XSCF> setdomainconfig -p ppar_id -i index
```

- -p ppar_id オプション
  物理パーティションを指定します。
- -i index オプション
  構成情報の管理番号を指定します。管理番号は showdomainconfigコマンドで確認できます。1から8の整数で指定できます。

注－Oracle Solarisのshutdownコマンドで制御ドメインを再起動しただけでは、XSCFに保存された論理ドメイン構成が有効になりません。必ずXSCFのpoweroff／poweronコマンドを使用して再起動してください。

3.1.19 物理パーティションを起動する

物理パーティションの電源を投入するには、poweronコマンドを使用します。

注－poweronコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
platadm, fieldeng, pparadm, pparmgr

```
XSCF> poweron -p ppar_id
```

- -p ppar_id オプション
  物理パーティション番号を指定します。

3.1.20 物理パーティションを停止する

物理パーティションを停止するには、poweroffコマンドを使用します。

```
XSCF> poweroff -p ppar_id
```

- -p ppar_id オプション
  物理パーティションを指定します。

注－論理ドメイン構成の場合は、物理パーティションの電源を切断する前に制御ドメイン上でOracle VM Server for SPARCのldm add-spconfigコマンドを実行し、最新の構成情報をXSCFに保存します。詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「6.2.2　システム停止前の論理ドメイン構成情報の保存」、および「10.11.1　論理ドメインの構成情報を保存する／表示する」を参照してください。

3.1.21 制御ドメインコンソールに接続する

物理パーティションの制御ドメインコンソールに接続するには、consoleコマンドを使用します。

注－consoleコマンドは次のいずれかの権限を持つユーザーアカウントで実行します。
platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> console -p ppar_id
```

- -p ppar_id オプション
  物理パーティション番号を指定します。

3.2 論理ドメイン構築に関する操作とコマンド

物理パーティションを構築したあとに論理ドメインを構築します。論理ドメインは、Oracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用して構築します。ここでは、論理ドメインの構築で使用する表示系コマンドと操作系コマンドについて、使用例を交えて説明します。なお、各コマンドのオプション、オペランド、および使用方法などの詳細な情報については、お使いのバージョンのOracle VM Server for SPARC および Oracle Solarisのマニュアルを参照してください。

論理ドメインの構築で使用するコマンドには、表示系コマンドと操作系コマンドがあります。

表 3-14　表示系コマンド

コマンド名	機能概要
svcs	Logical Domains Managerが起動していることを確認します。
ldm list-services	サービスを表示します。
ldm list-permits	割り当て可能な仮想CPU数を表示します。
ldm list-bindings	リソースの割り当て状況を表示します。
ldm list-devices ldm list-rsrc-group ldm list-socket	リソースの使用状況を表示します。
ldm list-domain	論理ドメインの状態を表示します。
ldm list-spconfig	論理ドメイン構成情報を表示します。
ldm list-io	I/Oデバイスの使用状況を表示します。

表 3-15　操作系コマンド

コマンド名	機能概要
ldm start-reconf	遅延再構成モードを開始します。
ldm add-vconscon ldm add-vdiskserver ldm add-vswitch	デフォルトのサービスを設定します。
ldm add-vcpu ldm set-vcpu ldm remove-vcpu ldm add-core ldm set-core ldm remove-core ldm grow-socket ldm shrink-socket	仮想CPUを設定します。

表 3-15　操作系コマンド (続き)

コマンド名	機能概要
ldm add-memory ldm set-memory ldm remove-memory ldm grow-socket ldm shrink-socket	仮想メモリを設定します。
ldm add-spconfig ldm set-spconfig ldm remove-spconfig	論理ドメイン構成情報を設定します。
ldm add-domain	論理ドメインを作成します。
ldm add-io ldm remove-io	I/Oデバイスを設定します。
ldm create-vf ldm destroy-vf	SR-IOVの仮想機能を作成・破棄します。
ldm add-vnet ldm remove-vnet	仮想ネットワークデバイスを設定します。
ldm add-vdiskserverdevice ldm remove-vdiskserverdevice	仮想ディスクサーバを設定します。
ldm add-vdisk ldm remove-vdisk	仮想ディスクを設定します。
ldm set-socket	CPUソケット制約を設定します。
ldm set-vconsole	仮想コンソールを設定します。
ldm set-variable	起動デバイスを設定します。
ldm bind-domain	リソースをバインドします。
ldm start-domain	ゲストドメインを起動します。
ldm set-domain	シャットダウングループを設定します。
devfsadm	デバイスを再構成します。
svccfg	復旧モード（recovery mode）を設定します。

3.2.1 Logical Domains Managerの起動を確認する

Logical Domains Managerが稼働していることを確認するには、Oracle Solarisのsvcsコマンドを使用します。

```
# svcs service
```

- service
  サービス名を指定します。

次の例では、Logical Domains Managerが稼働状態かどうかを確認しています。Logical Domains Managerのサービス名はldmdです。[STATE] 欄が「online」になっていれば稼働状態です。

```
# svcs ldmd
STATE STIME FMRI
online 16:25:31 svc:/ldoms/ldmd:default
```

注－Logical Domains ManagerはOracle VM Server for SPARCに含まれるソフトウェアです。svcsコマンドの詳細は、Oracle Solarisのリファレンスマニュアルを参照してください。

注－Logical Domains Managerが起動されていなければ、svcadmコマンドを実行し、起動します。svcadmコマンドの詳細は、Oracle Solarisのリファレンスマニュアルを参照してください。

3.2.2 サービスを確認する

設定したサービスを確認するには、ldm list-servicesコマンドを使用します。

```
# ldm list-services ldom
```

- ldom
  論理ドメイン名を指定します。制御ドメインの名前はprimaryです。

次の例では、仮想ディスクサーバとして「primary-vds0」、仮想コンソール端末集配信装置サービスとして「primary-vcc0」、仮想スイッチサービスとして「primary-vsw0」を表示しています。

```
# ldm list-services primary
VDS
   NAME         VOLUME           OPTIONS             DEVICE
   primary-vds0
VCC
   NAME         PORT-RANGE
   primary-vcc0 5000-5100
VSW
   NAME         MAC               NET-DEV  DEVICE   MODE
   primary-vsw0 xx:xx:xx:xx:xx:xx net0     switch@0 prog,promisc
```

3.2.3 CPUコア アクティベーションに基づく割り当て可能な仮想CPU数を確認する

割り当て可能なCPUコアの数を制御ドメインから確認するには、ldm list-permitsコマンドを使用します。

```
# ldm list-permits
```

3.2.4 リソースの割り当て状況を確認する

割り当て状況を確認するには、ldm list-bindingsコマンドを使用します。

```
# ldm list-bindings ldom
```

- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。

3.2.5 リソースの使用状況を確認する

物理パーティションのリソースの使用状況を確認するには、ldm list-devicesコマンドを使用します。

```
# ldm list-devices [-S] [-a] [cpu]
```

- -S オプション
  デバイスのステータスを表示します。
- -a オプション
  バインドされているリソースおよびバインドされていないリソースの、すべてのリソースの使用状況を表示します。
- cpu
  CPUスレッドおよび物理CPUリソースを一覧表示します。

表 3-16は、このコマンドの表示項目の意味を示しています。

表 3-16　ldm list-devicesコマンドの表示項目

表示項目	表示内容	意味
PID	0以上の整数	物理CPU番号
%FREE	0から100までの数値	未使用率
PM	yes	CPUが電源管理されている
	no	CPUの電源が投入されている
	---	論理ドメインに割り当てられていないCPU
STATUS(*1)	ok	CPUが正常な状態
	fail	CPUが故障している

*1:　STATUSは、-S オプションを指定した場合に表示されます。

次の例では、8つの仮想CPUの状態が表示されています。

```
# ldm list-devices -S -a cpu
VCPU
   PID %FREE PM  STATUS
   0   0     no  ok
   1   0     yes ok
   2   0     yes ok
   3   0     yes ok
   4   100   --- ok
   5   100   --- ok
   6   100   --- ok
   7   100   --- ok
```

次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID      %FREE   CPUSET
    0       0       (0, 1)
    4       0       (8, 9)
    8       0       (16, 17)
 (中略)
VCPU
    PID     %FREE   PM
    0       0       no
    1       0       no
    8       0       no
    9       0       no
 (中略)
MEMORY
    PA                  SIZE            BOUND
    0x7c0000000000      8G              guest2
    0x7c0200000000      24G
    0x7e0000800000      1G              _sys_
    0x7e0040800000      384M            _sys_
 (中略)
IO
    DEVICE                          PSEUDONYM        BOUND
OPTIONS
    pci@8000                        PCIE0            yes
    pci@8100                        PCIE1            yes
    pci@8200                        PCIE2            yes
    pci@8300                        PCIE3            yes
PCIEDEV
    pci@8000/pci@4/pci@0/pci@9      /BB0/CMUL/NET0   yes
    pci@8000/pci@4/pci@0/pci@a      /BB0/CMUL/NET2   yes
    pci@8000/pci@4/pci@0/pci@0      /BB0/CMUL/SASHBA yes
    pci@8100/pci@4/pci@0/pci@0      /BB0/PCI0        yes
 (以下略)
```

3.2.6 リソースグループごとに使用状況を確認する

ldm list-rsrc-groupコマンドを使用して、物理パーティション内のリソースグループごとにリソースの使用状況を確認します。リソースグループは、プロセッサコア、メモリ、I/Oバス間のベースとなる物理的な関係に基づいて構成されます。
SPARC M12/M10システムでは、このサブコマンドはOracle VM Server for SPARC 3.3以降でサポートされ、リソースグループとして筐体またはBB-IDが表示されます。
リソースグループの詳細は、お使いのバージョンの『Oracle VM Server for SPARC管理ガイド』の「リソースの管理」を参照してください。

```
# ldm list-rsrc-group [-a] [-d ldom] [-l] [-o core|memory|io] [-p] [resource-group]
```

- -a
  各リソースグループのすべてのリソースを表示します。どの論理ドメインにもバインドされていないリソースも含まれます。
- -d
  指定した論理ドメインの情報のみを表示します。
- -l
  各リソースグループの詳細な情報を表示します。
- -o
  指定したリソースの情報のみを表示します。
- -p
  解析可能な形式で結果を出力します。
- resource-group
  リソースグループを指定して情報を一覧表示します。

次の例は、2つのシステムボードで構成される物理パーティションでの各リソースグループについて、すべてのリソースを一覧表示しています。

```
# ldm list-rsrc-group
NAME     CORE     MEMORY     IO
/BB0     64       262136M    8
/BB1     64       262136M    8
```

3.2.7 CPUソケットごとにリソースの使用状況を確認する

ldm list-socketコマンドを使用して、物理パーティション内のCPUソケットごとにリソースの使用状況を確認します。このサブコマンドは、Oracle VM Server for SPARC 3.2以降でサポートされます。

```
# ldm list-socket [-l] [-p] [--free] [-o format] [socket_id=id[,id[,...]]]
[ldom...]
```

- -l
  物理CPUセットとコアIDを表示します。
- -p
  解析可能な形式で結果を出力します。
- --free
  空きリソースのみを表示します。
- -o
  出力結果を1以上のサブセットに限定します。サブセットについての詳細は、お使いのバージョンの『Oracle VM Server for SPARC リファレンスマニュアル』を参照してください。
- socket_id
  情報を一覧表示する1つ以上のCPUソケットを指定します。
- ldom
  情報を一覧表示する1つ以上の論理ドメインを指定します。

次の例は、CPUソケットごとに、すべてのリソースの使用状況を一覧表示しています。

```
# ldm list-socket
CONSTRAINTS
SOCKET
    TENANT     VCPUS    CORES    SOCKET_ID    GROUP
    primary    32       16       0            /BB0
    primary    32       16       1            /BB0
    primary    32       16       2            /BB0
    primary    32       16       3            /BB0
    primary    32       16       4            /BB1
    primary    32       16       5            /BB1
    primary    32       16       6            /BB1
    primary    32       16       7            /BB1
MEMORY
    PA                SIZE      SOCKET_ID    BOUND
    0x700000000000    64G       7            primary
    0x720000000000    64G       6            primary
    0x740000000000    64G       5            primary
    0x760050000000    31488M    4            primary
    0x780000000000    64G       3            primary
    0x7a0000000000    64G       2            primary
    0x7c0000000000    64G       1            primary
    0x7e0080000000    62G       0            primary
IO
    NAME      TYPE    BUS       SOCKET_ID    BOUND
    PCIE0     BUS     PCIE0     0            primary
    PCIE1     BUS     PCIE1     0            primary
    PCIE2     BUS     PCIE2     1            primary
    PCIE3     BUS     PCIE3     1            primary
    PCIE4     BUS     PCIE4     2            primary
```

```
    PCIE5     BUS     PCIE5     2            primary
    PCIE6     BUS     PCIE6     3            primary
    PCIE7     BUS     PCIE7     3            primary
    PCIE8     BUS     PCIE8     4            primary
    PCIE9     BUS     PCIE9     4            primary
    PCIE10    BUS     PCIE10    5            primary
    PCIE11    BUS     PCIE11    5            primary
    PCIE12    BUS     PCIE12    6            primary
    PCIE13    BUS     PCIE13    6            primary
    PCIE14    BUS     PCIE14    7            primary
    PCIE15    BUS     PCIE15    7            primary
```

3.2.8 論理ドメインの状態を確認する

物理パーティション上で稼働している論理ドメインの状態を確認したり、シャットダウングループの設定を確認したりするには、ldm list-domainコマンドを使用します。

```
# ldm list-domain [-o format] [ldom ...]
```

- format
  コマンドの形式を指定します。シャットダウングループの設定を確認するときにはdomainを指定します。
- ldom
  対象論理ドメインを指定します。

次の例では、制御ドメインと3つのゲストドメインが稼働していることがわかります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    14    8G        64%  2h 54m
guest0           active     -n----  5000    16    8G        42%  2h 54m
guest1           active     -n----  5001    16    8G        11%  2h 54m
guest2           active     -n----  5002    16    8G       7.3%  2h 54m
```

次の例では、論理ドメインldom1にシャットダウングループ7を設定して、その内容を確認しています。最終行に、シャットダウングループが7であることが示されています。

```
# ldm set-domain shutdown-group=7 ldom1
# ldm list-domain -o domain ldom1
NAME               STATE           FLAGS     UTIL
ldom1              active          -n----    0.1%

CONTROL
     failure-policy=ignore
     extended-mapin-space=on
     shutdown-group=7
```

なお、シャットダウングループの設定は、論理ドメインの構成情報に含まれます。こ

のため、シャットダウングループの設定内容を変更したときには、ldm add-spconfigコマンドを使って、その内容をXSCFに保存する必要があります。

3.2.9 論理ドメイン構成情報を表示する

XSCFに保存している論理ドメイン構成情報を制御ドメインから表示するには、ldm list-spconfigコマンドを使用します。

```
# ldm list-spconfig
```

次の例では、制御ドメインでldm list-spconfigコマンドを実行して、論理ドメイン構成情報を表示しています。

```
# ldm list-spconfig
ldm-set1 [current]
ldm-set2
factory-default
```

3.2.10 I/Oデバイスの使用状況を確認する

物理パーティション上の論理ドメインで構成されているI/Oデバイスを確認するには、ldm list-ioコマンドを使用します。

```
# ldm list-io [-l] [-p] [bus | device | pf-name]
# ldm list-io -d pf-name
```

- -l
  ルートコンプレックスとPCIeダイレクトI/Oデバイスの関係などの詳細を表示します。
- -p
  スクリプトなどで解析可能な形式で表示します。
- -d pf-name
  指定された物理機能に関する情報が一覧表示されます。
- bus、device、pf-name
  それぞれPCIeバス、ダイレクトI/O割り当て可能デバイス、およびPCIe SR-IOV物理機能です。

次の例では、論理ドメインでのI/Oデバイスの使用状況を表示しています。
SR-IOV機能をサポートしているPCIeデバイスは、[TYPE]に「PF」と表示されています。
SR-IOVの仮想機能を作成または使用しているデバイスは、[TYPE]に「VF」と表示されています。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
PCIE0                                     BUS   PCIE0     primary   IOV
PCIE1                                     BUS   PCIE1     primary   IOV
PCIE2                                     BUS   PCIE2     primary   IOV
PCIE3                                     BUS   PCIE3     primary   IOV
PCIE4                                     BUS   PCIE4     primary   IOV
PCIE5                                     BUS   PCIE5     primary   IOV
PCIE6                                     BUS   PCIE6     primary   IOV
PCIE7                                     BUS   PCIE7     primary   IOV
（中略）
/BB0/PCI6                                 PCIE  PCIE6     primary   EMP
/BB0/PCI9                                 PCIE  PCIE7     primary   EMP
/BB0/PCI10                                PCIE  PCIE7     primary   EMP
/BB0/PCI0/IOVNET.PF0                      PF    PCIE1     primary
/BB0/PCI0/IOVNET.PF1                      PF    PCIE1     primary
/BB0/CMUL/NET0/IOVNET.PF0                 PF    PCIE0     primary
/BB0/CMUL/NET0/IOVNET.PF1                 PF    PCIE0     primary
/BB0/CMUL/NET2/IOVNET.PF1                 PF    PCIE4     primary
/BB0/CMUL/NET2/IOVNET.PF0                 PF    PCIE4     primary
/BB0/PCI0/IOVNET.PF0.VF0                  VF    PCIE1     iodom00
/BB0/PCI0/IOVNET.PF0.VF1                  VF    PCIE1     iodom01
/BB0/PCI0/IOVNET.PF0.VF2                  VF    PCIE1     iodom02
```

ルートコンプレックスとPCIeダイレクトI/Oデバイスの関係などの詳細を確認するには、-lオプションを指定し実行します。
次の例では、論理ドメインでのI/Oデバイスの使用状況を表示しています。

- PCIeカードが割り当てられている論理ドメイン
  保守対象のPCIeカードのスロット（[NAME] に「/BBx/PCIx」と表示されている行）の [DOMAIN]に表示されている名前が、そのPCIeカードが割り当てられている論理ドメインとなります。

- PCIeカードが属するPCIeルートコンプレックスが割り当てられている論理ドメイン
  上記で確認した行の [BUS] に表示されている名前「PCIEx」が、PCIeカードが属するPCIeルートコンプレックスの名前です。これと同じPCIeルートコンプレックス（[NAME] に「PCIEx」と表示されている行）の [DOMAIN] に表示されている名前が、PCIeカードの属するPCIeルートコンプレックスが割り当てられている論理ドメインとなります。

```
# ldm list-io -l
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
PCIE0                                     BUS   PCIE0     primary   IOV
[pci@8000]
PCIE1                                     BUS   PCIE1     primary   IOV
[pci@8100]
PCIE2                                     BUS   PCIE2     primary   IOV
[pci@8200]
PCIE3                                     BUS   PCIE3     rootdom   IOV
[pci@8300]
PCIE4                                     BUS   PCIE4     primary   IOV
```

```
[pci@8400]
PCIE5                                   BUS   PCIE5     primary   IOV
[pci@8500]
PCIE6                                   BUS   PCIE6     rootdom   IOV
[pci@8600]
PCIE7                                   BUS   PCIE7     primary   IOV
[pci@8700]
/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@w50000393a82368b2,0
scsi@0/iport@f/smp@w500000e0e06d03bf
scsi@0/iport@f/enclosure@w500000e0e06d03bd,0
scsi@0/iport@v0
/BB0/PCI0                               PCIE  PCIE1     primary   OCC
[pci@8100/pci@4/pci@0/pci@0]
/BB0/PCI3                               PCIE  PCIE2     primary   EMP
[pci@8200/pci@4/pci@0/pci@0]
/BB0/PCI4                               PCIE  PCIE2     iodom1    OCC
[pci@8200/pci@4/pci@0/pci@8]
/BB0/PCI7                               PCIE  PCIE3     rootdom   EMP
[pci@8300/pci@4/pci@0/pci@0]
/BB0/PCI8                               PCIE  PCIE3     rootdom   EMP
[pci@8300/pci@4/pci@0/pci@8]
/BB0/CMUL/NET2                          PCIE  PCIE4     primary   OCC
[pci@8400/pci@4/pci@0/pci@a]
network@0
network@0,1
```

3.2.11 遅延再構成モードを開始する

制御ドメインまたはルートドメインに割り当てるハードウェアリソースを再構成するときには、遅延再構成モードに切り替えてから作業します。これは、動的再構成によって再構成したときに、仮想メモリの再構成に時間がかかることがあるためです。遅延再構成モードを開始するには、 ldm start-reconfコマンドを使用します。

```
# ldm start-reconf ldom
```

次の例では、制御ドメインprimaryを遅延再構成モードに切り替えています。

```
# ldm start-reconf primary
```

遅延再構成モードであるかどうかは、ldm list-bindingsコマンドを使用して確認します。次の例では、ldm list-bindingsコマンドの実行結果として実行結果として[FLAGS] 欄の値「-ndcv-」の先頭から3つ目の文字が遅延再構成モードを表す「d」になっています。

```
# ldm list-bindings primary
NAME          STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary       active     -ndcv-  SP      8     4G       0.1%  2h 16m
```

また、遅延再構成モードに切り替えると、ldmコマンドによるゲストドメインの操作が行えなくなります。次の例では、ゲストドメインを停止しようとして失敗しています。

```
# ldm stop-domain -a
------------------------------------------------------------------
Notice: LDom primary is in the process of a delayed
reconfiguration.
Any changes made to primary will only take effect after it
reboots.
------------------------------------------------------------------
Delayed reconfiguration of LDom ldom1 is not permitted
```

3.2.12 デフォルトのサービスを設定する

論理ドメインに対するデフォルトのサービスを設定するには、次のコマンドを使用します。

表 3-17 デフォルトのサービスを設定するコマンド

コマンド	機能概要
ldm add-vconscon	仮想コンソール端末集配信サービスを追加します。
ldm add-vdiskserver	仮想ディスクサーバを追加します。
ldm add-vswitch	仮想スイッチサービスを追加します。

ldm add-vconsconコマンド

制御ドメインに仮想コンソール端末集配信装置サービスを追加するには、ldm add-vconsconコマンドを使用します。

```
# ldm add-vconscon port-range=x-y vcc-name ldom
```

- x
  仮想コンソール端末集配信装置のポート番号の下限値を指定します。
- y
  仮想コンソール端末集配信装置のポートの上限値を指定します。
- vcc-name
  仮想コンソール端末集配信装置名を指定します。
- ldom
  論理ドメイン名を指定します。制御ドメインの名前はprimaryです。制御ドメイン

以外の論理ドメインを指定した場合、その論理ドメインがコンソールの出力先になります。

制御ドメイン以外の論理ドメインを指定した場合、ldm add-vccコマンドの実行後に、その論理ドメインに対してsvcadm enableコマンドを使用して仮想ネットワーク端末サーバデーモン（vntsd）を有効にします。

```
# svcadm enable svc:/ldoms/vntsd
```

ldm add-vdiskserverコマンド

制御ドメインに仮想ディスクサーバを追加するには、ldm add-vdiskserverコマンドを使用します。

```
# ldm add-vdiskserver service-name ldom
```

- service-name
  仮想ディスクサーバ名を指定します。この名前は、システム内にあるすべての仮想ディスクサーバ名の中で一意である必要があります。
- ldom
  論理ドメイン名を指定します。制御ドメインの名前はprimaryです。

ldm add-vswitchコマンド

制御ドメインに仮想スイッチサービスを追加するには、ldm add-vswitchコマンドを使用します。

```
# ldm add-vswitch net-dev=device vswitch-name ldom
```

- net-dev=device
  ネットワークデバイスの名前を指定します。
- vswitch-name
  仮想スイッチサービス名を指定します。
- ldom
  論理ドメイン名を指定します。制御ドメインの名前はprimaryです。

3.2.13 仮想CPUを設定する

仮想CPUを設定するには、次のコマンドを使用します。

表 3-18　仮想CPUを設定するコマンド

コマンド	機能概要
ldm add-vcpu	仮想CPUをスレッド単位で追加します。
ldm set-vcpu	仮想CPUをスレッド単位で設定します。
ldm remove-vcpu	仮想CPUをスレッド単位で削除します。
ldm add-core	仮想CPUをコア単位で追加します。
ldm set-core	仮想CPUをコア単位で設定します。
ldm remove-core	仮想CPUをコア単位で削除します。
ldm grow-socket	特定のCPUソケットに関連付けられた仮想CPUを追加します。
ldm shrink-socket	特定のCPUソケットに関連付けられた仮想CPUを削除します。

注－1つの論理ドメインに対して、コア単位の割り当てとスレッド単位の割り当てを両方行うことはできません。論理ドメインごとにどちらかに決めてください。

ldm add-vcpuコマンド

論理ドメインに仮想CPUを追加するには、ldm add-vcpuコマンドを使用します。

```
# ldm add-vcpu number ldom
```

- number
  追加する仮想CPU（スレッド）の数を指定します。
- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。

次の例では、制御ドメインに1つの仮想CPUを追加しています。

```
# ldm add-vcpu 1 primary
------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconf
iguration.
Any changes made to the primary domain will only take effect aft
er it reboots.
------------------------------------------------------------------
```

ldm set-vcpuコマンド

論理ドメインに仮想CPUを割り当てるには、ldm set-vcpuコマンドを使用します。

```
# ldm set-vcpu number ldom
```

- number
  割り当てる仮想CPU（スレッド）の数を指定します。

- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。

ldm remove-vcpuコマンド

論理ドメインから仮想CPUを削除するには、ldm remove-vcpuコマンドを使用します。

```
# ldm remove-vcpu number ldom
```

- number
  削除する仮想CPU（スレッド）の数を指定します。
- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。

次の例では、制御ドメインから1つの仮想CPUを削除しています。

```
# ldm remove-vcpu 1 primary
------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconf
iguration.
Any changes made to the primary domain will only take effect aft
er it reboots.
------------------------------------------------------------------
```

ldm add-coreコマンド

論理ドメインに特定のコアを追加するには、ldm add-coreコマンドを使用します。

```
# ldm add-core num ldom
# ldm add-core cid=core-ID [,core-ID [,...]] ldom
```

- num
  追加するCPUコアの数を指定します。
- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。
- cid=core-ID [,core-ID [,...]]
  追加するCPUコアのIDを指定します。

注－CIDを指定してCPUコアを追加した場合、動的再構成による構成変更はできません。

注－CPUコアID(CID)とCPUの物理位置との関係については、「2.4.1　論理ドメインの構築に関する留意点」の「CPUのコアIDの指定について」を参照してください。

注－add-coreサブコマンドは、whole-core constraint（コア全体の制約）が無効の状態で仮想CPUが割り当てられている論理ドメインには対応していません。whole-core constraint（コア全体の制約）については『Oracle VM Server for SPARC 管理ガイド』の「コア全体の制約

を適用する方法」を参照してください。

次の例では、制御ドメインに1コアのCPUを追加しています。

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

ldm set-coreコマンド

論理ドメインに特定のコアを設定するには、ldm set-coreコマンドを使用します。

```
# ldm set-core num ldom
# ldm set-core cid=core-ID [,core-ID [,...]] ldom
```

- num
  設定するCPUコアの数を指定します。
- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。
- cid=core-ID [,core-ID [,...]]
  設定するCPUコアのIDを指定します。

注－CIDを指定してCPUコアを設定した場合、動的再構成による構成変更はできません。

注－CPUコアID(CID)とCPUの物理位置との関係については、「2.4.1　論理ドメインの構築に関する留意点」の「CPUのコアIDの指定について」を参照してください。

---

ldm remove-coreコマンド

論理ドメインから特定のコアを削除するには、ldm remove-coreコマンドを使用します。

```
# ldm remove-core num ldom
# ldm remove-core cid=core-ID [,core-ID [,...]] ldom
# ldm remove-core -g resource-group [-n num] ldom
```

- num
  設定するCPUコアの数を指定します。
- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。
- cid=core-ID [,core-ID [,...]]
  設定するCPUコアのIDを指定します。
- -g resource-group
  リソースグループを指定して、CPUコアを削除します。SPARC M12/M10システムでは、このオプションはOracle VM Server for SPARC 3.3以降でサポートされます。

注ーremove-coreサブコマンドは、whole-core constraint（コア全体の制約）が無効の状態で仮想CPUが割り当てられている論理ドメインには対応していません。whole-core constraint（コア全体の制約）については『Oracle VM Server for SPARC 管理ガイド』の「コア全体の制約を適用する方法」を参照してください。

次の例では、制御ドメインから1つのCPUコアを削除しています。

```
# ldm remove-core 1 primary
```

ldm grow-socketコマンド

ldm grow-socketコマンドで使用して、CPUソケットに関連付けられた仮想CPUを論理ドメインに追加します。このサブコマンドは、Oracle VM for SPARC 3.2以降でサポートされます。

```
# ldm grow-socket vcpus=num socket_id=id ldom
# ldm grow-socket cores=num socket_id=id ldom
```

- num
  追加するCPUスレッドまたはCPUコアの数を指定します。
- id
  追加するCPUソケットID を指定します。
- ldom
  論理ドメインを指定します。

次の例は、CPUソケット#2から1つのCPUコアを制御ドメインに追加しています。

```
# ldm grow-socket cores=1 socket_id=2 primary
```

ldm shrink-socketコマンド

ldm shrink-socketコマンドを使用して、CPUソケットに関連付けられた仮想CPUを削除します。このサブコマンドは、Oracle VM for SPARC 3.2以降でサポートされます。

```
# ldm shrink-socket vcpus=num socket_id=id ldom
# ldm shrink-socket cores=num socket_id=id ldom
```

- num
  削除するCPUスレッドまたはCPUコアの数を指定します。
- id
  削除するCPUソケットID を指定します。
- ldom
  論理ドメインを指定します。

次の例は、CPUソケット#2に関連付けられた1つのCPUコアを制御ドメインから削除しています。

```
# ldm shrink-socket cores=1 socket_id=2 primary
```

3.2.14 仮想メモリを設定する

仮想メモリを設定するには、次のコマンドを使用します。

表 3-19 仮想メモリを設定するコマンド

コマンド	機能概要
ldm add-memory	仮想メモリを追加します。
ldm set-memory	仮想メモリを設定します。
ldm remove-memory	仮想メモリを削除します。
ldm grow-socket	CPUソケットに関連付けられた仮想メモリを追加します。
ldm shrink-socket	CPUソケットに関連付けられた仮想メモリを削除します。

ldm add-memoryコマンド

論理ドメインに仮想メモリを追加するには、ldm add-memoryを使用します。

```
# ldm add-memory [--auto-adj] size[unit] ldom
```

- --auto-adj オプション
  メモリサイズを256 MBの境界にそろえます。
- size
  追加するメモリのサイズを指定します。最少の割当単位は256 MBです。
- unit
  サイズの単位を指定します。指定できるのはG（ギガバイト）、M（メガバイト）、K（キロバイト）です。大文字小文字は区別されません。
- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。

次の例では、1 GBの仮想メモリを追加しています。

```
# ldm add-memory 1g 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-memoryコマンド

論理ドメインに仮想メモリを割り当てるには、ldm set-memoryコマンドを使用します。

```
# ldm set-memory [--auto-adj] size[unit] ldom
```

- --auto-adj オプション
  メモリサイズを256 MBの境界にそろえます。
- size
  割り当てるメモリのサイズを指定します。最少の割当単位は256 MBです。
- unit
  サイズの単位を指定します。指定できるのはG（ギガバイト）、M（メガバイト）、K（キロバイト）です。大文字小文字は区別されません。
- ldom
  論理ドメインを指定します。制御ドメインの場合はprimaryで固定です。

ldm remove-memoryコマンド

論理ドメインから仮想メモリを削除するには、ldm remove-memoryコマンドを使用します。

```
# ldm remove-memory [--auto-adj] size[unit] ldom
# ldm remove-memory -g resource-group [-s size[unit]] ldom
```

- --auto-adj オプション
  メモリサイズを256 Mバイトの境界にそろえます。
- size
  メモリサイズを指定します。
- unit
  サイズの単位を指定します。指定できるのはG（ギガバイト）、M（メガバイト）、K（キロバイト）です。大文字小文字は区別されません。
- -g resource-group
  リソースグループを指定してメモリを削除します。SPARC M12/M10システムでは、このオプションはOracle VM Server for SPARC 3.3以降でサポートされます。
- ldom
  論理ドメイン名を指定します。

次の例では、1 GBの仮想メモリを削除しています。

```
# ldm remove-memory 1g 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 grow-socketコマンド

ldm grow-socketコマンドを使用して、CPUソケットに関連付けられた仮想メモリを論理ドメインに追加します。このサブコマンドは、Oracle VM for SPARC 3.2以降でサポートされます。

```
# ldm grow-socket memory=size[unit] socket_id=id ldom
```

- size
  追加するメモリのサイズを指定します。最小単位は、256MBです。
- unit
  サイズの単位を指定します。G（ギガバイト）、M（メガバイト）、K（キロバイト）が指定できます。大文字小文字は区別されません。
- id
  CPUソケットIDを指定します。
- ldom
  論理ドメインを指定します。

次の例は、CPUソケット#2から1 GBの仮想メモリを制御ドメインに追加しています。

```
# ldm grow-socket memory=1G socket_id=2 primary
```

ldm shrink-socketコマンド

ldm shrink-socketコマンドを使用して、CPUソケットに関連付けられた仮想メモリを論理ドメインから削除します。このサブコマンドは、Oracle VM for SPARC 3.2以降でサポートされます。

```
# ldm shrink-socket memory=size[unit] socket_id=id ldom
```

- size
  削除するメモリのサイズを指定します。最小単位は、256MBです。
- unit
  サイズの単位を指定します。G（ギガバイト）、M（メガバイト）、K（キロバイト）が指定できます。大文字小文字は区別されません。
- id
  削除するCPUソケットIDを指定します。
- ldom
  論理ドメインを指定します。

次の例は、制御ドメインからCPUソケット#2に関連付けられている1 GBの仮想メモリを削除しています。

```
# ldm shrink-socket memory=1G socket_id=2 primary
```

3.2.15 CPUソケット制約を設定する

ldm set-socketコマンドを使用して、CPUソケット制約を設定します。CPUソケット制約によって、指定されたCPUソケットのみから仮想のCPU、コア、およびメモリが割り当てられます。このサブコマンドは、Oracle VM for SPARC 3.3以降でサポートされます。

```
# ldm set-socket [-f] [--remap] socket_id=[id[,id[,...]]] ldom
```

- -f
  アクティブではない論理ドメインに対して、すべての存在している仮想CPU、コア、メモリの制約を消去します。バインドされたドメインまたはアクティブなドメインは、指定したCPUソケットに制約よりも少ないリソースしかない場合に、リソースの制約、縮退CPUソケットの制約またはその両方が修正されることがあります。
- --remap
  仮想リソースを持っているドメインが動作している間に、アクティブな物理リソースに割り当てられている仮想リソースを別の物理リソースに移動します。
- id
  CPUソケットIDを指定します。

次の例は、論理ドメインldom1に対してCPUソケット#2の制約を設定しています。

```
# ldm set-socket socket_id=2 ldom1
```

3.2.16 論理ドメイン構成情報を設定する

論理ドメイン構成情報を設定するには、次のコマンドを使用します。

表 3-20 論理ドメイン構成情報を設定するコマンド

コマンド	機能概要
ldm add-spconfig	論理ドメイン構成情報を保存します。
ldm set-spconfig	論理ドメイン構成情報を切り替えます。
ldm remove-spconfig	論理ドメイン構成情報を削除します。

ldm add-spconfigコマンド

論理ドメインの構成情報を保存するには、ldm add-spconfigコマンドを使用します。

```
# ldm add-spconfig config-name
```

- config-name
  構成名を指定します。

構成情報は、工場出荷時のデフォルト値（構成名: factory-default）以外に、最大7個までXSCFに保存可能です。最後に保存した構成情報がデフォルトとなり、次回以降の起動に使われます。

次の例では、構成名config1として構成情報を保存しています。

```
# ldm add-spconfig config1
```

ldm set-spconfigコマンド

論理ドメイン構成情報を切り替えるには、ldm set-spconfigコマンドを使用します。

```
# ldm set-spconfig config-name
```

- config-name
  構成名を指定します。次回起動時に、この論理ドメイン構成情報に従って起動が行われます。

ldm remove-spconfigコマンド

XSCFから論理ドメイン構成情報を削除するには、ldm remove-spconfigコマンドを使用します。

```
# ldm remove-spconfig config-name
```

- config-name
  構成名を指定します。この論理ドメイン構成情報は削除されます。

3.2.17 論理ドメインを作成する

ゲストドメインを作成するには、ldm add-domainコマンドを使用します。

```
# ldm add-domain ldom
```

- ldom
  論理ドメイン名を指定します。

3.2.18 I/Oデバイスを設定する

I/Oデバイスを設定するには、次のコマンドを使用します。

表 3-21 I/Oデバイスを設定するコマンド

コマンド	機能概要
ldm add-io	I/Oデバイスを追加します。
ldm remove-io	I/Oデバイスを削除します。

なお、Oracle VM Server for SPARCのバージョンによって、I/Oデバイスを動的に再構成できます。詳細は、表 1-2、お使いのサーバの最新の『プロダクトノート』、お使いのバージョンの『Oracle VM Server for SPARC リリースノート』、および『Oracle VM Server for SPARC 管理ガイド』を参照してください。

ldm add-ioコマンド

論理ドメインにI/Oデバイスを追加するには、ldm add-ioコマンドを使用します。

```
# ldm add-io (bus | device | vf-name) ldom
```

- bus
  PCIeバス（ルートコンプレックス）を指定します。
- device
  ダイレクトI/O割り当て可能なデバイス名（PCIeエンドポイントデバイス名）を指定します。
- vf-name
  PCIe SR-IOV仮想機能名を指定します。
- ldom
  バス、デバイス、またはSR-IOV仮想機能を追加する論理ドメインを指定します。

ldm remove-ioコマンド

論理ドメインからI/Oデバイスを削除するには、ldm remove-ioコマンドを使用します。

```
# ldm remove-io [-n] (bus | device | vf-name) ldom
```

- -n
  コマンドが成功するかどうか事前確認します。
- bus
  PCIeバス（ルートコンプレックス）を指定します。
- device
  ダイレクトI/O割り当て可能なデバイス名（PCIeエンドポイントデバイス名）を指定します。

- vf-name
  PCIe SR-IOV仮想機能名を指定します。
- ldom
  バス、デバイス、またはSR-IOV仮想機能を削除する論理ドメインを指定します。

次の例では、制御ドメイン（primary）からPCIeバス（PCIE1）を削除し、I/Oドメインに割り当てる手順を説明します。また、PCIeエンドポイントについても、同様の手順となります。

注－工場出荷時のデフォルト値（factory-default）の場合、I/Oデバイスはすべてprimaryドメインに割り当てられています。

---

1. **Oracle Solarisのfmadm faultyコマンド、およびXSCFファームウェアのshowstatusコマンドを実行して、デバイスにエラー情報がないことを確認します。**

```
# fmadm faulty
```

```
XSCF> showstatus
```

エラー情報がある場合は、ハードウェアの保守作業を実施したあと、fmadm repairコマンドを実行してエラー情報をクリアします。

```
# fmadm repaired fmri | lable
```

2. **制御ドメインを遅延再構成状態にします。**
   動的に再構成する場合は、この手順は省略できます。

```
# ldm start-reconf primary
```

3. **制御ドメインから、PCIeエンドポイント（PCIE1）の割り当てを解除します。**

```
# ldm remove-io PCIE1 primary
```

4. **制御ドメインを再起動します。**
   動的に再構成する場合は、この手順は省略できます。

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

5. **割り当て先のI/OドメインのOracle Solarisが起動されている場合は停止します。**
   動的に再構成する場合は、この手順は省略できます。

```
# ldm stop-domain ldom1
```

6. **I/Oドメインに**PCIe**エンドポイント（**PCIE1**）の割り当てを追加します。**

```
# ldm add-io PCIE1 ldom1
```

7. **割り当て先のI/Oドメインの**Oracle Solaris**を起動します。**
   動的に再構成する場合は、この手順は省略できます。

```
# ldm start-domain ldom1
```

3.2.19 SR-IOVの仮想機能を作成・破棄する

SR-IOVの仮想機能を作成・破棄するには、次のコマンドを使用します。

表 3-22 SR-IOV機能を作成・破棄するコマンド

コマンド	機能概要
ldm create-vf	仮想機能（VF）を作成します。
ldm destroy-vf	仮想機能（VF）を破棄します。

なお、Oracle VM Server for SPARCのバージョンによって、I/Oデバイスを動的に再構成できます。詳細は、表 1-2、お使いのサーバの『プロダクトノート』、お使いのバージョンの『Oracle VM Server for SPARC リリースノート』、および『Oracle VM Server for SPARC 管理ガイド』を参照してください。

SR-IOVの仮想機能（VF）を作成する

仮想機能（VF）を作成するには、ldm create-vfコマンドを使用します。

ldm create-vfコマンド

```
複数の仮想機能（VF）をまとめて作成する場合
# ldm create-vf -n number | max pf-name
```

```
仮想機能（VF）を1つずつ作成する場合
# ldm create-vf pf-name
```

- -n
  仮想機能（VF）の数を指定します。
- pf-name
  SR-IOV機能をサポートするPCIeデバイス名です。

次の例では、SR-IOVの仮想機能（VF）を作成し、仮想デバイスをI/Oドメインに追加する手順を説明します。

1. **SR-IOV機能をサポートしているPCIeデバイスを確認します。**

次の実行例では、搭載されているPCIeカードのうち、PCIeデバイス名の"/BB0/PCI0/IOVNET.PF0"と"/BB0/PCI0/IOVNET.PF1"が該当し、制御ドメインに割り当てられています。

```
# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
PCIE0                                   BUS   PCIE0     primary   IOV
PCIE1                                   BUS   PCIE1     primary   IOV
PCIE2                                   BUS   PCIE2     primary   IOV
PCIE3                                   BUS   PCIE3     primary   IOV
PCIE4                                   BUS   PCIE4     primary   IOV
PCIE5                                   BUS   PCIE5     primary   IOV
PCIE6                                   BUS   PCIE6     primary   IOV
PCIE7                                   BUS   PCIE7     primary   IOV
（中略）
/BB0/PCI6                               PCIE  PCIE6     primary   EMP
/BB0/PCI9                               PCIE  PCIE7     primary   EMP
/BB0/PCI10                              PCIE  PCIE7     primary   EMP
/BB0/PCI0/IOVNET.PF0                    PF    PCIE1     primary
/BB0/PCI0/IOVNET.PF1                    PF    PCIE1     primary
/BB0/CMUL/NET0/IOVNET.PF0               PF    PCIE0     primary
/BB0/CMUL/NET0/IOVNET.PF1               PF    PCIE0     primary
/BB0/CMUL/NET2/IOVNET.PF1               PF    PCIE4     primary
/BB0/CMUL/NET2/IOVNET.PF0               PF    PCIE4     primary
```

2. **制御ドメインを遅延再構成状態にします。**
   動的に再構成する場合は不要です。手順3に進みます。

```
# ldm start-reconf primary
```

3. **仮想機能（VF）を作成します。**

```
# ldm create-vf /BB0/PCI0/IOVNET.PF0
# ldm create-vf /BB0/PCI0/IOVNET.PF1
```

4. **制御ドメインを再起動します。**
   動的に再構成する場合は不要です。手順6に進みます。

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

5. **割り当て先のI/Oドメイン（iodom00とiodom01）のOracle Solarisが起動されている場合は停止します。**
   動的に再構成する場合は不要です。手順6に進みます。

```
# ldm stop-domain ldom00
# ldm stop-domain ldom01
```

6. **I/Oドメインに仮想機能（/BB0/PCI0/IOVNET.PF0.VF0と/BB0/PCI0/IOVNET.PF1.VF1）を追加します。**

```
# ldm add-io /BB0/PCI0/IOVNET.PF0.VF0 ldom00
# ldm add-io /BB0/PCI0/IOVNET.PF1.VF0 ldom01
```

7. **I/OドメインのOracle Solarisを起動します。**
   動的に再構成する場合は不要です。

```
# ldm start-domain ldom00
# ldm start-domain ldom01
```

SR-IOVの仮想機能（VF）を破棄する

仮想機能（VF）を破棄するには、ldm destroy-vfコマンドを使用します。

ldm destroy-vfコマンド

- 複数の仮想機能（VF）を一度に破棄する場合

```
# ldm destroy-vf -n number | max pf-name
```

- 指定した仮想機能（VF）を破棄する場合

```
# ldm destroy-vf vf-name
```

- -n
  仮想機能（VF）の数を指定します。
- pf-name
  SR-IOV機能をサポートするPCIeデバイス名です。
- vf-name
  仮想機能（VF）の名前です。

次の例では、仮想デバイスをI/Oドメインから切り離し、SR-IOVの仮想機能（VF）を破棄する手順を説明します。

1. **論理ドメインでのI/Oデバイスの使用状況を確認します。**
   次の実行例では、仮想デバイスの"/BB0/PCI0/IOVNET.PF0.VF0"と"/BB0/PCI0/IOVNET.PF0.VF1"がI/Oドメイン（iodom00とiodom01）に割り当てられています。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
PCIE0                                     BUS   PCIE0     primary   IOV
PCIE1                                     BUS   PCIE1     primary   IOV
PCIE2                                     BUS   PCIE2     primary   IOV
PCIE3                                     BUS   PCIE3     primary   IOV
```

```
PCIE4                                   BUS   PCIE4     primary   IOV
PCIE5                                   BUS   PCIE5     primary   IOV
PCIE6                                   BUS   PCIE6     primary   IOV
PCIE7                                   BUS   PCIE7     primary   IOV
（中略）
/BB0/PCI6                               PCIE  PCIE6     primary   EMP
/BB0/PCI9                               PCIE  PCIE7     primary   EMP
/BB0/PCI10                              PCIE  PCIE7     primary   EMP
/BB0/PCI0/IOVNET.PF0                    PF    PCIE1     primary
/BB0/PCI0/IOVNET.PF1                    PF    PCIE1     primary
/BB0/CMUL/NET0/IOVNET.PF0               PF    PCIE0     primary
/BB0/CMUL/NET0/IOVNET.PF1               PF    PCIE0     primary
/BB0/CMUL/NET2/IOVNET.PF1               PF    PCIE4     primary
/BB0/CMUL/NET2/IOVNET.PF0               PF    PCIE4     primary
/BB0/PCI0/IOVNET.PF0.VF0                VF    PCIE1     iodom00
/BB0/PCI0/IOVNET.PF0.VF1                VF    PCIE1     iodom01
```

2. **仮想機能が割り当てられているI/Oドメインの**Oracle Solaris**が起動されている場合は停止します。**
   動的に再構成する場合は不要です。手順3に進みます。

```
# ldm stop-domain iodom00
# ldm stop-domain iodom01
```

3. **I/Oドメインから、仮想機能（/BB0/PCI0/IOVNET.PF0.VF0と/BB0/PCI0/IOVNET.PF0.VF1）の割り当てを解除します。**

```
# ldm remove-io /BB0/PCI0/IOVNET.PF0.VF0 iodom00
# ldm remove-io /BB0/PCI0/IOVNET.PF0.VF1 iodom01
```

4. **I/Oドメインの**Oracle Solaris**を起動します。**
   動的に再構成する場合は不要です。手順6に進みます。

```
# ldm start-domain ldom00
# ldm start-domain ldom01
```

5. **制御ドメインを遅延再構成状態にします。**
   動的に再構成する場合は不要です。手順6に進みます。

```
# ldm start-reconf primary
```

6. **仮想機能（VF）を破棄します。**

```
# ldm destroy-vf /BB0/PCI0/IOVNET.PF0
# ldm destroy-vf /BB0/PCI0/IOVNET.PF1
```

7. **制御ドメインを再起動します。**

動的に再構成する場合は不要です。

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

3.2.20 仮想ネットワークデバイスを設定する

仮想ネットワークデバイスを設定するには、次のコマンドを使用します。

表 3-23 仮想ネットワークデバイスを設定するコマンド

コマンド	機能概要
ldm add-vnet	仮想ネットワークデバイスを追加します。
ldm remove-vnet	仮想ネットワークデバイスを削除します。

ldm add-vnetコマンド

論理ドメインに仮想ネットワークデバイスを追加するには、ldm add-vnetを使用します。

```
# ldm add-vnet if-name vswitch-name ldom
```

- if-name
  仮想ネットワークデバイスのインスタンスに割り当てられるインターフェース名を指定します。
- vswitch-name
  仮想スイッチサービス名を指定します。
- ldom
  論理ドメイン名を指定します。

ldm remove-vnetコマンド

仮想ネットワークデバイスを削除するには、ldm remove-vnetを使用します。

```
# ldm remove-vnet if-name ldom
```

- if-name
  インターフェース名を指定します。
- ldom
  論理ドメイン名を指定します。

3.2.21 仮想ディスクサーバを設定する

仮想ディスクサーバを設定するには、次のコマンドを使用します。

表 3-24 仮想ディスクサーバを設定するコマンド

コマンド	機能概要
ldm add-vdiskserverdevice	仮想ディスクサーバにデバイスボリュームを追加します。
ldm remove-vdiskserverdevice	仮想ディスクサーバからデバイスボリュームを削除します。

ldm add-vdiskserverdeviceコマンド

仮想ディスクサーバにボリュームを追加するには、ldm add-vdiskserverdeviceコマンドを使用します。

```
# ldm add-vdiskserverdevice backend volume-name@service-name
```

- backend
  物理ディスクのパスを指定します。backendには、ファイルのほかに、ディスク、ディスクのスライスなどのパスが指定できます。
- volume-name
  仮想ディスクサーバ内で一意のボリューム名を指定します。
- service-name
  仮想ディスクサービスの名前を指定します。

次の例では、100 GBのファイルを作成して、それをディスクサーバprimary-vds0のボリュームとして追加しています。

```
# mkfile 100g /ldoms/domain/test/fdisk1
# ldm add-vdiskserverdevice /ldoms/domain/test/fdisk1 vol2@primary-vds0
```

ldm remove-vdiskserverdeviceコマンド

仮想ディスクサーバからボリュームを削除するには、ldm remove-vdiskserverdeviceコマンドを使用します。

```
# ldm remove-vdiskserverdevice volume-name@service-name
```

- volume-name
  ボリューム名を指定します。
- service-name
  仮想ディスクサーバ名を指定します。

次の例では、仮想ディスクサーバprimary-vds0からボリュームvol2を削除しています。

```
# ldm remove-vdiskserverdevice vol2@primary-vds0
```

3.2.22 仮想ディスクを設定する

仮想ディスクを設定するには、次のコマンドを使用します。

表 3-25 仮想ディスクを設定するコマンド

コマンド	機能概要
ldm add-vdisk	仮想ディスクを追加します。
ldm remove-vdisk	仮想ディスクを削除します。

ldm add-vdiskコマンド

仮想デバイスを仮想ディスクとしてゲストドメインに追加するには、ldm add-vdiskコマンドを使用します。

```
# ldm add-vdisk disk-name volume-name@service-name ldom
```

- disk-name
  仮想ディスク名を指定します。
- volume-name
  接続する既存の仮想ディスクサーバのデバイス名を指定します。
- service-name
  接続する既存の仮想ディスクサーバ名を指定します。
- ldom
  論理ドメインを指定します。

次の例では、仮想ディスクサーバprimary-vds0に追加したデバイスvol2を、仮想ディスクvdisk1としてゲストドメインldom1に追加しています。

```
# ldm add-vdisk vdisk1 vol2@primary-vds0 ldom1
```

ldm remove-vdiskコマンド

ゲストドメインから仮想ディスクを削除するには、ldm remove-vdiskコマンドを使用します。

```
# ldm remove-vdisk disk-name ldom
```

- disk-name

仮想ディスク名を指定します。

- ldom
  論理ドメイン名を指定します。

次の例では、ゲストドメインldom1から仮想ディスクvdisk2を削除しています。

```
# ldm remove-vdisk vdisk2 ldom1
```

3.2.23 仮想コンソールを設定する

ゲストドメインに仮想コンソールを割り当てたり、ゲストドメインのコンソールログをファイルに書き出すように設定したりするには、ldm set-vconsoleコマンドを使用します。

```
# ldm set-vconsole port=port-num ldom
# ldm set-vconsole [service=vcc-server] log=on ldom
```

- port-num
  仮想コンソールのポート番号を指定します。
- ldom
  論理ドメイン名を指定します。
- vcc-server
  仮想コンソール端末集配信装置サービスを制御ドメイン以外の論理ドメインに指定した場合、その論理ドメイン名を指定します。これにより、指定した論理ドメインに対して、*ldom* に指定したゲストドメインのコンソール出力およびログが出力されます。

注－service=vcc-serverによる論理ドメインの指定は、仮想コンソール端末配信装置サービスがOracle Solaris 11上で提供されているときにサポートされます。

注－ログの内容は、ldm add-vconsconで指定したサービスドメインの/var/log/vntsd/(ゲストドメイン名)/console-logファイルとして書き出されます。

3.2.24 起動デバイスを設定する

ゲストドメインの起動デバイスを設定するには、ldm set-variableコマンドを使用します。また、ゲストドメインの起動とともにOracle Solarisを自動的に起動するかどうかを設定するときにも、ldm set-variableコマンドを使用します。デフォルトでは、ゲストドメインとともにOracle Solarisが自動的に起動します。

```
# ldm set-variable boot-device=disk-name ldom
# ldm set-variable auto-boot?={true|false} ldom
```

- disk-name
  仮想ディスク名を指定します。
- ldom
  論理ドメイン名を指定します。
- true|false
  trueを指定すると自動的に起動します。falseを指定すると自動的に起動しません。

注－OpenBoot PROMのデバイスエイリアスには大文字は使用できません。ldm add-vdiskコマンドで追加した仮想ディスク名に大文字が含まれる場合は、仮想ディスク名をすべて小文字に変更してから、OpenBoot PROM環境変数boot-deviceに設定してください。

3.2.25 リソースをバインドする

ゲストドメインにリソースをバインドするには、ldm bind-domainコマンドを使用します。

```
# ldm bind-domain ldom
```

- ldom
  論理ドメイン名を指定します。

3.2.26 ゲストドメインを起動する

ゲストドメインを起動するには、ldm start-domainコマンドを使用します。

```
# ldm start-domain ldom
```

- ldom
  論理ドメイン名を指定します。

3.2.27 シャットダウングループを指定する

シャットダウングループを指定するには、ldm set-domainコマンドを使用します。

```
# ldm set-domain shutdown-group=num ldom
```

- num
  シャットダウングループを指定します。1から15までの数値です。
- ldom
  対象論理ドメインを指定します。primary（制御ドメイン）は指定できません。

注ーシャットダウングループを変更できるのは、ゲストドメインがinactiveまたはboundの状態のときだけです。

3.2.28 デバイスを再構成する

デバイスの再構成を行うときには、devfsadmコマンドを使用します。devfsadmコマンドは、仮想ディスクを割り当てたゲストドメインにログインして実行します。

```
guest# devfsadm
```

devfsadmコマンドは、すべてのドライバをシステムに読み込み、可能な限りのデバイスに接続しようとします。そして、デバイスファイルを /devicesディレクトリに作成し、論理リンクを/devディレクトリに作成します。devfsadmコマンドの詳細は、Oracle Solarisのリファレンスマニュアルを参照してください。

3.2.29 復旧モードを有効にする

復旧モードを有効にするには、svccfgコマンドを実行してldmd/recovery_modeプロパティーをautoに設定したあと、svcadmコマンドを実行して設定値を有効にします。なお、復旧モードは、Oracle VM Server for SPARC 3.3からデフォルトで有効になります。

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

XCP 2230からOracle VM Server for SPARC 3.1より提供された復旧モード（recovery mode）がサポートされました。
復旧モードの詳細は、お使いのバージョンの『Oracle VM Server for SPARC 管理ガイド』の「ハードウェアエラーの処理」参照してください。また、復旧モードに必要な修正は、お使いのバージョンの『Oracle VM Server for SPARC リリースノート』を参照してください。

第4章

物理パーティションの構築

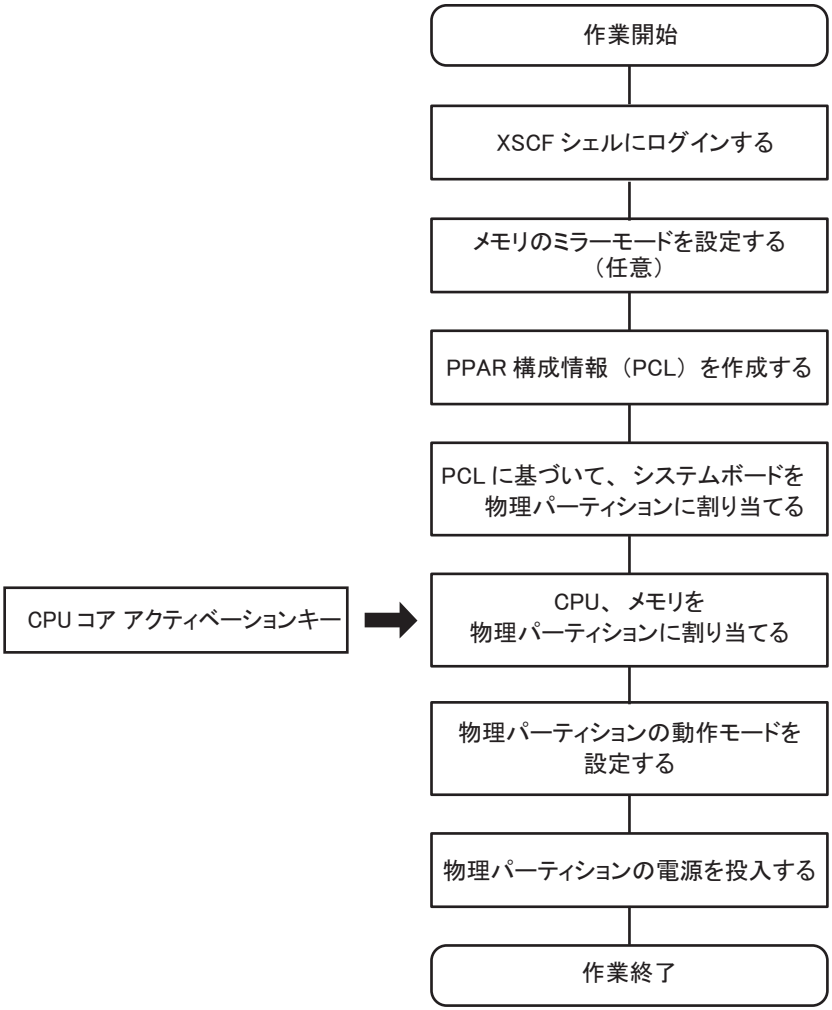
ここでは、物理パーティションの構築について説明します。

- 物理パーティション構築のながれ
- 物理パーティションの構築と操作

4.1 物理パーティション構築のながれ

ドメインを構築するときには、最初に物理パーティションを構築します。物理パーティションは、XSCFファームウェアを使用して構築します。図 4-1は、ビルディングブロック構成の場合における物理パーティションの構築のながれを示しています。

図 4-1　物理パーティションの構築



4.2　物理パーティションの構築と操作

ここでは、物理パーティションを構築する場合の留意点と構築例を説明します。

- 推奨される物理パーティション番号（PPAR-ID）の決定方法
  PPAR-IDには、システム構成によって0~15の任意の数字を指定できます。しかしながら、PPAR-IDは、SPARC M12/M10システムに搭載されている筐体のBB-IDを指定する必要があります。SPARC M12/M10システムに搭載されていない筐体のBB-IDをPPAR-IDに指定すると、その物理パーティションは電源の投入に失敗し、使用できません。

  例えば、システムに搭載されたSPARC M12-2S/M10-4SのBB-IDがそれぞれ

BB#00、BB#01、BB#02、BB#03の場合、物理パーティション構築時にPPAR-IDとして指定できる番号は0、1、2、または3です。

また、運用開始後にリソースを有効に活用するため、SPARC M12-2S/M10-4Sを減設することがあります。減設するSPARC M12-2S/M10-4SのBB-IDと同じPPAR-IDがシステムにある場合、その物理パーティションを停止させなければなりません。その場合は、物理パーティションの再構築が必要になる場合があります。そのため、以下を考慮して物理パーティションを構築することを推奨します。

【2BB構成の場合】

- 1つだけの物理パーティションでシステムを構築する場合は、PPAR-ID 0を設定します。PPAR-IDと異なるBB#01を減設できます。
- 2つの物理パーティションでシステムを構築する場合は、BB#00はPPAR-ID 0、BB#01はPPAR-ID 1に割り当てます。

これらにより、システムに残すべきSPARC M12-2S/M10-4Sの物理パーティションを停止せずに、SPARC M12-2S/M10-4Sを減設することができます。

例：2つの物理パーティションを構築する場合

【推奨するパターン】
PPAR-ID 0に、BB#00を、PPAR-ID 1にBB#01を割り当てます。
SPARC M12-2S/M10-4Sを減設するときに、そのBB-IDと異なる番号の物理パーティションは影響を受けません。

PPAR-ID 0 [BB#00]
PPAR-ID 1 [BB#01]

【推奨しないパターン】
PPAR-ID 0に、BB#01を、PPAR-ID 1にBB#00を割り当てます。SPARC M12-2S/M10-4Sを減設するときに、減設しない側の物理パーティションの停止も必要です。
その結果、物理パーティションの再構築が必要になります。

PPAR-ID 0 [BB#01]
PPAR-ID 1 [BB#00]

【3BB以上の筐体間直結の構成の場合】

- 1つだけの物理パーティションでシステムを構築する場合は、PPAR-ID 0を設定します。PPAR-IDと異なるBB-IDのSPARC M12-2S/M10-4Sを減設できます。
- 2つ以上の物理パーティションでシステムを構築する場合、減設を想定しているSPARC M12-2S/M10-4S(*1)のBB-IDをPPAR-IDに設定しないようにします。該当のSPARC M12-2S/M10-4Sを減設しても、そのBB-IDと異なる番号の物理パーティションは影響を受けません。
  *1: 減設後、マスタ／スタンバイのSPARC M12-2S/M10-4Sをシステムに残す必要があります。したがって、減設を想定したSPARC M12-2S/M10-4Sとは、BB-IDがBB#00/BB#01以外のものが対象です。

これらにより、システムに残すべきSPARC M12-2S/M10-4Sの物理パーティションを停止せずに、SPARC M12-2S/M10-4Sを減設することができます。

例：筐体間直結の4BB構成で、2つの物理パーティションを構築する場合

【推奨するパターン1: 各物理パーティションのリソースを減らすことを想定する場合】

PPAR-ID 0に、BB#00とBB#02を、PPAR-ID 1にBB#01とBB#03を割り当てます。
各物理パーティションを停止せず、PPAR-IDと異なるBB#02またはBB#03のSPARC M12-2S/M10-4Sを減設することができます。

PPAR-ID 0 [BB#00、BB#02]
PPAR-ID 1 [BB#01、BB#03]

【推奨するパターン2: 物理パーティションの数を減らすことを想定する場合】

PPAR-ID 0に、BB#00とBB#01を、PPAR-ID 2にBB#02とBB#03を割り当てます。
システムに残すべきSPARC M12-2S/M10-4Sの物理パーティションを停止せず、BB#02またはBB#03のSPARC M12-2S/M10-4Sを減設することができます。

PPAR-ID 0 [BB#00、BB#01]
PPAR-ID 2 [BB#02、BB#03]

【推奨しないパターン】

PPAR-ID 0に、BB#00とBB#01を、PPAR-ID 1にBB#02とBB#03を割り当てます。
PPAR-ID 0にあるSPARC M12-2S/M10-4Sはマスタ／スタンバイのため、減設できません。

PPAR-ID 0 [BB#00、BB#01]
PPAR-ID 1 [BB#02、BB#03]

【クロスバーボックス経由接続の構成の場合】

物理パーティションを2BB以上で構成する場合、減設を想定しているSPARC M12-2S/M10-4SのBB-IDをPPAR-IDに設定しないでください。

これにより、システムに残すべきSPARC M12-2S/M10-4Sの物理パーティションを停止せずに、SPARC M12-2S/M10-4Sを減設することができます。

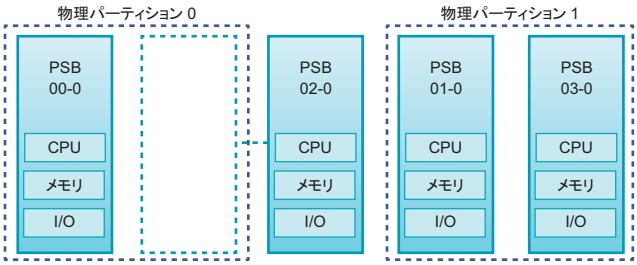
以降、SPARC M10-4S 4台による構成で、2つの物理パーティションを構築する例を使用して具体的な手順を説明します。

- PSB（BB）の割り当て
  システムボード（PSB 00-0とPSB 02-0）を物理パーティション0に、システムボード（PSB 01-0とPSB 03-0）を物理パーティション1に割り当てます。ただし、PSB 02-0は、実際には割り当てず、割り当て可能状態にしておきます。
- ミラーモード設定（任意）
  PSB 00-0だけメモリミラーを設定します。

- コンフィグレーションポリシー／メモリ無効化／I/O無効化
  物理パーティション0と物理パーティション1のコンフィグレーションポリシーはsystemとし、メモリ無効化とI/O無効化は行わないものとします。
- 物理パーティションの動作モード
  診断メッセージのメッセージレベルを「標準」に設定します。

図 4-2は、構築後の物理パーティションのイメージを示しています。

図 4-2　構築後の物理パーティションのイメージ



1. **platadm権限を持つユーザーアカウントで、XSCFシェルにログインします。**
2. **setupfruコマンドを実行し、メモリミラーを設定します。**

```
XSCF> setupfru -m y sb 00-0
```

3. **必要に応じて、showfruコマンドを実行し、メモリミラー状態を確認します。**
   システムボード（PSB 00-0）のメモリミラーの状態を確認します。[Memory Mirror Mode] がすべて「yes」になっていれば、正しくミラー状態になっています。

```
XSCF> showfru sb 00-0
Device     Location      Memory Mirror Mode
sb         00-0
   cpu     00-0-0        yes
   cpu     00-0-1        yes
   cpu     00-0-2        yes
   cpu     00-0-3        yes
XSCF>
```

4. **setpclコマンドを実行し、物理パーティション構成情報（PPAR構成情報）を作成します。**
   物理システムボード（PSB 00-0とPSB 02-0）をそれぞれ物理パーティション0の論理システムボード（LSB 0とLSB 1）に対応付け、物理システムボード（PSB 01-0とPSB 03-0）をそれぞれ物理パーティション1の論理システムボード（LSB 0とLSB 1）に対応付けます。なおコンフィグレーションポリシーはsystem、メモリ無効化とI/O無効化は行わないものとします。

```
XSCF> setpcl -p 0 -a 0=00-0 1=02-0
XSCF> setpcl -p 0 -s policy=system
XSCF> setpcl -p 0 -s no-mem=false
XSCF> setpcl -p 0 -s no-io=false
XSCF> setpcl -p 1 -a 0=01-0 1=03-0
XSCF> setpcl -p 1 -s policy=system
XSCF> setpcl -p 1 -s no-mem=false
XSCF> setpcl -p 1 -s no-io=false
```

5. **showpclコマンドを実行し、PPAR構成情報の内容を確認します。**
   [PPAR-ID] 欄は物理パーティション番号、 [LSB] 欄は論理システムボード番号、[PSB] 欄はPSB番号を表します。

```
XSCF> showpcl -v -a
PPAR-ID   LSB   PSB   Status   No-Mem   No-IO    Cfg-policy
00                  Powered Off
                                                System
          00    00-0           False    False
          01    02-0           False    False
01                  Powered Off
                                                System
          00    01-0           False    False
          01    03-0           False    False
```

6. **showboardsコマンドを実行し、PSBを確認します。**
   [PPAR-ID] 欄が「SP」と表示され、 [Assignment] 欄が「Available」になっているものが割り当て可能なPSBです。「SP」はシステムボードプール状態（どの物理パーティションにも割り当てられていない状態）を表します。

```
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
02-0 SP           Available   n    n    n    Passed  Normal
03-0 SP           Available   n    n    n    Passed  Normal
```

7. **addboardコマンドを実行し、PSBを割り当てます。**

```
XSCF> addboard -c assign -p 0 00-0
XSCF> addboard -c assign -p 1 01-0 03-0
```

8. **showboardsコマンドを実行し、PSBの割り当て状態を確認します。**
   割り当てたPSBの [R] 欄に「*」が表示され、 [PPAR-ID] 欄に設定したとおりの物理パーティションおよび論理システムボード（LSB）が設定されていれば、PSBは正しく割り当てられています。

```
XSCF> showboards -v -a
PSB   R PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- - ------------ ----------- ---- ---- ---- ------- --------
00-0 * 00(00)       Assigned    n    n    n    Passed  Normal
01-0 * 01(00)       Assigned    n    n    n    Passed  Normal
02-0   SP           Available   n    n    n    Passed  Normal
03-0 * 01(01)       Assigned    n    n    n    Passed  Normal
```

9. **showcodactivationコマンドを実行し、システムにCPUコア アクティベーションキーが登録されているかどうかを確認します。**
   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ません。手順10以降でCPUコア アクティベーションキーを追加します。インデックス番号が表示された場合はCPUコア アクティベーションキーはXSCFに登録されています。手順13からの作業に進んでください。

```
XSCF> showcodactivation
Index   Description Count
------- ----------- ------
```

10. **addcodactivationコマンドを実行し、CPUコア アクティベーションキーを登録します。**

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber: 10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtxgvZPsrtYguqiNUieB+mTDC
 :
 :
b1GCkFx1RH27FdVHiB2H0A=="
Above Key will be added, Continue?[y|n]:y
```

11. **showcodactivationコマンドを実行し、追加したCPUコア アクティベーションキーの内容を確認します。**
    購入したリソースのCPUコア アクティベーション数が表示されれば、CPUコア アクティベーションキーが正しく登録されています。

```
XSCF> showcodactivation
Index   Description Count
------- ----------- ------
      0 PROC            2
      1 PROC            2
      2 PROC            2
      3 PROC            2
```

12. **setcodコマンドを実行し、CPUコアリソースを物理パーティションに割り当てます。**

```
XSCF> setcod -p 0 -s cpu -c set 4
PROC Permits assigned for PPAR 0 : 0 -> 4

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.

XSCF> setcod -p 1 -s cpu -c set 4
PROC Permits assigned for PPAR 1 : 0 -> 4

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注ーXCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

---

13. **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
XSCF> resetdateoffset -p 1
```

14. **setpparmodeコマンドを実行し、診断メッセージの詳細レベルを「標準」にします。**

```
XSCF> setpparmode -p 0 -m message=normal
XSCF> setpparmode -p 1 -m message=normal
```

15. **poweronコマンドを実行し、物理パーティションの電源を投入します。**
次の例では、物理パーティション0と1の電源を投入しています。

```
XSCF> poweron -p 0
PPAR-IDs to power on:00
Continue? [y|n] :y
00 :Powering on

*Note*
This command only issues the instruction to power-on.
The result of the instruction can be checked by the "showlogs power".

XSCF> poweron -p 1
PPAR-IDs to power on:01
Continue? [y|n] :y
01 :Powering on
```

```
*Note*
This command only issues the instruction to power-on.
The result of the instruction can be checked by the "showlogs power".
```

16. **showboardsコマンドを実行し、PSBの状態を確認します。**
    [Pwr] 欄、 [Conn] 欄、 [Conf] 欄がすべて「y」になっていれば、物理パーティションが正しく稼働しています。

```
XSCF> showboards -a
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 01(00)       Assigned    y    y    y    Passed  Normal
02-0 SP           Available   n    n    n    Passed  Normal
03-0 01(01)       Assigned    y    y    y    Passed  Normal
```

第5章

論理ドメインの構築例

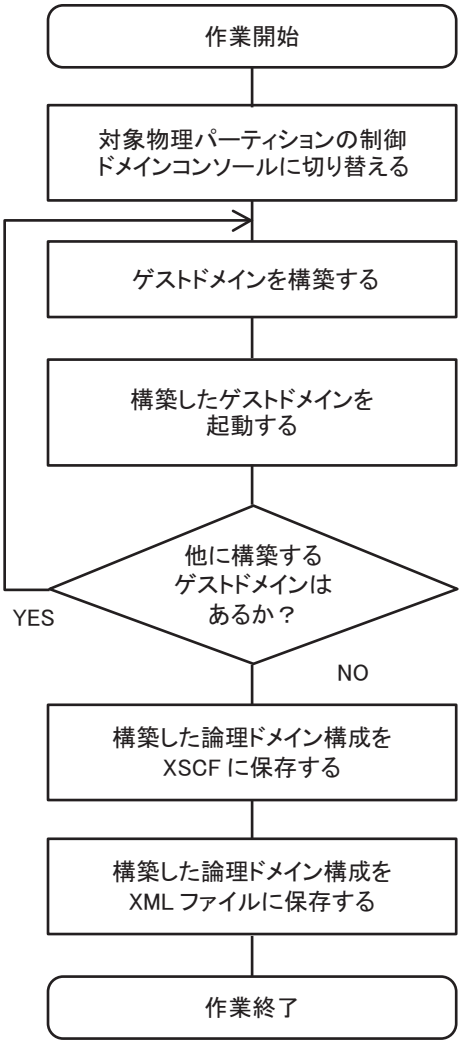
ここでは、論理ドメインの構築例を示します。

- 論理ドメイン構築のながれ
- 論理ドメイン構築の操作例

5.1 論理ドメイン構築のながれ

物理パーティションを構築し、電源を投入したあと、論理ドメインを構築します。論理ドメインは、Oracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用して構築します。図 5-1は、論理ドメインの構築のながれを示しています。

図 5-1　論理ドメインの構築



物理パーティションの構築時に割り当てたシステムボード上のハードウェアリソースは、すべて制御ドメインに割り当てられています。このため、論理ドメインを構築するときに、ハードウェアリソースを必要に応じてゲストドメインおよび各種サービスドメインに割り当てます。

また、論理ドメインの構築が完了したら、論理ドメインの構成情報をXSCFに保存します。物理パーティションの電源を投入したときには、この構成情報に従って物理パーティション内のすべての論理ドメインが起動されます。

5.2 論理ドメイン構築の操作例

ここでは、論理ドメイン構築の具体例を説明します。

論理ドメインを構築するには、各種名称を定義したり、各種リソースやその個数を指定したりする必要があります。それらの情報は作業を開始する前に整理しておいてください。

本節の例で使用するものを次に示します。表 5-1は各種名称の定義例、表 5-2は制御ドメインに割り当てるリソース例、表 5-3はゲストドメインに割り当てるリソース例をそれぞれ示しています。

表 5-1 各種名称の定義例

項目	名前
仮想コンソール端末集配信装置名	primary-vcc0
仮想ディスクサーバ名	primary-vds0
仮想スイッチサービス名	primary-vsw0
仮想ネットワークデバイスのインターフェース名	vnet1
仮想デバイス名	vol1
仮想ディスク名	vdisk1
ゲストドメインのドメイン名	ldom1
論理ドメイン構成情報名	config1

表 5-2 制御ドメインに割り当てるリソース例

項目	値
仮想コンソール端末のポート番号の範囲	5000から 5100
仮想スイッチサービスで使用するデバイス	net0
仮想ディスクサービスで使用するデバイス	/ldoms/domain/test/fdisk0
割り当てるCPUの数量	4コア
割り当てるメモリのサイズ	8 GB

表 5-3 ゲストドメインに割り当てるリソース例

項目	値
割り当てる仮想コンソール端末のポート番号	5000
割り当てるCPUの数量	2コア
割り当てるメモリのサイズ	4 GB

5.2.1 制御ドメインにログインする

ここでは、ターゲットの物理パーティション番号は0であるとして説明します。

1. **platadm権限を持つユーザーアカウントで、XSCFシェルにログインします。**
2. **poweronコマンドを実行し、物理パーティション0の電源を投入します。**

```
XSCF> poweron -p 0
PPAR-IDs to power on:00
Continue? [y|n] :y
00 :Powering on
*Note*
This command only issues the instruction to power-on.
The result of the instruction can be checked by the "showlogs power".
XSCF>
```

3. **consoleコマンドを実行し、物理パーティション0の制御ドメインコンソールに切り替えます。**

```
XSCF> console -p 0
Console contents may be logged. Connect to PPAR-ID 0?[y|n] :y
```

注ーOracle Solarisが自動的に起動しない設定のときは、起動の操作を行い、起動完了を待ちます。

---

4. **rootユーザーとして制御ドメインにログインします。**
5. **svcs ldmdコマンドを実行し、Logical Domains Managerが稼働していることを確認します。**
   Logical Domains Managerのサービス名はldmdです。[STATE]欄に「online」が表示されている場合、稼働状態です。

```
# svcs ldmd
STATE  STIME    FMRI
online 16:25:31 svc:/ldoms/ldmd:default
```

注ーLogical Domains Managerが起動されていなければ、svcadmコマンドを実行し、起動します。svcadmコマンドの詳細は、Oracle Solarisのリファレンスマニュアルを参照してください。

5.2.2 デフォルトサービスを設定する

1. **ldm add-vconsconコマンドを実行し、制御ドメインに仮想コンソール端末集配**

信装置サービスを追加します。
次の例では、制御ドメインに、ポート番号5000から5100までを扱うprimary-vcc0という名前の仮想コンソール端末集配信装置を追加しています。

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

2. **ldm add-vdiskserverコマンドを実行し、制御ドメインに仮想ディスクサーバを追加します。**
次の例では、制御ドメインにprimary-vds0という名前の仮想ディスクサーバを追加しています。

```
# ldm add-vdiskserver primary-vds0 primary
```

3. **ldm add-vswitchコマンドを実行し、制御ドメインに仮想スイッチサービスを追加します。**
次の例では、制御ドメインにprimary-vsw0という名前の仮想スイッチサービスを追加し、デバイスとしてnet0を使用しています。

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

4. **ldm list-servicesコマンドを実行し、設定したサービスを確認します。**
次の例では、制御ドメインのサービスを表示させています。

```
# ldm list-services primary
```

5.2.3 制御ドメインを初期設定する

1. **ldm start-reconfコマンドを実行し、遅延再構成モードを開始します。**
制御ドメイン名は、primaryで固定です。

```
# ldm start-reconf primary
```

2. **ldm list-permitsコマンドを実行し、割り当て可能なCPUコアの数を確認します。**

```
# ldm list-permits
```

3. **ldm set-coreコマンドを実行し、制御ドメインに仮想CPUを割り当てます。**
次の例では、制御ドメインにコア4個分のCPUを割り当てています。

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

4. **ldm set-memoryコマンドを実行し、仮想メモリを割り当てます。**

次の例では、制御ドメインに8 GBのメモリを割り当てています。

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

5. **ldm list-bindings**コマンドを実行し、リソースの割り当て状況を確認します。

```
# ldm list-bindings primary
```

6. **ldm add-spconfig**コマンドを実行し、論理ドメイン構成を**XSCF**に保存します。
   次の例では、論理ドメイン構成情報を構成名config1として保存しています。

```
# ldm add-spconfig config1
```

7. **XSCF**ファームウェアの**poweroff**コマンドと**poweron**コマンドを実行し、物理パーティションの電源を切断および再投入します。

```
XSCF> poweroff -p 0
XSCF> poweron -p 0
```

8. 再起動後、**list-bindings**コマンドを実行し、リソースの割り当て状況を確認します。
   次のコマンドの表示が、手順4の割り当て状況と同じであれば正しく設定が行われています。

```
# ldm list-bindings primary
```

5.2.4 ゲストドメインを構築する

1. **ldm list-devices**コマンドを実行し、ゲストドメインへ割り当て可能なリソースを確認します。
   ドメインに利用されていないリソースが表示されます。

```
# ldm list-devices
```

2. **ldm add-domain**コマンドを実行し、ゲストドメイン**ldom1**を作成します。

```
# ldm add-domain ldom1
```

3. **ldm list-permits**コマンドを実行し、割り当て可能な**CPU**コアの数を確認します。

```
# ldm list-permits
```

4. **ldm set-core**コマンドを実行し、**2**コア分の仮想**CPU**を追加します。

```
# ldm set-core 2 ldom1
```

5. **ldm add-memory**コマンドを実行し、**4 GB**の仮想メモリを追加します。

```
# ldm add-memory 4g ldom1
```

6. **ldm add-vnet**コマンドを実行し、仮想ネットワークデバイスを追加します。

```
# ldm add-vnet vnet1 primary-vsw0 ldom1
```

7. **ldm add-vdiskserverdevice**コマンドを実行し、仮想ディスクサーバにデバイスを追加します。
   次の例では、データの格納場所として、mkfileコマンドを使用して制御ドメイン内に100 GBの通常ファイルを作成し、それをデバイスとして追加しています。

```
# mkfile 100g /ldoms/domain/test/fdisk0
# ldm add-vdiskserverdevice /ldoms/domain/test/fdisk0 vol1@primary-vds0
```

8. **ldm add-vdisk**コマンドを実行し、仮想ディスクを追加します。
   手順6で追加した仮想ディスクをゲストドメインに追加します。

```
# ldm add-vdisk vdisk1 vol1@primary-vds0 ldom1
```

9. **ldm set-vconsole**コマンドを実行し、ポート番号**5000**の仮想コンソールを割り当てます。

```
# ldm set-vconsole port=5000 ldom1
```

10. **ldm set-variable**コマンドを実行し、ゲストドメインの起動デバイスを設定します。

```
# ldm set-variable boot-device=vdisk1 ldom1
```

注－OpenBoot PROMのデバイスエイリアスには大文字は使用できません。ldm add-vdiskコマンドで追加した仮想ディスク名に大文字が含まれる場合は、仮想ディスク名をすべて小文字に変更してから、OpenBoot PROM環境変数boot-deviceに設定してください。

---

11. **ldm list-bindings**コマンドを実行し、ゲストドメインの構成を確認します。

```
# ldm list-bindings ldom1
```

12. **ldm bind-domainコマンドとldm start-domainコマンドを実行し、ゲストドメインを起動します。**

```
# ldm bind-domain ldom1
# ldm start-domain ldom1
```

5.2.5 復旧モード（recovery mode）を設定する

復旧モードを設定するには、svccfgコマンドとsvcadmコマンドを使用します。

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

XCP 2230からOracle VM Server for SPARC 3.1より提供された復旧モード（recovery mode）がサポートされました。
復旧モードの詳細は、お使いのバージョンの『Oracle VM Server for SPARC 管理ガイド』の「ハードウェアエラーの処理」参照してください。また、復旧モードに必要な修正は、お使いのバージョンの『Oracle VM Server for SPARC リリースノート』を参照してください。

5.2.6 論理ドメインの構成情報を保存する

すべてのゲストドメインの設定が完了したら、論理ドメインの構成情報を保存します。

1. **ldm add-spconfigコマンドを実行し、論理ドメイン構成をXSCFに追加します。**
   次の例では、論理ドメイン構成情報を構成名config2として保存しています。

```
# ldm add-spconfig config2
```

2. **ldm list-constraintsコマンドを実行し、論理ドメインの構成情報をXMLファイルに保存します。**
   次の例では、論理ドメイン構成情報をconfig2.xml ファイルに保存しています。

```
# ldm list-constraints -x > config2.xml
```

第6章

物理パーティションの再構築

ここでは、ビルディングブロック構成時の物理パーティションの再構築例を示します。

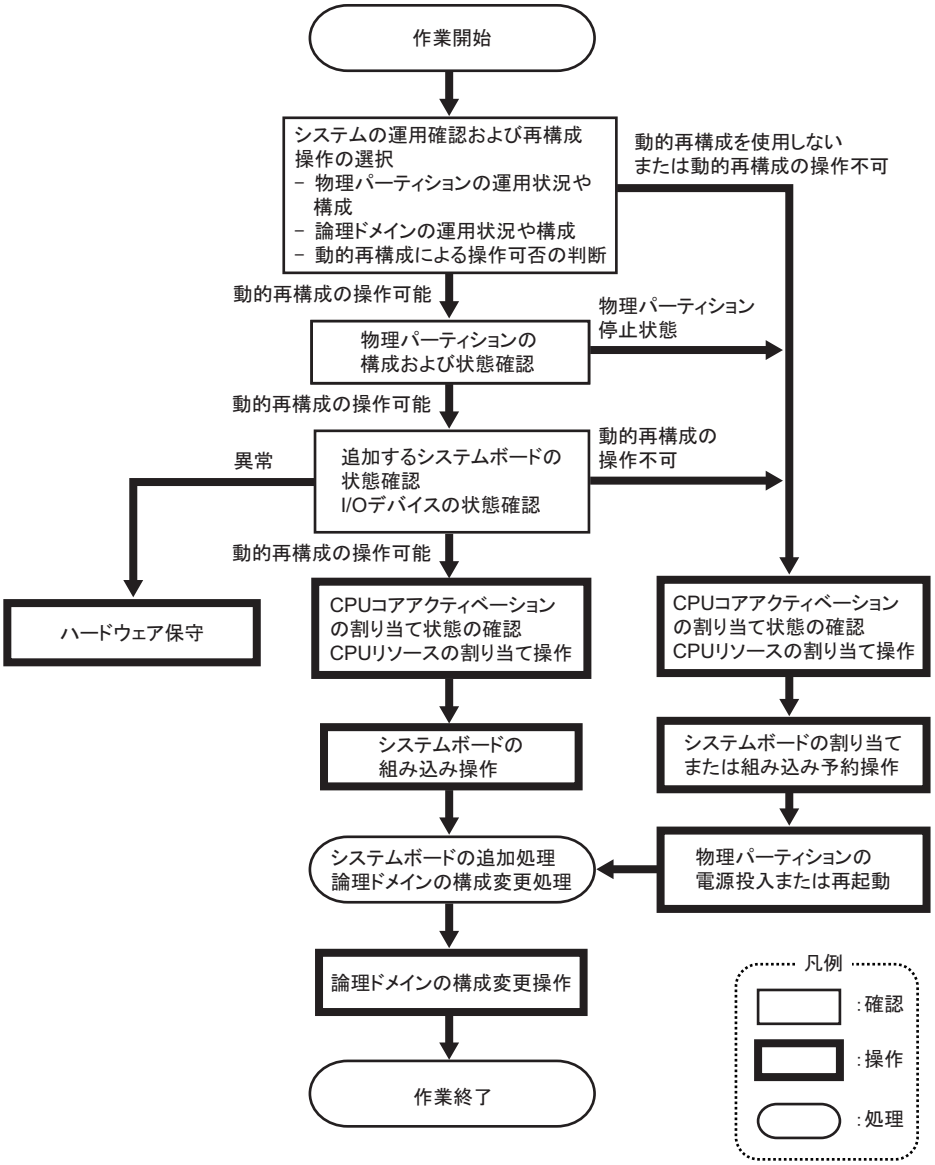
- 物理パーティション再構築のながれ
- システムボードを追加する操作例
- システムボードを削除する操作例
- システムボードを移動する操作例
- システムボードを交換する操作例
- 物理パーティション番号を変更する場合の留意点

6.1 物理パーティション再構築のながれ

ここでは、システムボード（PSB）の追加、削除、移動、交換といった物理パーティション再構築での操作のながれをフロー図で説明します。

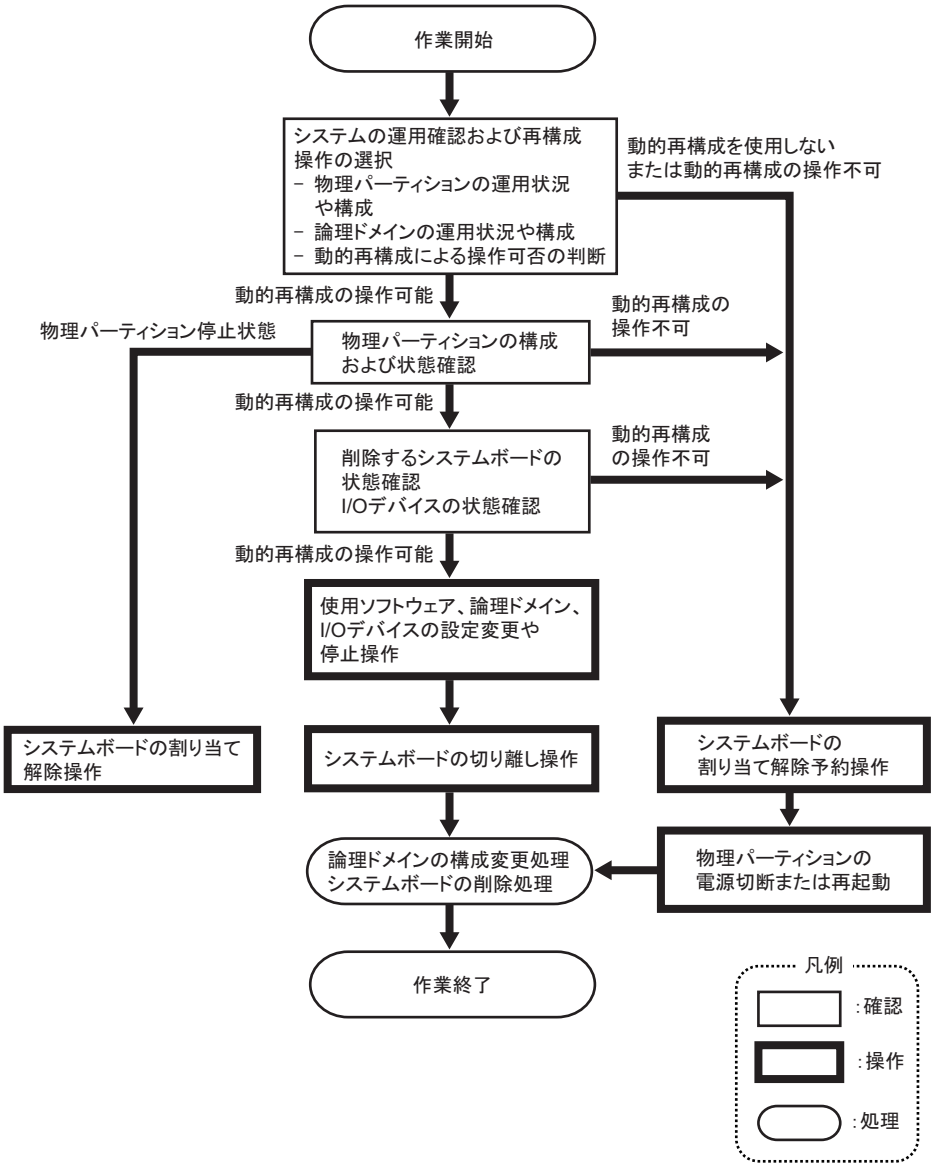
6.1.1 システムボードを追加する操作のながれ

図 6-1　　システムボードを追加する操作のながれ



6.1.2 システムボードを削除する操作のながれ

図 6-2 システムボードを削除する操作のながれ



6.1.3 システムボードを移動する操作のながれ

図 6-3 システムボードを移動する操作のながれ（1）

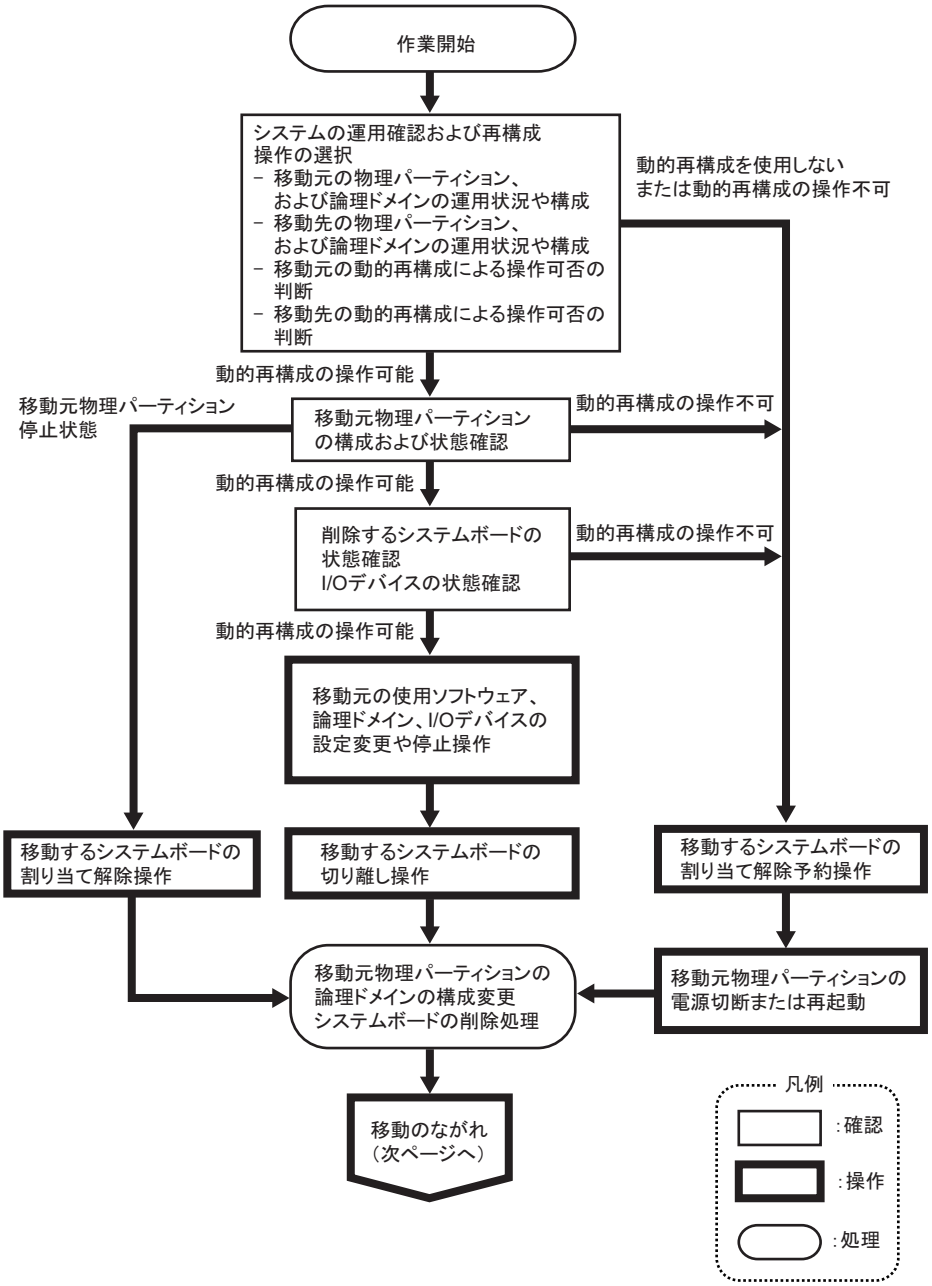
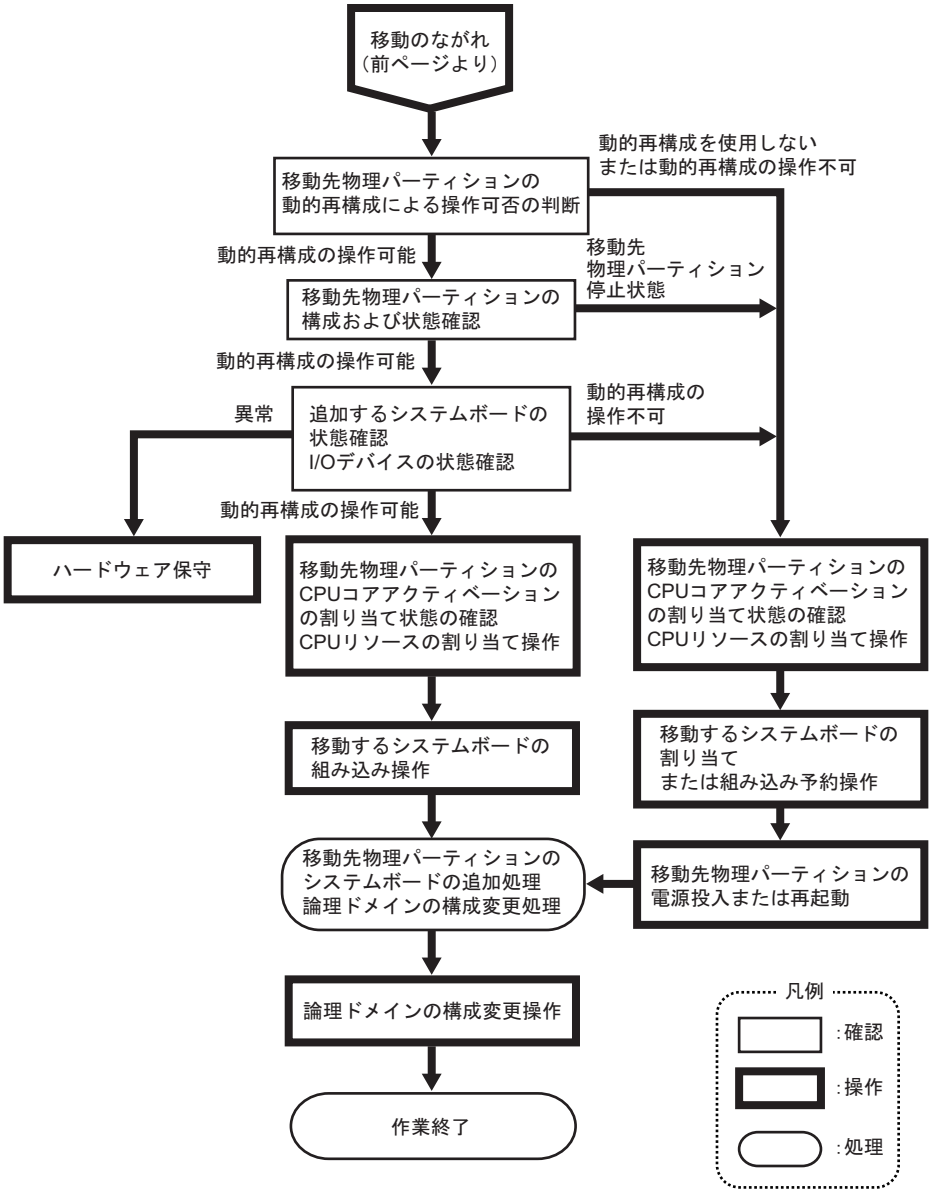


図 6-4　システムボードを移動する操作のながれ（2）



6.1.4 システムボードを交換する操作のながれ

図 6-5 システムボードを交換する操作のながれ（1）

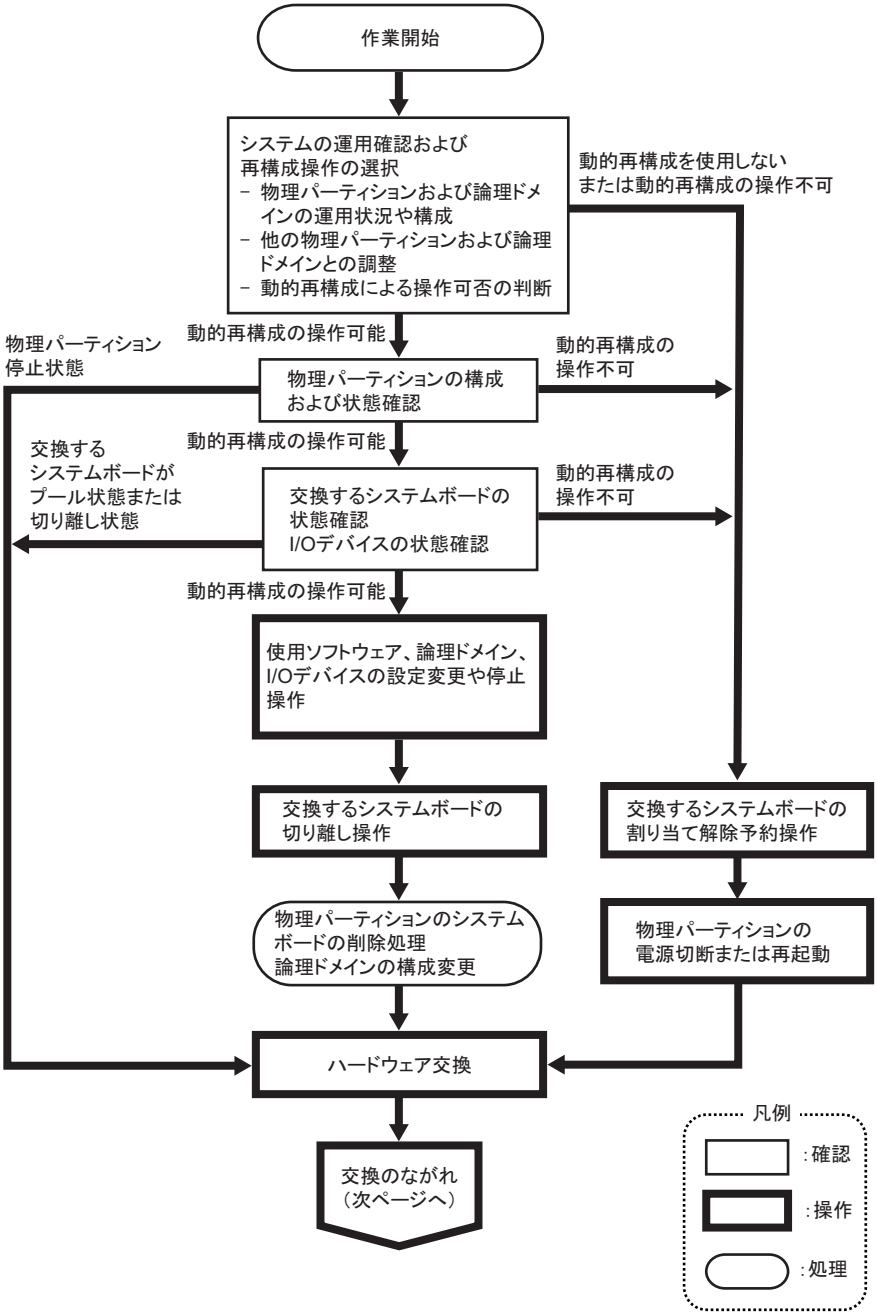
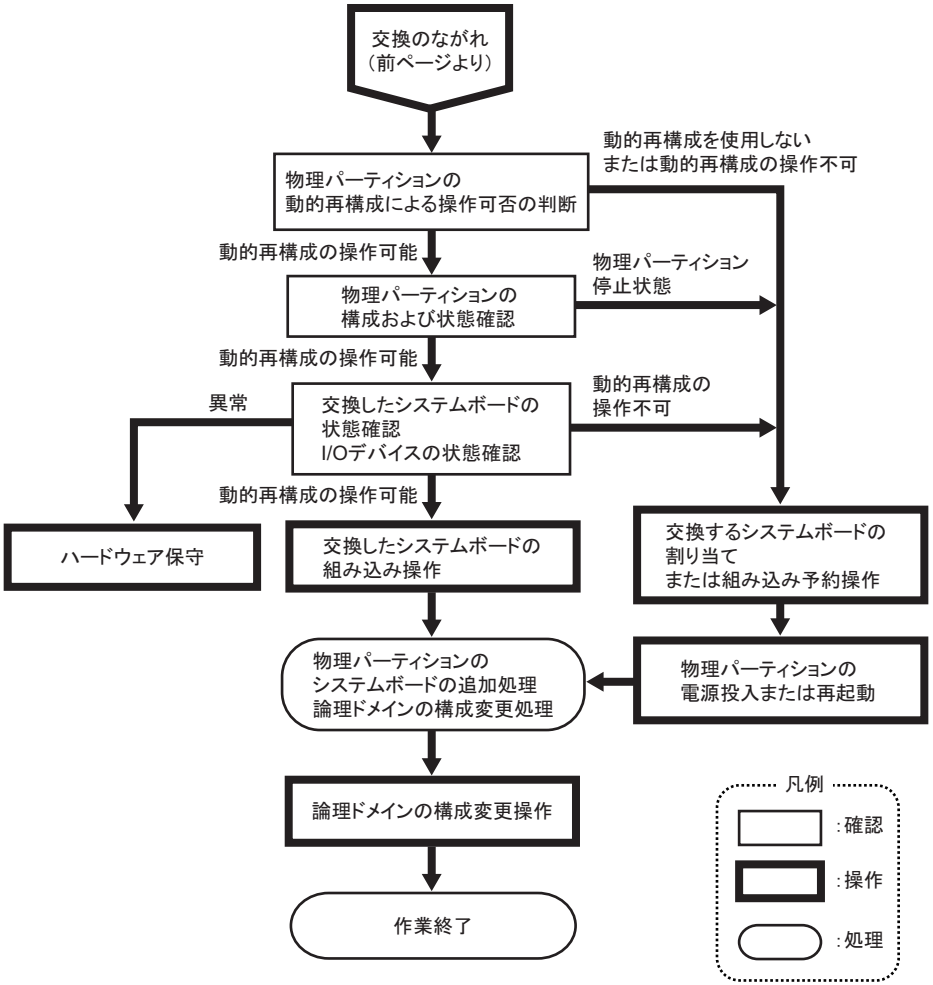


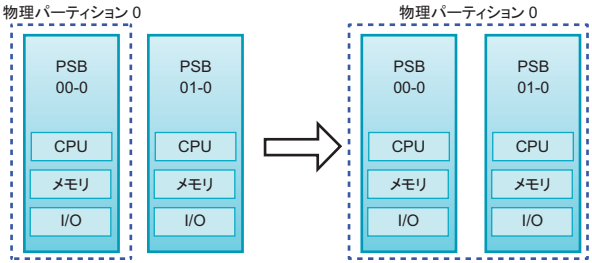
図 6-6　システムボードを交換する操作のながれ（2）



6.2　システムボードを追加する操作例

ここでは、図 6-7のようにシステムボード(PSB)を追加する場合の操作例を示します。

図 6-7　システムボードを追加する操作例



6.2.1　システムボードを割り当てる操作例

システムボード(PSB)を追加する物理パーティションが停止状態であった場合、addboardコマンドで-c assignオプションを指定して、物理パーティションにPSBを割り当てます。PSBを割り当てたあとに、物理パーティションを起動することで、PSBが物理パーティションに組み込まれます。

なお、記載しているSPARC M10-4Sの操作例では、XSCFシェルのコマンドを使用しています。SPARC M12-2Sでも同様の操作です。

1. **XSCFにログインします。**
2. **システムを増設してPSBを追加する場合、addfruコマンドを実行してSPARC M10-4Sをシステムに組み込みます。**

注－システムの増設については、『SPARC M12-2S インストレーションガイド』の「第8章 ビルディングブロック構成のシステムを増設する」または『SPARC M10-4S インストレーションガイド』の「第9章　ビルディングブロック構成のシステムを増設する」を参照してください。

---

3. **showpclコマンドを実行し、現在の物理パーティション構成情報（PPAR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
```

4. **setpclコマンドを実行し、PPAR構成情報にPSBを登録します。**
   次の例では、PSB 01-0を物理パーティション0の論理システムボード（LSB 1）に対応付けています。

```
XSCF> setpcl -p 0 -a 1=01-0
```

5. **showpclコマンドを実行し、設定したPPAR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
```

6. **showcodusageコマンドを実行して、CPUコアリソースの情報を表示します。**
   次の例では、CPUコアリソースの情報を表示しています。

   ここでは、システムには128個のCPUコアリソースが実装されており、128個のCPUコア アクティベーションが登録され、CPUコアリソースは使用されておらず、現在未使用のCPUコア アクティベーション数が128個であることを示しています。

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-------- ------ --------- ------------- ------
PROC          0       128             128 OK: 128 cores
available

Note:
  Please confirm the value of the "In Use" by the ldm command of
 Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
logical domains.
```

注－登録されているCPUコア アクティベーションの数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。CPUコア アクティベーションキーの追加については、『SPARC M12/M10 システム運用・管理ガイド』の「第5章　CPUコア アクティベーション」を参照してください。

---

7. **setcodコマンドを実行し、CPUコアリソースを物理パーティションに割り当てます。**
   次の例では、128個のCPUコアリソース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 -c set 128
PROC Permits assigned for PPAR 0 : 0 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注－XCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形

式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

8. **showboardsコマンドを実行し、PSBの状態を確認します。**
   次の例はPSB 01-0がシステムボードプール状態になっています。

```
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 SP           Available   n    n    n    Passed  Normal
```

9. **-c assignオプションを指定してaddboardコマンドを実行し、PSBを追加します。**

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

10. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**
    次の例では、終了ステータスとして0が返されているため、addboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

11. **showboardsコマンドを実行し、PSBの状態を確認します。**

```
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
```

12. **電源を投入し、物理パーティションを起動します。**
13. **showpparstatusコマンドを使用して物理パーティションの稼働状態を確認します。**
    次の例では、[PPAR Status]欄に「Running」が表示されているため、物理パーティションが正しく稼働しています。

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Running
```

14. **showboardsコマンドを使用してPSBの状態を確認します。**
    次の例では、PSB 01-0に対する[Conn]欄と[Conf]欄の表示がともに「y」となっているため、PSBが正しく追加されています。

```
XSCF> showboards -p 0
PSB   PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

15. **物理パーティションのリソースを論理ドメインに割り当て、論理ドメインを再構築します。**

6.2.2 システムボードを組み込む操作例

システムボード（PSB）を追加する物理パーティションが稼働状態である場合、addboardコマンドで-c configureオプションを指定して物理パーティションにPSBを組み込みます。

addboardコマンドを実行すると、物理パーティション構成情報（PPAR構成情報）に登録したPSB、または-c assignオプションで物理パーティションに割り当てたPSBが、物理パーティションに組み込まれます。このときには、物理パーティションの電源が投入され、制御ドメインが稼働している必要があります。

物理パーティションにPSBを組み込む際には、組み込み対象のPSBによる追加リソースの自動割り当て機能を有効にするかどうかを指定できます。

- **-mオプションを指定しなかった場合または-m bind=resourceを指定した場合**
  addboardコマンドの実行前にdeleteboardコマンドを実行してリソースを削除していた場合、deleteboardコマンド実行前のリソース割り当て状態に戻すことができます。ただし、addboardコマンド実行前に論理ドメイン構成を変更していた場合には、論理ドメイン構成情報に従ったリソースの割り当て状態になります。また、物理パーティションの論理ドメイン構成情報がfactory-defaultであった場合には、PSBによる追加リソースは制御ドメインに割り当てられます。
- **-m bind=noneを指定した場合**
  PSBによる追加リソースは、物理パーティションの空きリソースになります。

注－PPAR DR機能が無効となっている物理パーティションには、-c configureオプションを指定してPSBを組み込むことはできません。

注－addboardコマンドによるPSBの組み込みでは、対象となるPSBのハードウェア診断が行われたあとに、物理パーティションに組み込まれます。このため、コマンドの実行終了までに時間がかかることがあります。

なお、記載しているSPARC M10-4Sの操作例では、XSCFシェルのコマンドおよびOracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用しています。SPARC M12-2Sでの操作も同様です。

1. **XSCFにログインします。**
2. **システムを増設してPSBを追加する場合、addfruコマンドを実行してSPARC M10-4Sをシステムに組み込みます。**

注－システムの増設については、お使いのサーバの『インストレーションガイド』の「第9章　ビルディングブロック構成のシステムを増設する」を参照してください。

3. **showpclコマンドを実行し、現在の物理パーティション構成情報（PPAR構成情報）を確認します。**

```
XSCF> showpcl -p 0
PPAR-ID   LSB   PSB   Status
00                    Running
          00    00-0
```

4. **setpclコマンドを実行し、PPAR構成情報にPSBを登録します。**
   次の例では、PSB 01-0を物理パーティション0の論理システムボード（LSB 1）に対応付けています。

```
XSCF> setpcl -p 0 -a 1=01-0
```

5. **showpclコマンドを実行し、設定したPPAR構成情報を確認します。**

```
XSCF> showpcl -p 0
PPAR-ID   LSB   PSB   Status
00                    Running
          00    00-0
          01    01-0
```

6. **物理パーティションの制御ドメインコンソールに接続します。**
7. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**
   次の例では、制御ドメインと3つのゲストドメインが稼働していることがわかります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    14    8G        64%  2h 54m
guest0           active     -n----  5000    16    8G        42%  2h 54m
guest1           active     -n----  5001    16    8G        11%  2h 54m
guest2           active     -n----  5002    16    8G       7.3%  2h 54m
```

8. **XSCFシェルに戻り、showcodusageコマンドを実行し、CPUコアリソースの情報を表示します。**
   次の例では、CPUコアリソースの情報を表示しています。

   ここでは、システムには128個のCPUコアリソースが実装されており、128個のCPUコア アクティベーションが登録され、64個のCPUコアリソースが使用中であり、現在未使用のCPUコア アクティベーション数が64個であることを示しています。

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-------- ------ --------- ------------- ------
PROC         64       128             128 OK: 64 cores available

Note:
  Please confirm the value of the "In Use" by the ldm command of
 Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
logical domains.
```

注－登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。CPUコア アクティベーションキーの追加については、『SPARC M12/M10システム運用・管理ガイド』の「第5章　CPUコア アクティベーション」を参照してください。

---

9. **setcodコマンドを実行し、CPUコアリソースを物理パーティションに割り当てます。**
   次の例では、64個のCPUコアリソースを物理パーティション0に追加しています。

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 :64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注－XCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

---

10. **showboardsコマンドを実行し、PSBの状態を確認します。**
    次の例では、PSB 01-0がシステムボードプール状態になっています。

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
```

11. **-c configureオプションを指定してaddboardコマンドを実行し、PSBを物理パーティションに組み込みます。**
    次の例では、PSB 01-0を物理パーティション0に組み込んでいます。

```
XSCF> addboard -c configure -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
  0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager. [1800sec]
0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

12. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**
    次の例では、終了ステータスとして0が返されているため、addboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

13. **showboardsコマンドを実行し、PSBの状態を確認します。**
    次の例では、PSB 01-0に対する[Conn]欄と[Conf]欄の表示がともに「y」となっているため、PSBが正しく追加されています。

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

14. **物理パーティションの制御ドメインコンソールでldm list-domainコマンドを実行し、PSBの追加後に論理ドメインの稼働状況に変化がないことを確認します。**
15. **物理パーティションのリソースを論理ドメインに割り当て、論理ドメインを再構築します。**

6.2.3 システムボードの組み込みを予約する操作例

論理ドメインの運用状況や構成、物理パーティションの運用状況や構成、または物理パーティションの動的再構成の操作が行えない場合には、物理パーティションへのシステムボード(PSB)の組み込みを予約します。システムの運用状況から適切なタイミングで物理パーティションを再起動することで、PSBが物理パーティションに組み込まれます。

なお、記載しているSPARC M10-4Sの操作例では、XSCFシェルのコマンドを使用しています。SPARC M12-2Sでの操作も同様です。

1. **XSCFにログインします。**

2. **システムを増設してPSBを追加する場合、addfruコマンドを実行してSPARC M10-4Sをシステムに組み込みます。**

注－システムの増設については、『SPARC M12-2S インストレーションガイド』の「第8章 ビルディングブロック構成のシステムを増設する」または『SPARC M10-4S インストレーションガイド』の「第9章　ビルディングブロック構成のシステムを増設する」を参照してください。

---

3. **showpclコマンドを実行し、現在の物理パーティション構成情報（PPAR構成情報）を確認します。**

```
XSCF> showpcl -p 0
PPAR-ID   LSB   PSB   Status
00                    Running
          00    00-0
```

4. **setpclコマンドを実行し、PPAR構成情報にPSBを登録します。**
   次の例では、PSB 01-0を物理パーティション0の論理システムボード（LSB 1）に対応付けています。

```
XSCF> setpcl -p 0 -a 1=01-0
```

5. **showpclコマンドを実行し、設定したPPAR構成情報を確認します。**

```
XSCF> showpcl -p 0
PPAR-ID LSB PSB Status
00              Running
        00  00-0
        01  01-0
```

6. **showcodusageコマンドを実行して、CPUコアリソースの情報を表示します。**
   次の例では、CPUコアリソースの情報を表示しています。

   ここでは、システムには128個のCPUコアリソースが実装されており、128個のCPUコア アクティベーションが登録され、64個のCPUコアリソースが使用中であり、現在未使用のCPUコア アクティベーション数が64個であることを示しています。

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-------- ------ --------- ------------- ------
PROC          64       128            128 OK: 64 cores available
Note:
  Please confirm the value of the "In Use" by the ldm command of
 Oracle VM Server for SPARC.
  The XSCF may take up to 20 minutes to reflect the "In Use" of
logical domains.
```

注－登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。CPUコア アクティベーションキーの追加については、『SPARC M12/M10 システム運用・管理ガイド』の「第5章　CPUコア アクティベーション」を参照してください。

---

7. **setcodコマンドを実行し、CPUコアリソースを物理パーティションに割り当てます。**
   次の例では、64個のCPUコアリソースを物理パーティション0に追加しています。

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 :64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注－XCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

---

8. **showboardsコマンドを実行し、PSBの状態を確認します。**
   次の例では、PSB 01-0がシステムボードプール状態になっています。

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
```

9. **-c reserveオプションを指定してaddboardコマンドを実行し、PSBの追加を予約します。**

```
XSCF> addboard -c reserve -p 0 01-0
```

10. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**
    次の例では、終了ステータスとして0が返されているため、addboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

11. **showboardsコマンドを-vオプション付きで実行し、PSBの状態およびPSBが追**

加予約されていることを確認します。
次の例では、[R]欄に予約状態を表す「*」が表示されているため、PSBが正しく追加予約されています。

```
XSCF> showboards -v 01-0
PSB   R PPAR-ID(LSB) Assignment   Pwr   Conn Conf Test    Fault
---- - ------------ ----------- ---- ---- ---- ------- --------
01-0 * 00(01)       Assigned    n    n    n    Passed  Normal
```

12. 物理パーティションを再起動します。
13. **showpparstatus**コマンドを使用して物理パーティションの稼働状態を確認します。
次の例では、[PPAR Status]欄に「Running」が表示されているため、物理パーティションが正しく稼働しています。

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Running
```

14. **showboards**コマンドを使用して**PSB**の状態を確認します。
次の例では、PSB 01-0に対する[Conn]欄と[Conf]欄の表示がともに「y」となっているため、PSBが正しく追加されています。

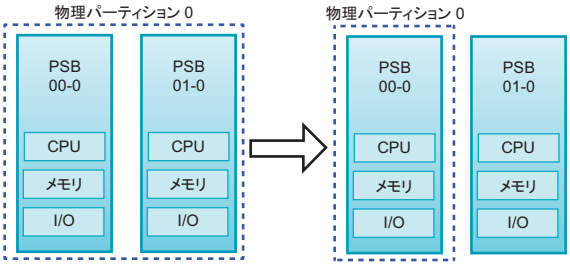
```
XSCF> showboards -p 0
PSB   PPAR-ID(LSB) Assignment   Pwr   Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

15. 物理パーティションのリソースを論理ドメインに割り当て、論理ドメインを再構築します。

6.3 システムボードを削除する操作例

ここでは、図 6-8のようにシステムボード(PSB)を削除する場合の操作例を示します。

図 6-8　システムボードを削除する操作例



6.3.1　システムボードの割り当てを解除する操作例

システムボード(PSB)を削除する物理パーティションが停止状態であった場合、deleteboardコマンドで-c unassignオプションを指定して、物理パーティションへのPSBの割り当てを解除します。PSBは、物理パーティションに割り当てられた状態が解除され、システムボードプール状態になります。

なお、記載している操作例では、XSCFシェルのコマンドを使用しています。

1. **XSCFにログインします。**
2. **showpparstatusコマンドを実行し、物理パーティションの電源が切断されていることを確認します。**

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Powered Off
```

3. **showboardsコマンドを実行し、削除するPSBの状態を確認します。**

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

4. **-c unassignオプションを指定してdeleteboardコマンドを実行し、PSBを削除します。**

```
XSCF> deleteboard -c unassign 01-0
```

5. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**
   次の例では、終了ステータスとして0が返されているため、deleteboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

6. **showboardsコマンドを実行し、PSBの状態を確認します。**
   次の例では、PSB 01-0がシステムボードプール状態になっています。

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

6.3.2 システムボードを切り離す操作例

システムボード(PSB)を切り離す物理パーティションが稼働状態である場合、deleteboardコマンドで-c disconnectオプションか-c unassignオプションを指定して物理パーティションからPSBを切り離します。

-c disconnectオプションを指定した場合、PSBは物理パーティションから切り離されますが、物理パーティションに割り当てられた状態のままです。このため、物理パーティションを再起動するか、addboardコマンドを実行することで、切り離したPSBが物理パーティションに再び組み込まれます。

-c unassignオプションを指定してPSBを物理パーティションから切り離した場合、PSBはシステムボードプール状態になり、ほかの物理パーティションに組み込んだり割り当てたりできるようになります。また、すでに-c disconnectオプションを指定して切り離したPSBに対して-c unassignオプションを指定してdeleteboardコマンドを実行すると、割り当て状態であったPSBがシステムボードプール状態になります。

物理パーティションからPSBを切り離す際は、物理パーティション内の空きリソースを使用して、切り離されるPSBのリソースを利用する論理ドメインを移動します。deleteboardコマンドに-mオプションを指定して、空きリソースの状況に応じた論理ドメイン移動時のリソースの確保方法を指定できます。

- **-mオプションを指定しなかった場合、または-m unbind=noneを指定した場合**

空きリソースを確保できない場合や、物理I/Oの割り当てを事前に解除していない場合は、deleteboardコマンドによる切り離しが失敗します。

- **-m unbind=resourceを指定した場合**

Oracle VM Server for SPARC 3.4以降では、ldmdサービスのldmd/fj_ppar_dr_policyプロパティーに設定されるPPAR DRポリシーにより、リソースをアンバインドするポリシーを選択できます。Oracle VM Server for SPARC 3.4よりも前のOracle VM Server for SPARCでは、このポリシーは常に「targeted」です。

ポリシーの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「8.15　物理パーティションの動的再構成ポリシーを設定する」を参照してください。

ldmd/fj_ppar_dr_policy = auto

このPPAR DRポリシー設定は、最新のポリシーを自動的に使用するものです。

Oracle VM Server for SPARC 3.4を使用するシステムの場合は、「auto」を設定すると、「ratio」設定時と同様に動作します。「auto」設定は、デフォルトのポリシーです。

ldmd/fj_ppar_dr_policy = ratio

このポリシーは、物理パーティション内のすべての論理ドメインがOracle Solaris 11.3以降を実行していて、かつXCP 2271以降がシステムに適用されている場合に機能します。それ以外の場合は、「targeted」設定時と同様にシステムが動作します。

ポリシーを「ratio」に設定した場合、切り離すPSBからリソースを移動するための十分な空きリソースが残りのPSB上にないと、既存ドメインからリソースを減らしてリソースを自動的に解放します。このとき、すべての既存ドメインから、できるだけ比率に応じたリソースが自動的に減らされます。リソースが使用中であるなどの理由で、正確な比率に応じた削減は必ずしも機能しないことがあり、比率に応じた削減はベストエフォートベースです。

ldmd/fj_ppar_dr_policy = targeted

空きリソースが確保されない場合は、移動先のPSB上のCPU/メモリを使用している各論理ドメインからリソースが削除され、移動のためのリソースが確保されます。リソースを削除する論理ドメインを選択する順序は、ポリシーで定義されます。このポリシーでは、デフォルト論理ドメイン（指定なし）、マスタが指定された論理ドメイン、I/Oドメイン、ルートドメイン、制御ドメインの順となります。

移動先のPSBから空きリソースを確保できない場合は、deleteboardコマンドによる切り離しが失敗します。

- **-m unbind=shutdownを指定した場合**

-m unbind=resourceを指定した場合と動作は同じです。使用可能なリソースがこれ以上確保できない場合は、移動元の論理ドメインがシャットダウンします。このオプションを使用する場合は、ldmd/fj_ppar_dr_policyが「targeted」に設定されている必要があります。この設定が行われていないと、論理ドメインがシャットダウンしないことがあります。

注－PPAR DR機能が無効となっている物理パーティションでは、物理パーティションの稼働中に-c disconnectオプションや-c unassignオプションを指定してPSBを切り離すことはできません。

注－deleteboardコマンドによるPSBの切り離しでは、PSB上のハードウェアリソースがOracle Solarisから切り離されます。したがって、コマンドの実行終了まで時間がかかることがあります。

なお、記載している操作例では、XSCFシェルのコマンドおよびOracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用しています。

1. **XSCFにログインします。**
2. **物理パーティションの制御ドメインコンソールに接続します。**
3. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**
   次の例では、制御ドメインと2つのゲストドメインと1つのルートドメインが稼働していることがわかります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    14    8G        64%  2h 54m
guest0           active     -n----  5000    16    8G        42%  2h 54m
guest1           active     -n----  5001    16    8G        11%  2h 54m
domain01         active     -n----  5002    16    8G       7.3%  2h 54m
```

4. **-aオプションを指定してldm list-devicesコマンドを実行し、リソースの使用状況を確認します。**
   次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID      %FREE   CPUSET
    0       0       (0, 1)
    4       0       (8, 9)
    8       0       (16, 17)
 (中略)
    184     0       (368, 369)
    188     100     (376, 377)
    512     100     (1024, 1025)
    516     100     (1032, 1033)
    520     100     (1040, 1041)
    524     100     (1048, 1049)
 (中略)
VCPU
    PID     %FREE   PM
    0       0       no
    1       0       no
    8       0       no
    9       0       no
 (中略)
    369     0       no
    376     100     ---
    377     100     ---
    1024    100     ---
    1025    100     ---
    1032    100     ---
    1033    100     ---
 (以下略)
```

5. **ldm list-ioコマンドを実行し、I/Oデバイスの使用状況を確認します。**
   次の例では、制御ドメインであるprimaryとルートドメインであるdomain01がI/Oデバイスを使用しています。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN   STATUS
----                                        ----   ---      ------   ------
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE1                                       BUS    PCIE1    primary  IOV
PCIE2                                       BUS    PCIE2    primary  IOV
```

```
PCIE3                                   BUS    PCIE3    primary  IOV
PCIE8                                   BUS    PCIE8    domain01 IOV
PCIE9                                   BUS    PCIE9    domain01 IOV
PCIE10                                  BUS    PCIE10   domain01 IOV
PCIE11                                  BUS    PCIE11   domain01 IOV
/BB0/CMUL/NET0                          PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                          PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                        PCIE   PCIE0    primary  OCC
/BB0/PCI0                               PCIE   PCIE1    primary  EMP
/BB0/PCI1                               PCIE   PCIE1    primary  EMP
（以下略）
```

6. **ルートドメインが提供するサービスのうち、削除対象のPSBのI/Oデバイスをすべて解除します。**
   次の例では、ルートドメインdomain01を停止し、domain01が所有しているPCIeルートコンプレックスのうち、削除対象のPSBに関係するものを解除しています。

注ー動的に再構成する場合は、ルートドメインを停止するためのldm stop-domainコマンドは不要です。I/Oデバイスの解除から操作してください。

```
# ldm stop-domain domain01
# ldm remove-io PCIE11 domain01
# ldm remove-io PCIE10 domain01
# ldm remove-io PCIE9 domain01
# ldm remove-io PCIE8 domain01
# ldm unbind-domain domain01
```

7. **再度ldm list-ioコマンドを実行し、I/Oデバイスの使用状況を確認します。**

```
# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
PCIE0                                   BUS    PCIE0    primary  IOV
PCIE1                                   BUS    PCIE1    primary  IOV
PCIE2                                   BUS    PCIE2    primary  IOV
PCIE3                                   BUS    PCIE3    primary  IOV
PCIE8                                   BUS    PCIE8
PCIE9                                   BUS    PCIE9
PCIE10                                  BUS    PCIE10
PCIE11                                  BUS    PCIE11
/BB0/CMUL/NET0                          PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                          PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                        PCIE   PCIE0    primary  OCC
/BB0/PCI0                               PCIE   PCIE1    primary  EMP
/BB0/PCI1                               PCIE   PCIE1    primary  EMP
（以下略）
```

8. **-c unassignオプションを指定してdeleteboardコマンドを実行し、PSBを物理パーティションから切り離します。**

次の例では、PSB 01-0を物理パーティションから切り離し、システムボードプール状態にしています。

```
XSCF> deleteboard -c unassign 01-0
PSB#01-0 will be unassigned from PPAR immediately. Continue?[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
  0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [43200sec]
  0..... 30.end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
  0..... 30..... 60..... 90.....120.....150.end
Operation has completed.
```

9. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**
   次の例では、終了ステータスとして0が返されているため、deleteboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

10. **showboardsコマンドを実行し、PSBの状態を確認します。**
    次の例では、PSB 01-0がシステムボードプール状態になっています。

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
```

11. **物理パーティションの制御ドメインコンソールでldm list-domainコマンドを実行し、PSBの削除後に論理ドメインの稼働状況に変化がないことを確認します。**
12. **ビルディングブロックを減設する場合は、マスタXSCFでinitbbコマンドを実行し、対象のSPARC M12-2S/M10-4Sをシステムから切り離します。**

注－システムの減設については、『SPARC M12-2Sインストレーションガイド』の「第9章 ビルディングブロック構成のシステムを減設する」または『SPARC M10-4S インストレーションガイド』の「第10章　ビルディングブロック構成のシステムを減設する」を参照してください。

6.3.3 システムボードの割り当て解除を予約する操作例

論理ドメインの運用状況や構成、物理パーティションの運用状況や構成、または物理パーティションの動的再構成の操作が行えない場合には、物理パーティションへのシ

ステムボード(PSB)の割り当て解除を予約します。システムの運用状況から適切なタイミングで物理パーティションを停止することで、物理パーティションへのPSBの割り当てが解除され、PSBがシステムボードプール状態になります。

なお、記載している操作例では、XSCFシェルのコマンドを使用しています。

1. **showpparstatusコマンドを実行し、物理パーティションの稼働状態を確認します。**

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Running
```

2. **showboardsコマンドを実行し、削除するPSBの状態を確認します。**

```
XSCF> showboards 01-0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

3. **-c reserveオプションを指定してdeleteboardコマンドを実行し、PSBの削除を予約します。**

```
XSCF> deleteboard -c reserve 01-0
```

4. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**
   次の例では、終了ステータスとして0が返されているため、deleteboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

5. **showboardsコマンドを-vオプション付きで実行し、PSBの状態およびPSBが削除予約されていることを確認します。**
   次の例では、[R]欄に予約状態を表す「*」が表示されているため、PSBが正しく削除予約されています。

```
XSCF> showboards -v 01-0
PSB  R PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- - ------------ ----------- ---- ---- ---- ------- -----
01-0 * 00(01)       Assigned    y    y    y    Passed  Normal
```

6. **物理パーティションを再起動します。**
7. **showpparstatusコマンドを使用して物理パーティションの稼働状態を確認します。**
   次の例では、[PPAR Status]欄に「Running」が表示されているため、物理パー

ティションが正しく稼働しています。

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Running
```

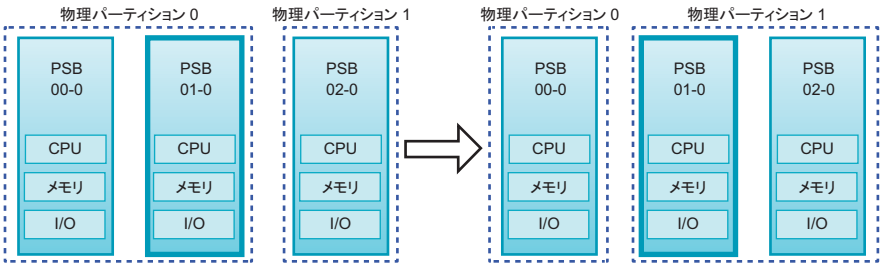
8. **showboardsコマンドを実行してPSBの状態を確認します。**
   次の例では、PSB 01-0がシステムボードプール状態になっています。

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
```

6.4 システムボードを移動する操作例

ここでは、図 6-9のように物理パーティション間でシステムボード（PSB）を移動する場合の操作例を示します。

図 6-9　システムボードを移動する操作例



物理パーティション間でPSBを移動するための操作は、PSBの割り当て解除の操作と、PSBの組み込みの操作を組み合わせたものになります。

移動元の物理パーティションの運用状況などによって動的再構成の操作を行えない場合には、移動元の物理パーティションでPSBの割り当て解除や割り当て解除予約をすることにより、PSBを削除します。また、移動先の物理パーティションで動的再構成の操作を行えない場合には、移動先の物理パーティションに対してPSBの割り当てや割り当て予約をすることにより、PSBを追加します。

注－PPAR DR機能が無効となっている物理パーティションでは、物理パーティションの稼働中に-c disconnectオプションや-c unassignオプションを指定してPSBを切り離したり、-c configureオプションを指定してPSBを組み込んだりすることはできません。

ここでは、移動元と移動先の物理パーティションで動的再構成の操作を行えるものと

して操作例を示します。 なお、記載しているSPARC M10-4Sの操作例では、XSCFシェルのコマンドおよびOracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用しています。SPARC M12-2Sでの操作も同様です。

1. **XSCFにログインします。**
2. **物理パーティションの制御ドメインコンソールに接続します。**
3. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**
   次の例では、制御ドメインと2つのゲストドメインと1つのルートドメインが稼働していることがわかります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    14    8G        64%  2h 54m
guest0           active     -n----  5000    16    8G        42%  2h 54m
guest1           active     -n----  5001    16    8G        11%  2h 54m
domain01         active     -n----  5002    16    8G       7.3%  2h 54m
```

4. **-aオプションを指定してldm list-devicesコマンドを実行し、リソースの使用状況を確認します。**
   次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID      %FREE   CPUSET
    0       0       (0, 1)
    4       0       (8, 9)
    8       0       (16, 17)
 （中略）
    184     0       (368, 369)
    188     100     (376, 377)
    512     100     (1024, 1025)
    516     100     (1032, 1033)
    520     100     (1040, 1041)
    524     100     (1048, 1049)
 （中略）
VCPU
    PID     %FREE   PM
    0       0       no
    1       0       no
    8       0       no
    9       0       no
 （中略）
    369     0       no
    376     100     ---
    377     100     ---
    1024    100     ---
    1025    100     ---
    1032    100     ---
    1033    100     ---
 （以下略）
```

5. **ldm list-ioコマンドを実行し、I/Oデバイスの使用状況を確認します。**
次の例では、制御ドメインであるprimaryとルートドメインであるdomain01がI/Oデバイスを使用しています。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN   STATUS
----                                        ----   ---      ------   ------
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE1                                       BUS    PCIE1    primary  IOV
PCIE2                                       BUS    PCIE2    primary  IOV
PCIE3                                       BUS    PCIE3    primary  IOV
PCIE8                                       BUS    PCIE8    domain01 IOV
PCIE9                                       BUS    PCIE9    domain01 IOV
PCIE10                                      BUS    PCIE10   domain01 IOV
PCIE11                                      BUS    PCIE11   domain01 IOV
/BB0/CMUL/NET0                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                            PCIE   PCIE0    primary  OCC
/BB0/PCI0                                   PCIE   PCIE1    primary  EMP
/BB0/PCI1                                   PCIE   PCIE1    primary  EMP
（以下略）
```

6. **ルートドメインが提供するサービスのうち、削除対象のPSBのI/Oデバイスをすべて解除します。**
次の例では、ルートドメインdomain01を停止し、domain01が所有しているPCIeルートコンプレックスのうち、削除対象のPSBに関係するものを解除しています。

注ー動的に再構成する場合は、ルートドメインを停止するためのldm stop-domainコマンドは不要です。I/Oデバイスの解除から操作してください。

```
# ldm stop-domain domain01
# ldm remove-io PCIE11 domain01
# ldm remove-io PCIE10 domain01
# ldm remove-io PCIE9 domain01
# ldm remove-io PCIE8 domain01
# ldm unbind-domain domain01
```

7. **再度ldm list-ioコマンドを実行し、I/Oデバイスの使用状況を確認します。**

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN   STATUS
----                                        ----   ---      ------   ------
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE1                                       BUS    PCIE1    primary  IOV
PCIE2                                       BUS    PCIE2    primary  IOV
PCIE3                                       BUS    PCIE3    primary  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9
```

```
PCIE10                                    BUS    PCIE10
PCIE11                                    BUS    PCIE11
/BB0/CMUL/NET0                            PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                            PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                          PCIE   PCIE0    primary  OCC
/BB0/PCI0                                 PCIE   PCIE1    primary  EMP
/BB0/PCI1                                 PCIE   PCIE1    primary  EMP
（以下略）
```

8. **-c unassignオプションを指定してdeleteboardコマンドを実行し、PSBを物理パーティションから切り離します。**
   次の例では、PSB 01-0を物理パーティションから切り離し、システムボードプール状態にしています。

```
XSCF> deleteboard -c unassign 01-0
```

9. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**
   次の例では、終了ステータスとして0が返されているため、deleteboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

10. **showboardsコマンドを実行し、PSBの状態を確認します。**
    次の例では、PSB 01-0がシステムボードプール状態になっています。

```
XSCF> showboards -a
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
02-0 01(00)       Assigned    y    y    y    Passed  Normal
```

11. **移動元の物理パーティションの制御ドメインコンソールでldm list-domainコマンドを実行し、PSBの削除後に論理ドメインの稼働状況に変化がないことを確認します。**
12. **showpclコマンドを実行し、移動先の物理パーティションにおける現在の物理パーティション構成情報（PPAR構成情報）を確認します。**

```
XSCF> showpcl -p 1
PPAR-ID   LSB   PSB   Status
01                    Running
          00    00-0
```

13. **setpclコマンドを実行し、移動先の物理パーティションのPPAR構成情報にPSBを登録します。**

次の例では、PSB 01-0を物理パーティション1の論理システムボード（LSB 1）に対応付けています。

```
XSCF> setpcl -p 1 -a 1=01-0
```

14. **showpclコマンドを実行し、設定したPPAR構成情報を確認します。**

```
XSCF> showpcl -p 1
PPAR-ID   LSB   PSB   Status
01                    Running
          00    02-0
          01    01-0
```

15. **物理パーティションの制御ドメインコンソールに接続します。**
16. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**
    次の例では、制御ドメインと3つのゲストドメインが稼働していることがわかります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    14    8G        64%  2h 54m
guest0           active     -n----  5000    16    8G        42%  2h 54m
guest1           active     -n----  5001    16    8G        11%  2h 54m
guest2           active     -n----  5002    16    8G       7.3%  2h 54m
```

17. **XSCFシェルに戻り、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コアリソースを確認します。**
    次の例では、CPUコアリソースの設定情報を表示しています。

ここでは、システムには192個のCPUコアリソースが実装されており、物理パーティション0に128個のCPUコア アクティベーションが登録され、物理パーティション1に64個のCPUコア アクティベーションが登録されていることを示しています。

```
XSCF> showcod -v -s cpu
PROC Permits installed : 192 cores
PROC Permits assigned for PPAR 0: 128 [Permanent 128cores]
PROC Permits assigned for PPAR 1: 64 [Permanent 64cores]
PROC Permits assigned for PPAR 2: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 3: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 4: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 5: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 6: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 7: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 8: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 9: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 10: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 11: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 12: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 13: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 14: 0 [Permanent 0cores]
```

```
PROC Permits assigned for PPAR 15: 0 [Permanent 0cores]
```

18. **showcodusageコマンドを実行し、CPUコアリソースの情報を確認します。**
    以下の実装状態であることを示しています。

    【物理パーティション0】

    - CPUコアリソース数：64
    - CPUコア アクティベーションの登録数：128
    - 使用中のCPUコアリソース数：64

    【物理パーティション1】

    - CPUコアリソース数：64
    - CPUコア アクティベーションの登録数：64
    - 使用中のCPUコアリソース数：64

    【システム全体】

    - 現在未使用のCPUコア アクティベーション数(Unused)：64

```
XSCF> showcodusage -p ppar
PPAR-ID/Resource In Use Installed Assigned
---------------- ------ --------- --------------
0 - PROC             64        64      128 cores
1 - PROC             64        64       64 cores
Unused - PROC         0         0       64 cores

Note:
  Please confirm the value of the "In Use" by the ldm command of
 Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
logical domains.
```

注－登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。CPUコア アクティベーションキーの追加については、『SPARC M12/M10 システム運用・管理ガイド』の「第5章　CPUコア アクティベーション」を参照してください。

---

19. **setcodコマンドを実行し、CPUコアリソースを移動先の物理パーティションに割り当てます。**
    次の例では、物理パーティション0から64個のCPUコア アクティベーションの割り当て数を減らし、物理パーティション1に追加しています。

```
XSCF> setcod -p 0 -s cpu -c del 64
PROC Permits assigned for PPAR 0 : 128 -> 64

PROC Permits assigned for PPAR will be changed.
```

```
Continue? [y|n] :y

Completed.

XSCF> setcod -p 1 -s cpu -c add 64
PROC Permits assigned for PPAR 1 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注ーXCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

---

20. **showboardsコマンドを実行し、PSBの状態を確認します。**

```
XSCF> showboards -a
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
02-0 01(00)       Assigned    y    y    y    Passed  Normal
```

21. **-c configureオプションを指定してaddboardコマンドを実行し、PSBを物理パーティションに組み込みます。**
    次の例では、PSB 01-0を物理パーティション1に組み込んでいます。

```
XSCF> addboard -c configure -p 1 01-0
```

22. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**
    次の例では、終了ステータスとして0が返されているため、addboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

23. **showboardsコマンドを実行し、PSBの状態を確認します。**
    次の例では、PSB 01-0に対する[Conn]欄と[Conf]欄の表示がともに「y」となっているため、PSBが正しく追加されています。

```
XSCF> showboards -a
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
```

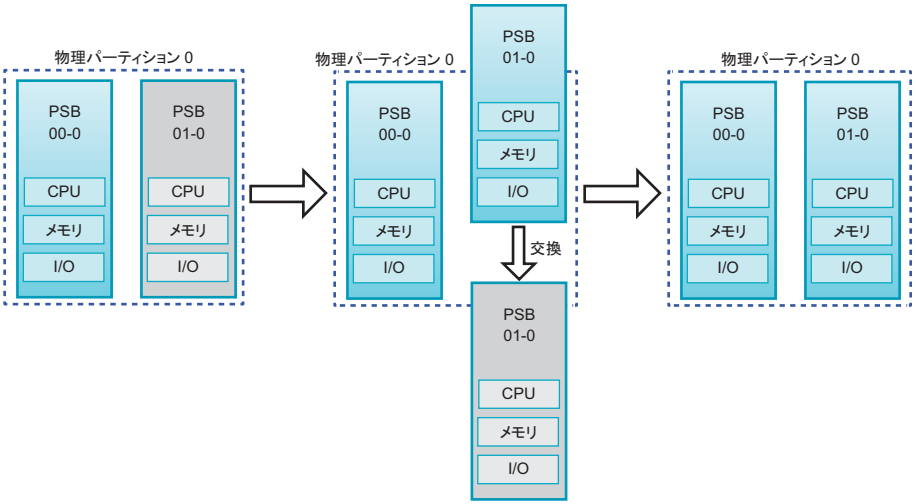
```
00-0 00(00)        Assigned    y    y    y    Passed  Normal
01-0 01(01)        Assigned    y    y    y    Passed  Normal
02-0 01(00)        Assigned    y    y    y    Passed  Normal
```

24. **移動先の物理パーティションの制御ドメインコンソールでldm list-domainコマンドを実行し、PSBの追加後に論理ドメインの稼働状況に変化がないことを確認します。**
25. **物理パーティションのリソースを論理ドメインに割り当て、論理ドメインを再構築します。**

6.5 システムボードを交換する操作例

ここでは、図 6-10のようにシステムボード(PSB)を交換する場合の操作例を示します。

図 6-10 システムボードを交換する操作例



PSBを交換するための操作は、PSBの切り離しの操作と、PSBの組み込みの操作を組み合わせたものになります。物理パーティションの運用状況などによって動的再構成の操作を行えない場合には、物理パーティションの電源を切断してPSBの交換を行ってください。

注－PPAR DR機能が無効となっている物理パーティションでは、物理パーティションの稼働中に-c disconnectオプションや-c unassignオプションを指定してPSBを切り離したり、-c configureオプションを指定してPSBを組み込んだりすることはできません。

ここでは、物理パーティションで動的再構成の操作を行えるものとして操作例を示します。なお、記載しているSPARC M10-4Sの操作例では、XSCFシェルのコマンドおよびOracle VM Server for SPARCの管理ソフトウェアであるLogical Domains Managerのldmコマンドを使用しています。SPARC M12-2Sでの操作も同様です。

1. **XSCFにログインします。**
2. **showbbstatusコマンドを実行し、交換するPSBのXSCFがマスタXSCFでないことを確認します。**

注－交換するPSBのXSCFがマスタXSCFとして動作している場合、switchscfコマンドを実行してXSCFをアクティブ状態からスタンバイ状態に切り替えてください。また、XSCFの切り替え、およびXSCFの再起動が完了したことを確認したあとに、PSBの切り離し操作を行ってください。

---

3. **物理パーティションの制御ドメインコンソールに接続します。**
4. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**
   次の例では、制御ドメインと2つのゲストドメインと1つのルートドメインが稼働していることがわかります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    14    8G        64%  2h 54m
guest0           active     -n----  5000    16    8G        42%  2h 54m
guest1           active     -n----  5001    16    8G        11%  2h 54m
domain01         active     -n----  5002    16    8G       7.3%  2h 54m
```

5. **-aオプションを指定してldm list-devicesコマンドを実行し、リソースの使用状況を確認します。**
   次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID      %FREE   CPUSET
    0       0       (0, 1)
    4       0       (8, 9)
    8       0       (16, 17)
 (中略)
    184     0       (368, 369)
    188     100     (376, 377)
    512     100     (1024, 1025)
    516     100     (1032, 1033)
    520     100     (1040, 1041)
    524     100     (1048, 1049)
 (中略)
VCPU
    PID     %FREE   PM
    0       0       no
    1       0       no
    8       0       no
    9       0       no
 (中略)
    369     0       no
    376     100     ---
    377     100     ---
```

```
    1024    100     ---
    1025    100     ---
    1032    100     ---
    1033    100     ---
（以下略）
```

6. **ldm list-ioコマンドを実行し、I/Oデバイスの使用状況を確認します。**
   次の例では、制御ドメインであるprimaryとルートドメインであるdomain01がI/Oデバイスを使用しています。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN   STATUS
----                                        ----   ---      ------   ------
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE1                                       BUS    PCIE1    primary  IOV
PCIE2                                       BUS    PCIE2    primary  IOV
PCIE3                                       BUS    PCIE3    primary  IOV
PCIE8                                       BUS    PCIE8    domain01 IOV
PCIE9                                       BUS    PCIE9    domain01 IOV
PCIE10                                      BUS    PCIE10   domain01 IOV
PCIE11                                      BUS    PCIE11   domain01 IOV
/BB0/CMUL/NET0                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                            PCIE   PCIE0    primary  OCC
/BB0/PCI0                                   PCIE   PCIE1    primary  EMP
/BB0/PCI1                                   PCIE   PCIE1    primary  EMP
（以下略）
```

7. **ルートドメインが提供するサービスのうち、削除対象のPSBのI/Oデバイスをすべて解除します。**
   次の例では、ルートドメインdomain01を停止し、domain01が所有しているPCIeルートコンプレックスのうち、削除対象のPSBに関係するものを解除しています。

注ー動的に再構成する場合は。ルートドメインを停止するためのldm stop-domainコマンドは不要です。I/Oデバイスの解除から操作してください。

```
# ldm stop-domain domain01
# ldm remove-io PCIE11 domain01
# ldm remove-io PCIE10 domain01
# ldm remove-io PCIE9 domain01
# ldm remove-io PCIE8 domain01
# ldm unbind-domain domain01
```

8. **再度ldm list-ioコマンドを実行し、I/Oデバイスの使用状況を確認します。**

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN   STATUS
----                                        ----   ---      ------   ------
```

```
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE1                                    BUS    PCIE1    primary  IOV
PCIE2                                    BUS    PCIE2    primary  IOV
PCIE3                                    BUS    PCIE3    primary  IOV
PCIE8                                    BUS    PCIE8
PCIE9                                    BUS    PCIE9
PCIE10                                   BUS    PCIE10
PCIE11                                   BUS    PCIE11
/BB0/CMUL/NET0                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                         PCIE   PCIE0    primary  OCC
/BB0/PCI0                                PCIE   PCIE1    primary  EMP
/BB0/PCI1                                PCIE   PCIE1    primary  EMP
（以下略）
```

9. **-c disconnectオプションを指定してdeleteboardコマンドを実行し、PSBを物理パーティションから切り離します。**
   次の例では、PSB 01-0を物理パーティションから切り離し、物理パーティション0に割り当てた状態にしています。

```
XSCF> deleteboard -c disconnect 01-0
```

10. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**
    次の例では、終了ステータスとして0が返されているため、deleteboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

11. **showboardsコマンドを実行し、PSBの状態を確認します。**

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

12. **replacefruコマンドを実行し、PSBを交換します。**

```
XSCF> replacefru
```

注－replacefruコマンドを使ったPSBの交換については、お使いのサーバの『サービスマニュアル』を参照してください。

---

13. **showboardsコマンドを実行し、PSBの状態を確認します。**

```
XSCF> showboards -p 0
PSB   PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

14. **-c configureオプションを指定してaddboardコマンドを実行し、PSBを物理パーティションに組み込みます。**
    次の例では、PSB 01-0を物理パーティション0に組み込んでいます。

```
XSCF> addboard -c configure -p 0 01-0
```

注－addboardコマンドの実行時に-mオプションを省略した場合、または-mオプションでbind=resourceを指定した場合、deleteboardコマンドを実行する前のリソース割り当て状態に戻らないことがあります。
CPUやメモリなど、交換前のPSBのリソース量が交換後のPSBのリソース量と異なっている場合、リソース割り当て状態を元の状態に戻すことができません。
元のリソース割り当て状態に戻らなかった場合、リソースは空きリソース状態になります。この場合、ldmコマンドを使用して、論理ドメインへのリソース割り当てを再度行ってください。

---

15. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**
    次の例では、終了ステータスとして0が返されているため、addboardコマンドの実行が正しく終了しています。

```
XSCF> showresult
0
```

16. **showboardsコマンドを実行し、PSBの状態を確認します。**

```
XSCF> showboards -p 0
PSB   PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

17. **物理パーティションの制御ドメインコンソールでldm list-domainコマンドを実行し、PSBの追加後に論理ドメインの稼働状況に変化がないことを確認します。**
18. **ルートドメインを起動して、ルートドメインによるサービス提供を再開します。**

6.6 物理パーティション番号を変更する場合の留意点

構築した物理パーティションの物理パーティション番号は、別の番号に変更することができます。変更する場合は、次の点に留意してください。

- 物理パーティションの構成情報の再設定が必要
  物理パーティション番号を変更した場合は、新たな物理パーティションの構成情報を設定する必要があります。物理パーティションの構成情報の設定に関する詳細は、「3.1.12　物理パーティション構成情報を設定する」および「第4章　物理パーティションの構築」を参照してください。
- CPUコア アクティベーションの再割り当てが必要
  物 理パーティション番号を変更しても、CPUコア アクティベーションはもともとの物理パーティション番号に割り当てられている状態です。したがって、いったん、CPUコア アクティベーションの割り当てを解除してから、新たな物理パーティション番号を指定して割り当てなおす必要があります。
- ホストID、イーサネット（MAC）アドレスの変更
  物理パーティション番号を変更すると、物理パーティション固有のホストIDやイーサネット（MAC）アドレスも変更されます。ホストIDやイーサネット（MAC）アドレスは、Oracle SolarisのZFSファイルシステムやアプリケーションのライセンス認証、設定情報などに使用されていることがあるため留意が必要です。

第7章

ゲストドメインを移行する

ここでは、ライブマイグレーションの機能を使って、ゲストドメインを別の物理パーティションに移行する方法を説明します。

- 概要
- ゲストドメインを移行する
- ゲストドメインの移行例

7.1 概要

ライブマイグレーションとは、ある物理パーティション上のゲストドメインを、稼働させた状態のままで別の物理パーティションに移行する機能です。ライブマイグレーションは、次のようなときに使用します。

- 物理パーティション間で負荷の調整をする
  負荷が高い物理パーティションから負荷が低い物理パーティションにゲストドメインを移行することで、システム全体での負荷を調整します。
- ゲストドメインを稼働させたまま、ハードウェア保守を実行する
  保守のためにシステムボードを物理パーティションから切り離す際に、その物理パーティションで稼働しているゲストドメインを別の物理パーティションに移行します。

ライブマイグレーションを実行すると、移行元と移行先のLogical Domains Managerが通信を行い、次のような処理を行います。

(1) 移行先の物理パーティションに、移行元と同じ構成のゲストドメインを構築します。

(2) 移行元のゲストドメインから移行先のゲストドメインに情報がコピーされます。移行元のゲストドメインの実行は継続されます。

(3) コピーが終了すると移行元の論理ドメインは削除され、移行先の論理ドメインの実行が開始されます。

論理ドメインの移行は、同一システム内でも、異なるシステム間でも可能です。図7-1は同一システム内の移行の様子、図 7-2は異なるシステム間の移行の様子を示して

います。

図 7-1　同一システム内の移行

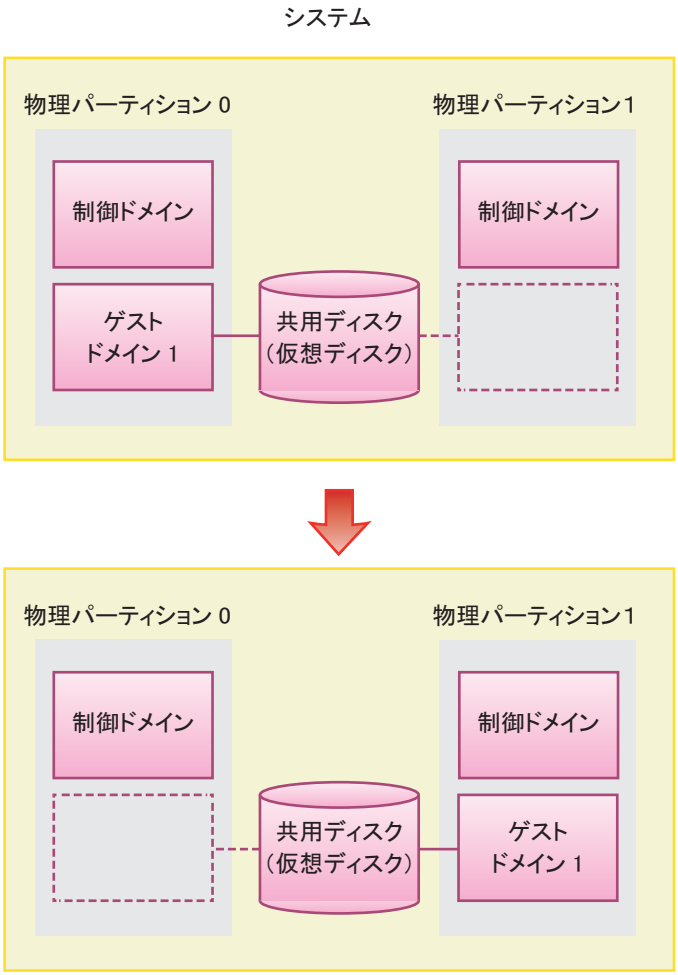
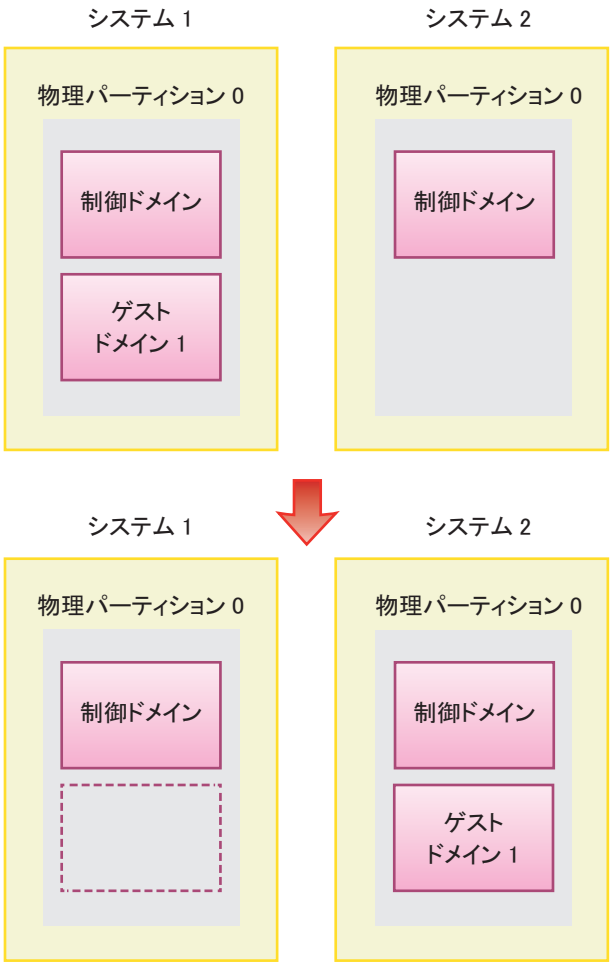


図 7-2　異なるシステム間の移行



7.1.1　ライブマイグレーションの要件

ライブマイグレーションを利用するときには、次の要件を満たす必要があります。

- 移行元および移行先で、Logical Domains Managerが稼働
  移行元および移行先に、バージョン2.1以降のOracle VM Server for SPARCがインストールされ、Logical Domains Managerが稼働している必要があります。
- 移行先に十分な空きリソースがある
  移行元のゲストドメインを受け入れるのに十分なハードウェアリソースが移行先の物理パーティションに用意されている必要があります。

このほかにもライブマイグレーションの要件が多くあります。それらの要件については、お使いのバージョンの『Oracle VM Server for SPARC 管理ガイド』の「ドメインの移行」、お使いのバージョンの『プロダクトノート』の「Oracle VM Server for SPARCのライブマイグレーションに関する留意点」を参照してください。なお、

ldm migrate-domainコマンドに-nオプションを指定して実行することで、使用する環境が要件を満たしているかどうかが確認できます。このとき、要件を満たしていない項目はエラーとして報告されます。

7.2 ゲストドメインを移行する

ライブマイグレーションによってゲストドメインを移行するときには、制御ドメインにログインして、ldm migrate-domainコマンドを実行します。

ldm migrate-domainコマンド

注ーここでは、主なオプションだけを示しています。詳細は、お使いのバージョンの『Oracle VM Server for SPARC リファレンスマニュアル』を参照してください。

```
# ldm migrate-domain [-n] source-ldom target-host[:target-ldom]
```

- -n オプション
  移行できるかどうかのテストだけを行います。実際の移行は行いません。
- source-ldom
  移行元のゲストドメイン名を指定します。
- target-host
  移行先の物理パーティションに属する制御ドメインのホスト名を指定します。ホスト名はネットワーク上で有効なホスト名です。
- target-ldom
  移行先のゲストドメイン名を指定します。指定しない場合は移行元のゲストドメイン名と同じになります。

7.3 ゲストドメインの移行例

ライブマイグレーションの具体例を示します。この例では、ldom1という名前のゲストドメインを、ホスト名がtarget1である制御ドメインの物理パーティションに移行しています。

1. **rootユーザーとして制御ドメインにログインします。**
2. **ldm migrate-domainコマンドを実行し、ゲストドメインldom1を、ホスト名がtarget1である制御ドメインの物理パーティションに移行します。**

```
# ldm migrate-domain ldom1 target1
```

移行先の物理パーティションにある制御ドメインでのルートユーザーのパスワード入力を求められます。

3. **パスワードを入力します。**

```
Target Password:*******
```

4. **ゲストドメインの移行が実行されます。**
移行先の物理パーティションにldom1という名前のゲストドメインが作成され、移行元のldom1が削除されます。

付録 A

物理パーティションの動的再構成を使用した環境構築・手順例

ここでは、物理パーティションの動的再構成を使用するための環境構築例およびその構築手順を説明します。
Oracle VM Server for SPARC 3.2以降で、ビルディングブロック(BB)構成を活用したシステムの事例をTechnical Parkで紹介しています。詳細は、Technical park の「Building Blockで実現できる低コストな高可用性システム」を参照してください。
https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/technical/document/index.html
以下の環境を構築する場合を例に挙げて説明しています。

- 物理パーティションの動的再構成が未サポートのシステムからサポートされるシステムへアップデートする場合
- XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがない論理ドメインの構成の場合
- XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがある論理ドメインの構成の場合
- XCP 2220以降の版数で新規に導入したシステムを1BBから2BB構成に増設する場合
- SPARC64 Xプロセッサだけで構成された物理パーティションにSPARC64 X+プロセッサで構成されたシステムボードを増設する場合
- 制御ドメインのみの構成でシステムボードを活性交換する場合（Oracle VM Server for SPARC 3.2以降）

このうち、物理パーティションの動的再構成を使用した活性交換の手順は、以下で説明しています。

- XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがない論理ドメインの構成の場合
- XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがある論理ドメインの構成の場合

また、物理パーティションの動的再構成を使用した活性増設の手順は、以下で説明しています。

- XCP 2220以降の版数で新規に導入したシステムを1BBから2BB構成に増設する場合
- SPARC64 Xプロセッサだけで構成された物理パーティションにSPARC64 X+プロセッサで構成されたシステムボードを増設する場合
- 制御ドメインのみの構成でシステムボードを活性交換する場合（Oracle VM Server for SPARC 3.2以降）

A.1 物理パーティションの動的再構成が未サポートのシステムからサポートされるシステムへアップデートする場合

ここでは、SPARC M10で物理パーティションの動的再構成がサポートされていないXCP 2210以前の版数が適用されたシステムを、物理パーティションの動的再構成が使用できるシステムにアップデートするための環境構築例と、その構築手順を説明します。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、「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。

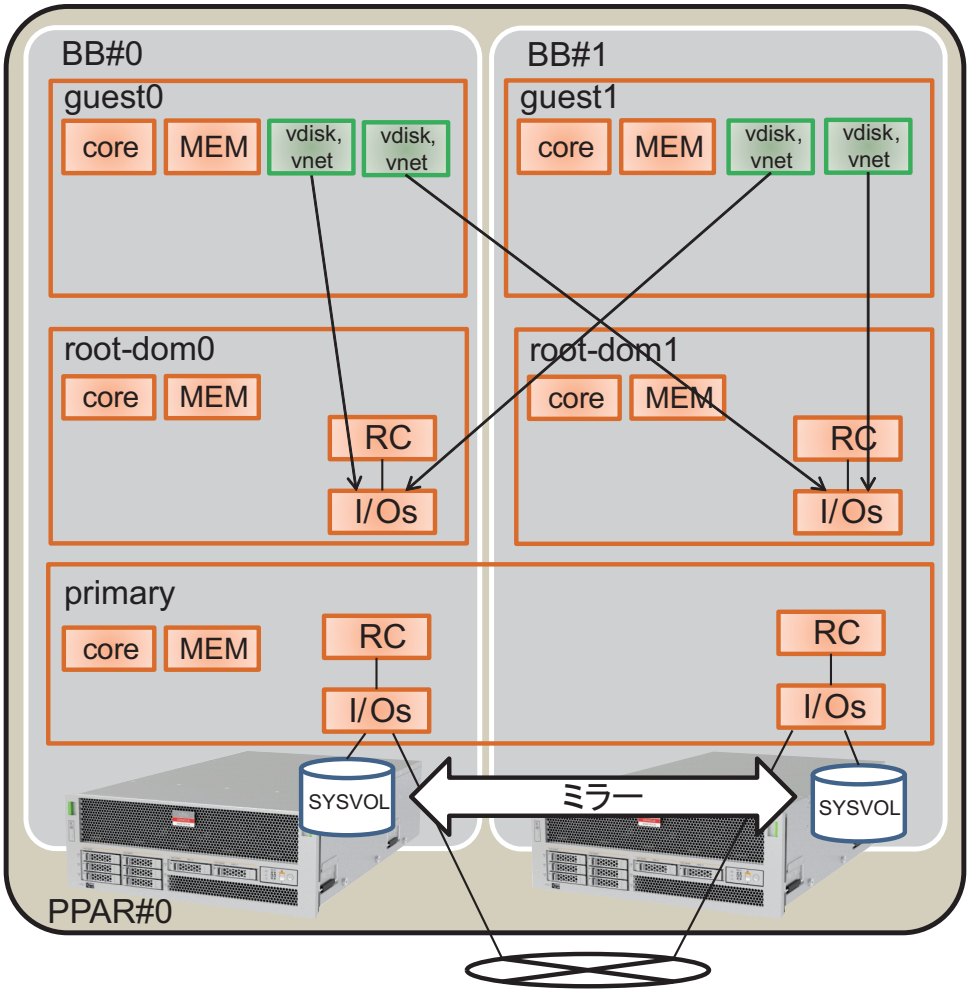
A.1.1 構築例

ここでは、2つのSPARC M10-4S、すなわち2つのシステムボード（PSB<BB>）で構成された（2BB構成）論理ドメインの構築例を示します。
物理パーティションの動的再構成でSPARC M10-4Sの活性交換を実施できるようにするには、以下の条件を満たす構成にしてください。

- 制御ドメインのシステムボリュームのI/Oデバイスやネットワークは、各SPARC M10-4Sのルートコンプレックス配下のI/Oデバイスを接続して、冗長構成にしてください。
  これはSPARC M10-4Sを削除した場合に、制御ドメインの再起動を伴ったI/Oデバイスの構成変更をしても、残りのSPARC M10-4SのI/Oデバイスを使って動作を継続できるようにするためです。
- SPARC M10-4Sごとにルートドメインを分けてください。
  これはSPARC M10-4Sを削除したことで、ほかのSPARC M10-4Sの物理I/Oデバイスを使ったサービスが使えなくなってしまうことを回避する必要があるためです。
- 業務は仮想I/Oデバイス（vdisk、vnet）で構成されるゲストドメインで運用してください。
  これにより、SPARC M10-4S削除時にルートドメインを停止して物理I/Oデバイスを切り離せるようにするとともに、業務を行っているゲストドメインには影響を与えないようにすることができます。

次の図は、物理パーティションの動的再構成が実施可能な構築例です。

図 A-1　物理パーティションの動的再構成が実施可能な構成例



A.1.2　構築手順

物理パーティションの動的再構成がサポートされていないシステムからサポートされるシステムへアップデートする手順は、次のとおりです。

1. **論理ドメイン構成情報をXMLファイルに保存します。**

   XCPファームウェアをアップデートしたあとPPAR DR機能を有効にすると、論理ドメイン構成は工場出荷時の状態（factory-default）に戻るため、XSCFに保存した論理ドメイン構成情報が使用できなくなります。そのため、現在の論理ドメイン構成情報をXMLファイルに保存してからXCPファームウェアをアップデートします。アップデート後、XMLファイルから論理ドメイン構成情報を復元することで、論理ドメインの再構築を簡単に実施できます。

   ここでは、XSCFに保存された論理ドメイン構成情報をXMLファイルに保存する

手順を説明します。

なお、XMLファイルに保存される構成情報の詳細は、『Oracle VM Server for SPARC 管理ガイド』を参照してください。

a. **制御ドメインにログインします。**

b. **保存する構成情報に切り替えます。**

制御ドメイン上でldm list-spconfigコマンドを実行し、論理ドメイン構成情報一覧を表示します。[current]と表示されている行が、現在適用されている構成情報を示しています。次の例ではldm-set3が適用されています。

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [current]
```

[next poweron]のみが表示されている場合は、XSCFと制御ドメインに保存されている論理ドメイン構成情報に差異があるため、ldm add-spconfigコマンドを実行し、現在の論理ドメイン構成情報を別の名前で保存しておきます。

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [next poweron]
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

[current]で示されている論理ドメイン構成情報が保存したい構成情報と一致している場合は、手順1.cに進んでください。

異なる場合は、ldm set-spconfigコマンドを実行し、保存したい構成情報を変更してください。

以下の例は、ldm-set1を保存するために、ldm set-spconfigコマンドで構成情報を切り替えています。

```
# ldm set-spconfig ldm-set1
# ldm list-spconfig
factory-default
ldm-set1 [next poweron]
ldm-set2
ldm-set3 [current]
```

XSCF上でpoweroffコマンドおよびpoweronコマンドを実行し、物理パーティション（PPAR）の電源を切断後、再投入します。このとき、論理ドメイン

の状態を、Oracle Solarisが起動している状態かinactive状態に移行させてから、poweroffコマンドを実行してください。

次の例は、PPAR-ID 0の電源を切断後、再投入しています。

```
XSCF> poweroff -p 0
XSCF> poweron -p 0
```

ldm list-spconfigコマンドを実行し、指定した論理ドメイン構成情報に設定されていることを確認します。次の例では、ldm-set1が現在の構成情報に設定されています。

```
# ldm list-spconfig
factory-default
ldm-set1 [current]
ldm-set2
ldm-set3
```

c. **ldm list-constraintsコマンドを実行し、現在の論理ドメイン構成情報をXMLファイルに保存します。**

ldm list-constraintsコマンドを実行し、現在の論理ドメイン構成情報をXMLファイルに保存します。

以下の例は、ldm-set1.xmlに現在の論理ドメイン構成情報を保存しています。保存したXMLファイルが消失することを防ぐため、別媒体などへバックアップしておきます。

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

d. **XMLファイルに保存されたかどうかを確認します。**

Oracle Solarisのmoreコマンドなどを実行し、XMLファイルに内容が保存されていることを確認します。

```
# more /ldm-set1.xml
<?xml version="1.0"?>
<LDM_interfaceversion="1.3" xmlns:xsi=http://www.w3.org/2001/XMLSchema-
instancce
```

e. **ほかの論理ドメイン構成情報を保存する場合は、手順1.b～1.eを繰り返します。**

ほかに保存したい構成情報がある場合、手順1.b～1.eを繰り返して保存してください。

2. **XSCF設定情報を保存します。**

XSCF設定情報は、USBデバイス、またはネットワーク経由で外部サーバに保存できます。以下に、それぞれの方法を説明します。

- **USBデバイスにXSCF設定情報を保存する場合**

a. **マスタXSCFにログインします。**

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。スタンバイ状態のXSCFの場合は、マスタXSCFへログインしなおしてください。

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

b. **マスタXSCFのUSBデバイスに設定情報を保存します。**

i. **マスタXSCFのXSCFユニットのパネル（背面パネル）にあるUSBポートにUSBデバイスを接続します。**

ii. **XSCF上のローカルなUSBデバイスに対して出力ファイル名を指定して、dumpconfigコマンドを実行します。**

指定したファイル名で、base64エンコーディングテキスト形式で保存されます。

```
XSCF> dumpconfig file:///media/usb_msd/backup-file.txt
operation completed
```

「operation completed」のメッセージが表示されたら、データの転送が正常に終了しています。

iii. **データの転送が終了したら、USBデバイスをUSBポートから取り外します。**

c. **PCのエディタ機能などを使用し、保存した設定ファイル内の以下の情報を確認します。**

- User-Comments: dumpconfigコマンドに-cオプションを指定したときのコメント
- Created: 保存した日時
- Platform: モデル名
- Serial-No: システムのシリアル番号

```
XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
```

```
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByaXNlAAAAAAAAAFLmZ6gAAPrfADhbdAAAAAIyMTExMjM0MDAzAAAA
...
```

- **ネットワーク経由で外部サーバに設定情報を保存する場合**

a. **マスタXSCFにログインします。**

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。スタンバイ状態のXSCFの場合は、マスタXSCFへログインしなおしてください。

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

b. **ネットワークを介してターゲットディレクトリを指定し、XSCF設定情報を保存します。**

ターゲットディレクトリおよび出力ファイル名を指定して、dumpconfigコマンドを実行します。指定したファイル名で、base64エンコーディングテキスト形式で保存されます。

```
XSCF> dumpconfig -u "user-name" "ftp://server/backup/backup-sca-
ff2-16.txt"
operation completed
```

「operation completed」のメッセージが表示されたら、データの転送が正常に終了しています。

c. **PCのエディタ機能などを使用し、保存した設定ファイルの情報を確認します。**

以下の情報を確認します。

- User-Comments: dumpconfigコマンドに-cオプションを指定したときのコメント
- Created: 保存した日時
- Platform: モデル名
- Serial-No: システムのシリアル番号

```
XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
```

```
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByaXNlAAAAAAAAAFLmZ6gAAPrfADhbdAAAAAIyMTExMjM0MDAzAAAA
...
```

3. **Oracle SolarisとOracle VM Server for SPARCをアップデートします。**
   物理パーティションの動的再構成をサポートしているOracle SolarisおよびOracle VM Server for SPARCを入手し、システムに適用します。

   a. **制御ドメインのOracle VM Server for SPARCをアップデートします。**

   お使いのサーバの最新の『プロダクトノート』を参照して最新のOracle SolarisおよびOracle VM Server for SPARCの版数を確認し、以下の手順でOracle VM Server for SPARCをアップデートします。

   i. **制御ドメインに使用する、Oracle VM Server for SPARCを入手します。**

   - **制御ドメインがOracle Solaris 10の場合**

     物理パーティションの動的再構成をサポートしているOracle VM Server for SPARCを入手します。入手方法は、お使いのサーバの最新の『プロダクトノート』を参照してください。

   - **制御ドメインがOracle Solaris 11の場合**

     SRU11.1.14.0以降を入手します。

   ii. **制御ドメインのOracle VM Server for SPARCやOracle Solarisをアップデートした場合は、制御ドメインを再起動します。**

   制御ドメインの再起動によって、I/Oドメインがパニックしたり、ゲストドメインのI/Oが停止したりすることがあるため、これらの論理ドメインがある場合は、事前にshutdownコマンドで停止させるか、制御ドメインからldm stop-domainコマンドを実行して停止させてください。

   停止させる必要のある論理ドメインかどうかは、次の方法で確認できます。

- **制御ドメインのからPCIeエンドポイントを割り当てているI/Oドメインの場合**

  ldm list-ioコマンドを実行し、論理ドメインに割り当てているPCIeエンドポイントのルートコンプレックス（BUS）が、primaryに割り当てられているかどうかで判断できます。次の例では、iodom0に割り当てているPCIeエンドポイント「/BB0/PCI3」と「/BB0/PCI4」のバス「PCIE2」がprimary（制御ドメイン）に割り当てられていることがわかります。なお、ここで説明する構成は「A.1.1　構築例」と異なるため、論理ドメインなどを変えて説明しています。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN   STATUS
----                                        ----   ---      ------   ------
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE1                                       BUS    PCIE1    primary  IOV
PCIE2                                       BUS    PCIE2    primary  IOV
PCIE3                                       BUS    PCIE3    primary  IOV
....
/BB0/CMUL/NET0                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                            PCIE   PCIE0    primary  OCC
/BB0/PCI0                                   PCIE   PCIE1    primary  OCC
/BB0/PCI3                                   PCIE   PCIE2    iodom0   OCC
/BB0/PCI4                                   PCIE   PCIE2    iodom0   OCC
/BB0/PCI7                                   PCIE   PCIE3    primary  OCC
/BB0/PCI8                                   PCIE   PCIE3    primary  EMP
....
```

- **制御ドメインの仮想サービスを割り当てているゲストドメインの場合**

  ldm list-bindings primaryを実行し、仮想ネットワークスイッチ（VSW）と接続先（PEER）の対応、仮想ディスクサービス（VDS）と仮想ディスクを使用している 論理ドメイン（CLIENT）の対応を確認します。次の例の場合、VSWのPEERと、VDSのCLIENTに「guestdom0」が設定されていることが確認できます。

```
# ldm list-bindings primary
....
VSW
    NAME             MAC               NET-DEV   ID   DEVICE     LINKPROP
DEFAULT-VLAN-ID PVID
VID                  MTU   MODE   INTER-VNET-LINK
    vsw0             xx:xx:xx:xx:xx:xx net0      0    switch@0             1
            1                        1500         on
        PEER                        MAC               PVID VID
  MTU   MAXBW      LINKPROP
 INTERVNETLINK
        vnet0@guestdom0             xx:xx:xx:xx:xx:xx 1
  1500
VDS
    NAME             VOLUME         OPTIONS          MPGROUP        DEVICE
....
        CLIENT                      VOLUME
```

```
        vdisk0@guestdom0            vol0
```

ldm stop-domainコマンドを実行して、上記で確認した論理ドメインを停止させます。次の例では、ldm stop-domainコマンドでiodom0とguestdom0を停止させ、その後、ldm list-domainコマンドを実行してbound状態になったことを確認しています。

```
# ldm stop-domain guestdom0
LDom guestdom0 stopped
# ldm stop-domain iodom0
LDom iodom0 stopped
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    58G      0.0%  6h 3m
guestdom0        bound      ------  5100    64    64G
iodom0           bound      ----v-  5000    32    32G
```

iii. **制御ドメインのOracle VM Server for SPARCをアップデートします。**

- **Oracle Solaris 10の場合**

  古いOracle VM Server for SPARCをアンインストールしてから、新しいOracle VM Server for SPARCをインストールします。詳細は、入手したOracle VM Server for SPARCのREADMEファイルを参照してください。

- **Oracle Solaris 11の場合**

  SRUを適用します。詳細は、入手したSRUのインストール手順書を参照してください。

iv. **shutdownコマンドを実行し、制御ドメインを再起動します。**

```
# shutdown -i6 -g0 -y
....
```

v. **ldm start-domainコマンドを実行し、上記iiで停止させた論理ドメインを起動させます。**

次の例では、ldm start-domainコマンドで「iodom0」と「guestdom0」を起動させたあと、ldm list-domainコマンドでactive状態になったことを確認しています。

```
# ldm start-domain guestdom0
# ldm start-domain iodom0
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    58G      0.0%  6h 3m
guestdom0        active     -n----  5100    64    64G
iodom0           active     -n--v-  5000    32    32G
```

b. **制御ドメイン以外の論理ドメインのOracle Solarisをアップデートします。**

お使いのサーバの最新の『プロダクトノート』を参照して、制御ドメイン以

外の論理ドメインのOracle Solarisをアップデートしてください。アップデートの手順は、各アップデート関連の情報を参照してください。

4. **XCPファームウェアをアップデートします。**
SPARC64 Xプロセッサで構成された物理パーティションに、SPARC64 X+プロセッサ搭載のPSB（BB）を増設するためには、SPARC64 Xプロセッサで構成された物理パーティションをXCP 2220以降のXCPファームウェアにアップデートします。

a. **最新XCPファームウェアを入手します。**

i. **XCPファームウェアのプログラムファイルをダウンロードします。**

ウェブサイトから本システムに接続しているPCの任意のフォルダにXCPファームウェアのプログラムファイル（XCPxxxx.tar.gzかXCPxxxx.exe）をダウンロードします。

お使いのサーバのファームウェアは、以下の方法で入手してください。

- **国内サイト**

  ファームウェアは、SupportDeskを契約されているお客さまは、SupportDesk-Webから入手が可能です。

- **グローバルサイト**

  ファームウェアの最新ファイルの入手方法については、営業担当者にお問い合わせください。

  以下のファイルが提供されます。

  - ファームウェアプログラムファイル（XSCF Control Package（XCP）ファイル）
  - XSCF拡張MIB（XSCF-SP-MIB）定義ファイル

ii. **ダウンロードしたプログラムファイルのXCP版数を確認します。**

ダウンロードしたXCPファームウェアのプログラムファイルの版数を確認します。XCP版数はファームウェアプログラム（tar.gz形式）のファイル名にある数字4桁を参照し、アップデートするXCPファームウェア版数であることを確認します。例えば、プログラムファイルの名前が「XCP2220.tar.gz」の場合は2220がXCP版数です。

iii. **ダウンロードしたプログラムファイルを解凍します。**

ダウンロードしたXCPファームウェアのプログラムファイルを解凍します。システムにインポートするためのXCPイメージファイルが展開されます。例えば、「XCP2220.tar.gz」を解凍した場合、「BBXCP2220.tar.gz」が展開されます。

b. **現在のXCPファームウェア版数を確認します。**

i. **マスタXSCFにログインします。**

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。スタンバイ状態のXSCFの場合は、マスタXSCFへログインしなおしてください。

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

ii. **versionコマンドを実行し、現在のシステムのXCP版数を確認します。**

ファームウェアをアップデートする前に、現在のシステムのXCP版数を必ず確認してください。次の例では、-c xcpオプションを付けてversionコマンドを実行し、XCP版数が物理パーティションの動的再構成をサポートしていないXCP 2041であることを確認しています。

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2041
XCP1 (Reserve): 2041
BB#01-XSCF#0(Standby)
XCP0 (Current): 2041
XCP1 (Reserve): 2041
```

c. **XCPファームウェアをアップデートします。**

i. **poweroffコマンドを実行し、すべての物理パーティションを電源切断します。**

```
XSCF> poweroff -a
```

showpparstatusコマンドを実行し、すべての物理パーティションの電源が切断されていることを確認します。

```
XSCF> showpparstatus -a
PPAR-ID         PPAR Status
0               Powered Off
```

ii. **showhardconfコマンドを実行し、マスタXSCF、スタンバイ状態のXSCFの [Status] が「Normal」になっていることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Service;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
   :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB;
   :
```

iii. **getflashimageコマンドを実行し、XCPイメージファイルをインポートします。**

次の例では、マスタXSCFのXSCFユニットのパネル（背面パネル）にあるUSBポート（MAINTENANCE ONLYと印字）にUSBデバイスを接続して、XCPイメージファイルをインポートしています。

```
XSCF> getflashimage file:///mnt/share/scf-firm/xscf/user/scfadmin/BBXCP2220.tar.gz
  0MB received
  1MB received
  2MB received
...
 86MB received
 87MB received
 88MB received
Download successful: 90539 Kbytes in 58 secs (1562.668 Kbytes/sec) Checking
file...
MD5: 2b89c06548205ce35a8ecb6c2321d999
```

「Download successful: ...」および「MD5: ...」の正常終了のメッセージが表示されたら、XCPイメージファイルのインポートは終了です。

注－XCPイメージファイルインポート時に「Warning: About to delete existing old versions.」のメッセージが表示されることがあります。これは、インポート済みである古いXCPイメージファイルを削除するかを確認するメッセージです。「Continue?」が表示されたら、「y」を入力して、インポート処理を継続してください。

注－XCPイメージファイルインポート後に「Error:File is invalid or corrupt」のメッセージが表示された場合は、インポートしたXCPイメージファイルが正しいファイルでなかったことを意味します。XCPイメージファイルが壊れている可能性があるため、再度、正しいXCPイメージファイルを入手し、インポートしてください

---

iv. **getflashimage -lコマンドを実行し、インポートしたXCPイメージファイルの版数を確認します。**

```
XSCF> getflashimage -l
Existing versions:
        Version                 Size  Date
        BBXCP2220.tar.gz    92712351  Thu May 23 15:01:42 JST 2014
```

v. **flashupdate -c checkコマンドを実行し、インポートしたXCPイメージファイルがアップデートに使用できるファイルかどうかを確認します。**

flashupdateコマンドの実行直後にshowresultコマンドを実行し、終了値が0であればアップデートできます。

```
XSCF> flashupdate -c check -m xcp -s 2220
XCP update is started. [3600sec]
0XSCF>
XSCF> showresult
0
```

```
XSCF>
```

vi. **flashupdateコマンドを実行し、ファームウェアをアップデートします。**

```
XSCF> flashupdate -c update -m xcp -s 2220
The XSCF will be reset. Continue? [y|n]: y
XCP update is started. [3600sec]
0.....30.....60.....90.....120.....150.....180.....210.....240.....-
270.....300.....330.....360.....390.....420.....450.....480.....510.....|
540.....570.....600.....630.....660.....690.....720.....750.....780.....-
810.....840.....870.....900.....930
:
```

ここでXSCFが再起動され、XSCFのセションが切断されます。この時点では、XCPファームウェアのアップデートはまだ完了していません。

注ーファームウェアの作業時間は、アップデートに約60分、アップデート完了後のXSCFの自動切り替えに約10分かかります。

注ーアップデートを安全に行うため、XCPファームウェアのアップデートが完了したことを表すメッセージ「XCP update has been completed」を確認されるまで、物理パーティションの電源は操作しないでください。

---

vii. **再度マスタXSCFに接続します。**

XSCFの再起動の直後は、マスタとスタンバイ状態のXSCFが元の状態と逆になっています。例えば、ファームウェアアップデートをBB-ID 0のマスタXSCFで実行した場合にXSCFに再接続すると、BB-ID 1がマスタ、BB-ID 0がスタンバイ状態になっています。

注ー引き継ぎIPアドレスを設定している場合は、引き継ぎIPアドレスで接続すると自動的にマスタXSCFに接続されます。

---

viii. **showbbstatusコマンドを実行し、マスタXSCFにログインしていることを確認します。スタンバイ状態のXSCFになっている場合、マスタXSCFに接続しなおします。**

```
XSCF> showbbstatus
BB#01 (Master)
```

ix. **showlogs monitorコマンドを実行し、XCPファームウェアアップデートの完了を確認します。**

「XCP update has been completed」のメッセージが表示されていれば、XCPファームウェアのアップデートが完了しています。

```
XSCF> showlogs monitor
May 29 09:38:05 M10-4S-0 Event: SCF:XCP update is started (XCP version=2220:
last version=2041)
```

```
May 29 09:40:31 M10-4S-0 Event: SCF:XSCF update is started (BBID=0, bank=0)
May 29 09:40:46 M10-4S-0 Event: SCF:XSCF update is started (BBID=1, bank=0)
May 29 09:41:03 M10-4S-0 Event: SCF:XSCF writing is performed (BBID=0, XSCF
version=02220000)
:
May 29 09:50:41 M10-4S-0 Event: SCF:XSCF bank apply has been completed
(BBID=1, bank=0, XCP version=2220: last version=2041)
May 29 10:01:50 M10-4S-0 Event: SCF:XSCF bank apply has been completed
(BBID=0, bank=0, XCP version=2220: last version=2041)
May 29 10:01:51 M10-4S-0 Event: SCF:Change Master Start (BB#01)May 29 10:03:26
 M10-4S-1 Event: SCF:Change Master Complete (BB#01)
May 29 10:05:00 M10-4S-1 Event: SCF:Standby XSCF Ready (BBID#00)
:
May 29 10:32:38 M10-4S-1 Event: SCF:XCP update has been completed (XCP
version=2220: last version=2041)
May 29 10:32:39 M10-4S-1 Event: SCF:This XSCF will be switched back to standby
mode after completion of firmware update
May 29 10:32:39 M10-4S-1 Event: SCF:Change Master Start (BB#00)
May 29 10:33:29 M10-4S-1 Event: SCF:Change Master Complete (BB#00)
May 29 10:42:29 M10-4S-1 Event: SCF:Standby XSCF Ready (BBID#01)
```

このあと、続いてXSCFのマスタ、スタンバイの切り替えが自動で行われます。「This XSCF will be switched back to standby mode after completion of firmware update」、「Change Master Complete」、および「Stand by XSCF Ready」のメッセージが表示されていれば、切り替えが完了しています。

注ー「XCP update has been completed」のメッセージが表示されていない場合は、まだアップデートが完了していません。再度、showlogs monitorコマンドを実行し、アップデートの完了を確認してください。

注ーファームウェアのアップデートを実行すると、マスタ、スタンバイ状態の2回切り替わります。その結果、マスタXSCFはflashupdateコマンドを実行したXSCFに戻ります。マスタ、スタンバイ状態の切り替われるときに、XSCFセションが切断されることがあります。XSCFセションが切断された場合は、接続しなおしてください。

切り替えが完了したことを確認するために、showhardconfコマンドを実行し、マスタXSCF、およびスタンバイ状態のXSCFの [Status] が「Normal」になっていることを確認します。

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Service;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
   :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                        ;
        + Power_Supply_System: ;
```

```
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB;
    :
```

注ーshowhardconfコマンドを実行したときに「Cannot communicate with the other XSCF. Check the other XSCF's state.」が表示された場合、XSCFのマスタとスタンバイの切り替えが完了していません。再度、showhardconfコマンドを実行し、切り替えの完了を確認してください。

---

x. **マスタとスタンバイ状態のXSCFをアップデート前の状態に戻したい場合、switchscfコマンドを実行します。**

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

XSCFの再起動によりXSCFのセションが切断されるため、マスタXSCFに接続しなおします。

切り替えの完了を確認するために、showhardconfコマンドを実行し、マスタXSCF、およびスタンバイ状態のXSCFの [Status] が「Normal」になっていることを確認します。

注ーshowhardconfコマンドを実行したときに「Cannot communicate with the other XSCF. Check the other XSCF's state.」が表示された場合、XSCFのマスタとスタンバイの切り替えが完了していません。再度、showhardconfコマンドを実行し、切り替えの完了を確認してください。

---

xi. **versionコマンドを実行し、ファームウェアの版数が新しくなっていることを確認します。**

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
BB#01-XSCF#0(Standby)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
```

注ー物理パーティションの電源を投入してファームウェアをアップデートした場合、CMUファームウェアのカレントバンク（Current bank）が、新しくなっています。物理パーティションの電源を停止してファームウェアアップデートした場合、CMUファームウェアのリザーブバンク（Reserve bank）およびカレントバンク（Current bank）の両方が新しくなっています。XCP版数と対応するCMUファームウェアの版数は、最新のプロダクトノートの「これまでのXCPファームウェア版数とサポート情報」を参照してください。

---

5. **XSCFのPPAR DRモードや起動する論理ドメインを指定し、物理パーティションを起動します。**

注ー物理パーティションの動的再構成を使用するためには、setpparmodeコマンドを使用してPPAR DRモードを有効に設定する必要があります。ただし、PPAR DRモードを有効に設定すると、論理ドメイン構成情報はfactory-defaultに変更され、既存の論理ドメイン構成情報は使用できなくなります。新たに論理ドメイン構成情報を作成する必要があります。
既存の論理ドメイン構成情報を使用する場合は、PPAR DRモードを無効にしてください。有効に設定し、かつ既存の論理ドメイン構成情報で物理パーティションの入力電源を投入すると、HypervisorでAbortする、Oracle VM Server for SPARCが正常に起動しない、などの現象が発生します。

---

a. **物理パーティションの動的再構成機能（PPAR DR機能）を有効にします。**

i. **showpparmodeコマンドを実行して、PPAR DRモードの設定を参照します。**

次の例は、物理パーティション番号0（PPAR ID 0）を指定しています。

```
XSCF> showpparmode -p 0
Host-ID                  :00000002
Diagnostic Level         :min
Message Level            :normal
Alive Check              :on
Watchdog Reaction        :reset
Break Signal             :on
Autoboot(Guest Domain)   :on
Elastic Mode             :off
IOreconfigure            :false
PPAR DR(Current)         :-
PPAR DR(Next)            :off
```

ii. **PPAR DRモードが無効の場合は、setpparmodeコマンドを実行し、有効に設定します。**

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

iii. **showpparmodeコマンドを実行して、PPAR DRモードが有効に設定されていることを確認します。**

```
XSCF> showpparmode -p 0
Host-ID                  :00000002
Diagnostic Level         :min
Message Level            :normal
Alive Check              :on
Watchdog Reaction        :reset
Break Signal             :on
Autoboot(Guest Domain)   :on
Elastic Mode             :off
IOreconfigure            :false
PPAR DR(Current)         :-
```

```
PPAR DR(Next)              :on
```

b. **物理パーティションの電源を投入します。**

i. **showdomainconfigコマンドを実行して、起動する論理ドメイン構成情報がfactory-defaultであることを確認します。**

次の例は、物理パーティション番号0（PPAR ID 0）を指定しています。

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) : factory-default
(Next) : factory-default
----------------------------------------------------------------
Index :1
config_name :factory-default
domains :1
date_created:-
----------------------------------------------------------------
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-0811:34:56'
----------------------------------------------------------------
Index :3
config_name :ldm-set2
domains :20
date_created:'2012-08-0912:43:56':
```

ii. **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。**

次の例は、物理パーティション番号0（PPAR ID 0）を指定しています。

```
XSCF> poweron -p 0
```

6. **XMLファイルを使って論理ドメインを再構築します。**

物理パーティションの起動が完了したあとに、手順1で保存したXMLファイルから論理ドメインを再構築します。

a. **制御ドメイン上でシステムがfactory-defaultで起動したことを確認します。**

ldm list-spconfigコマンドを実行し、factory-defaultの行に[current]が表示されていることを確認します。

```
# ldm list-spconfig
factory-default  [current]
ldm-set1
ldm-set2
ldm-set3
```

b. **ldm init-systemコマンドでXMLファイルから論理ドメイン構成情報を反映します。**

ldm init-systemコマンドを実行して保存したXMLファイルの設定を反映し、shutdownコマンドを実行して制御ドメインを再起動します。

```
# ldm init-system -i /ldm-set1.xml
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.
# shutdown -y -g0 -i6
```

c. **物理パーティションの動的再構成を可能にするために、各論理ドメインのメモリサイズを調整します。**

ldm list-domainコマンドなどを実行し、XMLファイルから論理ドメインが構成されたことを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    58G      0.0%  6m
guest0           inactive   ------          64    64G
guest1           inactive   ------          64    64G
root-dom0        inactive   ------          32    32G
root-dom1        inactive   ------          32    32G
```

Oracle VM Server for SPARCのバージョンが3.2よりも古い場合、物理パーティションの動的再構成が行えるようにするために、論理ドメインに割り当てるメモリサイズを、「論理ドメインに割り当てるコア数×256 MB」の倍数に設定してください。制御ドメイン以外の論理ドメインのメモリサイズを再設定する場合は、ldm set-memoryコマンドでメモリサイズを設定してください。

制御ドメインのメモリサイズを再設定する場合、ldm start-reconfコマンドで遅延再構成モードに移行して、ldm set-coreコマンドで同じコア数を設定してから、ldm set-memoryコマンドで再設定を行い、Oracle Solarisを再起動してください。

Oracle VM Server for SPARCのバージョンが3.2よりも古い場合は、まず、元の設定値（58 GB=59392 MB）に近い値にするために8192 MBの何倍にすればいいかを求めると、59392÷8192=7.25になるため、切り捨てて7にします。その結果、制御ドメインに割り当てなおすメモリサイズは、81924 MB×7=56 GBになります。

次に、制御ドメインのメモリを56 GBに再設定するコマンドの実行例を示します。

```
# 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 32 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-memory 56G 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.
--------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
....
```

ldm list-domainコマンドを実行し、メモリサイズ（MEMORY）が正しく設定されていることを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  6m
guest0           inactive   ------          64    64G
guest1           inactive   ------          64    64G
root-dom0        inactive   ------          32    32G
root-dom1        inactive   ------          32    32G
```

ldm bind-domainコマンドを実行し、各論理ドメインをバインドします。

```
# ldm bind-domain root-dom0
# ldm bind-domain root-dom1
# ldm bind-domain guest0
# ldm bind-domain guest1
```

なお、PPAR DR機能が無効の状態で保存したXMLファイルから論理ドメイン構成情報を復元すると、PPAR DR機能が有効な状態ではハイパーバイザ領域のメモリサイズが増えることで、メモリの割り当てが不足することがあります。メモリの割り当てが不足した状態でldm bind-domainコマンドを実行すると、以下のようなエラーメッセージとともにエラーで終了し、inactive状態のままになります。

```
# ldm bind-domain guest1
Not enough free memory present to meet this request.
Could not bind requested memory for LDom guest1.
```

この場合、手順6.cを参照して、論理ドメインに割り当てられているメモリサイズを再設定してください。すでにbound状態になった論理ドメインのメモリを調整する場合は、ldm unbind-domainコマンドを実行しバインドを解除してから設定してください。再設定が完了したら、再度、ldm bind-domainコマンドでバインドしなおしてください。

d. **メモリリソースの割り当て状況を確認します。**

ldm list-devices -aコマンドを実行し、各論理ドメインに割り当てられているメモリの連続領域（メモリブロック）の状態を確認して、細かいメモリブロックが多数存在しないことを確認してください。目安としては、SIZEが256 MB～512 MB程度のメモリブロックが多数存在しないことを確認してください。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom0
    0x720000000000      32G             root-dom1
    0x740000000000      32G             guest0
    0x760000800000      1272M           _sys_
    0x760050000000      2G              primary
    0x7600d0000000      29440M          guest1
    0x780000000000      32G             guest0
    0x7a0000000000      32G             guest1
    0x7c0000000000      32G             primary
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      3328M           guest1
    0x7e0150000000      4864M
    0x7e0280000000      22G             primary
```

メモリブロックが細かく分割されている場合は、ldm unbind-domainコマンド実行し、各論理ドメインをunbind状態に戻してから、手順6.cに従って再設定してください。

e. **すべての論理ドメインを起動します。**

ldm start-domainコマンドを実行して、制御ドメイン以外の論理ドメインを起動します。物理I/Oの割り当て関係（ルートドメインや、I/Oドメイン）や、仮想I/Oのサービスとクライアントの関係など、依存関係がある場合は、物理I/Oや仮想I/Oサービスの提供元のドメインから先に起動してください。

以下の例では、構成例を前提とした順序で起動しています。ldm start-domainコマンドで、root-dom0、root-dom1、 guest0、guest1の順に起動して、ldm list-domainコマンドで起動されたことを確認しています。 [STATE]が「active」になっていて、FLAGSの文字列の左から2番目の文字が「n」になっていれば、Oracle Solarisが起動されたことを示しています。

```
# ldm start-domain root-dom0
LDom root-dom0 started
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm start-domain guest0
LDom guest0 started
# ldm start-domain guest1
LDom guest1 started
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  15m
guest0           active     -n----  5002    64    64G      0.4%  18s
guest1           active     -n----  5003    64    64G      7.9%  9s
root-dom0        active     -n--v-  5000    32    32G      0.5%  24s
root-dom1        active     -n--v-  5001    32    32G      1.6%  16s
```

f. **XSCFに論理ドメイン構成情報を保存します。**

論理ドメイン構成情報を復元したあと、ldm add-spconfigコマンドを実行し、

XSCFに復元した論理ドメイン構成情報を保存します。次の例では、ldm-set4の構成情報をXSCFに保存しています。

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4   [current]
```

g. **復元する論理ドメイン構成情報が複数ある場合、factory-default構成に戻して手順6.bから6.gを繰り返します。**

次の例は、factory-default構成に戻しています。
XSCFからshowdomainconfigコマンドを実行し、起動する論理ドメイン構成情報を確認します。

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) :ldm-set1
(Next) :ldm-set1
----------------------------------------------------------------
Index :1
config_name :factory-default
domains :1
date_created:-
----------------------------------------------------------------
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-0811:34:56'
----------------------------------------------------------------
....
```

setdomainconfigコマンドを実行し、物理パーティション番号0（PPAR ID 0）にfactory-defaultのIndex1を指定します。

```
XSCF> setdomainconfig -p 0 -i 1
```

poweroff、poweronコマンドを実行し、物理パーティションを再起動します。起動が完了したら、手順6.bから6.gを繰り返してください。

```
XSCF> poweroff -p 0
...
XSCF> poweron -p 0
...
```

A.2 XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがない論理ドメインの構成の場合

「A.2　XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがない論理ドメインの構成の場合」から「A.4　XCP 2220以降の版数で新規に導入したシステムを1BBから2BB構成に増設する場合」では、SPARC M10-4Sの2BB構成で物理パーティションの動的再構成を使用できるシステムの構成例、および新規に構築する手順を説明します。
なお、前提として、SPARC M10-4SのファームウェアにはXCP 2220以降が使用され、『SPARC M10-4S インストレーションガイド』に従い、XSCFの設定や、2BB構成でシステムの設置が行われているものとして説明します。

ここでは、2つのSPARC M10-4S、すなわち2つのシステムボード（PSB<BB>）で構成される物理パーティションで、SPARC M10-4Sを切り離すときにリソースを移動するために必要とするハードウェアリソース（CPUコアとメモリ）の空きがない構成を構築する手順の例と、SPARC M10-4Sの活性交換を行う手順の例を説明します。
なお、この場合、SPARC M10-4Sの活性交換時に一時的に論理ドメインに割り当てたCPUコアやメモリを削減することになります。

ここでの説明は、SPARC M12でも同様です。

A.2.1 構成例

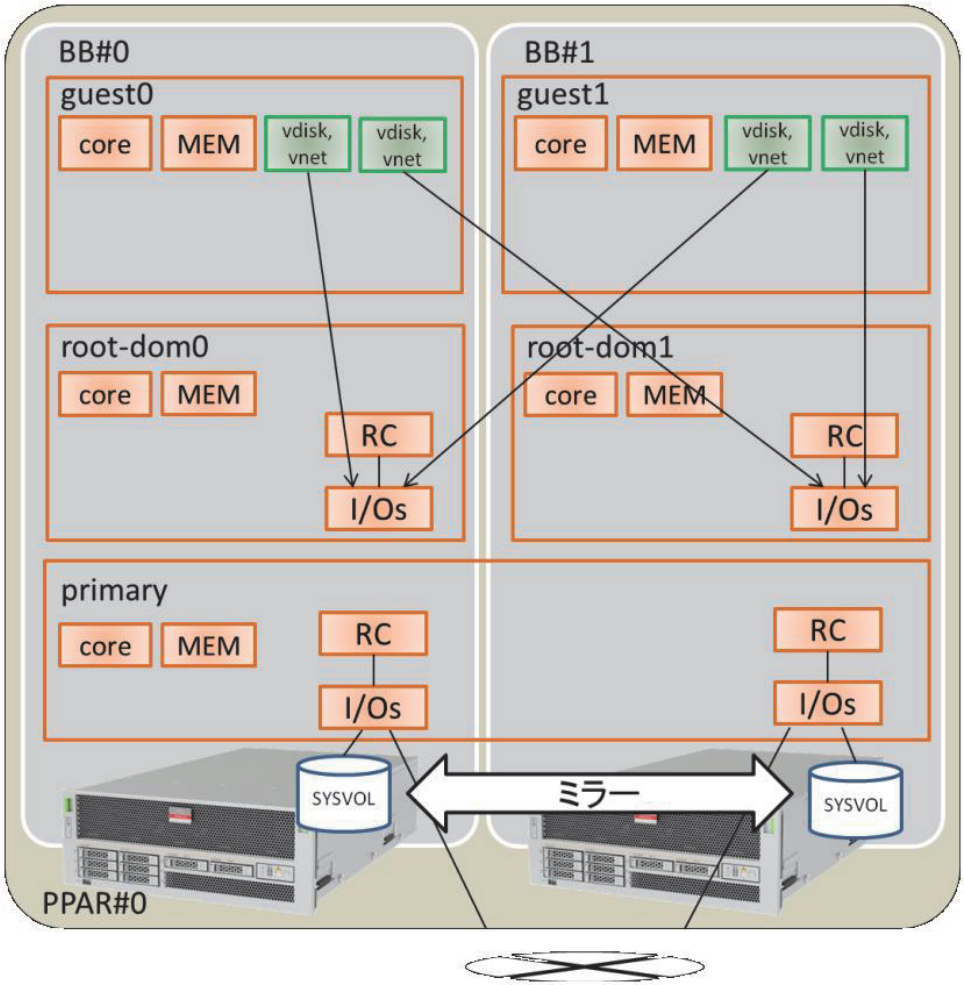
ここでは、2つのSPARC M10-4S、すなわち2つのシステムボード（PSB<BB>）を使った（2BB構成）論理ドメインの構成例を示します。
物理パーティションで動的再構成を実施できるようにするために、以下の条件を満たすI/O構成にする必要があります。

- 制御ドメインのシステムボリュームディスクやネットワークインターフェースは、各SPARC M10-4SのI/Oを接続して冗長構成にする必要があります。これはSPARC M10-4Sを削除した場合に、残りのSPARC M10-4SのI/Oを使って動作を継続できるようにするためです。
- SPARC M10-4Sごとにルートドメインを配置して、ゲストドメインへ仮想I/Oを提供するための専用のルートドメインにする必要があります。これは、ルートドメインを業務で利用していると、SPARC M10-4Sを切り離すときに、ルートドメインを停止する必要があり、業務が停止してしまうためです。
- 業務はゲストドメイン上で運用する必要があります。このゲストドメインの仮想I/Oデバイス（vdisk、vnet）は、各ルートドメインの仮想I/Oサービスを割り当てて、冗長構成にする必要があります。これは、片方のSPARC M10-4Sが切り離されても、残りのSPARC M10-4S上のルートドメインの仮想I/Oサービスを使って、I/Oアクセスを継続できるようにするためです。これにより、SPARC M10-4Sを切り離しても、ゲストドメインには影響を与えず、業務の運用を継続できるようになります。

次の図は、上記の条件を満たした構成例の概略図です。なお、各論理ドメインのI/O

構成は簡略化しています。

図 A-2　2BB構成で運用継続可能な構成例



以下の表では、2BB構成で制御ドメイン、ルートドメイン、ゲストドメインに割り当てるCPUコア、メモリ、I/O構成の例を示しています。CPUコアとメモリについては、SPARC M10-4Sの切り離しの際に各論理ドメインに割り当てられたCPUコアとメモリを移動できるだけの空きリソースがない状態になっています。

表 A-1　2BB構成で空きハードウェアリソースがない構成の例

論理ドメイン名	primary	root-dom0	root-dom1	guest0	guest1
CPUコア（アクティベーション：128）	32	16	16	32	32
メモリ（合計：256 GB）	56 GB(*1)	32 GB	32 GB	64 GB	64 GB

表 A-1　2BB構成で空きハードウェアリソースがない構成の例 (続き)

論理ドメイン名	primary	root-dom0	root-dom1	guest0	guest1
物理I/O （ルートコンプレックスの割り当て）	オンボード#0 （PCIE0、PCIE4） オンボード#1 （PCIE8、PCIE12）	PCIE1、PCIE2, PCIE3、PCIE5, PCIE6、PCIE7	PCIE9、PCIE10, PCIE11、PCIE13, PCIE14、PCIE15	－	－
仮想I/Oデバイス	－	vds0 vsw0、vsw1	vds1 vsw10、vsw11	vdisk0、vdisk10 vnet0、vnet10	vdisk1、vdisk11 vnet1、vnet11

*1:　上記の構成例では、制御ドメインに割り当てるメモリサイズは、ほかの論理ドメインに割り当てたメモリの残りとして割り当てる例として算出しています。
まず、物理的なメモリ全体の容量から、制御ドメイン以外に割り当てたサイズを引いた残りサイズを求めると、64 GBになります。
256 GB － 32 GB － 32 GB － 64 GB － 64GB ＝ 64 GB
実際に制御ドメインに割り当てられる最大サイズは、ここから、ハイパーバイザに割り当てられるメモリサイズ（2 GB＋1.25 GB）を引いたサイズ（60.75 GB）までになります。
64 GB － 2 GB － 1.25 GB ＝ 60.75 GB
さらにOracle VM Server for SPARCのバージョンが3.2よりも古い場合、物理パーティションの動的再構成を行うための条件として（コア数×256 MB）の倍数にするため、60.75 GB未満で計算すると、56 GBとなります。
60.75 GB ÷ （32コア×256 MB） ＝ 7.59
整数倍にするため、切り捨てて「7」にします。
（32コア × 256 MB） × 7 ＝ 56 GB

注－論理ドメインに割り当てるメモリサイズは、以下の2つの点について注意して設計する必要があります。詳細は、「2.4.1　論理ドメインの構築に関する留意点」を参照してください。
- 論理ドメインに割り当てられるメモリサイズは、物理的なメモリサイズより小さくなります。
- Oracle VM Server for SPARCのバージョンが3.2よりも古い場合は、物理パーティションの動的再構成を可能にするために、（コア数×256 MB）の倍数にしてください。

A.2.2　物理パーティションの構築手順例

1. **マスタXSCFにログインします。**
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の場合は、マスタXSCFへログインしなおしてください。

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

2. **メモリのミラーモードを設定します。**
a. **showfruコマンドを実行し、SPARC M10-4Sのシステムボード（PSB<BB>）のメモリのミラーモード設定を確認します。**

次の例では、PSB 00-0配下のデバイス設定情報を表示しています。

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
    cpu 00-0-0 no
    cpu 00-0-1 no
    cpu 00-0-2 no
    cpu 00-0-3 no
```

b. **メモリのミラーモードを使用する場合、setupfruコマンドを使用し、メモリのミラーモードを設定します。**

メモリのミラーモードを使用しない場合は、この手順は不要です。

メモリのミラーモードの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「14.1　メモリをミラー構成にする」を参照してください。

次の例では、PSB 00-0のすべてのCPUをミラーモードに設定しています。

```
XSCF> setupfru -m y sb 00-0
```

注－SPARC M12をお使いの場合、メモリをミラー構成にする場合は、-c mirror=yesを指定します。
例： XSCF> **setupfru -c mirror=yes sb 00-0**

---

showfruコマンドを実行し、メモリのミラーモード設定を確認します。

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
    cpu 00-0-0 yes
    cpu 00-0-1 yes
    cpu 00-0-2 yes
    cpu 00-0-3 yes
```

3. **物理パーティション構成情報を作成します。**

a. **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
```

b. **setpclコマンドを実行し、物理パーティション構成情報にシステムボードを登録します。**

setpclコマンドを実行し、SPARC M10-4Sのシステムボード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 00-0とPSB 01-0）を、物理パーティション0の論理システムボード（LSB 00とLSB 01）に対応付けています。

```
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。**

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                          Running
            00      00-0
            01      01-0
```

コンフィグレーションポリシー、IO無効化オプション（No-IO）、メモリ無効化オプション（No-Mem）の詳細を表示するには、-vオプションを指定します。

```
XSCF> showpcl -v -p 0
PPAR-ID   LSB    PSB    Status   No-Mem   No-IO    Cfg-policy
00                      Running
                                                  System
          00     00-0            False    False
          01     01-0            False    False
```

コンフィグレーションポリシー、IO無効化オプション（No-IO）、メモリ無効化オプション（No-Mem）の設定を変更する場合は、setpclコマンドを使用します。

setpclコマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

4. **物理パーティションにシステムボード（PSB<BB>）を割り当てます。**

a. **showboards -aコマンドを実行し、PSBの状態を確認します。**

showboards -aコマンドを実行し、それぞれのPSB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、PSB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、PSB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PSBの状態を確認します。

この例では、それぞれのPSB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、それぞれのPSBの [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
```

5. **CPUコア アクティベーションキーを登録し、CPUコアリソースを割り当てます。**
   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。

      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ません。

```
XSCF> showcodactivation
Index Description Count
------- ----------- ------
```

注ー登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。

---

   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登録します。**

      CPUコア アクティベーションキーを登録する方法の詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「5.3　CPUコアリソースを追加する」を参照してください。

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber:10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtxgvZPsrtYguqiNUieB+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            2
      1 PROC            2
      2 PROC            2
      3 PROC            2

--- 中略 ---

     62 PROC            2
     63 PROC            2
```

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コア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、128個のCPUコアリソース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 -c set 128
PROC Permits assigned for PPAR 0 : 0 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注－XCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

---

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コア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128個のCPUコアリソースが、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ています。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

6. **resetdateoffsetコマンドを実行し、XSCFが管理する時刻の差分をリセットします。**

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
```

7. **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                  :00000002
Diagnostic Level         :min
Message Level            :normal
Alive Check              :on
Watchdog Reaction        :reset
Break Signal             :on
Autoboot(Guest Domain)   :on
Elastic Mode             :off
IOreconfigure            :false
PPAR DR(Current)         :-
PPAR DR(Next)            :on
```

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。

setpparmodeコマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

```
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
```

8. **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> poweron -p 0
```

9. **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
```

10. **Oracle SolarisとOracle VM Server for SPARCをインストールします。**
制御ドメインにOracle SolarisとOracle VM Server for SPARCをインストールします。

物理パーティションの動的再構成に必要なOracle Solarisの版数や条件などについては、表 1-3および表 1-4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/en/）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.xシステムのインストール』

- Oracle VM Server for SPARC

  『Oracle VM Server for SPARC 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」

11. **論理ドメインを構築します。**

    ここでは、「表 A-1　2BB構成で空きハードウェアリソースがない構成の例」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ほか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て、コンソールサービスを設定します。

    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。

```
# 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、構成例に従って、PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7、PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15を削除するコマンドの例の一部を示します。

```
# ldm remove-io PCIE1 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 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、構成例に従って制御ドメインのCPUコアを32個、メモリサイズ

を56 GBに設定する例を示します。

```
# ldm set-core 32 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-memory 56G 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.
------------------------------------------------------------------------------
```

注ー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、物理パーティションの動的再構成が可能なCPUコアとメモリの配置の条件を満たしやすくなります。CPUコアとメモリの配置に関する条件の詳細は、「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 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。

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

構成情報を保存してから、Oracle Solarisを再起動します。

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で保存されたことを確認して、最後に、Oracle Solarisを再起動する例を示しています。

```
# ldm list-spconfig
factory-default [current]
# ldm add-spconfig ldm-set1
# ldm list-spconfig
factory-default
ldm-set1 [current]
# shutdown -i6 -g0 -y
```

b. **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。**

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、ZFSでミラーリング構成にするコマンドの例を説明します。詳細は、Oracle社のホー

ムページ（https://docs.oracle.com/en/）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の場合

  『Oracle Solaris ZFS 管理ガイド』の「ミラー化されたZFSルートプールを作成する方法（インストール後）」

- Oracle Solaris 11の場合

  『Oracle Solaris 11.xの管理: ZFSファイルシステム』の「ミラー化ルートプールを構成する方法（SPARCまたはx86/VTOC）」

その他の冗長構成ソフトウェアをご利用になる場合は、対象ソフトウェアのマニュアルを参照してください。

zpool statusコマンドを実行して、ルートプールの状態を確認します。

次の例では、デフォルトのルートプール（rpool）にc2t50000393E802CCE2d0s0が割り当てられていることを示しています。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

formatコマンドを実行して、追加可能なディスクを確認します。

次の例では、もう1つのディスクとしてc3t50000393A803B13Ed0が存在することがわかります。

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
```

zpool attachコマンドを実行して、rpoolに2本目のディスクを追加してミラー構成にします。

次の例では、zpool attachコマンドでc3t50000393A803B13Ed0s0を追加して、zpool statusコマンドで同期処理（resilver）の状況を確認しています。status、actionを見ると、同期処理中であることがわかるので、同期処理が完了するまで、定期的にzpool statusコマンドを実行して状況を確認します。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Wed Jan 29 21:35:39 2014
    3.93G scanned out of 70.6G at 71.9M/s, 0h15m to go
    3.90G resilvered, 5.56% done
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       DEGRADED     0     0     0
          mirror-0                  DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0 ONLINE       0     0     0
            c3t50000393A803B13Ed0s0 DEGRADED     0     0     0  (resilvering)
```

同期処理が完了すると以下のように、[state] が「ONLINE」となります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h10m with 0 errors on Wed Jan 29 21:45:42 2014
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          mirror-0                  ONLINE       0     0     0
            c2t50000393E802CCE2d0s0 ONLINE       0     0     0
            c3t50000393A803B13Ed0s0 ONLINE       0     0     0
errors: No known data errors
```

c. **ルートドメインを構成します。**

ここでは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。

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

ldm set-variableコマンドを実行し、Oracle Solaris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Oracle Solaris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。

```
# 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、構成例に従って、ldm set-coreコマンドでCPUコアを16、ldm set-memoryコマンドでメモリを32 GB割り当てています。

```
# ldm set-core 16 root-dom0
# ldm set-memory 32G 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、物理パーティションの動的再構成が可能なCPUコアとメモリの配置の条件を満たしやすくなります。CPUコアとメモリの配置に関する条件の詳細は、「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（vcons）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。

```
# 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、SPARC M10-4S0（BB0）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、PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7が、BB0上の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。

```
# ldm list-io -l
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
（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
/BB0/PCI0                                 PCIE   PCIE1             EMP
[pci@8100/pci@4/pci@0/pci@0]
/BB0/PCI3                                 PCIE   PCIE2             EMP
```

```
[pci@8200/pci@4/pci@0/pci@0]
/BB0/PCI4                                   PCIE   PCIE2             EMP
[pci@8200/pci@4/pci@0/pci@8]
/BB0/PCI7                                   PCIE   PCIE3             EMP
[pci@8300/pci@4/pci@0/pci@0]
/BB0/PCI8                                   PCIE   PCIE3             EMP
[pci@8300/pci@4/pci@0/pci@8]
/BB0/CMUL/NET2                              PCIE   PCIE4    primary  OCC
[pci@8400/pci@4/pci@0/pci@a]
    network@0
    network@0,1
/BB0/PCI1                                   PCIE   PCIE5             EMP
[pci@8500/pci@4/pci@0/pci@8]
/BB0/PCI2                                   PCIE   PCIE5             EMP
[pci@8500/pci@4/pci@0/pci@9]
/BB0/PCI5                                   PCIE   PCIE6             EMP
[pci@8600/pci@4/pci@0/pci@9]
/BB0/PCI6                                   PCIE   PCIE6             EMP
[pci@8600/pci@4/pci@0/pci@11]
/BB0/PCI9                                   PCIE   PCIE7             EMP
[pci@8700/pci@4/pci@0/pci@9]
/BB0/PCI10                                  PCIE   PCIE7             EMP
[pci@8700/pci@4/pci@0/pci@11]
（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.6　SPARC M10-4S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ルートドメインに割り当てるかを決めます。

構成例の場合、上記で確認したBB0上の未割り当てのルートコンプレックスすべて（PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7）になるため、ldm add-ioコマンドを実行してroot-dom0に割り当てます。

次の例は、コマンド実行例です。

```
# ldm add-io PCIE1 root-dom0
# ldm add-io PCIE2 root-dom0
# ldm add-io PCIE3 root-dom0
# ldm add-io PCIE5 root-dom0
# ldm add-io PCIE6 root-dom0
# ldm add-io PCIE7 root-dom0
```

ldm bind-domainコマンドでルートドメインを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、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、PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7がroot-dom0

に割り当てられていることが確認できます。

```
# ldm bind-domain root-dom0
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN      STATUS
----                                        ----   ---      ------      ------
PCIE0                                       BUS    PCIE0    primary     IOV
PCIE1                                       BUS    PCIE1    root-dom0   IOV
PCIE2                                       BUS    PCIE2    root-dom0   IOV
PCIE3                                       BUS    PCIE3    root-dom0   IOV
PCIE4                                       BUS    PCIE4    primary     IOV
PCIE5                                       BUS    PCIE5    root-dom0   IOV
PCIE6                                       BUS    PCIE6    root-dom0   IOV
PCIE7                                       BUS    PCIE7    root-dom0   IOV
PCIE8                                       BUS    PCIE8    primary     IOV
（以下省略）
```

手順11.cに従って、ほかのSPARC M10-4Sのルートドメイン（例の場合、root-dom1）も同様に構成してください。

d. **ルートドメインにOracle Solarisをインストールします。**

ここでは、ルートドメインにOracle Solarisをインストールする手順を説明します。

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

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

ldm set-variableコマンドを実行し、Oracle Solaris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。

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

telnetコマンドを実行して、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。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  7h 7m
root-dom0        active     -t----  5000    32    32G      0.0%  20s
root-dom1        bound      ------  5001    32    32G
# telnet localhost 5000
```

```
....
{0} ok
```

ルートドメインにOracle Solarisをインストールします。

物理パーティションの動的再構成に必要なOracle Solarisの版数や条件などについては、表 1-3および表 1-4を参照してください。

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/en/）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.xシステムのインストール』

手順11.dに従って、ほかのルートドメイン（例の場合、root-dom1）も同様にインストールします。

e. **ルートドメインに仮想I/Oサービスを割り当てます。**

ここでは、ルートドメインに仮想I/Oサービスを設定する手順の例を説明します。

仮想ディスクサービス（vds）には、ルートドメインの物理ディスク全体を仮想ディスクとして貸し出す例を示します。

仮想スイッチネットワークデバイス（vsw）には、Ethernetカードのポートを割り当てる例を示します。

仮想I/Oサービスの詳細については、Oracle社の『Oracle VM Server for SPARC 管理ガイド』を参照してください。

ルートドメインにroot権限でログインします。

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

仮想ディスクサービス（vds）に割り当てるバックエンドデバイス（物理ディスク）を特定するために、formatコマンドを実行して、ディスクとそのデバイスパスを表示します。

cXtXXXXXXXXXXXXXXXXdXの形式の文字列がディスク、「/pci@」で始まる文字列がデバイスパスを示しています。

```
root-dom0# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
```

```
       0. c2t50000393A802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8100/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
       1. c3t50000393D8285226d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393d8285226,0
       2. c4t50000393A804B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@3200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
```

上記の結果で表示されるデバイスパスと、『SPARC M12/M10 システム運用・管理ガイド』の「A.6　SPARC M10-4Sのデバイスパス」を参照してディスクの物理位置を確認して、仮想ディスクサービスに割り当てるバックエンドのディスク（cXtXXXXXXXXXXXXXXXXdX）を決めます。

構成例として、2つのディスク（c3t50000393D8285226d0, c4t50000393A804B13Ed0）をそれぞれ、ゲストドメイン（guest0、guest1）の仮想ディスクのバックエンドとして割り当てるようにします。

次に、仮想スイッチサービス（vsw）に割り当てるネットワークインターフェースを特定するために、dladm show-physコマンドを実行して、PCIeスロットの位置（LOC）に対応するネットワークインターフェース名（LINK）を表示します。

表示結果から、ネットワークインターフェースの位置を確認して、仮想スイッチサービスに割り当てるネットワークインターフェースを決めます。

構成例として、2つのネットワークインターフェース（net1、net2）を、各ゲストドメインの仮想ネットワークに対応する仮想スイッチに割り当てるようにします。

```
root-dom0# dladm show-phys -L
LINK         DEVICE      LOC
net0         igb0        BB#0-PCI#0
net1         igb1        BB#0-PCI#1
net2         igb2        BB#0-PCI#2
net3         igb3        BB#0-PCI#3
...
```

ldm add-vdiskserverコマンドを実行して、ルートドメインに仮想ディスクサービスを追加します。

次の例は、ldm add-vdiskserverコマンドでルートドメイン（root-dom0）に仮想ディスクサービス（vds0）を追加しています。

```
# ldm add-vdiskserver vds0 root-dom0
```

ldm add-vdiskserverdeviceコマンドを実行して、仮想ディスクのバックエンドをルートドメインからエクスポートします。

次の例は、ldm add-vdiskserverdeviceコマンドを実行して、2つバックエンドのディスクをvds0に割り当ててエクスポートしています。

```
# ldm add-vdiskserverdevice /dev/dsk/c3t50000393D8285226d0s2 vol0@
vds0
# ldm add-vdiskserverdevice /dev/dsk/c4t50000393A804B13Ed0s2 vol1@
vds0
```

ldm add-vswitchコマンドを実行して、ルートドメインに仮想スイッチを追加します。

次の例では、ゲストドメインごとに仮想スイッチを追加するため、vsw0、vsw1 を追加して、それぞれに異なる物理ネットワークインターフェース（net1、net2）を割り当てています。

```
# ldm add-vswitch net-dev=net1 vsw0 root-dom0
# ldm add-vswitch net-dev=net2 vsw1 root-dom0
```

同様に、BB 1側のSPARC M10-4SのI/Oを割り当てたルートドメイン（root-dom1）にも仮想I/Oサービスを追加します。

root-dom1に割り当てる仮想ディスクサービス名、仮想スイッチ名の例として、以下のように割り当てます。手順は11.eを参照してください。

仮想ディスクサービス名：vds1（バックエンドのボリューム名はvol10、vol11の2つを割り当てる）

仮想スイッチ名　　：vsw10、vsw11

f. **ゲストドメインを構築します。**

ここでは、ゲストドメインの構築手順を説明します。

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。

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

dm set-variableコマンドを実行し、Oracle Solaris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Oracle Solaris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。

```
# 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、構成例に従って、ldm set-coreコマンドでCPUコアを32、ldm set-memoryコマンドでメモリを64 GB割り当てています。

```
# ldm set-core 32 root-dom0
# ldm set-memory 64G root-dom0
```

注ー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、物理パーティションの動的再構成が可能なCPUコアとメモリの配置の条件を満たしやすくなります。CPUコアとメモリの配置に関する条件の詳細は、「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（vcons）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「5100」を仮想コンソールに割り当てています。

```
# ldm set-vconsole service=vcc0 port=5100 guest0
```

ldm add-vdiskコマンドを実行して、仮想ディスク（vdisk）を割り当てます。

次の例では、構成例に従って2つの仮想ディスク（vdisk0、vdisk10）を割り当てます。それぞれの仮想ディスクのバックエンドは、2つのルートドメイン（root-dom0、root-dom1）に追加したバックエンド（vds0のvol0、vds1のvol10）を指定しています。

```
# ldm add-vdisk vdisk0 vol0@vds0 guest0
# ldm add-vdisk vdisk10 vol10@vds1 guest0
```

ldm add-vnetコマンドを実行して、仮想ネットワークデバイス（vnet）を割り当てます。

次の例では、構成例に従って2の仮想ネットワークデバイス（vnet0、vnet10）を割り当てます。それぞれの仮想ネットワークデバイスに接続する仮想スイッチは、2つのルートドメイン（root-dom0、root-dom1）に追加した仮想スイッチ（vsw0、vsw10）を指定しています。

```
# ldm add-vnet vnet0 vsw0 guest0
# ldm add-vnet vnet10 vsw10 guest0
```

注ー複数の仮想ディスクや仮想ネットワークデバイスを割り当てた場合、ldm list -lコマンドの実行結果から、仮想デバイスに割り当てたIDの値を記録してください。動的に削除したあと、再度追加する際に仮想デバイスパスが変わらないようにするためにこのIDが必要になります。

---

g. **ゲストドメインにOracle Solarisをインストールします。**

ここでは、ゲストドメインにOracle Solarisをインストールする手順を説明します。

ldm bind-domainコマンドを実行してゲストドメインをバインドし、さらにstart-domainコマンドを実行してゲストドメインを起動します。

次の例は、guest0を起動するコマンドの実行例です。

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

ldm set-variableコマンドを実行し、Oracle Solaris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。

```
# ldm set-variable auto-boot¥?=true guest0
```

telnetコマンドを実行して、ゲストドメイン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「5100」であることを確認し、telnetコマンドでlocalhostのポート番号「5100」に接続し、guest0がOpenBoot PROMの状態で止まっていることが確認できます。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  8h 7m
guest0           active     -t----  5100    64    64G      0.0%  20s
root-dom0        active     -n--v-  5000    32    32G      0.0%  43s
root-dom1        active     -n--v-  5001    32    32G      0.0%  20s
guest1           inactive   ------          64    64G
# telnet localhost 5100
....
{0} ok
```

物理パーティションの動的再構成に必要なOracle Solarisの版数や条件などについては、表 1-3および表 1-4を参照してください。

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/en/）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.xシステムのインストール』

手順11.gに従って、ほかのゲストドメイン（例の場合、guest1）も同様にインストールします。

h. **ゲストドメインの仮想I/Oを冗長構成にします。**

ここでは、ゲストドメインguest0に割り当てた2つの仮想ネットワークインターフェース（vnet）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。

ゲストドメインguest0にログインします。

次の例では、ldm list-domainコマンドでguest0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「5100」に接続する実行例を示しています。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  8h 7m
guest0           active     -n----  5100    32    32G      0.0%  20s
guest1           active     -n----  5101    32    32G      0.0%  19s
root-dom0        active     -n--v-  5000    24    24G      0.0%  43s
root-dom1        active     -n--v-  5001    24    24G      0.0%  20s
# telnet localhost 5100
....
guest0 console login: root
Password:
...
guest0#
```

dladmコマンドを実行して、仮想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、net1として仮想ネットワークデバイスが参照できています。

```
guest0# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         0      unknown   vnet1
```

ここでは、Oracle Solaris 11での手順を説明します。ipadm show-ifコマンドを実行して、net0、net1が表示されていないことを確認します。

```
guest0# 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、net1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。

```
guest0# ipadm create-ip net0
guest0# ipadm create-ip net1
guest0# ipadm show-if
IFNAME     CLASS    STATE    ACTIVE OVER
```

```
lo0         loopback ok       yes     --
net0        ip       down     no      --
net1        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、net1をIPMPグループに追加します。

```
guest0# ipadm create-ipmp ipmp0
guest0# ipadm add-ipmp -i net0 -i net1 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。

```
guest0# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/
v4
guest0# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
ipmp0/v4          static   ok           xx.xx.xx.xx/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。

```
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net1        no      ipmp0       is-----   up        disabled  ok
net0        yes     ipmp0       --mbM--   up        disabled  ok
```

ほかのゲストドメイン（例の場合、guest1）も同様に設定を行います。

i. **メモリリソースの割り当て状況を確認します。**

ここでは、各論理ドメインに割り当てられたメモリの配置を確認し、調整する手順を説明します。

物理パーティションの動的再構成でSPARC M10-4Sを切り離せるようにするためには、「2.5.2　運用時の留意点」の「CPUコアとメモリの配置」に書かれているメモリ配置の条件を満たす必要があります。

いっぽう、構成例の場合、あらかじめメモリブロックの移動先となる空きリソースを確保しない構成になっています。そのため、「A.2.3　活性交換手順例（Oracle VM Server for SPARC 3.1.xの場合）」に説明されている手順で、メモリ配置を考慮しやすくするように、メモリブロックの配置を事前に調整しておくことが大切です。

以下に手順の例を説明します。

ldm list-devices -aコマンドを実行して、各論理ドメインに割り当てられているメモリの連続領域（メモリブロック）の状態を確認して、細かいメモリブ

ロックが多数存在しないことを確認してください。

目安としては、SIZEが256 MB～512 GB程度のメモリブロックが多数存在しないことを確認してください。

以下に、ldm list-devices -a memoryコマンドの実行例を示します。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      4G              root-dom0
    0x700100000000      4G              root-dom1
    0x700200000000      8G              root-dom0
    0x700400000000      4G              root-dom1
    0x700500000000      4G              root-dom0
    0x700600000000      8G              root-dom1
    0x720000000000      8G              root-dom0
    0x720200000000      8G              root-dom0
    0x720400000000      8G              root-dom1
    0x720600000000      8G              root-dom1
    0x740000000000      32G             guest0
    0x760000800000      1272M           _sys_
    0x760050000000      31488M          guest1
    0x780000000000      16G             guest0
    0x780400000000      16G             guest1
    0x7a0000000000      16G             guest0
    0x7a0400000000      16G             guest1
    0x7c0000000000      28G             primary
    0x7c0700000000      256M            guest1
    0x7c0710000000      3840M
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      8G              primary
    0x7e0280000000      20G             primary
    0x7e0780000000      2G              guest1
```

上記の場合、root-dom0、root-dom1のメモリブロックが比較的細かく分割されているため、手順11.jに従って、メモリブロックを割り当てなおします。

以下のように、各論理ドメインのメモリブロックが細かく分散していない場合は、再構成をする必要はありません。手順11.kへ進んでください。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom0
    0x720000000000      32G             root-dom1
    0x740000000000      32G             guest0
    0x760000800000      1272M           _sys_
    0x760050000000      2G              primary
    0x7600d0000000      29440M          guest1
    0x780000000000      32G             guest0
    0x7a0000000000      32G             guest1
    0x7c0000000000      32G             primary
```

```
    0x7e0000800000       1272M           _sys_
    0x7e0050000000       512M            _sys_
    0x7e0070000000       256M            _sys_
    0x7e0080000000       3328M           guest1
    0x7e0150000000       4864M
    0x7e0280000000       22G             primary
```

j. **論理ドメインのメモリブロックの配置を調整します。**

手順11.iで確認した結果、論理ドメインのメモリブロックが細かく分割されている場合は、論理ドメインに割り当てられているリソースを再配置します。以下で手順の例を示します。

ldm stop-domainコマンドで各論理ドメインを停止し、ldm unbind-domainコマンドでinactive状態にします。

注ー物理I/Oの割り当ての依存関係（ルートコンプレックスと、PCIeエンドポイントやSR-IOVのVirtual Function）や、仮想I/Oの依存関係がある場合は、仮想I/Oを割り当てたゲストドメイン、PCIeエンドポイントやSR-IOVのVFを割り当てたI/Oドメイン、ルートドメインの順で停止とアンバインドを行ってください。

次の例は、guest0、guest1、root-dom0、root-dom1をinactive状態にするコマンド例を示しています。

```
# ldm stop-domain guest0
LDom guest0 stopped
# ldm stop-domain guest1
LDom guest1 stopped
# ldm stop-domain root-dom0
LDom root-dom0 stopped
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain guest0
# ldm unbind-domain guest1
# ldm unbind-domain root-dom0
# ldm unbind-domain root-dom1
```

ldm list-domainコマンドで、inactive状態になったことを確認します。

次の例は、guest0、guest1、root-dom0、root-dom1がinactive状態になったこと示しています。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  5h 28m
guest0           inactive   ------          64    64G
guest1           inactive   ------          64    64G
root-dom0        inactive   ------          32    32G
root-dom1        inactive   ------          32    32G
```

ldm start-reconfコマンドで制御ドメインを遅延再構成モードに移行します。

```
# 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コマンドの順に、すでに割り当てているコア数とメモリサイズに同じ値を再設定して、制御ドメインのOracle Solarisを再起動します。

次の例は、ldm set-coreコマンドでCPUコア数を32に再設定し、ldm set-memoryコマンドでメモリサイズを56 GBに再設定し、shutdownコマンドでOracle Solarisを再起動しています。

```
# ldm set-core 32 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-memory 56G 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.
------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
```

ldm bind-domainコマンド、ldm start-domainコマンドを実行して、各論理ドメインをバインドして起動します。

バインド、起動の順は、停止を行ったときの逆の順序で行います。

次の例は、root-dom0、root-dom1、guest0、guest1の順でldm bind-domain, ldm start-domainコマンドを実行して起動しています。

```
# ldm bind-domain root-dom0
# ldm bind-domain root-dom1
# ldm bind-domain guest0
# ldm bind-domain guest1
# ldm start-domain root-dom0
LDom root-dom0 started
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm start-domain guest0
LDom guest0 started
# ldm start-domain guest1
LDom guest1 started
```

ldm list-domainコマンドを実行し、すべての論理ドメインがactive状態になったことを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.4%  5h 55m
guest0           active     -n----  5100    64    64G      1.7%  12s
guest1           active     -n----  5101    64    64G       10%  9s
root-dom0        active     -n--v-  5000    32    32G      3.6%  15s
root-dom1        active     -n--v-  5001    32    32G      2.2%  11s
```

ldm list-devicesコマンドを実行し、メモリ割り当ての状態を確認します。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom0
    0x720000000000      32G             root-dom1
    0x740000000000      32G             guest0
    0x760000800000      1272M           _sys_
    0x760050000000      31488M          guest1
    0x780000000000      32G             guest0
    0x7a0000000000      32G             guest1
    0x7c0000000000      28G             primary
    0x7c0700000000      4G
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      28G             primary
    0x7e0780000000      1280M           guest1
    0x7e07d0000000      768M
```

12. **構築した論理ドメイン構成情報をXSCFに保存します。**

    ldm set-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、すでにある構成情報と同じ名前で再度保存しています。

    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。

```
# 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。

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

ldm add-spconfigコマンドを実行して、構成情報を保存しなおします。

```
# 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。

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

13. **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。

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。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  6h 9m
guest0           active     -n----  5100    64    64G      0.0%  15m
guest1           active     -n----  5101    64    64G      0.0%  15m
root-dom0        active     -n--v-  5000    32    32G      0.0%  15m
root-dom1        active     -n--v-  5001    32    32G      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。

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

A.2.3 活性交換手順例（Oracle VM Server for SPARC 3.1.xの場合）

ここでは、図 A-2に記載された2BB構成システムに対して、PPAR DRを使用して、BB#01のSPARC M10-4Sを活性交換する手順例を記載します。

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

1. **マスタXSCFにログインします。**
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の場合は、マスタXSCFへログインしなおしてください。

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

2. **showhardconfコマンドを実行し、交換するSPARC M10-4SのXSCFの [Status] が「Normal」であることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Locked;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
           + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
            CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
                + Freq:3.000 GHz; Type:0x10;
                + Core:16; Strand:2;
      :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
      :
```

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

3. **showbbstatusコマンドを実行し、交換するSPARC M10-4SのXSCFがマスタでないことを確認します。**

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

交換するSPARC M10-4SがマスタXSCFの場合、switchscfコマンドを実行し、XSCFを切り替えます。

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

注ーXSCFの切り替え、およびXSCFの再起動が完了したことを確認したあとに、PSBの切り離し操作を行ってください。

---

4. **console**コマンドを実行し、制御ドメインのコンソールに接続し、ログインします。

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

5. 論理ドメインの稼働状況やリソース使用状況を確認します。

   a. **ldm list-domain**コマンドを実行し、論理ドメインの稼働状況を確認します。

   論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、[FLAGS] の文字列の左から2文字目が次の意味を持ちます。

   「n」: Oracle Solarisが動作中

   「t」: OpenBoot PROMの状態

   「-」: その他の状態（ [STATE] が「active」以外のときも含む）

   この例では、制御ドメイン、2つのルートドメイン、2つのゲストドメインが稼働しています。

すべてのドメインが「active」でOracle Solarisが動作状態、または「inactive」状態であることを確認します。OpenBoot PROM状態またはbound状態のドメインが存在すると、物理パーティションの動的再構成に失敗することがあります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  1h 33m
guest0           active     -n----  5100    64    64G      3.1%  2s
guest1           active     -n----  5101    64    64G      1.6%  18m
root-dom0        active     -n--v-  5000    32    32G      3.1%  17m
root-dom1        active     -n--v-  5001    32    32G      3.1%  17m
```

   b. **-a**オプションを指定して**ldm list-devices**コマンドを実行し、リソースの使用状況を確認します。

   次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID      %FREE    CPUSET
    0       0        (0, 1)
    4       0        (8, 9)
    8       0        (16, 17)
（中略）
    944     0       (1888, 1889)
    948     0       (1896, 1897)
    952     0       (1904, 1905)
    956     0       (1912, 1913)
VCPU
    PID     %FREE    PM
    0       0        no
    1       0        no
    8       0        no
```

```
    9        0         no
（中略）
    1904     0        no
    1905     0        no
    1912     0        no
    1913     0        no
（以下略）
```

6. **制御ドメインのシステムボリュームやI/Oデバイスの冗長構成を切り離します。**
   BB#01を切り離せるようにするために、制御ドメインで使用している、交換するSPARC M10-4SのI/Oデバイスを切り離します。冗長構成を解除する手順の詳細は、各冗長化ソフトウェアのドキュメントを参照してください。

   a. **制御ドメインのシステムボリュームの冗長化を解除します。**

      次の例では制御ドメインのシステムボリュームのZFSミラー機能を解除する手順を示します。

      制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

      zpool detachコマンドを実行し、ディスクをミラー構成から切り離します。

```
# zpool detach rpool c3t50000393A803B13Ed0
```

      zpool statusコマンドで、ミラー構成が解除されたことを確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

      ほかにもBB#01上のデバイスを使用している場合、冗長構成を解除したり、デバイスの利用を停止したりしてください。冗長構成の解除方法やデバイスの

利用停止方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

b. **制御ドメインのI/O構成を削除します。**

制御ドメインに割り当てられている物理I/Oデバイスのうち、BB#01のルートコンプレックスを遅延再構成で削除します。

はじめに、制御ドメインを遅延再構成モードに移行します。

```
# 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 list-ioコマンドを実行し、primaryに割り当てられているルートコンプレックスを確認します。

次の例では、BB1のデバイスを持つルートコンプレックスがPCIE8とPCIE12であることがわかります。

```
# ldm list-io | grep primary
PCIE0                                     BUS    PCIE0    primary  IOV
PCIE4                                     BUS    PCIE4    primary  IOV
PCIE8                                     BUS    PCIE8    primary  IOV
PCIE12                                    BUS    PCIE12   primary  IOV
/BB0/CMUL/NET0                            PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                          PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                            PCIE   PCIE4    primary  OCC
/BB1/CMUL/NET0                            PCIE   PCIE8    primary  OCC
/BB1/CMUL/SASHBA                          PCIE   PCIE8    primary  OCC
/BB1/CMUL/NET2                            PCIE   PCIE12   primary  OCC
```

ldm remove-ioコマンドを実行し、primaryからPCIE8とPCIE12を削除して、Oracle Solarisを再起動します。

```
# ldm remove-io PCIE8 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 remove-io PCIE12 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.
------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
....
```

Oracle Solarisが起動されたら、ldm list-ioコマンドを実行し、制御ドメインからBB#01のルートコンプレックスが削除されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE4                                       BUS    PCIE4    primary  IOV
/BB0/CMUL/NET0                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                            PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                              PCIE   PCIE4    primary  OCC
```

c. **ゲストドメインに割り当てている仮想I/Oデバイスの冗長構成を解除します。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）を事前にシャットダウンするために、各ゲストドメインにログインして、root-dom1からの仮想I/Oデバイスの冗長構成を解除します。

冗長構成ソフトウェアの使用方法については、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。

次の例では、仮想ネットワークデバイス（vnet1）をIPMPの構成から解除する例を示しています。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  4h 17m
guest0           active     -n----  5100    64    64G      0.0%  1h 13m
guest1           active     -n----  5101    64    64G      0.0%  1h 4m
root-dom0        active     -n--v-  5000    32    32G      0.0%  1h 47m
root-dom1        active     -n--v-  5001    32    32G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         0      unknown   vnet1
```

ipmpstat -iコマンドを実行し、IPMPを構成しているネットワークインターフェースの構成情報を確認します。

```
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       is-----   up        disabled  ok
```

if_mpadm -d コマンドを実行し、IPMPのグループからnet1を切り離して、ipmpstat -i コマンドを実行し、切り離されたことを確認します。次の例の場合、STATEがofflineになったことを確認します。

```
guest0# if_mpadm -d net1
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       -s---d-   up        disabled  offline
```

ipadm delete-ipコマンドを実行して、net1を削除します。

```
guest0# ipadm delete-ip net1
```

ゲストドメイン（guest1）についても、同様の切り離しを行います。

d. **停止するルートドメインから割り当てている仮想I/Oデバイスを削除します。**

ldm remove-vdiskコマンド、ldm remove-vnetコマンドを実行して、次の手順で停止するルートドメインから割り当てている仮想ディスク（vdisk）と仮想ネットワークデバイス（vnet）を削除します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている仮想ディスク（vdisk10）と仮想ネットワークデバイス（vnet10）を削除するコマンドの実行例を示しています。

```
# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0
```

ゲストドメイン（guest1）についても、同様の削除を行います。

7. **I/Oデバイスのリソース使用状況を確認し、交換するSPARC M10-4SのI/Oデバイスをすべて解除します。**

a. **切り離すSPARC M10-4Sのルートコンプレックスを割り当てている論理ドメインを確認します。**

ldm list-ioコマンドを実行し、BB#01のルートコンプレックスを割り当てている論理ドメインを確認します。

次の例では、root-dom1だけが「/BB1/」で始まるPCIeエンドポイントを所有しています。そして、そのPCIeエンドポイントのルートコンプレックス（BUS）である、PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15がroot-dom1に割り当てられていることがわかります。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
```

```
PCIE10                                    BUS    PCIE10   root-dom1  IOV
PCIE11                                    BUS    PCIE11   root-dom1  IOV
PCIE12                                    BUS    PCIE12
PCIE13                                    BUS    PCIE13   root-dom1  IOV
PCIE14                                    BUS    PCIE14   root-dom1  IOV
PCIE15                                    BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                            PCIE   PCIE8             UNK
/BB1/CMUL/SASHBA                          PCIE   PCIE8             UNK
/BB1/PCI0                                 PCIE   PCIE9    root-dom1OCC
/BB1/PCI3                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI4                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI7                                 PCIE   PCIE11   root-dom1OCC
/BB1/PCI8                                 PCIE   PCIE11   root-dom1OCC
/BB1/CMUL/NET2                            PCIE   PCIE12            UNK
/BB1/PCI1                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI2                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI5                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI6                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI9                                 PCIE   PCIE15   root-dom1OCC
/BB1/PCI10                                PCIE   PCIE15   root-dom1OCC
```

b. **切り離す SPARC M10-4S のルートコンプレックスを割り当てているルートドメインを停止して切り離します。**

次の例では、ldm stop-domainコマンド、ldm unbind-domainコマンドを実行し、ルートドメイン（root-dom1）を切り離し、inactive状態になったことが確認できます。

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  4h 59m
guest0           active     -n----  5100    64    64G      0.0%  1h 55m
guest1           active     -n----  5101    64    64G      0.0%  1h 46m
root-dom0        active     -n--v-  5000    32    32G      0.0%  2h 29m
root-dom1        inactive   ------          32    32G
```

c. **交換する SPARC M10-4S の I/O デバイスがすべて解除されたことを確認します。**

ldm list-ioコマンドを実行し、I/Oデバイスが切り離されたことを確認します。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN     STATUS
----                                      ----   ---      ------     ------
PCIE0                                     BUS    PCIE0    primary    IOV
PCIE1                                     BUS    PCIE1    root-dom0  IOV
PCIE2                                     BUS    PCIE2    root-dom0  IOV
PCIE3                                     BUS    PCIE3    root-dom0  IOV
PCIE4                                     BUS    PCIE4    primary    IOV
PCIE5                                     BUS    PCIE5    root-dom0  IOV
PCIE6                                     BUS    PCIE6    root-dom0  IOV
```

```
PCIE7                                          BUS     PCIE7     root-dom0   IOV
PCIE8                                          BUS     PCIE8
PCIE9                                          BUS     PCIE9
PCIE10                                         BUS     PCIE10
PCIE11                                         BUS     PCIE11
PCIE12                                         BUS     PCIE12
PCIE13                                         BUS     PCIE13
PCIE14                                         BUS     PCIE14
PCIE15                                         BUS     PCIE15
（以下省略）
```

8. **論理ドメインに割り当てられているCPUコアとメモリリソースを手動で削減します。**

   ここでは、-m unbind=resourceオプションを指定する代わりに、論理ドメインに割り当てられているCPUコアとメモリリソースを手動で削減して、「2.5.2　運用時の留意点」の「CPUコアとメモリの配置」の条件を満たすようにする手順を説明します。

   deleteboardコマンドがエラーで失敗したときなどに、この手順を実施することで、deleteboardコマンドが実施できる場合があります。

   deleteboardコマンドの-m unbind=resourceオプションを指定する場合は、手順9に進んでください。

注－Oracle VM Server for SPARC 3.2よりも古いバージョンのOracle VM Server for SPARCでは、deleteboard -c disconnectコマンドの-m unbind=resourceオプションが未サポートのため、この手順を実施してから、deleteboardコマンドで-m unbind=resourceを指定しないでSPARC M10-4Sの切り離しを行ってください。

---

a. **CPUコアの数を確認して削除します。**

   SPARC M10-4Sの切り離しによって、利用できるCPUコア数が減るため、次の手順で論理ドメインに割り当てているCPUコア数を削減しておきます。

   i. **SPARC M10-4Sを切り離したあとのCPUコア数を確認します。**

      XSCFから、showpparinfoコマンドを実行し、切り離される予定のSPARC M10-4Sを除いたCPUコア数を確認します。

      次の例では、物理システムボード(PSB)番号01-0のSPARC M10-4Sを切り離す場合、他PSB番号00-0のCoresの合計値を計算することになるため、16+16+16+16＝64コアとなります。

```
XSCF> showpparinfo -p 0
PPAR#00 Information:
--------------------
 CPU(s)                        :        8
 CPU Cores                     :      128
 CPU Threads                   :      256
 Memory size (GB)              :      256
 CoD Assigned (Cores)          :      256
CPU(s):
-------
 PID   PSB    CPU#   Cores   Threads
 00    00-0   0         16        32
```

```
 00    00-0  1         16        32
 00    00-0  2         16        32
 00    00-0  3         16        32
 00    01-0  0         16        32
 00    01-0  1         16        32
 00    01-0  2         16        32
 00    01-0  3         16        32
（以下省略）
```

ii. **各論理ドメインに割り当てられているCPUコア数の総数を確認します。**

ldm list-devices -a core を実行し、%FREEの列が100以外になっている行の数が、論理ドメインに割り当てられているCPUコア数の総数になります。

次の例では、ldm list-devices -a coreコマンドの実行例と、-pオプションを使って確認する実行例を示します。この結果、112コアを論理ドメイン全体にバインドしていることがわかります。

```
# ldm list-devices -a core
CORE
    ID      %FREE   CPUSET
    0       0      (0, 1)
    4       0      (8, 9)
    8       0      (16, 17)
    12      0      (24, 25)
（以下省略）
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
     112
```

iii. **SPARC M10-4Sの切り離しによって不足するコア数を計算します。**

以下の計算式から、SPARC M10-4Sの切り離し後に不足するCPUコア数を計算します。

不足するCPUコア数＝論理ドメインで使用するコア数（手順ii）－切り離し後の物理コア数（手順i）

手順i、iiの例の場合、112コア（使用中）－64コア（残り）=48コア不足することがわかります。

iv. **CPUコアが不足する場合、削除対象の論理ドメインを検討します。**

手順iiiの結果で、CPUコアが不足する場合、論理ドメインからCPUコアを削除しておく必要があります。

ldm list-domainコマンドを実行し、activeまたはbound状態の各論理ドメインに割り当てられているCPUコア数を確認し、CPUコアを削除する論理ドメインを確認します。

次の例では、primaryに32コア（64 vcpu）、guest0に32コア（64 vcpu）、guest1に32コア（64 vcpu）、root-dom0に16コア（32 vcpu）、root-dom1に16コア（32 vcpu）割り当てられていることがわかります。例では、48コアを削除する必要があるため、primary、guest0、guest1から16コアずつ削除することにします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.1%  18h 17m
guest0           active     -n----  5100    64    64G      0.0%  15h 13m
guest1           active     -n----  5101    64    64G      0.0%  15h 4m
root-dom0        active     -n--v-  5000    32    32G      0.0%  15h 47m
root-dom1        inactive   ------          32    32G
```

v. **ldm remove-coreコマンドを実行し、対象の論理ドメインからCPUコアを削除します。**

次の例は、primary、guest0、guest1からそれぞれ16コアずつ減らして、実際に減ったかどうか確認するコマンドの実行例を示しています。

```
# ldm remove-core 16 primary
# ldm remove-core 16 guest0
# ldm remove-core 16 guest1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    56G      0.0%  18h 19m
guest0           active     -n----  5100    32    64G      0.0%  15h 15m
guest1           active     -n----  5101    32    64G      0.0%  15h 5m
root-dom0        active     -n--v-  5000    32    32G      0.0%  15h 49m
root-dom1        inactive   ------  5001    32    32G
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
      64
```

b. **メモリリソースを確認して削除します。**

SPARC M10-4Sの切り離しによって、利用できるメモリ領域が減るため、以下の手順で論理ドメインに割り当てられているメモリリソースを削除しておきます。

i. **メモリの連続領域（メモリブロック）の使用状況を確認します。**

prtdiagコマンドとldm list-devices -a memoryコマンドを実行し、各論理ドメインに割り当てられているメモリブロックと、未割り当てのメモリブロックが、どのSPARC M10-4Sに割り当てられているかを確認します。

最初に、prtdiagコマンドを実行し、メモリの物理アドレスとSPARC M10-4Sの対応を確認します。

```
# prtdiag
(略)
======================== Physical Memory Configuration ========================
Segment Table:
----------------------------------------------------------------
Base            Segment  Interleave   Bank     Contains
Address         Size     Factor       Size     Modules
----------------------------------------------------------------
0x7e0000000000 32 GB    4            8 GB     /BB0/CMUL/CMP0/MEM00A
(略)
0x7c0000000000 32 GB    4            8 GB     /BB0/CMUL/CMP1/MEM10A
(略)
```

```
0x7a0000000000 32 GB    4            8 GB     /BB0/CMUU/CMP0/MEM00A
（略）
0x780000000000 32 GB    4            8 GB     /BB0/CMUU/CMP1/MEM10A
（略）
0x760000000000 32 GB    4            8 GB     /BB1/CMUL/CMP0/MEM00A
（略）
0x740000000000 32 GB    4            8 GB     /BB1/CMUL/CMP1/MEM10A
（略）
0x720000000000 32 GB    4            8 GB     /BB1/CMUU/CMP0/MEM00A
（略）
0x700000000000 32 GB    4            8 GB     /BB1/CMUU/CMP1/MEM10A
（以下省略）
```

この例の結果を、メモリの物理アドレスを小さい順に並べ直して、物理アドレスとSPARC M10-4Sの対応を表にすると以下のようになります。

表 A-2 物理アドレスとSPARC M10-4Sの対応例

Base Address（物理アドレス）	ビルディングブロック構成中のSPARC M10-4S
0x700000000000～	BB1
0x720000000000～	BB1
0x740000000000～	BB1
0x760000000000～	BB1
0x780000000000～	BB0
0x7a0000000000～	BB0
0x7c0000000000～	BB0
0x7e0000000000～	BB0

次に、ldm list-devices -a memory コマンドを実行し、論理ドメインに割り当てられているメモリの連続領域（以降、メモリブロックと呼びます）と、未割り当てのメモリブロックを確認します。

次の例は、ldm list-devices -a memoryコマンドの実行結果の例です。

各パラメーターの意味は以下のとおりです。

PA　: メモリブロックの先頭の物理アドレス

SIZE : メモリブロックのサイズ

BOUND: メモリブロックを割り当てている論理ドメイン名。空欄は未割り当ての領域、_sys_は論理ドメインに割り当てられない制御用の領域です。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom0
    0x720000000000      32G
    0x740000000000      32G             guest0
```

```
0x760000800000        1272M           _sys_
0x760050000000        31488M          guest1
0x780000000000        32G             guest0
0x7a0000000000        32G             guest1
0x7c0000000000        28G             primary
0x7c0700000000        4G
0x7e0000800000        1272M           _sys_
0x7e0050000000        512M            _sys_
0x7e0070000000        256M            _sys_
0x7e0080000000        28G             primary
0x7e0780000000        1280M           guest1
0x7e07d0000000        768M
```

上記の結果と、prtdiagコマンドで確認した物理位置を合わせると、以下のようなメモリブロックの使用状況であることがわかります。

表 A-3　メモリブロックの使用状況の例

SPARC M10-4S	物理アドレス	サイズ	論理ドメイン
BB1（交換対象）	0x700000000000	32 GB	root-dom0
	0x720000000000	32 GB	未割り当て
	0x740000000000	32 GB	guest0
	0x760050000000	31,488 MB	guest1
BB0	0x780000000000	32 GB	guest0
	0x7a0000000000	32 GB	guest1
	0x7c0000000000	28 GB	primary
	0x7c0700000000	4 GB	未割り当て
	0x7e0080000000	28 GB	primary
	0x7e0780000000	1,280 MB	guest1
	0x7e07d0000000	768 MB	未割り当て

ii. **移動元のメモリブロックのサイズと個数を確認します。**

メモリブロックの使用状況の確認結果を参照し、交換対象のSPARC M10-4S上で割り当てられているメモリブロック（以降、移動元のメモリブロックと呼びます）を確認します。

「表 A-3　メモリブロックの使用状況の例」の場合、BB1側で論理ドメインに割り当てられているメモリブロックとして、32 GB×2個、31,488 MB×1個のメモリブロックが使用中であることがわかります。

iii. **メモリを削除する論理ドメインとサイズを決定します。**

次に、各論理ドメインに割り当てているメモリブロックの配置を確認し、メモリブロック単位で削除したり、メモリブロックのサイズを減らしたりすることで、移動元のメモリブロックを、切り離し対象ではないSPARC M10-4Sの未割り当てのメモリブロック（以降、移動先のメモリブロックと呼びます）に移動できるようにします。

その結果として、最終的にどの論理ドメインからどのくらいのメモリを削

除すればいいかを決定します。

方法は以下があります。

- 移動元のメモリブロックをまるごと削除することで、移動する必要のあるメモリブロックを減らす。
- 切り離し対象ではないSPARC M10-4S上の、論理ドメインに割り当てられているメモリブロックをまるごと削除して空き領域を増やすことで、移動先を増やす。
- 移動元のメモリブロックのサイズを減らして、移動先の空き領域に入るサイズにする。
- 移動元のメモリブロックのサイズを減らして、かつ移動先で使用中のメモリブロックのサイズを減らすことで、移動先の空きメモリブロックを増やして、移動できるようにする。

注ーサイズを減らしたあとの空きメモリブロックは、連続領域にならない（フラグメント化する）ことに注意してください。小さいメモリブロックを複数削除して空き領域を増やしても、空き領域が連続領域ではないため、移動元のメモリブロックの連続領域が大きい場合は、移動できません。そのような場合は、移動元のメモリブロックも削除してサイズを調整してください。

注ー検討をする際、なるべく存在しているメモリブロックと同じサイズで削除できるものを削除対象として選択することで、削除後に連続領域が細分化（フラグメント化）する可能性を減らすことができます。それにより、メモリブロックの移動処理が成功する可能性が高まります。

注ーメモリを削除しすぎると、論理ドメイン内のメモリを圧迫することになり、Oracle Solarisがハングアップするなどの問題が発生する可能性があります。vmstatコマンドで"free"のサイズを目安にして、削除しすぎないよう注意が必要です。

「表 A-3　メモリブロックの使用状況の例」を元に削除プランを検討します。

検討の結果、例として「表 A-4　メモリブロックの削除プラン」で示すとおり、root-dom0から4 GB削除、guest0から32 GB削除、guest1から31,488 MB削除、primaryから28 GB削除するプランを立てます。

表 A-4　メモリブロックの削除プラン

SPARC M10-4S	サイズ	論理ドメイン	削除プラン
BB1（交換対象）	32 GB	root-dom0	この領域を4 GB減らして28 GBにして、BB0側のprimaryの28 GBを削除して、移動させる
	32 GB	未割り当て	ー
	32 GB	guest0	BB0側に32 GBのguest0のメモリがあるので、削除
	31,488 MB	guest1	BB0側に32 GBのguest1のメモリがあるので、削除
BB0	32 GB	guest0	残す

表 A-4　メモリブロックの削除プラン (続き)

SPARC M10-4S	サイズ	論理ドメイン	削除プラン
	32 GB	guest1	残す
	28 GB	primary	残す
	4 GB	未割り当て	ー
	28 GB	primary	root-dom0の28 GBを移動させるため、削除
	1,280 MB	guest1	残す
	768 MB	未割り当て	ー

iv. **論理ドメインからメモリを手動で削除します。**

手順iiiで決定した、メモリの削除プランに従って、ldm remove-memoryコマンドを使って論理ドメインからメモリを削除します。

次の例は、「表 A-4　メモリブロックの削除プラン」に従った、メモリを削除するコマンドの実行例です。

```
# ldm remove-memory 4G root-dom0
# ldm remove-memory 32G guest0
# ldm remove-memory 31488M guest1
# ldm remove-memory 28G primary
```

v. **削除されたメモリブロックの状態を確認します。**

ldm list-devices -a memoryコマンドを実行し、削除された結果を参照して、移動できる配置になったかどうかを確認します。移動できないメモリブロックが残った場合、この結果を元に、さらに削除するメモリブロックを検討して、削除します。

次の例の場合、BB1側で割り当てられている大きいサイズと、BB0側の空き領域の大きいサイズを並べて比較することで、移動できるかどうかが確認しやすくなります。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
(BB1側)
    0x700000000000      256M            root-dom0   →4GBの移動先を分割して移動可能
    0x700010000000      4G
    0x700110000000      28416M          root-dom0   →0x780000000000(28GB)へ移
                                                      動可能
    0x720000000000      32G
    0x740000000000      256M            guest0      →4GBの移動先を分割して移動可能
    0x740010000000      4G
    0x740110000000      28416M          guest0      →移動先がないため、再度削除が必
                                                      要(*)
    0x760000800000      1272M           _sys_
    0x760050000000      256M            guest1      →4 GBの領域を分割して移動可能
```

```
    0x760060000000       1792M
    0x7600d0000000       29440M          guest1     →0x7a0000000000(29 GB)へ移
                                                     動可能
(BB0側)
    0x780000000000       28G                        ←root-dom0
                                                     (0x700110000000)から移動可
                                                     能
    0x780700000000       4G              guest0
    0x7a0000000000       29G                        ←guest1(0x7600d0000000)か
                                                     ら移動可能
    0x7a0740000000       3G              guest1
    0x7c0000000000       256M            primary
    0x7c0010000000       4G                         ←root-dom0、guest0、guest1
                                                     の256 MBから移動可能
    0x7c0110000000       24320M          primary
    0x7c0700000000       4G
    0x7e0000800000       1272M           _sys_
    0x7e0050000000       512M            _sys_
    0x7e0070000000       256M            _sys_
    0x7e0080000000       24G                        ←guest0(0x740110000000)が
                                                     移動するには不足(*)
    0x7e0680000000       4G              primary
    0x7e0780000000       1280M           guest1
    0x7e07d0000000       768M
```

上記の例の（*）部分が、移動先として24 GB（24,576 MB）しか空いていないため、移動元（BB1側）の28,416 MBの領域（guest0）を3,840 MB削除して、再度確認します。下記の例で、すべてのメモリブロックが移動可能な状態になったことがわかります。

```
# ldm remove-memory 3840M guest0
# ldm list-devices -a memory
MEMORY
    PA                   SIZE            BOUND
(BB1側)
    0x700000000000       256M            root-dom0
    0x700010000000       4G
    0x700110000000       28416M          root-dom0
    0x720000000000       32G
    0x740000000000       256M            guest0
    0x740010000000       7936M
    0x740200000000       24G             guest0  →0x7e0080000000(24G)へ移動可能
    0x760000800000       1272M           _sys_
    0x760050000000       256M            guest1
    0x760060000000       1792M
    0x7600d0000000       29440M          guest1
(BB0側)
    0x780000000000       28G
    0x780700000000       4G              guest0
    0x7a0000000000       29G
    0x7a0740000000       3G              guest1
    0x7c0000000000       256M            primary
    0x7c0010000000       4G
    0x7c0110000000       24320M          primary
```

```
    0x7c0700000000       4G
    0x7e0000800000       1272M          _sys_
    0x7e0050000000       512M           _sys_
    0x7e0070000000       256M           _sys_
    0x7e0080000000       24G                    ←guest0(0x740200000000)から
                                                移動可能
    0x7e0680000000       4G             primary
    0x7e0780000000       1280M          guest1
    0x7e07d0000000       768M
```

9. **SPARC M10-4S**に該当するシステムボード（**PSB<BB>**）を物理パーティションから切り離します。

   a. **deleteboard -c disconnect**コマンドを実行し、**PSB**を物理パーティションから切り離します。

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

   b. **showresult**コマンドを実行し、直前に実行した**deleteboard**コマンドの終了ステータスを確認します。

   終了値が0の場合、deleteboardコマンドは正常終了です。

   終了値が0以外の場合、およびdeleteboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.2 deleteboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

   c. **showboardsコマンドを実行し、PSBの状態を確認します。**

   交換するSPARC M10-4SのPSBが「Assigned」状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

10. **replacefru**コマンドを実行し、**SPARC M10-4S**を交換します。

```
XSCF> replacefru
```

注－replacefruコマンドを使用したSPARC M10-4Sの交換の詳細は、『SPARCM10-4/M10-4Sサービスマニュアル』の「5.8　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムから切り離す」および「6.2　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムに組み込む」を参照してください。

---

11. **交換するSPARC M10-4Sのシステムボード（PSB<BB>）を物理パーティションに組み込みます。**

    a. **showboardsコマンドを実行し、PSBの状態を確認します。**

    交換するSPARC M10-4SのPSBがAssigned状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

    b. **addboard -c configureコマンドを実行し、PSBを物理パーティションに組み込みます。**

    論理ドメイン構成を元の構成に戻すため、-m bind=resourceオプションを指定して、addboard -c configureコマンドを実行します。

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

注－addboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

---

    c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

    終了値が0の場合、addboardコマンドは正常終了です。

    終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換したSPARC M10-4SのPSBが正常に組み込まれたことで、 [Conn] 欄と [Conf] 欄の表示がともに「y」と表示され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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

12. **論理ドメインの稼働状況を確認します。**

a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

b. **ldm list-domainコマンドを実行し、PSBの追加後に論理ドメインの稼働状況に変化がないことを確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

```
# ldm list-domain
NAME             STATE      FLAGS   CONS   VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART   32    28G      64%   2h 54m
guest0           active     -n----  5100   32    28928M   42%   2h 54m
guest1           active     -n----  5101   32    34048M   11%   2h 54m
root-dom0        active     -n--v-  5000   32    28G      10%   2h 54m
root-dom1        inactive   ------         32    32G
```

13. **削除されたリソースを復元します。**

deleteboardコマンドによって論理ドメインに割り当てられたリソースが削減されている場合、ldm add-core, ldm add-memoryコマンドで追加して、リソースを復元します。

次の例では、削減されたCPUコアとメモリを追加して、SPARC M10-4Sを交換する前のリソースに戻しています。

```
# ldm add-core 16 primary
# ldm add-core 16 guest0
# ldm add-core 16 guest1
```

```
# ldm add-memory 28G primary
# ldm add-memory 4G root-dom0
# ldm add-memory 36608M guest0
# ldm add-memory 31488M guest1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  4h 59m
guest0           active     -n----  5100    64    64G      0.0%  1h 55m
guest1           active     -n----  5101    64    64G      0.0%  1h 46m
root-dom0        active     -n--v-  5000    32    32G      0.0%  2h 29m
root-dom1        inactive   ------          32    32G
```

14. **I/Oデバイスの使用を再開します。**
    a. **ルートコンプレックスを再割り当てします。**

    ldm bind-domain, ldm start-domain コマンドを実行し、交換したSPARC M10-4Sのルートコンプレックスを割り当てていたルートドメインを、unbind状態から起動します。

    次の例では、unbind状態にしたルートドメイン（root-dom1）を起動して、起動されたことを確認しています。

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  3h 8m
guest0           active     -n----  5100    64    64G      0.0%  3h 8m
guest1           active     -n----  5101    64    64G      0.0%  3h 8m
root-dom0        active     -n--v-  5000    32    32G      0.0%  3h 8m
root-dom1        active     -n--v-  5001    32    32G      7.3%  8s
```

    ldm list-ioコマンドを実行し、起動したルートドメインに物理I/Oデバイスが割り当てられていることを確認します。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN     STATUS
----                                      ----   ---      ------     ------
PCIE0                                     BUS    PCIE0    primary    IOV
PCIE1                                     BUS    PCIE1    root-dom0  IOV
PCIE2                                     BUS    PCIE2    root-dom0  IOV
PCIE3                                     BUS    PCIE3    root-dom0  IOV
PCIE4                                     BUS    PCIE4    primary    IOV
PCIE5                                     BUS    PCIE5    root-dom0  IOV
PCIE6                                     BUS    PCIE6    root-dom0  IOV
PCIE7                                     BUS    PCIE7    root-dom0  IOV
PCIE8                                     BUS    PCIE8
PCIE9                                     BUS    PCIE9    root-dom1  IOV
PCIE10                                    BUS    PCIE10   root-dom1  IOV
PCIE11                                    BUS    PCIE11   root-dom1  IOV
PCIE12                                    BUS    PCIE12
PCIE13                                    BUS    PCIE13   root-dom1  IOV
```

```
PCIE14                                    BUS    PCIE14   root-dom1  IOV
PCIE15                                    BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                            PCIE   PCIE8             UNK
/BB1/CMUL/SASHBA                          PCIE   PCIE8             UNK
/BB1/PCI0                                 PCIE   PCIE9    root-dom1OCC
/BB1/PCI3                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI4                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI7                                 PCIE   PCIE11   root-dom1OCC
/BB1/PCI8                                 PCIE   PCIE11   root-dom1OCC
/BB1/CMUL/NET2                            PCIE   PCIE12             UNK
/BB1/PCI1                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI2                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI5                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI6                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI9                                 PCIE   PCIE15   root-dom1OCC
/BB1/PCI10                                PCIE   PCIE15   root-dom1OCC
```

b. **ルートドメインからの仮想I/Oデバイスをゲストドメインに追加します。**

ldm add-vdiskコマンド、ldm add-vnetコマンドを実行して、起動したルートドメインの仮想I/Oサービスに対応した、仮想ディスク（vdisk）と、仮想ネットワークデバイス（vnet）を、各ゲストドメインに追加します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている、仮想ディスク（vdisk10）と、仮想ネットワークデバイス（vnet10）を追加するコマンドの実行例を示しています。

ゲストドメイン（guest1）についても、同様の追加を行います。

```
# ldm add-vdisk id=1 vdisk10 vol10@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

注－仮想I/Oデバイスを再度追加する場合、あらかじめ割り当てられたIDを指定する必要があります。IDは、仮想I/Oデバイスを削除する前の状態でldm list -lコマンドの実行結果から確認できます。

---

c. **ゲストドメインに割り当てている仮想I/Oデバイスを冗長構成に組み込みます。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）が起動されると、各ゲストドメインに対する仮想I/Oデバイスのサービスも起動されます。

各ゲストドメインにログインして、事前に解除しておいたroot-dom1からの仮想I/Oデバイスを冗長構成に組み込みます。冗長構成ソフトウェアの使用方法については、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。

ここでは、仮想ネットワークデバイス（vnet1）をIPMPの構成に組み込む例を説明します。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

まず、ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  4h 17m
guest0           active     -n----  5100    64    64G      0.0%  1h 13m
guest1           active     -n----  5101    64    64G      0.0%  1h 4m
root-dom0        active     -n----  5000    32    32G      0.0%  1h 47m
root-dom1        active     -n----  5001    32    32G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            up         0      unknown   vnet0
net1              Ethernet            up         0      unknown   vnet1
```

ipadm create-ipコマンド、ipadm set-ifpropコマンド、ipadm add-ipmpコマンドを実行し、net1をipmp0の待機デバイスとして登録します。

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

ipmpstat -iコマンドを実行し、IPMPを構成しているネットワークインターフェースのSTATEがokになっていることを確認します。

ゲストドメイン（guest1）についても、同様の対応を行います。

```
guest0# ipmpstat -i
INTERFACE ACTIVE GROUP FLAGS LINK PROBE STATE
net0 yes ipmp0 -smbM-- up disabled ok
net1 no ipmp0 -s---d- up disabled ok
```

15. **制御ドメインのシステムボリュームやI/Oデバイスを冗長構成に戻します。**
    a. **制御ドメインのルートコンプレックス構成を追加します。**

    制御ドメインから事前に削除した、BB#01のルートコンプレックスを遅延再構成で追加します。

    まず、制御ドメインを遅延再構成モードに移行します。

```
# 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 list-ioコマンドを実行し、割り当てられていないルートコンプレックスを

確認します。

次の例では、BB1のデバイスを持つルートコンプレックスがPCIE8とPCIE12が割り当てられていないことがわかります。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
（以下省略）
```

ldm add-ioコマンドを実行し、primaryへPCIE8とPCIE12を追加して、Oracle Solarisを再起動します。

```
# ldm add-io PCIE8 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 add-io PCIE12 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.
--------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
....
```

Oracle Solarisが起動されたら、ldm list-ioコマンドを実行し、制御ドメインにBB#01のルートコンプレックスが追加されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE4                                       BUS    PCIE4    primary  IOV
PCIE8                                       BUS    PCIE8    primary  IOV
PCIE12                                      BUS    PCIE12   primary  IOV
/BB0/CMUL/NET0                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                            PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                              PCIE   PCIE4    primary  OCC
```

b. **制御ドメインのシステムボリュームの冗長化を構成します。**

制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

次の例では制御ドメインのシステムボリュームのZFSミラー機能を構成する手順を示します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

zpool attachコマンドを実行し、ディスクをミラー構成に組み込みます。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

zpool statusコマンドで、ミラー構成が組み込まれたことを確認します。

なお、同期処理（resilver）が完了したかどうか、zpool statusコマンドで状態を確認してください。

次の例は、同期処理中の表示例です。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         DEGRADED     0     0     0
          mirror-0                    DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   DEGRADED     0     0     0  (resilvering)
errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を構築したり、デバイスの利用を再開したりしてください。冗長構成の構築方法やデバイスの利用再開方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

A.2.4 活性交換手順例（PCIeバスを動的に割り当て可能な場合）

ここでは、PCIeバスを動的に割り当て可能な環境（XCP 2240以降、かつ、Oracle VM Server for SPARC 3.2以降で、ルートドメインがOracle Solaris 11.2 SRU11.2.8.4.0以降）で、「図 A-2　2BB構成で運用継続可能な構成例」に記載された2BB構成システムに対して、PPAR DRを使用して、BB#01を活性交換する手順例を記載します。

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

1. **マスタXSCFにログインします。**
   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の場合は、マスタXSCFへログインしなおしてください。

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

2. **showhardconfコマンドを実行し、交換するSPARC M10-4SのXSCFの [Status] が「Normal」であることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Locked;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
```

```
        + FRU-Part-Number: CA07361-D202 A1                          ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
           + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
            CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
                + Freq:3.000 GHz; Type:0x10;
                + Core:16; Strand:2;
      :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                          ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
      :
```

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

3. **showbbstatusコマンドを実行し、交換するSPARC M10-4SのXSCFがマスタでないことを確認します。**

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

交換するSPARC M10-4SがマスタXSCFの場合、switchscfコマンドを実行し、XSCFを切り替えます。

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

注ーXSCFの切り替え、およびXSCFの再起動が完了したことを確認したあとに、システムボードの切り離し操作を行ってください。

---

4. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

5. **論理ドメインの稼働状況やリソース使用状況を確認します。**
   a. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

この例では、制御ドメイン、2つのルートドメイン、2つのゲストドメインが稼働しています。

すべてのドメインが「active」でOracle Solarisが動作状態、または「inactive」状態であることを確認します。OpenBoot PROM状態またはbound状態のドメインが存在すると、物理パーティションの動的再構成に失敗することがあります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  1h 33m
guest0           active     -n----  5100    64    64G      3.1%  2s
guest1           active     -n----  5101    64    64G      1.6%  18m
root-dom0        active     -n--v-  5000    32    32G      3.1%  17m
root-dom1        active     -n--v-  5001    32    32G      3.1%  17m
```

b. **-aオプションを指定してldm list-devicesコマンドを実行し、リソースの使用状況を確認します。**

次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID      %FREE    CPUSET
    0       0        (0, 1)
    4       0        (8, 9)
    8       0        (16, 17)
 （中略）
    944     0       (1888, 1889)
    948     0       (1896, 1897)
    952     0       (1904, 1905)
    956     0       (1912, 1913)
VCPU
    PID     %FREE    PM
    0       0        no
    1       0        no
    8       0        no
    9       0        no
 （中略）
    1904    0       no
    1905    0       no
    1912    0       no
    1913    0       no
 （以下略）
```

6. **制御ドメインのシステムボリュームやI/Oデバイスの冗長構成を切り離します。**
BB#01を切り離せるようにするために、制御ドメインで使用している、交換するSPARC M10-4SのI/Oデバイスを切り離します。冗長構成を解除する手順の詳細は、各冗長化ソフトウェアのドキュメントを参照してください。

a. **制御ドメインのシステムボリュームの冗長化を解除します。**

次の例では制御ドメインのシステムボリュームのZFSミラー機能を解除する手順を示します。

制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

zpool detachコマンドを実行し、ディスクをミラー構成から切り離します。

```
# zpool detach rpool c3t50000393A803B13Ed0
```

zpool statusコマンドで、ミラー構成が解除されたことを確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を解除したり、デバイスの利用を停止したりしてください。冗長構成の解除方法やデバイスの利用停止方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

b. **制御ドメインのI/O構成を削除します。**

制御ドメインに割り当てられている物理I/Oデバイスのうち、BB#01のルートコンプレックスを動的再構成で削除します。

ldm list-ioコマンドを実行し、primaryに割り当てられているルートコンプレックスを確認します。

次の例では、BB1のデバイスを持つルートコンプレックスがPCIE8とPCIE12であることがわかります。

```
# ldm list-io | grep primary
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE4                                    BUS    PCIE4    primary  IOV
PCIE8                                    BUS    PCIE8    primary  IOV
PCIE12                                   BUS    PCIE12   primary  IOV
/BB0/CMUL/NET0                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                         PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                           PCIE   PCIE4    primary  OCC
/BB1/CMUL/NET0                           PCIE   PCIE8    primary  OCC
/BB1/CMUL/SASHBA                         PCIE   PCIE8    primary  OCC
/BB1/CMUL/NET2                           PCIE   PCIE12   primary  OCC
```

ldm remove-ioコマンドを実行し、primaryからPCIE8とPCIE12を削除します。

```
# ldm remove-io PCIE8 primary
# ldm remove-io PCIE12 primary
```

ldm list-ioコマンドを実行し、制御ドメインからBB#01のルートコンプレックスが削除されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE4                                    BUS    PCIE4    primary  IOV
/BB0/CMUL/NET0                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                         PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                           PCIE   PCIE4    primary  OCC
```

c. **ゲストドメインに割り当てている仮想I/Oデバイスの冗長構成を解除します。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）を事前にシャットダウンするために、各ゲストドメインにログインして、root-dom1からの仮想I/Oデバイスの冗長構成を解除します。

冗長構成ソフトウェアの使用方法については、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。

次の例では、仮想ネットワークデバイス（vnet1）をIPMPの構成から解除する例を示しています。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  4h 17m
guest0           active     -n----  5100    64    64G      0.0%  1h 13m
guest1           active     -n----  5101    64    64G      0.0%  1h 4m
root-dom0        active     -n--v-  5000    32    32G      0.0%  1h 47m
```

```
root-dom1        active     -n--v-  5001    32    32G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         0      unknown   vnet1
```

ipmpstat -iコマンドを実行し、IPMPを構成しているネットワークインターフェースの構成情報を確認します。

```
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       is-----   up        disabled  ok
```

if_mpadm -dコマンドを実行し、IPMPのグループからnet1を切り離して、ipmpstat -iコマンドを実行し、切り離されたことを確認します。次の例の場合、STATEがofflineになったことを確認します。

```
guest1# if_mpadm -d net1
guest1# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       -s---d-   up        disabled  offline
```

ipadm delete-ipコマンドを実行して、net1を削除します。

```
guest0# ipadm delete-ip net1
```

ゲストドメイン（guest1）についても、同様の切り離しを行います。

d. **停止するルートドメインから割り当てている仮想I/Oデバイスを削除します。**

ldm remove-vdiskコマンド、ldm remove-vnetコマンドを実行して、次の手順で停止するルートドメインから割り当てている仮想ディスク（vdisk）と仮想ネットワークデバイス（vnet）を削除します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている仮想ディスク（vdisk10）と仮想ネットワークデバイス（vnet10）を削除するコマンドの実行例を示しています。

```
# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0
```

ゲストドメイン（guest1）についても、同様の削除を行います。

7. **I/Oデバイスのリソース使用状況を確認し、交換するSPARC M10-4SのI/Oデバイスをすべて解除します。**

   a. **切り離すSPARC M10-4Sのルートコンプレックスを割り当てている論理ドメインを確認します。**

   ldm list-ioコマンドを実行し、BB#01であるSPARC M10-4Sのルートコンプレックスを割り当てている論理ドメインを確認します。

   次の例では、root-dom1だけが「/BB1/」で始まるPCIeエンドポイントを所有しています。そして、そのPCIeエンドポイントのルートコンプレックス（BUS）である、PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15がroot-dom1に割り当てられていることがわかります。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
PCIE14                                      BUS    PCIE14   root-dom1  IOV
PCIE15                                      BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                              PCIE   PCIE8             UNK
/BB1/CMUL/SASHBA                            PCIE   PCIE8             UNK
/BB1/PCI0                                   PCIE   PCIE9    root-dom1OCC
/BB1/PCI3                                   PCIE   PCIE10   root-dom1OCC
/BB1/PCI4                                   PCIE   PCIE10   root-dom1OCC
/BB1/PCI7                                   PCIE   PCIE11   root-dom1OCC
/BB1/PCI8                                   PCIE   PCIE11   root-dom1OCC
/BB1/CMUL/NET2                              PCIE   PCIE12             UNK
/BB1/PCI1                                   PCIE   PCIE13   root-dom1OCC
/BB1/PCI2                                   PCIE   PCIE13   root-dom1OCC
/BB1/PCI5                                   PCIE   PCIE14   root-dom1OCC
/BB1/PCI6                                   PCIE   PCIE14   root-dom1OCC
/BB1/PCI9                                   PCIE   PCIE15   root-dom1OCC
/BB1/PCI10                                  PCIE   PCIE15   root-dom1OCC
```

   b. **切り離すSPARC M10-4Sのルートコンプレックスを割り当てているルートド**

メインを停止して切り離します。

次の例では、ldm stop-domainコマンド、ldm unbind-domainコマンドを実行し、ルートドメイン（root-dom1）を切り離し、inactive状態になったことが確認できます。

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  4h 59m
guest0           active     -n----  5100    64    64G      0.0%  1h 55m
guest1           active     -n----  5101    64    64G      0.0%  1h 46m
root-dom0        active     -n--v-  5000    32    32G      0.0%  2h 29m
root-dom1        inactive   ------          32    32G
```

c. **交換するSPARC M10-4SのI/Oデバイスがすべて解除されたことを確認します。**

ldm list-ioコマンドを実行し、I/Oデバイスが切り離されたことを確認します。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9
PCIE10                                      BUS    PCIE10
PCIE11                                      BUS    PCIE11
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13
PCIE14                                      BUS    PCIE14
PCIE15                                      BUS    PCIE15
（以下省略）
```

8. **論理ドメインに割り当てられているCPUコアとメモリリソースを手動で削減します。**

ここでは、-m unbind=resourceオプションを指定する代わりに、論理ドメインに割り当てられているCPUコアとメモリリソースを手動で削減して、「2.5.2　運用時の留意点」の「CPUコアとメモリの配置」の条件を満たすようにする手順を説明します。

deleteboardコマンドがエラーで失敗したときなどに、この手順を実施することで、deleteboardコマンドが実施できる場合があります。

deleteboardコマンドの-m unbind=resourceオプションを指定する場合は、手順9に進んでください。

a. **CPUコアの数を確認して削除します。**

SPARC M10-4Sの切り離しによって、利用できるCPUコア数が減るため、次の手順で論理ドメインに割り当てているCPUコア数を削減しておきます。

i. **SPARC M10-4Sを切り離したあとのCPUコア数を確認します。**

XSCFから、showpparinfoコマンドを実行し、切り離される予定のSPARC M10-4Sを除いたCPUコア数を確認します。

次の例では、物理システムボード(PSB)番号 01-0を切り離す場合、他PSB番号 00-0のCoresの合計値を計算することになるため、16+16+16+16＝64コアとなります。

```
XSCF> showpparinfo -p 0
PPAR#00 Information:
--------------------
 CPU(s)                        :       8
 CPU Cores                     :     128
 CPU Threads                   :     256
 Memory size (GB)              :     256
 CoD Assigned (Cores)          :     256
CPU(s):
-------
 PID  PSB   CPU#  Cores  Threads
 00   00-0  0        16       32
 00   00-0  1        16       32
 00   00-0  2        16       32
 00   00-0  3        16       32
 00   01-0  0        16       32
 00   01-0  1        16       32
 00   01-0  2        16       32
 00   01-0  3        16       32
（以下省略）
```

ii. **各論理ドメインに割り当てられているCPUコア数の総数を確認します。**

ldm list-devices -a core を実行し、%FREEの列が100以外になっている行の数が、論理ドメインに割り当てられているCPUコア数の総数になります。

次の例では、ldm list-devices -a coreコマンドの実行例と、-pオプションを使って確認する実行例を示します。この結果、112コアを論理ドメイン全体にバインドしていることがわかります。

```
# ldm list-devices -a core
CORE
    ID       %FREE   CPUSET
    0        0      (0, 1)
    4        0      (8, 9)
    8        0      (16, 17)
    12       0      (24, 25)
（以下省略）
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
     112
```

iii. **SPARC M10-4Sの切り離しによって不足するコア数を計算します。**

以下の計算式から、SPARC M10-4S切り離し後に不足するCPUコア数を計算します。

不足するCPUコア数＝論理ドメインで使用するコア数（手順ⅱ）－切り離し後の物理コア数（手順ⅰ）

手順i、iiの例の場合、112コア（使用中）－64コア（残り）=48コア不足することがわかります。

iv. **CPUコアが不足する場合、削除対象の論理ドメインを検討します。**

手順iiiの結果で、CPUコアが不足する場合、論理ドメインからCPUコアを削除しておく必要があります。

ldm list-domainコマンドを実行し、activeまたはbound状態の各論理ドメインに割り当てられているCPUコア数を確認し、CPUコアを削除する論理ドメインを確認します。

次の例では、primaryに32コア（64 vcpu）、guest0に32コア（64 vcpu）、guest1に32コア（64 vcpu）、root-dom0に16コア（32 vcpu）割り当てられていることがわかります。例では、48コアを削除する必要があるため、primary、guest0、guest1から16コアずつ削除することにします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.1%  18h 17m
guest0           active     -n----  5100    64    64G      0.0%  15h 13m
guest1           active     -n----  5101    64    64G      0.0%  15h 4m
root-dom0        active     -n--v-  5000    32    32G      0.0%  15h 47m
root-dom1        inactive   ------          32    32G
```

v. **ldm remove-coreコマンドを実行し、対象の論理ドメインからCPUコアを削除します。**

次の例は、primary、guest0、guest1からそれぞれ16コアずつ減らして、実際に減ったかどうか確認するコマンドの実行例を示しています。

```
# ldm remove-core 16 primary
# ldm remove-core 16 guest0
# ldm remove-core 16 guest1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    56G      0.0%  18h 19m
guest0           active     -n----  5100    32    64G      0.0%  15h 15m
guest1           active     -n----  5101    32    64G      0.0%  15h 5m
root-dom0        active     -n--v-  5000    32    32G      0.0%  15h 49m
root-dom1        inactive   ------  5001    32    32G
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
      64
```

b. **メモリリソースを確認して削除します。**

SPARC M10-4Sの切り離しによって、利用できるメモリ領域が減るため、以下の手順で論理ドメインに割り当てられているメモリリソースを削除してお

きます。

i. **メモリの連続領域（メモリブロック）の使用状況を確認します。**

prtdiagコマンドとldm list-devices -a memoryコマンドを実行し、各論理ドメインに割り当てられているメモリブロックと、未割り当てのメモリブロックが、どのSPARC M10-4Sに割り当てられているかを確認します。

最初に、prtdiagコマンドを実行し、メモリの物理アドレスとSPARC M10-4Sの対応を確認します。

```
# prtdiag
(略)
======================= Physical Memory Configuration ========================
Segment Table:
----------------------------------------------------------------
Base            Segment  Interleave   Bank     Contains
Address         Size     Factor       Size     Modules
----------------------------------------------------------------
0x7e0000000000 32 GB    4            8 GB     /BB0/CMUL/CMP0/MEM00A
(略)
0x7c0000000000 32 GB    4            8 GB     /BB0/CMUL/CMP1/MEM10A
(略)
0x7a0000000000 32 GB    4            8 GB     /BB0/CMUU/CMP0/MEM00A
(略)
0x780000000000 32 GB    4            8 GB     /BB0/CMUU/CMP1/MEM10A
(略)
0x760000000000 32 GB    4            8 GB     /BB1/CMUL/CMP0/MEM00A
(略)
0x740000000000 32 GB    4            8 GB     /BB1/CMUL/CMP1/MEM10A
(略)
0x720000000000 32 GB    4            8 GB     /BB1/CMUU/CMP0/MEM00A
(略)
0x700000000000 32 GB    4            8 GB     /BB1/CMUU/CMP1/MEM10A
(以下省略)
```

この例の結果を、メモリの物理アドレスを小さい順に並べ直して、物理アドレスとSPARC M10-4Sの対応を表にすると以下のようになります。

表 A-5 物理アドレスとSPARC M10-4Sの対応例

Base Address（物理アドレス）	ビルディングブロック構成中の**SPARC M10-4S**
0x700000000000～	BB1
0x720000000000～	BB1
0x740000000000～	BB1
0x760000000000～	BB1
0x780000000000～	BB0
0x7a0000000000～	BB0
0x7c0000000000～	BB0
0x7e0000000000～	BB0

次に、ldm list-devices -a memory コマンドを実行し、論理ドメインに割り当てられているメモリの連続領域（以降、メモリブロックと呼びます）と、未割り当てのメモリブロックを確認します。

次の例は、ldm list-devices -a memoryコマンドの実行結果の例です。

各パラメーターの意味は以下のとおりです。

PA　: メモリブロックの先頭の物理アドレス

SIZE : メモリブロックのサイズ

BOUND: メモリブロックを割り当てている論理ドメイン名。空欄は未割り当ての領域、_sys_は論理ドメインに割り当てられない制御用の領域です。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom0
    0x720000000000      32G
    0x740000000000      32G             guest0
    0x760000800000      1272M           _sys_
    0x760050000000      31488M          guest1
    0x780000000000      32G             guest0
    0x7a0000000000      32G             guest1
    0x7c0000000000      28G             primary
    0x7c0700000000      4G
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      28G             primary
    0x7e0780000000      1280M           guest1
    0x7e07d0000000      768M
```

上記の結果と、prtdiagコマンドで確認した物理位置を合わせると、以下のようなメモリブロックの使用状況であることがわかります。

表 A-6　メモリブロックの使用状況の例

SPARC M10-4S	物理アドレス	サイズ	論理ドメイン
BB1（交換対象）	0x700000000000	32 GB	root-dom0
	0x720000000000	32 GB	未割り当て
	0x740000000000	32 GB	guest0
	0x760050000000	31,488 MB	guest1
BB0	0x780000000000	32 GB	guest0
	0x7a0000000000	32 GB	guest1
	0x7c0000000000	28 GB	primary
	0x7c0700000000	4 GB	未割り当て
	0x7e0080000000	28 GB	primary
	0x7e0780000000	1,280 MB	guest1

表 A-6　メモリブロックの使用状況の例 (続き)

SPARC M10-4S	物理アドレス	サイズ	論理ドメイン
	0x7e07d0000000	768 MB	未割り当て

ii. **移動元のメモリブロックのサイズと個数を確認します。**

メモリブロックの使用状況の確認結果を参照し、交換対象のSPARC M10-4S上で割り当てられているメモリブロック（以降、移動元のメモリブロックと呼びます）を確認します。

「表 A-6　メモリブロックの使用状況の例」の場合、BB1側で論理ドメインに割り当てられているメモリブロックとして、32 GB×2個、31,488 MB×1個のメモリブロックが使用中であることがわかります。

iii. **メモリを削除する論理ドメインとサイズを決定します。**

次に、各論理ドメインに割り当てているメモリブロックの配置を確認し、メモリブロック単位で削除したり、メモリブロックのサイズを減らしたりすることで、移動元のメモリブロックを、切り離し対象ではないSPARC M10-4Sの未割り当てのメモリブロック（以降、移動先のメモリブロックと呼びます）に移動できるようにします。

その結果として、最終的にどの論理ドメインからどのくらいのメモリを削除すればいいかを決定します。

方法は以下があります。

- 移動元のメモリブロックをまるごと削除することで、移動する必要のあるメモリブロックを 減らす。
- 切り離し対象ではないSPARC M10-4S上の、論理ドメインに割り当てられているメモリブロックをまるごと削除して空き領域を増やすことで、移動先を増やす。
- 移動元のメモリブロックのサイズを減らして、移動先の空き領域に入るサイズにする。
- 移動元のメモリブロックのサイズを減らして、かつ移動先で使用中のメモリブロックのサイズを減らすことで、移動先の空きメモリブロックを増やして、移動できるようにする。

注ーサイズを減らしたあとの空きメモリブロックは、連続領域にならない（フラグメント化する）ことに注意してください。小さいメモリブロックを複数削除して空き領域を増やしても、空き領域が連続領域ではないため、移動元のメモリブロックの連続領域が大きい場合は、移動できません。そのような場合は、移動元のメモリブロックも削除してサイズを調整してください。

注ー検討をする際、なるべく存在しているメモリブロックと同じサイズで削除できるものを削除対象として選択することで、削除後に連続領域が細分化（フラグメント化）する可能性を減らすことができます。それにより、メモリブロックの移動処理が成功する可能性が高まります。

注ーメモリを削除しすぎると、論理ドメイン内のメモリを圧迫することになり、Oracle Solarisがハングアップするなどの問題が発生する可能性があります。vmstatコマンドで"free"のサイズを目安にして、削除しすぎないよう注意が必要です。

「表 A-6　メモリブロックの使用状況の例」を元に削除プランを検討します。

検討の結果、例として「表 A-7　メモリブロックの削除プラン」で示すとおり、root-dom0から4 GB削除、guest0から32 GB削除、guest1から31,488 MB削除、primaryから28 GB削除するプランを立てます。

表 A-7　メモリブロックの削除プラン

SPARC M10-4S	サイズ	論理ドメイン	削除プラン
BB1（交換対象）	32 GB	root-dom0	この領域を4 GB減らして28 GBにして、BB0側のprimaryの28 GBを削除して、移動させる
	32 GB	未割り当て	ー
	32 GB	guest0	BB0側に32 GBのguest0のメモリがあるので、削除
	31,488 MB	guest1	BB0側に32 GBのguest1のメモリがあるので、削除
BB0	32 GB	guest0	残す
	32 GB	guest1	残す
	28 GB	primary	残す
	4 GB	未割り当て	ー
	28 GB	primary	root-dom0の28 GBを移動させるため、削除
	1,280 MB	guest1	残す
	768 MB	未割り当て	ー

iv. **論理ドメインからメモリを手動で削除します。**

手順iiiで決定した、メモリの削除プランに従って、ldm remove-memoryコマンドを使って論理ドメインからメモリを削除します。

次の例は、「表 A-7　メモリブロックの削除プラン」に従った、メモリを削除するコマンドの実行例です。

```
# ldm remove-memory 4G root-dom0
# ldm remove-memory 32G guest0
# ldm remove-memory 31488M guest1
# ldm remove-memory 28G primary
```

v. **削除されたメモリブロックの状態を確認します。**

ldm list-devices -a memoryコマンドを実行し、削除された結果を参照して、移動できる配置になったかどうかを確認します。移動できないメモリブロックが残った場合、この結果を元に、さらに削除するメモリブロックを検討して、削除します。

次の例の場合、BB1側で割り当てられている大きいサイズと、BB0側の空き領域の大きいサイズを並べて比較することで、移動できるかどうかが確認

しやすくなります。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
(BB1側)
    0x700000000000      256M            root-dom0  →4 GBの移動先を分割して移動可能
    0x700010000000      4G
    0x700110000000      28416M          root-dom0  →0x780000000000(28 GB)へ移
                                                     動可能
    0x720000000000      32G
    0x740000000000      256M            guest0     →4 GBの移動先を分割して移動可能
    0x740010000000      4G
    0x740110000000      28416M          guest0     →移動先がないため、再度削除が必
                                                     要(*)
    0x760000800000      1272M           _sys_
    0x760050000000      256M            guest1     →4 GBの領域を分割して移動可能
    0x760060000000      1792M
    0x7600d0000000      29440M          guest1     →0x7a0000000000(29 GB)へ移
                                                     動可能
(BB0側)
    0x780000000000      28G                        ←root-dom0
                                                     (0x700110000000)から移動可
                                                     能
    0x780700000000      4G              guest0
    0x7a0000000000      29G                        ←guest1(0x7600d0000000)か
                                                     ら移動可能
    0x7a0740000000      3G              guest1
    0x7c0000000000      256M            primary
    0x7c0010000000      4G                         ←root-dom0、guest0、guest1
                                                     の256 MBから移動可能
    0x7c0110000000      24320M          primary
    0x7c0700000000      4G
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      24G                        ←guest0(0x740110000000)が
                                                     移動するには不足(*)
    0x7e0680000000      4G              primary
    0x7e0780000000      1280M           guest1
    0x7e07d0000000      768M
```

上記の例の（*）部分が、移動先として24 GB（24,576 MB）しか空いていないため、移動元（BB1側）の28,416 MBの領域（guest0）を3,840 MB削除して、再度確認します。下記の例で、すべてのメモリブロックが移動可能な状態になったことがわかります。

```
# ldm remove-memory 3840M guest0
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
(BB1側)
    0x700000000000      256M            root-dom0
```

```
    0x700010000000       4G
    0x700110000000       28416M          root-dom0
    0x720000000000       32G
    0x740000000000       256M            guest0
    0x740010000000       7936M
    0x740200000000       24G             guest0  →0x7e0080000000(24 G)へ移動可
                                                 能
    0x760000800000       1272M           _sys_
    0x760050000000       256M            guest1
    0x760060000000       1792M
    0x7600d0000000       29440M          guest1
(BB0側)
    0x780000000000       28G
    0x780700000000       4G              guest0
    0x7a0000000000       29G
    0x7a0740000000       3G              guest1
    0x7c0000000000       256M            primary
    0x7c0010000000       4G
    0x7c0110000000       24320M          primary
    0x7c0700000000       4G
    0x7e0000800000       1272M           _sys_
    0x7e0050000000       512M            _sys_
    0x7e0070000000       256M            _sys_
    0x7e0080000000       24G                     ←guest0(0x740200000000)から
                                                 移動可能
    0x7e0680000000       4G              primary
    0x7e0780000000       1280M           guest1
    0x7e07d0000000       768M
```

9. **SPARC M10-4S**のシステムボード（**PSB<BB>**）を物理パーティションから切り離します。
   a. **deleteboard -c disconnect**コマンドを実行し、**PSB**を物理パーティションから切り離します。

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

   b. **showresult**コマンドを実行し、直前に実行した**deleteboard**コマンドの終了ステータスを確認します。

      終了値が0の場合、deleteboardコマンドは正常終了です。

      終了値が0以外の場合、およびdeleteboardコマンド実行中にエラーメッセー

ジが出力された場合は、異常終了です。エラーメッセージより「C.1.2 deleteboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

c. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換するSPARC M10-4SのPSBが「Assigned」状態であり、 [Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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     y    y    y    Passed   Normal
01-0 00(01)       Assigned     n    n    n    Passed   Normal
```

10. **replacefruコマンドを実行し、SPARC M10-4Sを交換します。**

```
XSCF> replacefru
```

注－replacefruコマンドを使用したSPARC M10-4Sの交換の詳細は、『SPARCM10-4/M10-4Sサービスマニュアル』の「5.8　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムから切り離す」および「6.2　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムに組み込む」を参照してください。

---

11. **PSBを物理パーティションに組み込みます。**

a. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換するSPARC M10-4SであるPSBがAssigned状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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     y    y    y    Passed   Normal
01-0 00(01)       Assigned     n    n    n    Passed   Normal
```

b. **addboard -c configureコマンドを実行し、PSBを物理パーティションに組み込みます。**

論理ドメイン構成を元の構成に戻すため、-m bind=resourceオプションを指定して、addboard -c configureコマンドを実行します。

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
```

```
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

注ーaddboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

---

c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

終了値が0の場合、addboardコマンドは正常終了です。

終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換したSPARC M10-4SのPSBが正常に組み込まれたことで、 [Conn] 欄と [Conf] 欄の表示がともに「y」と表示され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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

12. **論理ドメインの稼働状況を確認します。**

a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

b. **ldm list-domainコマンドを実行し、SPARC M10-4SのPSB（BB）の追加後に論理ドメインの稼働状況に変化がないことを確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

```
# ldm list-domain
NAME             STATE      FLAGS   CONS   VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART   32    28G      64%   2h 54m
guest0           active     -n----  5100   32    61876M   42%   2h 54m
guest1           active     -n----  5101   32    62388M   11%   2h 54m
root-dom0        active     -n--v-  5000   32    28G      0.0%  2h 54m
root-dom1        inactive   ------         32    32G      0.0%  2h 54m
```

13. **削除されたリソースを復元します。**
    deleteboardコマンドによって論理ドメインに割り当てられたリソースが削減されている場合、ldm add-core、ldm add-memoryコマンドで追加して、リソースを復元します。

    次の例では、削減されたCPUコアとメモリを追加して、SPARC M10-4Sを交換する前のリソースに戻しています。

```
# ldm add-core 16 primary
# ldm add-core 16 guest0
# ldm add-core 16 guest1
# ldm add-memory 28G primary
# ldm add-memory 4G root-dom0
# ldm add-memory 36608M guest0
# ldm add-memory 31488M guest1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  4h 59m
guest0           active     -n----  5100    64    64G      0.0%  1h 55m
guest1           active     -n----  5101    64    64G      0.0%  1h 46m
root-dom0        active     -n--v-  5000    32    32G      0.0%  2h 29m
root-dom1        inactive   ------          32    32G
```

14. **I/Oデバイスの使用を再開します。**
    a. **ルートコンプレックスを再割り当てします。**

    ldm bind-domain, ldm start-domain コマンドを実行し、交換したSPARC M10-4Sのルートコンプレックスを割り当てていたルートドメインを、unbind状態から起動します。

    次の例では、unbind状態にしたルートドメイン（root-dom1）を起動して、起動されたことを確認しています。

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.2%  3h 8m
guest0           active     -n----  5100    64    64G      0.0%  3h 8m
guest1           active     -n----  5101    64    64G      0.0%  3h 8m
root-dom0        active     -n--v-  5000    32    32G      0.0%  3h 8m
```

```
root-dom1         active     -n--v-  5001    32    32G      7.3%  8s
```

ldm list-ioコマンドを実行し、起動したルートドメインに物理I/Oデバイスが割り当てられていることを確認します。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
PCIE14                                      BUS    PCIE14   root-dom1  IOV
PCIE15                                      BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                              PCIE   PCIE8             UNK
/BB1/CMUL/SASHBA                            PCIE   PCIE8             UNK
/BB1/PCI0                                   PCIE   PCIE9    root-dom1OCC
/BB1/PCI3                                   PCIE   PCIE10   root-dom1OCC
/BB1/PCI4                                   PCIE   PCIE10   root-dom1OCC
/BB1/PCI7                                   PCIE   PCIE11   root-dom1OCC
/BB1/PCI8                                   PCIE   PCIE11   root-dom1OCC
/BB1/CMUL/NET2                              PCIE   PCIE12            UNK
/BB1/PCI1                                   PCIE   PCIE13   root-dom1OCC
/BB1/PCI2                                   PCIE   PCIE13   root-dom1OCC
/BB1/PCI5                                   PCIE   PCIE14   root-dom1OCC
/BB1/PCI6                                   PCIE   PCIE14   root-dom1OCC
/BB1/PCI9                                   PCIE   PCIE15   root-dom1OCC
/BB1/PCI10                                  PCIE   PCIE15   root-dom1OCC
```

b. **ルートドメインからの仮想I/Oデバイスをゲストドメインに追加します。**

ldm add-vdiskコマンド、ldm add-vnetコマンドを実行して、起動したルートドメインの仮想I/Oサービスに対応した、仮想ディスク（vdisk）と、仮想ネットワークデバイス（vnet）を、各ゲストドメインに追加します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている、仮想ディスク（vdisk10）と、仮想ネットワークデバイス（vnet10）を追加するコマンドの実行例を示しています。

```
# ldm add-vdisk id=1 vdisk10 vol10@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

ゲストドメイン（guest1）についても、同様の追加を行います。

注－仮想I/Oデバイスを再度追加する場合、あらかじめ割り当てられたIDを指定する必要があります。IDは、仮想I/Oデバイスを削除する前の状態でldm list -lコマンドの実行結果から確認できます。

---

c. **ゲストドメインに割り当てている仮想I/Oデバイスを冗長構成に組み込みます。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）が起動されると、各ゲストドメインに対する仮想I/Oデバイスのサービスも起動されます。

各ゲストドメインにログインして、事前に解除しておいたroot-dom1からの仮想I/Oデバイスを冗長構成に組み込みます。冗長構成ソフトウェアの使用方法については、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。

ここでは、仮想ネットワークデバイス（vnet1）をIPMPの構成に組み込む例を説明します。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

まず、ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  4h 17m
guest0           active     -n----  5100    64    64G      0.0%  1h 13m
guest1           active     -n----  5101    64    64G      0.0%  1h 4m
root-dom0        active     -n--v-  5000    32    32G      0.0%  1h 47m
root-dom1        active     -n--v-  5001    32    32G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         0      unknown   vnet1
```

ipadm create-ipコマンド、ipadm set-ifpropコマンド、ipadm add-ipmpコマンドを実行し、net1をipmp0の待機デバイスとして登録します。

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

ipmpstat -iコマンドを実行し、IPMPを構成しているネットワークインターフェースのSTATEがokになっていることを確認します。

```
guest0# ipmpstat -i
INTERFACE ACTIVE GROUP FLAGS LINK PROBE STATE
net0 yes  ipmp0 -smbM-- up   disabled   ok
net1 no   ipmp0 -s---d- up   disabled   ok
```

ゲストドメイン（guest1）についても、同様の対応を行います。

15. **制御ドメインのシステムボリュームやI/Oデバイスを冗長構成に戻します。**

    a. **制御ドメインのルートコンプレックス構成を追加します。**

    制御ドメインから事前に削除した、BB#01のルートコンプレックスを追加します。

    ldm list-ioコマンドを実行し、割り当てられていないルートコンプレックスを確認します。

    次の例では、BB1のデバイスを持つルートコンプレックスがPCIE8とPCIE12が割り当てられていないことがわかります。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
（以下省略）
```

ldm add-ioコマンドを実行し、primaryへPCIE8とPCIE12を追加します。

```
# ldm add-io PCIE8 primary
# ldm add-io PCIE12 primary
```

ldm list-ioコマンドを実行し、制御ドメインにBB#01のルートコンプレックスが追加されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                       BUS    PCIE0    primary  IOV
PCIE4                                       BUS    PCIE4    primary  IOV
PCIE8                                       BUS    PCIE8    primary  IOV
PCIE12                                      BUS    PCIE12   primary  IOV
/BB1/CMUL/NET0                              PCIE   PCIE0    primary  OCC
/BB1/CMUL/SASHBA                            PCIE   PCIE0    primary  OCC
```

```
/BB1/CMUL/NET2                              PCIE    PCIE4    primary  OCC
```

b. **制御ドメインのシステムボリュームの冗長化を構成します。**

制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

次の例では制御ドメインのシステムボリュームのZFSミラー機能を構成する手順を示します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

zpool attachコマンドを実行し、ディスクをミラー構成に組み込みます。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

zpool statusコマンドで、ミラー構成が組み込まれたことを確認します。

なお、同期処理（resilver）が完了したかどうか、zpool statusコマンドで状態を確認してください。

次の例は、同期処理中の表示例です。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       DEGRADED     0     0     0
          mirror-0                  DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0 ONLINE       0     0     0
            c3t50000393A803B13Ed0s0 DEGRADED     0     0     0  (resilvering)
errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を構築したり、デバイスの利用を再開したりしてください。冗長構成の構築方法やデバイスの利用再開方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

A.3 XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがある論理ドメインの構成の場合

ここでは、2つのSPARC M10-4S、すなわち2つのシステムボード（PSB<BB>）で構成される物理パーティションで、物理パーティションの動的再構成時にリソースを移動するために必要とするハードウェアリソース（CPUコアとメモリ）の空きがある構成、つまり、1つのシステムボード分の量のハードウェアリソースを論理ドメインに割り当てた構成を構築する手順の例と、物理パーティションの動的再構成でSPARC M10-4Sの活性交換を行う手順の例を説明します。
この場合、論理ドメインに割り当てたCPUコアやメモリが削減されずにシステムボードの活性交換ができます。ここでの説明は、SPARC M12でも同様です。

A.3.1 構成例

ここでは、搭載されているCPU数とメモリ容量が同じ2つのSPARC M10-4S、すなわち2つのシステムボード（PSB<BB>）を使った（2BB構成）論理ドメインの構成例を示します。
物理パーティションで動的再構成を実施できるようにするために、以下の条件を満たすI/O構成にする必要があります。

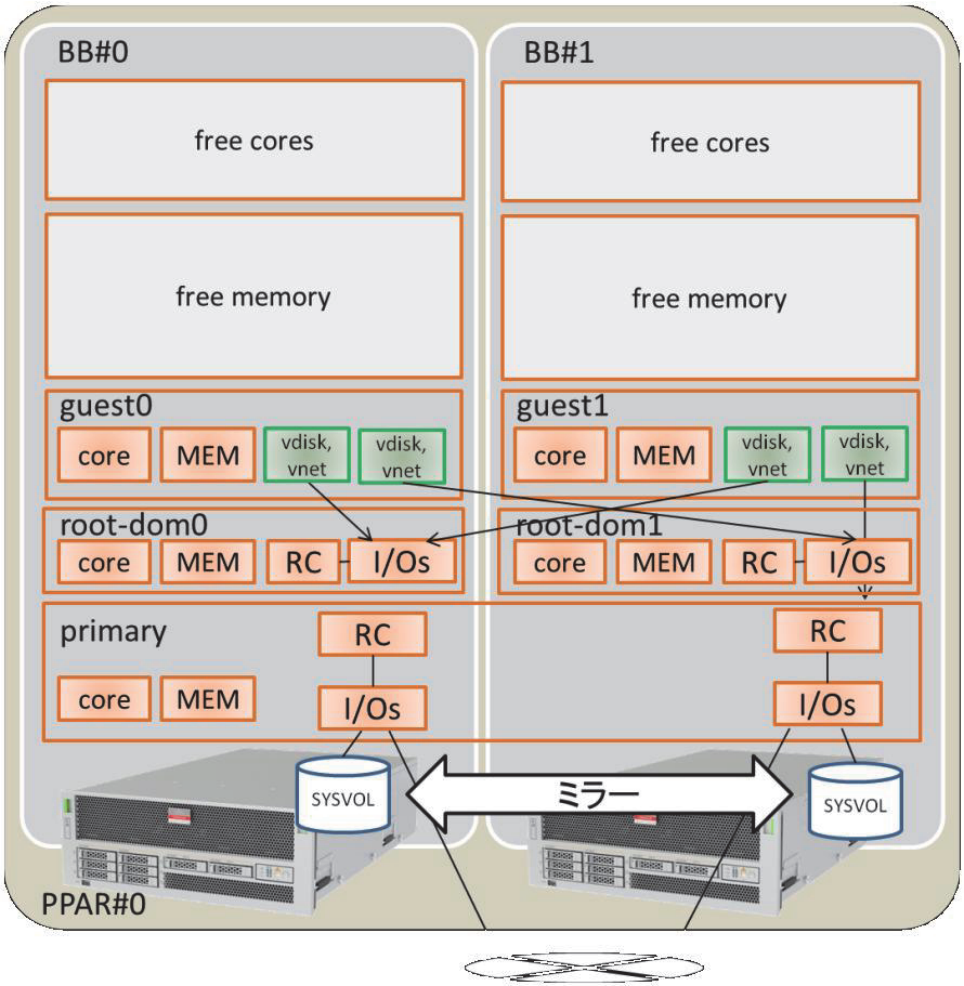
- 制御ドメインのシステムボリュームディスクやネットワークインターフェースは、各SPARC M10-4SのI/Oを接続して冗長構成にする必要があります。これはSPARC M10-4Sを削除した場合に、残りのSPARC M10-4SのI/Oを使って動作を継続できるようにするためです。
- SPARC M10-4Sごとにルートドメインを配置して、ゲストドメインへ仮想I/Oを提供するための専用のルートドメインにする必要があります。これは、ルートドメ

インを業務で利用していると、SPARC M10-4Sを切り離すときに、ルートドメインを停止する必要があり、業務が停止してしまうためです。

- 業務はゲストドメイン上で運用する必要があります。このゲストドメインの仮想I/Oデバイス（vdisk、vnet）は、各ルートドメインの仮想I/Oサービスを割り当てて、冗長構成にする必要があります。これは、片方のSPARC M10-4Sが切り離されても、残りのSPARC M10-4S上のルートドメインの仮想I/Oサービスを使って、I/Oアクセスを継続できるようにするためです。これにより、SPARC M10-4Sを切り離しても、ゲストドメインには影響を与えず、業務の運用を継続できるようになります。

次の図は、上記の条件を満たした構成例の概略図です。なお、各論理ドメインのI/O構成は簡略化しています。

図 A-3　2BB構成で運用継続可能な構成例（空きリソースあり）



以下の表では、2BB構成で制御ドメイン、ルートドメイン、ゲストドメインに割り当てるCPUコア、メモリ、I/O構成の例を示しています。
CPUコアとメモリについては、SPARC M10-4Sの切り離しの際に、各論理ドメインに

割り当てられたCPUコア数とメモリの合計は、1つのSPARC M10-4Sでまかなえるリソースに設定します。

表 A-8 2BB構成で空きハードウェアリソースがある構成の例

論理ドメイン名	primary	root-dom0	root-dom1	guest0	guest1	空き	合計
CPUコア	8	12	12	16	16	64	128（アクティベーション：64）
メモリ	14 GB(*1)	24 GB	24 GB	32 GB	32 GB	128 GB	256 GB
物理I/O（ルートコンプレックスの割り当て）	オンボード#0（PCIE0、PCIE4） オンボード#1（PCIE8、PCIE12）	PCIE1、PCIE2, PCIE3、PCIE5, PCIE6、PCIE7	PCIE9、PCIE10, PCIE11、PCIE13, PCIE14、PCIE15	—	—	—	—
仮想I/Oデバイス	—	vds0 vsw0、vsw1	vds1 vsw10、vsw11	vdisk0、vdisk10 vnet0、vnet10	vdisk1、vdisk11 vnet1、vnet11	—	—

*1: 上記の構成例では、制御ドメインに割り当てるメモリサイズは、ほかの論理ドメインに割り当てたメモリの残りとして割り当てる例として算出しています。
まず、1つのSPARC M10-4Sの物理的なメモリ容量（128 GB）から、制御ドメイン以外に割り当てたサイズを引いた残りサイズを求めると、16 GBになります。
128 GB − 32 GB − 32 GB − 24 GB − 24 GB ＝ 16 GB
実際に制御ドメインに割り当てられる最大サイズは、ここから、1つのSPARC M10-4S上のハイパーバイザに割り当てられる最大のメモリサイズ（2 GB）を引いたサイズ（14 GB）までになります。
16 GB − 2 GB ＝ 14 GB
さらにOracle VM Server for SPARCのバージョンが3.2よりも古い場合、物理パーティションの動的再構成を行うための条件として（コア数×256 MB）の倍数かどうかを確認すると、条件を満たしているため、14 GBとなります。
14 GB ÷ (8コア×256 MB) ＝ 7
割り切れるため、14 GBで問題ありません。

注－論理ドメインに割り当てるメモリサイズは、以下の2つの点について注意して設計する必要があります。詳細は、「2.4.1 論理ドメインの構築に関する留意点」を参照してください。
- 論理ドメインに割り当てられるメモリサイズは、物理的なメモリサイズより小さくなります。
- Oracle VM Server for SPARC のバージョンが3.2よりも古い場合は、物理パーティションの動的再構成を可能にするために、（コア数×256 MB）の倍数にしてください。

A.3.2 物理パーティションの構築手順例

1. **マスタXSCFにログインします。**
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の場合は、マスタXSCFへログインしなおしてください。

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

2. **メモリのミラーモードを設定します。**
   a. **showfruコマンドを実行し、SPARC M10-4Sのシステムボード（PSB<BB>）のメモリのミラーモード設定を確認します。**

      次の例では、物理システムボード（PSB 00-0）配下のデバイス設定情報を表示しています。

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
    cpu 00-0-0 no
    cpu 00-0-1 no
    cpu 00-0-2 no
    cpu 00-0-3 no
```

   b. **メモリのミラーモードを使用する場合、setupfruコマンドを使用し、メモリのミラーモードを設定します。**

      メモリのミラーモードを使用しない場合は、この手順は不要です。

      メモリのミラーモードの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「14.1　メモリをミラー構成にする」を参照してください。

      次の例では、PSB 00-0のすべてのCPUをミラーモードに設定しています。

```
XSCF> setupfru -m y sb 00-0
```

      showfruコマンドを実行し、メモリのミラーモード設定を確認します。

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
    cpu 00-0-0 yes
    cpu 00-0-1 yes
    cpu 00-0-2 yes
    cpu 00-0-3 yes
```

3. **物理パーティション構成情報を作成します。**
   a. **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
```

   b. **setpclコマンドを実行し、物理パーティション構成情報にシステムボードを登録します。**

setpclコマンドを実行し、SPARC M10-4Sのシステムボード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。

```
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。**

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                         Running
            00      00-0
            01      01-0
```

コンフィグレーションポリシー、IO無効化オプション（No-IO）、メモリ無効化オプション（No-Mem）の詳細を表示するには、-vオプションを指定します。

```
XSCF> showpcl -v -p 0
PPAR-ID   LSB    PSB    Status   No-Mem   No-IO    Cfg-policy
00                      Running
                                                  System
          00     00-0            False    False
          01     01-0            False    False
```

コンフィグレーションポリシー、IO無効化オプション（No-IO）、メモリ無効化オプション（No-Mem）の設定を変更する場合は、setpclコマンドを使用します。

setpclコマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

4. **物理パーティションにシステムボード（PSB<BB>）を割り当てます。**

a. **showboards -aコマンドを実行し、PSBの状態を確認します。**

showboards -aコマンドを実行し、それぞれのPSB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、PSB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、PSB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PSBの状態を確認します。

この例では、それぞれのPSB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、それぞれのPSBの [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
```

5. **CPUコア アクティベーションキーを登録し、CPUコアリソースを割り当てます。**

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。

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ません。

```
XSCF> showcodactivation
Index Description Count
------- ----------- ------
```

注ー登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。

---

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登録します。**

CPUコア アクティベーションキーを登録する方法の詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「5.3　CPUコアリソースを追加する」を参照してください。

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber:10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtxgvZPsrtYguqiNUieB+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            2
      1 PROC            2
      2 PROC            2
      3 PROC            2

--- 中略 ---

     30 PROC            2
     31 PROC            2
```

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コア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、64個のCPUコアリソース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 -c set 64
PROC Permits assigned for PPAR 0 : 0 -> 64

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注ーXCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

---

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コア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64個のCPUコアリソースが、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: 64
```

6. **resetdateoffsetコマンドを実行し、XSCFが管理する時刻の差分をリセットします。**

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
```

7. **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                  :00000002
Diagnostic Level         :min
Message Level            :normal
Alive Check              :on
Watchdog Reaction        :reset
Break Signal             :on
Autoboot(Guest Domain)   :on
Elastic Mode             :off
IOreconfigure            :false
PPAR DR(Current)         :-
PPAR DR(Next)            :on
```

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。

setpparmodeコマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

```
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
```

8. **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> poweron -p 0
```

9. **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
```

10. **Oracle SolarisとOracle VM Server for SPARCをインストールします。**
制御ドメインにOracle SolarisとOracle VM Server for SPARCをインストールし

ます。

物理パーティションの動的再構成に必要なOracle Solarisの版数や条件などについては、表 1-3および表 1-4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/en/）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.xシステムのインストール』

- Oracle VM Server for SPARC

  『Oracle VM Server for SPARC 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」

11. **論理ドメインを構築します。**

    ここでは、「図 A-3　2BB構成で運用継続可能な構成例（空きリソースあり）」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ほか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て、コンソールサービスを設定します。

    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。

```
# 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、構成例に従って、PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7、PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15を削除するコマンドの例の一部を示します。

```
# ldm remove-io PCIE1 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 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、構成例に従って制御ドメインのCPUコアを8個、メモリサイズを14 GBに設定する例を示します。

```
# ldm set-core 8 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-memory 14G 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、物理パーティションの動的再構成が可能なCPUコアとメモリの配置の条件を満たしやすくなります。CPUコアとメモリの配置に関する条件の詳細は、「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 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。

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

構成情報を保存してから、Oracle Solarisを再起動します。

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で保存されたことを確認して、最後に、Oracle Solarisを再起動する例を示しています。

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

```
ldm-set1 [current]
# shutdown -i6 -g0 -y
```

b. **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。**

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、ZFSでミラーリング構成にするコマンドの例を説明します。詳細は、Oracle社のホームページ（https://docs.oracle.com/en/）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の場合

『Oracle Solaris ZFS 管理ガイド』の「ミラー化されたZFSルートプールを作成する方法（インストール後）」

■ Oracle Solaris 11の場合

『Oracle Solaris 11.1の管理: ZFSファイルシステム』の「ミラー化ルートプールを構成する方法 （SPARCまたは x86/VTOC）」

その他の冗長構成ソフトウェアをご利用になる場合は、対象ソフトウェアのマニュアルを参照してください。

zpool statusコマンドを実行して、ルートプールの状態を確認します。

次の例では、デフォルトのルートプール（rpool）にc2t50000393E802CCE2d0s0が割り当てられていることを示しています。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

formatコマンドを実行して、追加可能なディスクを確認します。

次の例では、もう1つのディスクとしてc3t50000393A803B13Ed0が存在することがわかります。

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
```

zpool attachコマンドを実行して、rpoolに2本目のディスクを追加してミラー構成にします。

次の例では、zpool attachコマンドでc3t50000393A803B13Ed0s0を追加して、zpool statusコマンドで同期処理（resilver）の状況を確認しています。status、actionを見ると、同期処理中であることがわかるので、同期処理が完了するまで、定期的にzpool statusコマンドを実行して状況を確認します。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Wed Jan 29 21:35:39 2014
    3.93G scanned out of 70.6G at 71.9M/s, 0h15m to go
    3.90G resilvered, 5.56% done
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         DEGRADED     0     0     0
          mirror-0                    DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   DEGRADED     0     0     0  (resilvering)
```

同期処理が完了すると以下のように、 [state] が「ONLINE」となります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h10m with 0 errors on Wed Jan 29 21:45:42 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

c. ルートドメインを構成します。

ここでは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。

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

ldm set-variableコマンドを実行し、Oracle Solaris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Oracle

Solaris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。

```
# 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、構成例に従って、ldm set-coreコマンドでCPUコアを12、ldm set-memoryコマンドでメモリを24 GB割り当てています。

```
# ldm set-core 12 root-dom0
# ldm set-memory 24G root-dom0
```

注ー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、物理パーティションの動的再構成が可能なCPUコアとメモリの配置の条件を満たしやすくなります。CPUコアとメモリの配置に関する条件の詳細は、「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（vcons）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。

```
# 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、SPARC M10-4S(BB0)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、PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7が、BB0上の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。

```
# ldm list-io -l
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
(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
/BB0/PCI0                                   PCIE   PCIE1             EMP
[pci@8100/pci@4/pci@0/pci@0]
/BB0/PCI3                                   PCIE   PCIE2             EMP
[pci@8200/pci@4/pci@0/pci@0]
/BB0/PCI4                                   PCIE   PCIE2             EMP
[pci@8200/pci@4/pci@0/pci@8]
/BB0/PCI7                                   PCIE   PCIE3             EMP
[pci@8300/pci@4/pci@0/pci@0]
/BB0/PCI8                                   PCIE   PCIE3             EMP
[pci@8300/pci@4/pci@0/pci@8]
/BB0/CMUL/NET2                              PCIE   PCIE4    primary  OCC
[pci@8400/pci@4/pci@0/pci@a]
    network@0
    network@0,1
/BB0/PCI1                                   PCIE   PCIE5             EMP
[pci@8500/pci@4/pci@0/pci@8]
/BB0/PCI2                                   PCIE   PCIE5             EMP
[pci@8500/pci@4/pci@0/pci@9]
/BB0/PCI5                                   PCIE   PCIE6             EMP
[pci@8600/pci@4/pci@0/pci@9]
/BB0/PCI6                                   PCIE   PCIE6             EMP
[pci@8600/pci@4/pci@0/pci@11]
/BB0/PCI9                                   PCIE   PCIE7             EMP
[pci@8700/pci@4/pci@0/pci@9]
/BB0/PCI10                                  PCIE   PCIE7             EMP
[pci@8700/pci@4/pci@0/pci@11]
（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.6　SPARC M10-4S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ルートドメインに割り当てるかを決めます。

構成例の場合、上記で確認したBB0上の未割り当てのルートコンプレックスすべて（PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7）になるため、ldm add-ioコマンドを実行してroot-dom0に割り当てます。

次の例は、コマンド実行例です。

```
# ldm add-io PCIE1 root-dom0
# ldm add-io PCIE2 root-dom0
# ldm add-io PCIE3 root-dom0
# ldm add-io PCIE5 root-dom0
# ldm add-io PCIE6 root-dom0
# ldm add-io PCIE7 root-dom0
```

ldm bind-domainコマンドでルートドメインを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、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、PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7がroot-dom0に割り当てられていることが確認できます。

```
# ldm bind-domain root-dom0
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8    primary    IOV
（以下省略）
```

手順11.cに従って、ほかのSPARC M10-4Sのルートドメイン（例の場合、root-dom1）も同様に構成してください。

d. **ルートドメインにOracle Solarisをインストールします。**

ここでは、ルートドメインにOracle Solarisをインストールする手順を説明します。

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

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

ldm set-variableコマンドを実行し、Oracle Solaris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。

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

telnetコマンドを実行して、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。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  7h 7m
root-dom0        active     -t----  5000    24    24G      0.0%  20s
root-dom1        bound      ------  5001    24    24G
# telnet localhost 5000
....
{0} ok
```

ルートドメインにOracle Solarisをインストールします。

物理パーティションの動的再構成に必要なOracle Solarisの版数や条件などについては、表 1-3および表 1-4を参照してください。

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/en/）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.xシステムのインストール』

手順11.dに従って、ほかのルートドメイン（例の場合、root-dom1）も同様にインストールします。

e. **ルートドメインに仮想I/Oサービスを割り当てます。**

ここでは、ルートドメインに仮想I/Oサービスを設定する手順の例を説明します。

仮想ディスクサービス（vds）には、ルートドメインの物理ディスク全体を仮想ディスクとして貸し出す例を示します。

仮想スイッチネットワークデバイス（vsw）には、Ethernetカードのポートを割り当てる例を示します。

仮想I/Oサービスの詳細については、Oracle社の『Oracle VM Server for SPARC 管理ガイド』を参照してください。

ルートドメインにroot権限でログインします。

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

仮想ディスクサービス（vds）に割り当てるバックエンドデバイス（物理ディスク）を特定するために、formatコマンドを実行して、ディスクとそのデバイスパスを表示します。

cXtXXXXXXXXXXXXXXXXdXの形式の文字列がディスク、「/pci@」で始まる文字列がデバイスパスを示しています。

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c2t50000393A802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8100/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
       1. c3t50000393D8285226d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393d8285226,0
       2. c4t50000393A804B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@3200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
```

上記の結果で表示されるデバイスパスと、『SPARC M12/M10 システム運用・管理ガイド』の「A.6　SPARC M10-4Sのデバイスパス」を参照してディスクの物理位置を確認して、仮想ディスクサービスに割り当てるバックエンドのディスク（cXtXXXXXXXXXXXXXXXXdX）を決めます。

構成例として、2つのディスク（c3t50000393D8285226d0, c4t50000393A804B13Ed0）をそれぞれ、ゲストドメイン（guest0、guest1）の仮想ディスクのバックエンドとして割り当てるようにします。

次に、仮想スイッチサービス（vsw）に割り当てるネットワークインターフェースを特定するために、dladm show-physコマンドを実行して、PCIeスロットの位置（LOC）に対応するネットワークインターフェース名（LINK）を表示します。

表示結果から、ネットワークインターフェースの位置を確認して、仮想スイッチサービスに割り当てるネットワークインターフェースを決めます。

構成例として、2つのネットワークインターフェース（net1、net2）を、各ゲストドメインの仮想ネットワークに対応する仮想スイッチに割り当てるようにします。

```
# dladm show-phys -L
LINK         DEVICE       LOC
net0         igb0         BB#0-PCI#0
net1         igb1         BB#0-PCI#1
net2         igb2         BB#0-PCI#2
net3         igb3         BB#0-PCI#3
...
```

ldm add-vdiskserverコマンドを実行して、ルートドメインに仮想ディスクサービスを追加します。

次の例は、ldm add-vdiskserverコマンドでルートドメイン（root-dom0）に仮想ディスクサービス（vds0）を追加しています。

```
# ldm add-vdiskserver vds0 root-dom0
```

ldm add-vdiskserverdeviceコマンドを実行して、仮想ディスクのバックエンドをルートドメインからエクスポートします。

次の例は、ldm add-vdiskserverdeviceコマンドを実行して、2つバックエンドのディスクをvds0に割り当ててエクスポートしています。

```
# ldm add-vdiskserver vds0 root-dom0
# ldm add-vdiskserverdevice /dev/dsk/c3t50000393D8285226d0s2 vol0@
vds0
# ldm add-vdiskserverdevice /dev/dsk/c4t50000393A804B13Ed0s2 vol1@
vds0
```

ldm add-vswitchコマンドを実行して、ルートドメインに仮想スイッチを追加します。

次の例では、ゲストドメインごとに仮想スイッチを追加するため、vsw0、vsw1 を追加して、それぞれに異なる物理ネットワークインターフェース（net1、net2）を割り当てています。

```
# ldm add-vswitch net-dev=net1 vsw0 root-dom0
# ldm add-vswitch net-dev=net2 vsw1 root-dom0
```

同様に、BB1側のSPARC M10-4SのI/Oを割り当てたルートドメイン（root-dom1）にも仮想I/Oサービスを追加します。

root-dom1に割り当てる仮想ディスクサービス名、仮想スイッチ名の例として、以下のように割り当てます。手順は11.eを参照してください。

仮想ディスクサービス名：vds1（バックエンドのボリューム名はvol10、vol11の2つを割り当てる）

仮想スイッチ名　　：vsw10、vsw11

f. **ゲストドメインを構築します。**

ここでは、ゲストドメインの構築手順を説明します。

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。

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

ldm set-variableコマンドを実行し、Oracle Solaris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Oracle Solaris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。

```
# 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、構成例に従って、ldm set-coreコマンドでCPUコアを16、ldm set-memoryコマンドでメモリを32 GB割り当てています。

```
# ldm set-core 16 root-dom0
# ldm set-memory 32G 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、物理パーティションの動的再構成が可能なCPUコアとメモリの配置の条件を満たしやすくなります。CPUコアとメモリの配置に関する条件の詳細は、「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（vcons）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「5100」を仮想コンソールに割り当てています。

```
# ldm set-vconsole service=vcc0 port=5100 guest0
```

ldm add-vdiskコマンドを実行して、仮想ディスク（vdisk）を割り当てます。

次の例では、構成例に従って2つの仮想ディスク（vdisk0、vdisk10）を割り当てます。それぞれの仮想ディスクのバックエンドは、2つのルートドメイン（root-dom0、root-dom1）に追加したバックエンド（vds0のvol0、vds1のvol10）を指定しています。

```
# ldm add-vdisk vdisk0 vol0@vds0 guest0
# ldm add-vdisk vdisk10 vol10@vds1 guest0
```

ldm add-vnetコマンドを実行して、仮想ネットワークデバイス（vnet）を割り当てます。

次の例では、構成例に従って2の仮想ネットワークデバイス（vnet0、vnet10）を割り当てます。それぞれの仮想ネットワークデバイスに接続する仮想スイッチは、2つのルートドメイン（root-dom0、root-dom1）に追加した仮想スイッチ（vsw0、vsw10）を指定しています。

```
# ldm add-vnet vnet0 vsw0 guest0
# ldm add-vnet vnet10 vsw10 guest0
```

手順11.fに従って、ほかのゲストドメイン（例の場合、guest1）も同様に構築します。

構成例に従って割り当てるデバイスは、以下のようになります。

仮想ディスク　: vdisk1（バックエンドはvol1@vds0）、vdisk11（バックエンドはvol11@vds1）

仮想ネットワーク: vnet1（仮想スイッチはvsw1）、vnet11（仮想スイッチは

vsw11）

注－複数の仮想ディスクや仮想ネットワークデバイスを割り当てた場合、ldm list -lコマンドの実行結果から、仮想デバイスに割り当てたIDの値を記録してください。動的に削除したあとに、再度追加する際にIDを指定することで、仮想デバイスパスが変わらないようにするために必要になります。

---

g. **ゲストドメインにOracle Solarisをインストールします。**

ここでは、ゲストドメインにOracle Solarisをインストールする手順を説明します。

ldm bind-domainコマンドを実行してゲストドメインをバインドし、さらにstart-domainコマンドを実行してゲストドメインを起動します。

次の例は、guest0を起動するコマンドの実行例です。

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

ldm set-variableコマンドを実行し、Oracle Solaris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。

```
# ldm set-variable auto-boot¥?=true guest0
```

telnetコマンドを実行して、ゲストドメイン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「5100」であることを確認し、telnetコマンドでlocalhostのポート番号「5100」に接続し、guest0がOpenBoot PROMの状態で止まっていることが確認できます。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  8h 7m
guest0           active     -t----  5100    32    32G      0.0%  20s
root-dom0        active     -n--v-  5000    24    24G      0.0%  43s
root-dom1        active     -n--v-  5001    24    24G      0.0%  20s
guest1           inactive   ------          32    32G
# telnet localhost 5100
....
{0} ok
```

物理パーティションの動的再構成に必要なOracle Solarisの版数や条件などについては、表 1-3および表 1-4を参照してください。

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/en/）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.xシステムのインストール』

手順11.gに従って、ほかのゲストドメイン（例の場合、guest1）も同様にインストールします。

h. **ゲストドメインの仮想I/Oを冗長構成にします。**

ここでは、ゲストドメインguest0に割り当てた2つの仮想ネットワークインターフェース（vnet）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。

ゲストドメインguest0にログインします。

次の例では、ldm list-domainコマンドでguest0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「5100」に接続する実行例を示しています。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  8h 7m
guest0           active     -n----  5100    32    32G      0.0%  20s
guest1           active     -n----  5101    32    32G      0.0%  19s
root-dom0        active     -n--v-  5000    24    24G      0.0%  43s
root-dom1        active     -n--v-  5001    24    24G      0.0%  20s
# telnet localhost 5100
....
guest0 console login: root
Password:
...
guest0#
```

dladmコマンドを実行して、仮想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、net1として仮想ネットワークデバイスが参照できています。

```
guest0# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            up         0      unknown   vnet0
net1              Ethernet            up         0      unknown   vnet1
```

ipadm show-ifコマンドを実行して、net0、net1が表示されていないことを確

認します。

```
guest0# 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、net1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。

```
guest0# ipadm create-ip net0
guest0# ipadm create-ip net1
guest0# ipadm show-if
IFNAME     CLASS    STATE    ACTIVE OVER
lo0        loopback ok       yes    --
net0       ip       down     no     --
net1       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、net1をIPMPグループに追加します。

```
guest0# ipadm create-ipmp ipmp0
guest0# ipadm add-ipmp -i net0 -i net1 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。

```
guest0# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/
v4
guest0# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
ipmp0/v4          static   ok           xx.xx.xx.xx/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。

```
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net1        no      ipmp0       is-----   up        disabled  ok
net0        yes     ipmp0       --mbM--   up        disabled  ok
```

ほかのゲストドメイン（例の場合、guest1）も同様に設定を行います。

i. メモリリソースの割り当て状況を確認して分散したメモリブロックをまとめます。

まず、「A.2.2　物理パーティションの構築手順例」の手順11.h、手順11.iを参照して、分散しているメモリブロックをまとめます。

次の例は、構成例に従ったメモリ配分でldm list-devicesコマンドを実行した結果の例です。

論理ドメインごとにメモリブロックが1つだけ存在しており、256 MB～512 MB程度の分散されたメモリブロックが多数存在しないことが確認できます。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      24G             root-dom1
    0x700600000000      8G
    0x720000000000      32G             guest0
    0x740000000000      32G             guest1
    0x760000800000      1272M           _sys_
    0x760050000000      24G             root-dom0
    0x760650000000      6912M
    0x780000000000      32G
    0x7a0000000000      32G
    0x7c0000000000      32G
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      14G             primary
    0x7e0400000000      16G
```

j. **空きメモリブロックに余裕があるかどうかを確認して調整します。**

ここでは、SPARC M10-4Sの切り離しによって、論理ドメインのメモリブロックが空きメモリブロックに移動できるかどうかを確認して、調整する手順を説明します。

i. **メモリの連続領域（メモリブロック）の使用状況を確認します。**

prtdiagコマンドを実行し、メモリの物理アドレスとSPARC M10-4Sの対応を確認します。

```
# prtdiag
(略)
======================= Physical Memory Configuration ========================
Segment Table:
--------------------------------------------------------------
Base            Segment  Interleave   Bank     Contains
Address         Size     Factor       Size     Modules
--------------------------------------------------------------
0x7e0000000000 32 GB    4            8 GB     /BB0/CMUL/CMP0/MEM00A
(略)
0x7c0000000000 32 GB    4            8 GB     /BB0/CMUL/CMP1/MEM10A
(略)
0x7a0000000000 32 GB    4            8 GB     /BB0/CMUU/CMP0/MEM00A
(略)
0x780000000000 32 GB    4            8 GB     /BB0/CMUU/CMP1/MEM10A
(略)
```

```
0x760000000000 32 GB    4            8 GB     /BB1/CMUL/CMP0/MEM00A
（略）
0x740000000000 32 GB    4            8 GB     /BB1/CMUL/CMP1/MEM10A
（略）
0x720000000000 32 GB    4            8 GB     /BB1/CMUU/CMP0/MEM00A
（略）
0x700000000000 32 GB    4            8 GB     /BB1/CMUU/CMP1/MEM10A
（以下省略）
```

この例の結果を、メモリの物理アドレスを小さい順に並べ直して、物理アドレスとSPARC M10-4Sの対応を表にすると以下のようになります。

表 A-9　物理アドレスとSPARC M10-4Sの対応例

Base Address（物理アドレス）	ビルディングブロック構成中のSPARC M10-4S
0x700000000000～	BB1
0x720000000000～	BB1
0x740000000000～	BB1
0x760000000000～	BB1
0x780000000000～	BB0
0x7a0000000000～	BB0
0x7c0000000000～	BB0
0x7e0000000000～	BB0

手順11.iで表示した結果と、prtdiagコマンドで確認した物理位置を合わせると、以下のようなメモリブロックの使用状況であることがわかります。

表 A-10　メモリブロックの使用状況の例

SPARC M10-4S	物理アドレス	サイズ	論理ドメイン
BB1（交換対象）	0x700000000000	24 GB	root-dom1
	0x700600000000	8 GB	未割り当て
	0x720000000000	32 GB	guest0
	0x740000000000	32 GB	guest1
	0x760050000000	24 GB	root-dom0
	0x760650000000	6,912 MB	未割り当て
BB0	0x780000000000	32 GB	未割り当て
	0x7a0000000000	32 GB	未割り当て
	0x7c0000000000	32 GB	未割り当て
	0x7e0080000000	14 GB	primary
	0x7e0400000000	16 GB	未割り当て

ii. **移動元のメモリブロックのサイズと個数を確認します。**

メモリブロックの使用状況の確認結果を参照し、交換対象のSPARC M10-4S上で割り当てられているメモリブロック（以降、移動元のメモリブロックと呼びます）を確認します。

「表 A-10　メモリブロックの使用状況の例」の場合、BB1側で論理ドメインに割り当てられているメモリブロックとして、32 GB×2個（guest0、guest1に割り当て）、24 GB×1個（root-dom0）であることがわかります。

注－切り離されるSPARC M10-4SのI/Oを割り当てているルートドメイン（例えば、BB1を切り離す場合は、root-dom1）は、SPARC M10-4Sの切り離し時に、アンバインドしてinactive状態にするため、移動対象から除外します。

---

iii. **空きメモリブロックを確認します。**

次に、手順 i で確認した結果から、切り離されないSPARC M10-4S上の、論理ドメインに割り当てられていないメモリブロック（以降、空きメモリブロックと呼びます）を確認します。

「表 A-10　メモリブロックの使用状況の例」の場合、32 GB×3個、16 GB×1個が空きメモリブロックであることがわかります。

iv. **メモリブロックが移動可能かどうかを確認します。**

手順iiと手順iiiで確認した結果より、移動元のメモリブロックが空きメモリブロックに移動できるかどうかを確認します。

移動元のメモリブロックと同じかそれ以上のサイズの空きメモリブロックがあれば、移動可能です。

例えば「表 A-10　メモリブロックの使用状況の例」の場合、guest0（32 GB）、guest1（32 GB）、root-dom0（24 GB）の移動先として、32 GB×3個の空きメモリブロックがあるため、BB1を切り離せるメモリ配置であることがわかります。表にすると、「表 A-11　メモリブロックの移動先候補」のようになります。

この場合は、手順12へ進んでください。

表 A-11　メモリブロックの移動先候補

SPARC M10-4S	サイズ	論理ドメイン	移動先候補
BB1（交換対象）	24 GB	root-dom0	－
	8 GB	未割り当て	－
	32 GB	guest0	BB0側の32 GB
	32 GB	guest1	BB0側の32 GB
	24 GB	root-dom0	BB0側の32 GB
	6,912 MB	未割り当て	BB0側に32 GBのguest1のメモリがあるので、削除
BB0	32 GB	未割り当て	ここに移動
	32 GB	未割り当て	ここに移動

表 A-11　メモリブロックの移動先候補 (続き)

SPARC M10-4S	サイズ	論理ドメイン	移動先候補
	32 GB	未割り当て	ここに移動
	14 GB	primary	―
	16 GB	未割り当て	サイズ不足で移動先の候補にならない

v. **移動先がない場合、論理ドメインのメモリ割り当てを減らします。**

ここでは、移動先候補になる大きさの空きメモリブロックがない場合の対処方法を説明します。

「2.5.2　運用時の留意点」の「CPUコアとメモリの配置」に書かれている条件を満たさない場合、メモリブロックの移動に失敗します。

例えば、次の「表 A-12　移動先がないメモリブロックの配置例」の構成の場合、移動元のメモリブロックは、32 GB×1個と、24 GB×3個になります。いっぽう、移動先の空きメモリブロックは、32 GB×3個と、16 GB×1個です。

その結果、BB1の32 GBのメモリブロック1個（guest0）と、24 GBのメモリブロック2個（guest1、guest2、root-dom0のいずれか2つ）は移動可能です。

しかし、残りの移動先の空きメモリブロックは、16 GB×1個と、32 GBのメモリブロックに24 GBのメモリブロックが移動したあとに残る8 GB×2個になるため、24 GBのメモリブロックを割り当てられているguest1、guest2、root-dom0のどれか1つが移動できなくなります。

表 A-12　移動先がないメモリブロックの配置例

SPARC M10-4S	サイズ	論理ドメイン	移動先候補
BB1（交換対象）	24 GB	guest2	移動できない可能性あり
	8 GB	root-dom1	―
	32 GB	guest0	BB0側の32 GB
	32 GB	guest1	移動できない可能性あり
	24 GB	root-dom0	移動できない可能性あり
	6,912 MB	未割り当て	―
BB0	32 GB	未割り当て	ここに移動
	32 GB	未割り当て	guest1、guest2、root-dom0のどれか（24 GB）がここに移動して、8 GB残る
	32 GB	未割り当て	guest1、guest2、root-dom0のどれか（24 GB）がここに移動して、8 GB残る
	14 GB	primary	―
	16 GB	未割り当て	サイズ不足で移動先の候補にならない

この場合、移動できない可能性のある論理ドメインのメモリブロックを、移動先のメモリブロックのサイズに合わせて減らしておく必要があります。

上記の例の場合、論理ドメインguest1、guest2、root-dom0のうちどれか1つのメモリを24 GBから16 GB以下に変更します。

以下の例では、ゲストドメインguest2のメモリを16 GB以下に変更する手順の例を示します。

ldm list-domainコマンドで、論理ドメインの状態を確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.5%  40m
guest0           active     -n----  5100    32    32G      0.0%  26m
guest1           active     -n----  5101    32    24G      0.0%  26m
guest2           active     -n----  5102    12    24G      0.1%  1m
root-dom0        active     -n--v-  5000    24    24G      0.0%  26m
root-dom1        active     -n--v-  5001    16    8G       0.0%  26m
```

ldm stop-domainコマンドで停止し、ldm unbind-domainコマンドでinactive状態にします。

次の例では、ゲストドメインguest2をinactive状態にするコマンドの実行例を示しています。

```
# ldm stop-domain guest2
Ldom guest2 stopped
# ldm unbind-domain guest2
```

注ー仮想I/Oサービスを設定しているドメインを停止する場合、対応する仮想I/Oデバイスを割り当てているゲストドメインもあらかじめldm stop-domainコマンドで停止してください。

Oracle VM Server for SPARCのバージョンが3.2よりも古い場合は、ldm set-memoryコマンドで、対象の論理ドメインのメモリを再設定します。

ゲストドメインguest2には、6コア（12vcpu）割り当てられているため、16 GB（16,384 MB）以下で、（コア数×256 MB）の倍数になるようにする必要があります。

16384÷（6×256）＝10.66　切り捨てて10

したがって、ゲストドメインguest2には、（6×256 MB）×10＝15,360 MB（15 GB）のメモリを再設定します。

次の例は、ldm set-memoryコマンドでメモリを15 GBに再設定するコマンドの実行例です。

```
# ldm set-memory 15G guest2
```

ldm list-domainコマンドを実行して、変更したサイズのメモリに設定されたことを確認します。

次の例は、ldm list-domainコマンドを実行して、ゲストドメインguest2のメモリが15 GBに設定されたことを示しています。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.5%  40m
guest0           active     -n----  5100    32    32G      0.0%  26m
guest1           active     -n----  5101    32    24G      0.0%  26m
root-dom0        active     -n--v-  5000    24    24G      0.0%  26m
root-dom1        active     -n--v-  5001    16    8G       0.0%  26m
guest2           inactive   ------          12    15G
```

ldm bind-domain, ldm start-domainコマンドを実行して、論理ドメインをバインドして起動します。

次の例は、ldm bind-domain, ldm start-domainコマンドを実行して、ゲストドメインguest2を起動するコマンドの実行例です。

```
# ldm bind-domain guest2
# ldm start-domain guest2
LDom guest2 started
```

ldm list-domainコマンドを実行して、論理ドメインが起動されたことを確認します。

次の例は、ldm list-domainコマンドを実行して、ゲストドメインguest2が再設定されたメモリで起動されたことを示しています。

[STATE] が「active」、 [FLAGS] の文字列の左から2文字目が「n」になったことを確認してください。

```
# ldm bind-domain guest2 root-dom0
-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.5%  44m
guest0           active     -n----  5100    32    32G      0.0%  30m
guest1           active     -n----  5101    32    24G      0.0%  30m
guest2           active     -n----  5102    12    15G      0.1%  1m
root-dom0        active     -n----  5000    24    24G      0.0%  30m
root-dom1        active     -n----  5001    16    8G       0.0%  30m
```

ldm list-devices -a memoryコマンドを実行して、メモリブロックの配置が、「2.5.2　運用時の留意点」の「CPUコアとメモリの配置」に書かれている条件を満たしているかどうか確認してください。

次の例は、ldm list-devices -a memoryコマンドの実行例を示しています。

```
# ldm ls-devices -a mem
MEMORY
    PA                    SIZE            BOUND
```

```
(BB1側)
    0x700000000000       32G             guest0  →0x780000000000(32GB)へ移動
    0x720000000000       24G             root-dom0 →0x7a0000000000(32GB)へ移動
    0x720600000000       8G
    0x740000000000       8G              root-dom1 (対象外)
    0x740200000000       24G
    0x760000800000       1272M           _sys_
    0x760050000000       24G             guest1  →0x7c0000000000(32GB)へ移動
    0x760650000000       6912M
(BB0側)
    0x780000000000       32G
    0x7a0000000000       32G
    0x7c0000000000       32G
    0x7e0000800000       1272M           _sys_
    0x7e0050000000       512M            _sys_
    0x7e0070000000       256M            _sys_
    0x7e0080000000       14G             primary
    0x7e0400000000       15G             guest2
    0x7e07c0000000       1G
```

12. **構築した論理ドメイン構成情報をXSCFに保存します。**
    ldm set-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、すでにある構成情報と同じ名前で再度保存しています。

    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。

```
# 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。

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

    ldm add-spconfigコマンドを実行して、構成情報を保存しなおします。

```
# 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。

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

13. **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。

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。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  10h 7m
guest0           active     -n----  5100    32    32G      0.0%  2h 20s
guest1           active     -n----  5101    32    32G      0.0%  2h 5s
root-dom0        active     -n----  5000    24    24G      0.0%  2h 43s
root-dom1        active     -n----  5001    24    24G      0.0%  2h 20s
```

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。

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

A.3.3 活性交換手順例（Oracle VM Server for SPARC 3.1.xの場合）

ここでは、「図 A-3　2BB構成で運用継続可能な構成例（空きリソースあり）」に記載された2BB構成システムに対して、PPAR DRを使用して、BB#01のSPARC M10-4Sを活性交換する手順例を記載します。

注－活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

1. **マスタXSCFにログインします。**
   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の場合は、マスタXSCFへログインしなおしてください。

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

2. **showhardconfコマンドを実行し、交換するSPARC M10-4SのXSCFの [Status] が「Normal」であることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Locked;
    + System_Power:On; System_Phase:Cabinet Power On;
```

```
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
    + FRU-Part-Number: CA07361-D202 A1                        ;
    + Power_Supply_System: ;
    + Memory_Size:256 GB;
    CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
       + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
        + Memory_Size:128 GB; Type: A ;
        CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
            + Freq:3.000 GHz; Type:0x10;
            + Core:16; Strand:2;
  :
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
    + FRU-Part-Number: CA20393-B50X A2                        ;
    + Power_Supply_System: ;
    + Memory_Size:256 GB;
    CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
        + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
        + Memory_Size:128 GB; Type: A ;
 :
```

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

3. **showbbstatusコマンドを実行し、交換するSPARC M10-4SのXSCFがマスタでないことを確認します。**

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

交換するSPARC M10-4SがマスタXSCFの場合、switchscfコマンドを実行し、XSCFを切り替えます。

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

注ーXSCFの切り替え、およびXSCFの再起動が完了したことを確認したあとに、SPARC M10-4Sの切り離し操作を行ってください。

---

4. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

5. **論理ドメインの稼働状況やリソース使用状況を確認します。**

a. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

この例では、制御ドメイン、2つのルートドメイン、2つのゲストドメインが稼働しています。

すべてのドメインが「active」でOracle Solarisが動作状態、または「inactive」状態であることを確認します。OpenBoot PROM状態またはbound状態のドメインが存在すると、物理パーティションの動的再構成に失敗することがあります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  10h 7m
guest0           active     -n----  5100    32    32G      0.0%  2h 20s
guest1           active     -n----  5101    32    32G      0.0%  2h 5s
root-dom0        active     -n--v-  5000    24    24G      0.0%  2h 43s
root-dom1        active     -n--v-  5001    24    24G      0.0%  2h 20s
```

b. **-aオプションを指定してldm list-devicesコマンドを実行し、リソースの使用状況を確認します。**

次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID      %FREE    CPUSET
    0       0        (0, 1)
    4       0        (8, 9)
    8       0        (16, 17)
 （中略）
    944     0       (1888, 1889)
    948     0       (1896, 1897)
    952     0       (1904, 1905)
    956     0       (1912, 1913)
VCPU
    PID     %FREE    PM
    0       0        no
    1       0        no
    8       0        no
    9       0        no
 （中略）
    1904    0        no
    1905    0        no
    1912    0        no
```

```
    1913    0        no
（以下略）
```

6. **制御ドメインのシステムボリュームやI/Oデバイスの冗長構成を切り離します。**
ここでは、BB-ID#01のSPARC M10-4Sを切り離せるようにするために、制御ドメインで使用している、交換するSPARC M10-4SのI/Oデバイスを切り離す手順を示します。その他の冗長化ソフトウェアを使用している場合、構成を解除する手順の詳細は、各冗長化ソフトウェアのドキュメントを参照してください。

   a. **制御ドメインのシステムボリュームの冗長化を解除します。**

   次の例では制御ドメインのシステムボリュームのZFSミラー機能を解除する手順を示します。

   制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

   zpool detachコマンドを実行し、ディスクをミラー構成から切り離します。

```
# zpool detach rpool c3t50000393A803B13Ed0
```

   zpool statusコマンドで、ミラー構成が解除されたことを確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

   ほかにもBB#01上のデバイスを使用している場合、冗長構成を解除したり、デバイスの利用を停止したりしてください。冗長構成の解除方法やデバイスの利用停止方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

   b. **制御ドメインのI/O構成を削除します。**

制御ドメインに割り当てられている物理I/Oデバイスのうち、BB#01のルートコンプレックスを遅延再構成で削除します。

はじめに、制御ドメインを遅延再構成モードに移行します。

```
# 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 list-ioコマンドを実行し、primaryに割り当てられているルートコンプレックスを確認します。

次の例では、BB1のデバイスを持つルートコンプレックスがPCIE8とPCIE12であることがわかります。

```
# ldm list-io | grep primary
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE4                                    BUS    PCIE4    primary  IOV
PCIE8                                    BUS    PCIE8    primary  IOV
PCIE12                                   BUS    PCIE12   primary  IOV
/BB0/CMUL/NET0                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                         PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                           PCIE   PCIE4    primary  OCC
/BB1/CMUL/NET0                           PCIE   PCIE8    primary  OCC
/BB1/CMUL/SASHBA                         PCIE   PCIE8    primary  OCC
/BB1/CMUL/NET2                           PCIE   PCIE12   primary  OCC
```

ldm remove-ioコマンドを実行し、primaryからPCIE8とPCIE12を削除して、Oracle Solarisを再起動します。

```
# ldm remove-io PCIE8 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 remove-io PCIE12 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.
------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
....
```

Oracle Solarisが起動されたら、ldm list-ioコマンドを実行し、制御ドメインからBB#01のルートコンプレックスが削除されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE4                                    BUS    PCIE4    primary  IOV
```

```
/BB0/CMUL/NET0                              PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                            PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                              PCIE   PCIE4    primary  OCC
```

c. **ゲストドメインに割り当てている仮想I/Oデバイスの冗長構成を解除します。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）を事前にシャットダウンするために、各ゲストドメインにログインして、root-dom1からの仮想I/Oデバイスの冗長構成を解除します。

冗長構成ソフトウェアの使用方法については、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。

次の例では、仮想ネットワークデバイス（vnet1）をIPMPの構成から解除する例を示しています。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  4h 17m
guest0           active     -n----  5100    64    64G      0.0%  1h 13m
guest1           active     -n----  5101    64    64G      0.0%  1h 4m
root-dom0        active     -n--v-  5000    32    32G      0.0%  1h 47m
root-dom1        active     -n--v-  5001    32    32G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         0      unknown   vnet1
```

ipmpstat -iコマンドを実行し、IPMPを構成しているネットワークインターフェースの構成情報を確認します。

```
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       is-----   up        disabled  ok
```

if_mpadm -dコマンドを実行し、IPMPのグループからnet1を切り離して、ipmpstat -i コマンドを実行し、切り離されたことを確認します。次の例の場合、STATEがofflineになったことを確認します。

```
guest0# if_mpadm -d net1
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       -s---d-   up        disabled  offline
```

ipadm delete-ipコマンドを実行して、net1を削除します。

```
guest0# ipadm delete-ip net1
```

ゲストドメイン（guest1）についても、同様の切り離しを行います。

d. **停止するルートドメインから割り当てている仮想I/Oデバイスを削除します。**

ldm remove-vdiskコマンド、ldm remove-vnetコマンドを実行して、次の手順で停止するルートドメインから割り当てている仮想ディスク（vdisk）と仮想ネットワークデバイス（vnet）を削除します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている仮想ディスク（vdisk10）と仮想ネットワークデバイス（vnet10）を削除するコマンドの実行例を示しています。

ゲストドメイン（guest1）についても、同様の削除を行います。

```
# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0
```

7. **I/Oデバイスのリソース使用状況を確認し、交換するSPARC M10-4SのI/Oデバイスをすべて解除します。**

a. **切り離すSPARC M10-4Sのルートコンプレックスを割り当てている論理ドメインを確認します。**

ldm list-ioコマンドを実行し、BB#01のルートコンプレックスを割り当てている論理ドメインを確認します。

次の例では、root-dom1だけが「/BB1/」で始まるPCIeエンドポイントを所有しています。そして、そのPCIeエンドポイントのルートコンプレックス（BUS）である、PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15がroot-dom1に割り当てられていることがわかります。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
```

```
PCIE10                                    BUS    PCIE10   root-dom1  IOV
PCIE11                                    BUS    PCIE11   root-dom1  IOV
PCIE12                                    BUS    PCIE12
PCIE13                                    BUS    PCIE13   root-dom1  IOV
PCIE14                                    BUS    PCIE14   root-dom1  IOV
PCIE15                                    BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                            PCIE   PCIE8             UNK
/BB1/CMUL/SASHBA                          PCIE   PCIE8             UNK
/BB1/PCI0                                 PCIE   PCIE9    root-dom1OCC
/BB1/PCI3                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI4                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI7                                 PCIE   PCIE11   root-dom1OCC
/BB1/PCI8                                 PCIE   PCIE11   root-dom1OCC
/BB1/CMUL/NET2                            PCIE   PCIE12            UNK
/BB1/PCI1                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI2                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI5                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI6                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI9                                 PCIE   PCIE15   root-dom1OCC
/BB1/PCI10                                PCIE   PCIE15   root-dom1OCC
```

b. **切り離す**SPARC M10-4S**のルートコンプレックスを割り当てているルートドメインを停止して切り離します。**

次の例では、ldm stop-domainコマンド、ldm unbind-domainコマンドを実行し、ルートドメイン（root-dom1）を切り離し、inactive状態になったことが確認できます。

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.2%  4h 59m
guest0           active     -n----  5100    32    32G      0.0%  1h 55m
guest1           active     -n----  5101    32    32G      0.0%  1h 46m
root-dom0        active     -n--v-  5000    24    24G      0.0%  2h 29m
root-dom1        inactive   ------          24    24G
```

c. **交換する**SPARC M10-4S**のI/Oデバイスがすべて解除されたことを確認します。**

ldm list-ioコマンドを実行し、I/Oデバイスが切り離されたことを確認します。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN     STATUS
----                                      ----   ---      ------     ------
PCIE0                                     BUS    PCIE0    primary    IOV
PCIE1                                     BUS    PCIE1    root-dom0  IOV
PCIE2                                     BUS    PCIE2    root-dom0  IOV
PCIE3                                     BUS    PCIE3    root-dom0  IOV
PCIE4                                     BUS    PCIE4    primary    IOV
PCIE5                                     BUS    PCIE5    root-dom0  IOV
PCIE6                                     BUS    PCIE6    root-dom0  IOV
```

```
PCIE7                                       BUS     PCIE7     root-dom0   IOV
PCIE8                                       BUS     PCIE8
PCIE9                                       BUS     PCIE9
PCIE10                                      BUS     PCIE10
PCIE11                                      BUS     PCIE11
PCIE12                                      BUS     PCIE12
PCIE13                                      BUS     PCIE13
PCIE14                                      BUS     PCIE14
PCIE15                                      BUS     PCIE15
（以下省略）
```

8. **SPARC M10-4Sのシステムボード（PSB<BB>）を物理パーティションから切り離します。**

   a. **deleteboard -c disconnectコマンドを実行し、PSBを物理パーティションから切り離します。**

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

   b. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**

   終了値が0の場合、deleteboardコマンドは正常終了です。

   終了値が0以外の場合、およびdeleteboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.2 deleteboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

   c. **showboardsコマンドを実行し、PSBの状態を確認します。**

   交換するSPARC M10-4SのPSBが「Assigned」状態であり、 [Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

9. **replacefruコマンドを実行し、SPARC M10-4Sを交換します。**

```
XSCF> replacefru
```

注ーreplacefruコマンドを使用したSPARC M10-4Sの交換の詳細は、『SPARCM10-4/M10-4Sサービスマニュアル』の「5.8　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムから切り離す」および「6.2　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムに組み込む」を参照してください。

---

10. **PSBを物理パーティションに組み込みます。**

a. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換するSPARC M10-4SのPSBがAssigned状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

b. **addboard -c configureコマンドを実行し、PSBを物理パーティションに組み込みます。**

論理ドメイン構成を元の構成に戻すため、-m bind=resourceオプションを指定して、addboard -c configureコマンドを実行します。

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

注ーaddboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

---

c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

終了値が0の場合、addboardコマンドは正常終了です。

終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」

を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換したSPARC M10-4SのPSBが正常に組み込まれたことで、 [Conn] 欄と [Conf] 欄の表示がともに「y」と表示され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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

11. **論理ドメインの稼働状況を確認します。**

a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

b. **ldm list-domainコマンドを実行し、SPARC M10-4SのPSB（BB）の追加後に論理ドメインの稼働状況に変化がないことを確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  10h 30m
guest0           active     -n----  5100    32    32G      0.0%  2h 50s
guest1           active     -n----  5101    32    32G      0.0%  2h 30s
root-dom0        active     -n--v-  5000    24    24G      0.0%  3h 20s
root-dom1        inactive   ------          24    24G
```

12. **I/Oデバイスの使用を再開します。**

a. **ルートコンプレックスを再割り当てします。**

ldm bind-domain, ldm start-domain コマンドを実行し、交換したSPARC M10-4Sのルートコンプレックスを割り当てていたルートドメインを、unbind状態から起動します。

次の例では、unbind状態にしたルートドメイン（root-dom1）を起動して、起動されたことを確認しています。

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    16G      0.2%  3h 8m
guest0           active     -n----  5100    32    32G      0.0%  3h 8m
guest1           active     -n----  5101    32    32G      0.0%  3h 8m
root-dom0        active     -n--v-  5000    24    24G      0.0%  3h 8m
root-dom1        active     -n--v-  5001    24    24G      7.3%  8s
```

ldm list-ioコマンドを実行し、起動したルートドメインに物理I/Oデバイスが割り当てられていることを確認します。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN     STATUS
----                                      ----   ---      ------     ------
PCIE0                                     BUS    PCIE0    primary    IOV
PCIE1                                     BUS    PCIE1    root-dom0  IOV
PCIE2                                     BUS    PCIE2    root-dom0  IOV
PCIE3                                     BUS    PCIE3    root-dom0  IOV
PCIE4                                     BUS    PCIE4    primary    IOV
PCIE5                                     BUS    PCIE5    root-dom0  IOV
PCIE6                                     BUS    PCIE6    root-dom0  IOV
PCIE7                                     BUS    PCIE7    root-dom0  IOV
PCIE8                                     BUS    PCIE8
PCIE9                                     BUS    PCIE9    root-dom1  IOV
PCIE10                                    BUS    PCIE10   root-dom1  IOV
PCIE11                                    BUS    PCIE11   root-dom1  IOV
PCIE12                                    BUS    PCIE12
PCIE13                                    BUS    PCIE13   root-dom1  IOV
PCIE14                                    BUS    PCIE14   root-dom1  IOV
PCIE15                                    BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                            PCIE   PCIE8             UNK
/BB1/CMUL/SASHBA                          PCIE   PCIE8             UNK
/BB1/PCI0                                 PCIE   PCIE9    root-dom1OCC
/BB1/PCI3                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI4                                 PCIE   PCIE10   root-dom1OCC
/BB1/PCI7                                 PCIE   PCIE11   root-dom1OCC
/BB1/PCI8                                 PCIE   PCIE11   root-dom1OCC
/BB1/CMUL/NET2                            PCIE   PCIE12             UNK
/BB1/PCI1                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI2                                 PCIE   PCIE13   root-dom1OCC
/BB1/PCI5                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI6                                 PCIE   PCIE14   root-dom1OCC
/BB1/PCI9                                 PCIE   PCIE15   root-dom1OCC
/BB1/PCI10                                PCIE   PCIE15   root-dom1OCC
```

b. **ルートドメインからの仮想I/Oデバイスをゲストドメインに追加します。**

ldm add-vdiskコマンド、ldm add-vnetコマンドを実行して、起動したルートドメインの仮想I/Oサービスに対応した、仮想ディスク（vdisk）と、仮想ネットワークデバイス（vnet）を、各ゲストドメインに追加します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている、仮想ディスク（vdisk10）と、仮想ネットワークデバイス（vnet10）を追加するコマンドの実行例を示しています。

ゲストドメイン（guest1）についても、同様の追加を行います。

```
# ldm add-vdisk id=1 vdisk10 vol10@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

注－仮想I/Oデバイスを再度追加する場合、あらかじめ割り当てられたIDを指定する必要があります。IDは、仮想I/Oデバイスを削除する前の状態でldm list -lコマンドの実行結果から確認できます。

---

c. **ゲストドメインに割り当てている仮想I/Oデバイスを冗長構成に組み込みます。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）が起動されると、各ゲストドメインに対する仮想I/Oデバイスのサービスも起動されます。

各ゲストドメインにログインして、事前に解除しておいたroot-dom1からの仮想I/Oデバイスを冗長構成に組み込みます。冗長構成ソフトウェアの使用方法については、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。

ここでは、仮想ネットワークデバイス（vnet1）をIPMPの構成に組み込む例を説明します。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

まず、ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  4h 17m
guest0           active     -n----  5100    32    32G      0.0%  1h 13m
guest1           active     -n----  5101    32    32G      0.0%  1h 4m
root-dom0        active     -n----  5000    24    24G      0.0%  1h 47m
root-dom1        active     -n----  5001    24    24G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            up         0      unknown   vnet0
net1              Ethernet            up         0      unknown   vnet1
```

ipadm create-ipコマンド、ipadm set-ifpropコマンド、ipadm add-ipmpコマンドを実行し、net1をipmp0の待機デバイスとして登録します。

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

ipmpstat -iコマンドを実行し、IPMPを構成しているネットワークインターフェースのSTATEがokになっていることを確認します。

ゲストドメイン（guest1）についても、同様の対応を行います。

```
guest0# ipmpstat -i
INTERFACE ACTIVE GROUP FLAGS LINK PROBE STATE
net0 yes ipmp0 -smbM-- up disabled ok
net1 no ipmp0 -s---d- up disabled ok
```

13. **制御ドメインのシステムボリュームやI/Oデバイスを冗長構成に戻します。**
    a. **制御ドメインのルートコンプレックス構成を追加します。**

制御ドメインから事前に削除した、BB#01のルートコンプレックスを遅延再構成で追加します。

まず、制御ドメインを遅延再構成モードに移行します。

```
# 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 list-ioコマンドを実行し、割り当てられていないルートコンプレックスを確認します。

次の例では、BB1のデバイスを持つルートコンプレックスがPCIE8とPCIE12が割り当てられていないことがわかります。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN     STATUS
----                                      ----   ---      ------     ------
PCIE0                                     BUS    PCIE0    primary    IOV
PCIE1                                     BUS    PCIE1    root-dom0  IOV
PCIE2                                     BUS    PCIE2    root-dom0  IOV
PCIE3                                     BUS    PCIE3    root-dom0  IOV
PCIE4                                     BUS    PCIE4    primary    IOV
PCIE5                                     BUS    PCIE5    root-dom0  IOV
PCIE6                                     BUS    PCIE6    root-dom0  IOV
PCIE7                                     BUS    PCIE7    root-dom0  IOV
PCIE8                                     BUS    PCIE8
PCIE9                                     BUS    PCIE9    root-dom1  IOV
PCIE10                                    BUS    PCIE10   root-dom1  IOV
PCIE11                                    BUS    PCIE11   root-dom1  IOV
PCIE12                                    BUS    PCIE12
PCIE13                                    BUS    PCIE13   root-dom1  IOV
（以下省略）
```

ldm add-ioコマンドを実行し、primaryへPCIE8とPCIE12を追加して、Oracle Solarisを再起動します。

```
# ldm add-io PCIE8 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 add-io PCIE12 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.
--------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
....
```

Oracle Solarisが起動されたら、ldm list-ioコマンドを実行し、制御ドメインにBB#01のルートコンプレックスが追加されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE4                                    BUS    PCIE4    primary  IOV
PCIE8                                    BUS    PCIE8    primary
PCIE12                                   BUS    PCIE12   root-dom1
/BB0/CMUL/NET0                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                         PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                           PCIE   PCIE4    primary  OCC
```

b. **制御ドメインのシステムボリュームの冗長化を構成します。**

制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

次の例では制御ドメインのシステムボリュームのZFSミラー機能を構成する手順を示します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

zpool attachコマンドを実行し、ディスクをミラー構成に組み込みます。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

zpool statusコマンドで、ミラー構成が組み込まれたことを確認します。

なお、同期処理（resilver）が完了したかどうか、zpool statusコマンドで状態を確認してください。

次の例は、同期処理中の表示例です。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         DEGRADED     0     0     0
          mirror-0                    DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   DEGRADED     0     0     0  (resilvering)
errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を構築したり、デバイスの利用を再開したりしてください。冗長構成の構築方法やデバイスの利用再開方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

A.3.4 活性交換手順例（PCIeバスを動的に割り当て可能な場合）

ここでは、PCIeバスを動的に割り当て可能な環境（XCP 2240以降、かつ、Oracle VM Server for SPARC 3.2以降で、ルートドメインがOracle Solaris 11.2 SRU11.2.8.4.0以降）で、「図 A-3　2BB構成で運用継続可能な構成例（空きリソースあり）」に記載

された2BB構成システムに対して、PPAR DRを使用して、BB#01を活性交換する手順例を記載します。

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

1. **マスタXSCFにログインします。**
   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の場合は、マスタXSCFへログインしなおしてください。

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

2. **showhardconfコマンドを実行し、交換するSPARC M10-4SのXSCFの [Status] が「Normal」であることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Locked;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
           + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
            CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
                + Freq:3.000 GHz; Type:0x10;
                + Core:16; Strand:2;
     :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
     :
```

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

3. **showbbstatusコマンドを実行し、交換するSPARC M10-4SのXSCFがマスタでないことを確認します。**

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

交換するSPARC M10-4SがマスタXSCFの場合、switchscfコマンドを実行し、XSCFを切り替えます。

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

注ーXSCFの切り替え、およびXSCFの再起動が完了したことを確認したあとに、システムボードの切り離し操作を行ってください。

---

4. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

5. **論理ドメインの稼働状況やリソース使用状況を確認します。**

a. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**

論理ドメインの稼働状況は、 [STATE] が「active」であり、FLAGSの文字列のうち、左から2つ目の文字を組み合わせて確認します。表示される文字と意味は以下のとおりです

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（ [STATE] が「active」以外のときも含む）

この例では、制御ドメイン、2つのルートドメイン、2つのゲストドメインが稼働しています。

すべてのドメインが「active」でOracle Solarisが動作状態、または「inactive」状態であることを確認します。OpenBoot PROM状態またはbound状態のドメインが存在すると、物理パーティションの動的再構成に失敗することがあります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  10h 7m
guest0           active     -n----  5100    32    32G      0.0%  2h 20s
guest1           active     -n----  5101    32    32G      0.0%  2h 5s
root-dom0        active     -n--v-  5000    24    24G      0.0%  2h 43s
root-dom1        active     -n--v-  5001    24    24G      0.0%  2h 20s
```

b. **-aオプションを指定してldm list-devicesコマンドを実行し、リソースの使用**

状況を確認します。

次の例では、-aオプションを指定して、論理ドメインにバインドされているリソースとバインドされていないリソースをすべて表示しています。

```
# ldm list-devices -a
CORE
    ID       %FREE    CPUSET
    0        0        (0, 1)
    4        0        (8, 9)
    8        0        (16, 17)
 （中略）
    944      0       (1888, 1889)
    948      0       (1896, 1897)
    952      0       (1904, 1905)
    956      0       (1912, 1913)
VCPU
    PID      %FREE    PM
    0        0        no
    1        0        no
    8        0        no
    9        0        no
 （中略）
    1904     0       no
    1905     0       no
    1912     0       no
    1913     0       no
 （以下略）
```

6. **制御ドメインのシステムボリュームやI/Oデバイスの冗長構成を切り離します。**
ここでは、ビルディングブロックBB-ID#01を切り離せるようにするために、制御ドメインで使用している、交換するSPARC M10-4SのI/Oデバイスを切り離す手順を示します。その他の冗長化ソフトウェアを使用している場合、構成を解除する手順の詳細は、各冗長化ソフトウェアのドキュメントを参照してください。

   a. **制御ドメインのシステムボリュームの冗長化を解除します。**

   次の例では制御ドメインのシステムボリュームのZFSミラー機能を解除する手順を示します。

   制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

zpool detachコマンドを実行し、ディスクをミラー構成から切り離します。

```
# zpool detach rpool c3t50000393A803B13Ed0
```

zpool statusコマンドで、ミラー構成が解除されたことを確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を解除したり、デバイスの利用を停止したりしてください。冗長構成の解除方法やデバイスの利用停止方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

b. **制御ドメインのI/O構成を削除します。**

制御ドメインに割り当てられている物理I/Oデバイスのうち、BB#01のルートコンプレックスを削除します。

ldm list-ioコマンドを実行し、primaryに割り当てられているルートコンプレックスを確認します。

次の例では、BB1のデバイスを持つルートコンプレックスがPCIE8とPCIE12であることがわかります。

```
# ldm list-io | grep primary
PCIE0                                     BUS    PCIE0    primary  IOV
PCIE4                                     BUS    PCIE4    primary  IOV
PCIE8                                     BUS    PCIE8    primary  IOV
PCIE12                                    BUS    PCIE12   primary  IOV
/BB0/CMUL/NET0                            PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                          PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                            PCIE   PCIE4    primary  OCC
/BB1/CMUL/NET0                            PCIE   PCIE8    primary  OCC
/BB1/CMUL/SASHBA                          PCIE   PCIE8    primary  OCC
/BB1/CMUL/NET2                            PCIE   PCIE12   primary  OCC
```

ldm remove-ioコマンドを実行し、primaryからPCIE8とPCIE12を削除します。

```
# ldm remove-io PCIE8 primary
# ldm remove-io PCIE12 primary
```

ldm list-ioコマンドを実行し、制御ドメインからBB#01のルートコンプレック

スが削除されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE4                                    BUS    PCIE4    primary  IOV
/BB0/CMUL/NET0                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                         PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                           PCIE   PCIE4    primary  OCC
```

c. **ゲストドメインに割り当てている仮想I/Oデバイスの冗長構成を解除します。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）を事前にシャットダウンするために、各ゲストドメインにログインして、root-dom1からの仮想I/Oデバイスの冗長構成を解除します。

冗長構成ソフトウェアの使用方法については、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。

次の例では、仮想ネットワークデバイス（vnet1）をIPMPの構成から解除する例を示しています。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  4h 17m
guest0           active     -n----  5100    64    64G      0.0%  1h 13m
guest1           active     -n----  5101    64    64G      0.0%  1h 4m
root-dom0        active     -n--v-  5000    32    32G      0.0%  1h 47m
root-dom1        active     -n--v-  5001    32    32G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            up         0      unknown   vnet0
net1              Ethernet            up         0      unknown   vnet1
```

ipmpstat -i コマンドを実行し、IPMPを構成しているネットワークインターフェースの構成情報を確認します。

```
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       is-----   up        disabled  ok
```

if_mpadm -d コマンドを実行し、IPMPのグループからnet1を切り離して、

ipmpstat -i コマンドを実行し、切り離されたことを確認します。次の例の場合、STATEがofflineになったことを確認します。

```
guest0# if_mpadm -d net1
guest0# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net1        no      ipmp0       -s---d-   up        disabled  offline
```

ipadm delete-ipコマンドを実行して、net1を削除します。

```
guest0# ipadm delete-ip net1
```

ゲストドメイン（guest1）についても、同様の切り離しを行います。

d. **停止するルートドメインから割り当てている仮想I/Oデバイスを削除します。**

ldm remove-vdiskコマンド、ldm remove-vnetコマンドを実行して、次の手順で停止するルートドメインから割り当てている仮想ディスク（vdisk）と仮想ネットワークデバイス（vnet）を削除します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている仮想ディスク（vdisk10）と仮想ネットワークデバイス（vnet10）を削除するコマンドの実行例を示しています。

```
# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0
```

ゲストドメイン（guest1）についても、同様の削除を行います。

7. **I/Oデバイスのリソース使用状況を確認し、交換するSPARC M10-4SのI/Oデバイスをすべて解除します。**

a. **切り離すSPARC M10-4Sのルートコンプレックスを割り当てている論理ドメインを確認します。**

ldm list-ioコマンドを実行し、BB#01のルートコンプレックスを割り当てている論理ドメインを確認します。

次の例では、root-dom1だけが「/BB1/」で始まるPCIeエンドポイントを所有しています。そして、そのPCIeエンドポイントのルートコンプレックス（BUS）である、PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15がroot-dom1に割り当てられていることがわかります。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN     STATUS
----                                      ----   ---      ------     ------
PCIE0                                     BUS    PCIE0    primary    IOV
PCIE1                                     BUS    PCIE1    root-dom0  IOV
PCIE2                                     BUS    PCIE2    root-dom0  IOV
PCIE3                                     BUS    PCIE3    root-dom0  IOV
PCIE4                                     BUS    PCIE4    primary    IOV
PCIE5                                     BUS    PCIE5    root-dom0  IOV
PCIE6                                     BUS    PCIE6    root-dom0  IOV
```

```
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
PCIE14                                      BUS    PCIE14   root-dom1  IOV
PCIE15                                      BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                              PCIE   PCIE8               UNK
/BB1/CMUL/SASHBA                            PCIE   PCIE8               UNK
/BB1/PCI0                                   PCIE   PCIE9    root-dom1  OCC
/BB1/PCI3                                   PCIE   PCIE10   root-dom1  OCC
/BB1/PCI4                                   PCIE   PCIE10   root-dom1  OCC
/BB1/PCI7                                   PCIE   PCIE11   root-dom1  OCC
/BB1/PCI8                                   PCIE   PCIE11   root-dom1  OCC
/BB1/CMUL/NET2                              PCIE   PCIE12              UNK
/BB1/PCI1                                   PCIE   PCIE13   root-dom1  OCC
/BB1/PCI2                                   PCIE   PCIE13   root-dom1  OCC
/BB1/PCI5                                   PCIE   PCIE14   root-dom1  OCC
/BB1/PCI6                                   PCIE   PCIE14   root-dom1  OCC
/BB1/PCI9                                   PCIE   PCIE15   root-dom1  OCC
/BB1/PCI10                                  PCIE   PCIE15   root-dom1  OCC
```

b. **切り離す SPARC M10-4S のルートコンプレックスを割り当てているルートドメインを停止して切り離します。**

次の例では、ldm stop-domainコマンド、ldm unbind-domainコマンドを実行し、ルートドメイン（root-dom1）を切り離し、inactive状態になったことが確認できます。

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.2%  4h 59m
guest0           active     -n----  5100    32    32G      0.0%  1h 55m
guest1           active     -n----  5101    32    32G      0.0%  1h 46m
root-dom0        active     -n--v-  5000    24    24G      0.0%  2h 29m
root-dom1        inactive   ------          24    24G
```

c. **交換する SPARC M10-4S の I/O デバイスがすべて解除されたことを確認します。**

ldm list-ioコマンドを実行し、I/Oデバイスが切り離されたことを確認します。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
```

```
PCIE4                                       BUS     PCIE4     primary     IOV
PCIE5                                       BUS     PCIE5     root-dom0   IOV
PCIE6                                       BUS     PCIE6     root-dom0   IOV
PCIE7                                       BUS     PCIE7     root-dom0   IOV
PCIE8                                       BUS     PCIE8
PCIE9                                       BUS     PCIE9
PCIE10                                      BUS     PCIE10
PCIE11                                      BUS     PCIE11
PCIE12                                      BUS     PCIE12
PCIE13                                      BUS     PCIE13
PCIE14                                      BUS     PCIE14
PCIE15                                      BUS     PCIE15
（以下省略）
```

8. **SPARC M10-4Sのシステムボード（PSB<BB>）を物理パーティションから切り離します。**

   a. **deleteboard -c disconnectコマンドを実行し、PSBを物理パーティションから切り離します。**

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

   b. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**

   終了値が0の場合、deleteboardコマンドは正常終了です。

   終了値が0以外の場合、およびdeleteboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.2 deleteboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

   c. **showboardsコマンドを実行し、PSBの状態を確認します。**

   交換するSPARC M10-4SのPSBが「Assigned」状態であり、 [Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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     y    y    y    Passed   Normal
01-0 00(01)       Assigned     n    n    n    Passed   Normal
```

9. **replacefruコマンドを実行し、SPARC M10-4Sを交換します。**

```
XSCF> replacefru
```

注－replacefruコマンドを使用したSPARC M10-4Sの交換の詳細は、『SPARCM10-4/M10-4Sサービスマニュアル』の「5.8　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムから切り離す」および「6.2　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムに組み込む」を参照してください。

---

10. **PSBを物理パーティションに組み込みます。**

a. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換したSPARC M10-4SのPSBがAssigned状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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     y    y    y    Passed   Normal
01-0 00(01)       Assigned     n    n    n    Passed   Normal
```

b. **addboard -c configureコマンドを実行し、PSBを物理パーティションに組み込みます。**

論理ドメイン構成を元に戻すため、-m bind=resourceオプションを指定して、addboard -c configureコマンドを実行します。

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

注－addboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

---

c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

終了値が0の場合、addboardコマンドは正常終了です。

終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換したSPARC M10-4SのPSBが正常に組み込まれたことで、 [Conn] 欄と [Conf] 欄の表示がともに「y」と表示され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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

11. **論理ドメインの稼働状況を確認します。**

a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

b. **ldm list-domainコマンドを実行し、SPARC M10-4SのPSB（BB）の組み込み後に論理ドメインの稼働状況に変化がないことを確認します。**

論理ドメインの稼働状況は、 [STATE] が「active」であり、 [FLAGS] の文字列のうち、左から2つ目の文字を組み合わせて確認します。表示される文字と意味は以下のとおりです。

「n」：Oracle Solarisが動作中

「t」：OpenBoot PROMの状態

「-」：その他の状態（ [STATE] が「active」以外のときも含む）

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  10h 30m
guest0           active     -n----  5100    32    32G      0.0%  2h 50s
guest1           active     -n----  5101    32    32G      0.0%  2h 30s
root-dom0        active     -n--v-  5000    24    24G      0.0%  3h 20s
root-dom1        inactive   ------          24    24G
```

12. **I/Oデバイスの使用を再開します。**

a. **ルートコンプレックスを再割り当てします。**

ldm bind-domain, ldm start-domain コマンドを実行し、交換したSPARC M10-4Sのルートコンプレックスを割り当てていたルートドメインを、unbind状態から起動します。

次の例では、unbind状態にしたルートドメイン（root-dom1）を起動して、起動されたことを確認しています。

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    16G      0.2%  3h 8m
guest0           active     -n----  5100    32    32G      0.0%  3h 8m
guest1           active     -n----  5101    32    32G      0.0%  3h 8m
root-dom0        active     -n--v-  5000    24    24G      0.0%  3h 8m
root-dom1        active     -n--v-  5001    24    24G      7.3%  8s
```

ldm list-ioコマンドを実行し、起動したルートドメインに物理I/Oデバイスが割り当てられていることを確認します。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN     STATUS
----                                      ----   ---      ------     ------
PCIE0                                     BUS    PCIE0    primary    IOV
PCIE1                                     BUS    PCIE1    root-dom0  IOV
PCIE2                                     BUS    PCIE2    root-dom0  IOV
PCIE3                                     BUS    PCIE3    root-dom0  IOV
PCIE4                                     BUS    PCIE4    primary    IOV
PCIE5                                     BUS    PCIE5    root-dom0  IOV
PCIE6                                     BUS    PCIE6    root-dom0  IOV
PCIE7                                     BUS    PCIE7    root-dom0  IOV
PCIE8                                     BUS    PCIE8
PCIE9                                     BUS    PCIE9    root-dom1  IOV
PCIE10                                    BUS    PCIE10   root-dom1  IOV
PCIE11                                    BUS    PCIE11   root-dom1  IOV
PCIE12                                    BUS    PCIE12
PCIE13                                    BUS    PCIE13   root-dom1  IOV
PCIE14                                    BUS    PCIE14   root-dom1  IOV
PCIE15                                    BUS    PCIE15   root-dom1  IOV
....
/BB1/CMUL/NET0                            PCIE   PCIE8               UNK
/BB1/CMUL/SASHBA                          PCIE   PCIE8               UNK
/BB1/PCI0                                 PCIE   PCIE9    root-dom1  OCC
/BB1/PCI3                                 PCIE   PCIE10   root-dom1  OCC
/BB1/PCI4                                 PCIE   PCIE10   root-dom1  OCC
/BB1/PCI7                                 PCIE   PCIE11   root-dom1  OCC
/BB1/PCI8                                 PCIE   PCIE11   root-dom1  OCC
/BB1/CMUL/NET2                            PCIE   PCIE12              UNK
/BB1/PCI1                                 PCIE   PCIE13   root-dom1  OCC
/BB1/PCI2                                 PCIE   PCIE13   root-dom1  OCC
/BB1/PCI5                                 PCIE   PCIE14   root-dom1  OCC
/BB1/PCI6                                 PCIE   PCIE14   root-dom1  OCC
```

```
/BB1/PCI9                                   PCIE    PCIE15    root-dom1   OCC
/BB1/PCI10                                  PCIE    PCIE15    root-dom1   OCC
```

b. **ルートドメインからの仮想I/Oデバイスをゲストドメインに追加します。**

ldm add-vdiskコマンド、ldm add-vnetコマンドを実行して、起動したルートドメインの仮想I/Oサービスに対応した、仮想ディスク（vdisk）と、仮想ネットワークデバイス（vnet）を、各ゲストドメインに追加します。

次の例では、BB#01のルートドメイン（root-dom1）の仮想I/Oサービスを使っている、仮想ディスク（vdisk10）と、仮想ネットワークデバイス（vnet10）を追加するコマンドの実行例を示しています。

```
# ldm add-vdisk id=1 vdisk10 vol10@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

注－仮想I/Oデバイスを再度追加する場合、あらかじめ割り当てられたIDを指定する必要があります。IDは、仮想I/Oデバイスを削除する前の状態でldm list -lコマンドの実行結果から確認できます。

ゲストドメイン（guest1）についても、同様の追加を行います。

c. **ゲストドメインに割り当てている仮想I/Oデバイスを冗長構成に組み込みます。**

BB#01のルートコンプレックスを割り当てているルートドメイン（root-dom1）が起動されると、各ゲストドメインに対する仮想I/Oデバイスのサービスも起動されます。

各ゲストドメインにログインして、事前に解除しておいたroot-dom1からの仮想I/Oデバイスを冗長構成に組み込みます。冗長構成ソフトウェアの使用方法については、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。

ここでは、仮想ネットワークデバイス（vnet1）をIPMPの構成に組み込む例を説明します。コマンドの詳細は、Oracle Solarisのマニュアルを参照してください。

まず、ゲストドメイン（guest0）にログインします。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  4h 17m
guest0           active     -n----  5100    32    32G      0.0%  1h 13m
guest1           active     -n----  5101    32    32G      0.0%  1h 4m
root-dom0        active     -n--v-  5000    24    24G      0.0%  1h 47m
root-dom1        active     -n--v-  5001    24    24G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

dladm show-physコマンドを実行し、仮想ネットワークインターフェース（vnet1）とネットワークインターフェース名（net1）の対応を確認します。

```
guest0# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         0      unknown   vnet1
```

ipadm create-ipコマンド、ipadm set-ifpropコマンド、ipadm add-ipmpコマンドを実行し、net1をipmp0の待機デバイスとして登録します。

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

ipmpstat -iコマンドを実行し、IPMPを構成しているネットワークインターフェースのSTATEがokになっていることを確認します。

```
guest0# ipmpstat -i
INTERFACE ACTIVE GROUP FLAGS   LINK PROBE    STATE
net0      yes    ipmp0 -smbM-- up   disabled ok
net1      no     ipmp0 -s---d- up   disabled ok
```

ゲストドメイン（guest1）についても、同様の対応を行います。

13. **制御ドメインのシステムボリュームやI/Oデバイスを冗長構成に戻します。**
    a. **制御ドメインのルートコンプレックス構成を追加します。**

    制御ドメインから事前に削除した、BB#01のルートコンプレックスを追加します。

    ldm list-ioコマンドを実行し、割り当てられていないルートコンプレックスを確認します。

    次の例では、BB1のデバイスを持つルートコンプレックスにPCIE8とPCIE12が割り当てられていないことがわかります。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9    root-dom1  IOV
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
（以下省略）
```

ldm add-ioコマンドを実行し、primaryへPCIE8とPCIE12を追加します。

```
# ldm add-io PCIE8 primary
# ldm add-io PCIE12 primary
```

ldm list-ioコマンドを実行し、制御ドメインにBB#01のルートコンプレックスが追加されていることを確認します。

```
# ldm list-io | grep primary
PCIE0                                    BUS    PCIE0    primary  IOV
PCIE4                                    BUS    PCIE4    primary  IOV
PCIE8                                    BUS    PCIE8    primary  IOV
PCIE12                                   BUS    PCIE12   primary  IOV
/BB0/CMUL/NET0                           PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                         PCIE   PCIE0    primary  OCC
/BB0/CMUL/NET2                           PCIE   PCIE4    primary  OCC
（以下省略）
```

b. **制御ドメインのシステムボリュームの冗長化を構成します。**

制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

次の例では制御ドメインのシステムボリュームのZFSミラー機能を構成する手順を示します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
        NAME                       STATE     READ WRITE CKSUM
        rpool                      ONLINE       0     0     0
          c2t50000393E802CCE2d0s0  ONLINE       0     0     0
errors: No known data errors
```

zpool attachコマンドを実行し、ディスクをミラー構成に組み込みます。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

zpool statusコマンドで、ミラー構成が組み込まれたことを確認します。

なお、同期処理（resilver）が完了したかどうか、zpool statusコマンドで状態を確認してください。

次の例は、同期処理中の表示例です。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         DEGRADED     0     0     0
          mirror-0                    DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   DEGRADED     0     0     0  (resilvering)
errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を構築したり、デバイスの利用を再開したりしてください。冗長構成の構築方法やデバイスの利用再開方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

A.4 XCP 2220以降の版数で新規に導入したシステムを1BBから2BB構成に増設する場合

ここでは、物理パーティションが動作した状態で物理パーティションの動的再構成により、1BB構成のSPARC M10-4Sを2BB構成に増設する環境構築例と、その構築手順を説明します。実施する例にして説明します。
なお、1BB構成のSPARC M10-4Sには、物理パーティションの動的再構成を使用するための、XCPファームウェアやOracle Solaris関連ソフトウェアの条件、およびXSCF

の設定などの、システム構築が完了していることを前提としています。

ここでの説明は、SPARC M12でも同様です。

A.4.1 構成例

ここでは、1BB構成のSPARC M10-4Sを、物理パーティションの動的再構成により2BB構成に増設して、ゲストドメインguest1、および増設したSPARC M10-4S（BB#1）のルートコンプレックスを割り当てたルートドメインroot-dom1を追加します。各ゲストドメインの仮想I/Oを冗長構成にして、さらに制御ドメインのシステムボリュームを内蔵ハードディスクでミラーリングした構成例を記載しています。

図 A-4　1BB構成時（増設前）の構成例

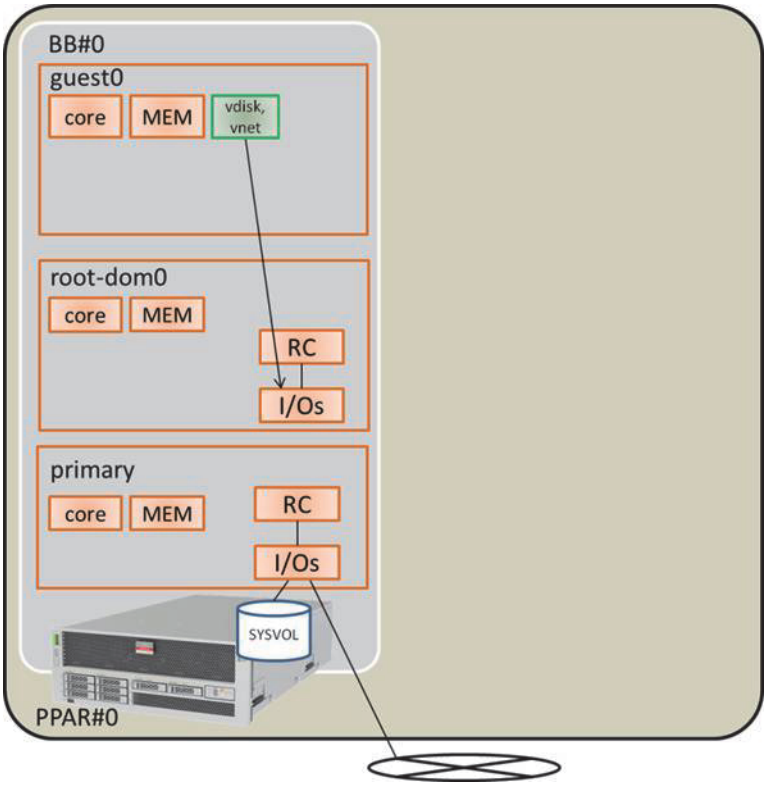


図 A-5　1BBから2BBへ増設後の構成例

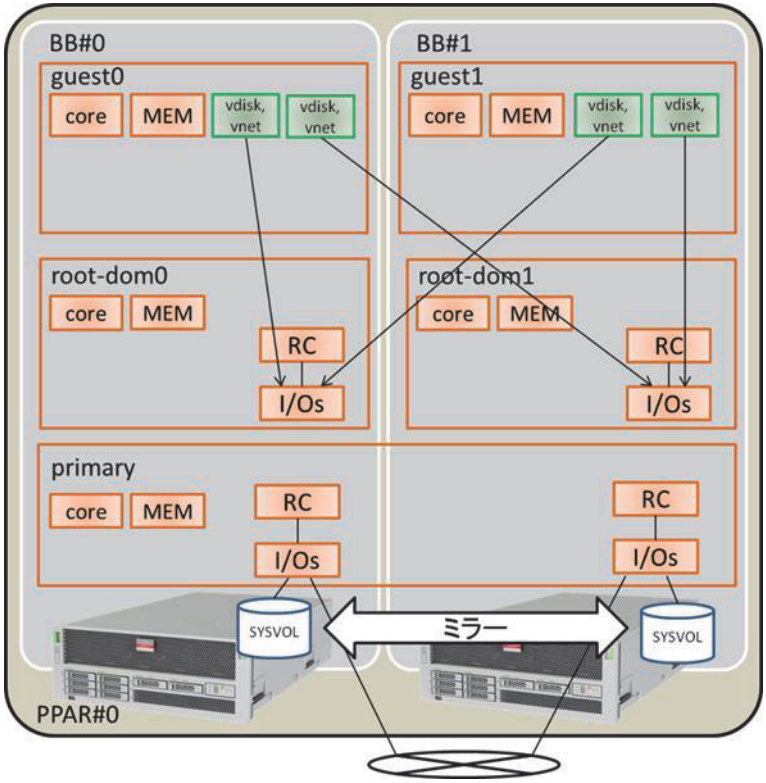


表 A-13　1BBから2BBへ増設する構成例

論理ドメイン	増設前			増設後		
	CPUコア	メモリ	I/O構成	CPUコア	メモリ	I/O構成
制御ドメイン（primary）	16	28 GB	オンボード#0（PCIE0、PCIE4）	32	56 GB	オンボード#0（PCIE0、PCIE4） オンボード#1（PCIE8、PCIE12）
guest0	32	64 GB	vdisk0、vnet0	32	64 GB	vdisk0、vdisk10 vnet0、vnet10
guest1	-	-	-	32	64 GB	vdisk1、vdisk11 vnet1、vnet11
root-dom0	16	32 GB	PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7、vds0、vsw0	16	32 GB	PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7、vds0、vsw0
root-dom1	-	-	-	16	32 GB	PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15、vds1、vsw1

表 A-13　1BBから2BBへ増設する構成例 (続き)

論理ドメイン	増設前			増設後		
	CPUコア	メモリ	I/O構成	CPUコア	メモリ	I/O構成
未割り当てのリソース	0	約2 GB (*1)	-	0	約4.75 GB(*1)	-
合計	64	128 GB	-	128	256 GB	-

*1:　各SPARC M10-4S上の約2 GBまたは約1.25 GBのメモリ領域はハイパーバイザのために割り当てられるため、論理ドメインに割り当て可能なメモリリソースは、物理的に搭載されたメモリ容量よりも少なくなります。

A.4.2　増設手順（Oracle VM Server for SPARC 3.1.xの場合）

「図 A-5　1BBから2BBへ増設後の構成例」に記載される構成例に合わせ、1BB構成のSPARC M10-4Sを、動的再構成により2BB構成に増設する手順は次のとおりです。

1. **マスタXSCFにログインします。**
   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。

   スタンバイ状態のXSCFの場合は、マスタXSCFへログインしなおしてください。

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

2. **showsscpコマンドを実行し、サービスプロセッサ間通信プロトコル（SSCP）のIPアドレスがデフォルト値か、ユーザーの設定した値か確認します。**

```
XSCF> showsscp
```

   IPアドレスがデフォルト値で、増設するSPARC M10-4SのIPアドレスもデフォルト値にする場合、次の手順へ進みます。

   ユーザー値を設定する場合、setsscpコマンドでIPアドレスを設定し、applynetworkコマンドで増設するSPARC M10-4SのSSCPのIPアドレスを反映させて確認します。その後、rebootxscfコマンドを実行し、設定を完了させてから次の手順へ進みます。手順の詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.6　ネットワーク設定を適用する」を参照してください。

3. **SPARC M10-4Sを増設します。**
   a. **addfruコマンドを実行し、メッセージに従ってSPARC M10-4Sを増設します。**

注－SSCPのIPアドレスが設定されていない場合、addfruコマンドはエラーとなります。

注－addfruコマンドを実行すると、増設するSPARC M10-4Sのファームウェアの版数は、マスタXSCFが動作しているSPARC M10-4Sのファームウェアの版数に自動的に合わされます。この例では、増設されるSPARC M10-4S（BB-ID 1）のファームウェアの版数は、マスタ

XSCFが動作するSPARC M10-4S（BB-ID 0）のファームウェアに、自動的に版数が合わされます。

次の例では、BB#1が増設されています。

```
XSCF> addfru
--------------------------------------------------------------------------------
Maintenance/Addition Menu
Please select the chassis including added FRU.
No. FRU                  Status
--- -------------------- --------------
 1  /BB#0                Normal
 2  /BB#1                Unmount
 3  /BB#2                Unmount
 4  /BB#3                Unmount
--------------------------------------------------------------------------------
Select [1-16|c:cancel] :2
Maintenance/Addition Menu
Please select the BB or a type of FRU to be added.
1. BB itself
2. PSU (Power Supply Unit)
--------------------------------------------------------------------------------
Select [1,2|c:cancel] :1
Maintenance/Addition Menu
Please select a FRU to be added.
No. FRU                  Status
--- -------------------- --------------
 1  /BB#1                Unmount
Select [1|b:back] :1
```

b. **増設するSPARC M10-4Sをラックに搭載します。**

前述のaddfruコマンド実行中に、「After the added device is connected with the system, please turn on the breaker of the BB#1.」のメッセージが表示されたら、増設するSPARC M10-4Sをラックに搭載します。

搭載手順は、『SPARC M10-4S インストレーションガイド』の「3.4.1 SPARC M10-4Sをラックに搭載する」を参照してください。

c. **増設するSPARC M10-4Sの識別ID（BB-ID）を設定します。**

『SPARC M10-4S インストレーションガイド』の「4.1　筐体の識別ID（BB-ID）を設定する」を参照してください。

d. **クロスバーケーブルを接続します。**

既存のSPARC M10-4Sと増設したBBのクロスバーケーブルを接続します。接続ケーブルのルート図とケーブル一覧表は、『SPARC M10-4S インストレーションガイド』の「付録 B　ビルディングブロック構成のケーブル接続資料」を参照してください。筐体間直結で接続する場合は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

e. **XSCF BB制御ケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4SをXSCF BB制御ケーブルで接続します。XSCF BB制御ケーブルのルート図とケーブル一覧表は、『SPARC M10-4S インストレーションガイド』の「付録 B　ビルディングブロック構成のケーブル接続資料」を参照してください。筐体間直結で接続する場合は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

f. **XSCF DUAL制御ケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4SをXSCF DUAL制御ケーブルで接続します。詳細は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

g. **シリアルケーブルやLANケーブルを接続します。**

増設するSPARC M10-4SのXSCFシリアルポートへシリアルケーブルを接続します。また、XSCF-LAN、GbEポート、PCIeカードの各LANポートへケーブルを接続します。詳細は、『SPARC M10-4S インストレーションガイド』の「5.1　SPARC M10-4Sにケーブルを接続する」を参照してください。

h. **増設するSPARC M10-4Sの入力電源を接続します。**

増設するSPARC M10-4Sの電源コードを入力電源に接続します。

i. **addfruコマンドの入力画面から「f」を入力します。**

手順a.で実行しているaddfruコマンドの入力画面で「f」を入力し、SPARC M10-4Sの増設処理を実施します。

```
2) Please select[f:finish] :f

Waiting for BB#1 to enter install state.
[This operation may take up to 20 minute(s)] (progress scale
 reported in seconds)
0.....  30.... done


Waiting for BB#1 to enter ready state.
[This operation may take up to 45 minute(s)] (progress scale
 reported in seconds)
0.....  30.....  60... done

Do you want to start to diagnose BB#1? [s:start|c:cancel] :
```

j. **増設するSPARC M10-4Sの診断をスキップし、addfruコマンドを終了させます。**

マスタXSCFにあるaddfruコマンドの入力画面で「c」を入力し、増設するSPARC M10-4Sの診断をスキップします。「The addition of BB#1 has completed.」が出力されたら「f」を入力し、最後に「c」を入力して、addfruコマンドを終了させます。

```
Do you want to start to diagnose BB#1? [s:start|c:cancel] :c

Diagnostic tests are about to be skipped.
Running diagnostic tests are strongly recommended before using BB#1.
Are you sure you want to skip testing? [y:yes|n:no] :y
```

```
------------------------------------------------------------------------------
Maintenance/Addition Menu Status of the added FRU.

FRU                  Status
------------------- --------------
/BB#1                Normal
------------------------------------------------------------------------------
[Warning:007]
Running diagnostic tests on BB#1 is strongly recommended after addfru has
completed.
The addition of BB#1 has completed.[f:finish] :f

------------------------------------------------------------------------------
Maintenance/Addition Menu
Please select the chassis including added FRU.

No. FRU                  Status
--- ------------------- --------------
 1   /BB#0                Normal
 2   /BB#1                Normal


------------------------------------------------------------------------------
Select [1,2|c:cancel] :c
```

4. **増設したSPARC M10-4Sを診断します。**

   a. **testsbコマンドを実行し、増設したSPARC M10-4Sの診断テストを実施します。**

   testsbコマンドを実行し、診断テストを実施します。増設したSPARC M10-4Sの物理システムボード(PSB)番号を指定し、初期診断と接続I/Oの確認を実施します。

```
XSCF> testsb -v -p -s -y 01-0
Initial diagnosis is about to start, Continue?[y|n] :y
PSB#01-0 power on sequence started.
```

   エラーが表示された場合は、『SPARC M10-4S インストレーションガイド』の「A.2.4　診断結果を確認する」を参照してください。

   b. **diagxbuコマンドを実行し、クロスバーケーブル診断テストを実施します。**

   diagxbuコマンドを実行し、組み込み先の物理パーティションと増設したSPARC M10-4Sのクロスバーケーブルの診断テストを実施します。増設したSPARC M10-4Sの識別ID（BB-ID）、および組み込み先の物理パーティションID（PPAR-ID）を指定します。

```
XSCF> diagxbu -y -b 01 -p 00
XBU diagnosis is about to start, Continue?[y|n] :y
Power on sequence started. [7200sec]
0..... 30..... 60..end
XBU diagnosis started. [7200sec]
```

```
0..... 30..... 60..... 90.....120.....150.....180.....210.....240...../
270.....300.....330.....360.....390.....420.....450.....480.....510.....¥
540.....570.....600.....630.....660.....690.....720.....750.....780.....¥
810.....840.....870.....900.....930...end
completed.
Power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....180.end
completed.
```

c. **showlogs errorコマンドを実行し、エラーが表示されないことを確認します。**

```
XSCF> showlogs error
```

エラーが表示された場合は、『SPARC M10-4S インストレーションガイド』の「A.2.4　診断結果を確認する」を参照してください。

d. **showhardconfコマンドを実行し、増設したSPARC M10-4Sの構成や状態を確認します。**

showhardconfコマンドを実行し、増設したSPARC M10-4SのCPUやメモリなどのハードウェア構成や、各ユニットのStatusがNormalになっていることを確認します。

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Service;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
           + FRU-Part-Number:CA07361-D941 C4   /7060911             ;
            + Memory_Size:128 GB; Type: A ;
            CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
                + Freq:3.000 GHz; Type:0x10;
                + Core:16; Strand:2;
     :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
     :
```

5. **増設したSPARC M10-4SのXSCFネットワークを設定します。**
増設したSPARC M10-4SのXSCF用のネットワークを設定します。詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.2　イーサネット（XSCF-LAN）のIPアドレスを設定する」と「7.5.3　引き継ぎIPアドレスを設定する」を参照してください。設定後は、applynetworkコマンドで実行し、設定

内容を反映させ確認します。その後、rebootxscfコマンドを実行し、設定を完了させてから次の手順へ進みます。手順の詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.6　ネットワーク設定を適用する」を参照してください。

6. **増設したSPARC M10-4Sのメモリのミラーモードを設定します。**
メモリのミラーモードを使用しない場合は、この手順は不要です。

メモリのミラーモードの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「14.1　メモリをミラー構成にする」を参照してください。

a. **増設したSPARC M10-4Sのメモリをミラー構成にする場合、setupfruコマンドを使用し、メモリのミラーモードを設定します。**

次の例では、SPARC M10-4Sの物理システムボード（PSB 01-0）のすべてのCPUをミラーモードに設定しています。

```
XSCF> setupfru -m y sb 01-0
```

b. **showfruコマンドを使用し、メモリのミラーモード設定を確認します。**

```
XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb     01-0
   cpu 01-0-0   yes
   cpu 01-0-1   yes
   cpu 01-0-2   yes
   cpu 01-0-3   yes
```

7. **物理パーティション構成情報に増設したSPARC M10-4Sのシステムボードを登録します。**

a. **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                          Running
            00      00-0
```

b. **setpclコマンドを実行し、物理パーティション構成情報にシステムボードを登録します。**

setpclコマンドを実行し、増設したSPARC M10-4Sのシステムボード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 01-0）を、物理パーティション0の論理システムボード（LSB 01）に対応付けています。

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

8. **CPUコア アクティベーションキーを登録し、CPUコアリソースを割り当てます。**
   a. **showcodusageコマンドを実行し、CPUコア アクティベーションの情報を確認します。**

   showcodusage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コアリソースがあるかを確認します。

   ここでは、システムには128個のCPUコアリソースが実装されており、128個のCPUコア アクティベーションが登録され、64個のCPUコアリソースが使用中であり、現在未使用のCPUコア アクティベーション数が64個であることを示しています。

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-------- ------ --------- ------------- ------
PROC         64       128           128 OK: 64 cores available

Note:
  Please confirm the value of the "In Use" by the ldm command of Oracle VM
Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of logical
domains.
```

注ー登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。CPUコア アクティベーションキーの追加については、『SPARC M12/M10 システム運用・管理ガイド』の「5.3　CPUコアリソースを追加する」を参照してください。

---

   b. **showcodコマンドを実行し、物理パーティションのCPUコアリソース情報を確認します。**

   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コアリソース情報を確認します。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 64
```

   割り当てが不足している場合は、setcodコマンドを実行し、CPUコア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、64個のCPUコアリソースを物理パーティション0に追加しています。

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注ーXCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

CPUコアリソースを割り当てた場合は、再度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コアリソース情報を確認します。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

9. **論理ドメインの稼働状況を確認します。**
   a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

   b. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

この例では、制御ドメインと1つのルートドメイン、1つのゲストドメインが稼働しています。

すべてのドメインが「active」でOracle Solarisが動作状態、または「inactive」状態であることを確認します。OpenBoot PROM状態またはbound状態のドメインが存在すると、物理パーティションの動的再構成に失敗することがあります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    28G      0.0%  1h 33m
guest0           active     -n----  5100    64    64G      3.1%  2s
root-dom0        active     -n--v-  5000    32    32G      3.1%  17m
```

10. **増設したSPARC M10-4Sのシステムボード（PSB<BB>）を物理パーティションに組み込みます。**

a. **XSCFシェルに戻り、showboardsコマンドを実行し、PSBの状態を確認します。**

showboardsコマンドを実行し、増設したSPARCM10-4SのPSB（BB）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 -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
```

b. **addboardコマンドを実行し、PSBを物理パーティションに組み込みます。**

addboard -c configureコマンドを実行し、PSBを物理パーティションに組み込みます。

```
XSCF> addboard -c configure -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
 [1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

注ーaddboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

---

c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

終了値が0の場合、addboardコマンドは正常終了です。

終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

増設したSPARC M10-4SのPSBが「Assigned」状態であり、Pwr欄、Conn欄とConf欄の表示がすべて「y」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

11. **論理ドメインの稼働状況を確認します。**
    a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

    b. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**

    論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

    「n」: Oracle Solarisが動作中

    「t」: OpenBoot PROMの状態

    「-」: その他の状態（[STATE] が「active」以外のときも含む）

    この例では、制御ドメインと1つのルートドメイン、1つのゲストドメインが稼働しています。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    28G      0.0%  2h 3m
guest0           active     -n----  5100    64    64G      3.1%  33m
root-dom0        active     -n--v-  5000    32    32G      3.1%  47m
```

12. **既存の論理ドメインを構成変更し、新規に論理ドメインを追加します。**
    追加されたSPARC M10-4Sのリソースを、既存の論理ドメインに割り当てたり、新規に論理ドメインを構築して割り当てたります。次の例では、構成例に従って、制御ドメインのCPUコアとメモリを追加し、ゲストドメインguest1、ルートドメインroot-dom1を追加し、各ゲストドメインの仮想I/Oを冗長構成にし、さらに制御ドメインのシステムボリュームを内蔵ハードディスクでミラーリング構成にしています。

    a. **ldm list-devicesコマンドを実行し、追加されたハードウェアリソースの状態を確認します。**

```
# ldm list-devices
CORE
    ID      %FREE   CPUSET
    128     100    (256, 257)
    132     100    (264, 265)
    136     100    (272, 273)
    140     100    (280, 281)
....
MEMORY
```

```
    PA                    SIZE            BOUND
    0x700000000000        32G
    0x720000000000        32G
    0x740000000000        32G
    0x760050000000        31488M
....
```

b. **ldm add-coreコマンド、ldm add-memoryコマンドを実行し、制御ドメインのCPUコアとメモリリソースを追加します。**

ldm add-core、ldm add-memoryコマンドを実行し、制御ドメインのCPUコアとメモリを追加し、ldm list-domainコマンドで追加されたことを確認します。

```
# ldm add-core 16 primary
# ldm add-memory 38G primary
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  2h 13m
guest0           active     -n----  5100    64    64G      0.1%  43m
root-dom0        active     -n--v-  5000    32    32G      0.1%  57m
```

c. **新規に論理ドメインを作成し、CPUコアとメモリリソースを割り当てます。**

ここでは、ルートドメイン（root-dom1）と、ゲストドメイン（guest1）を追加し、CPUコア、メモリを割り当てる例を示します。なお、ドメインを構築するときに、ldm set-coreコマンドで先にCPUコア数を設定し、その後、ldm set-memoryコマンドで割り当てるメモリサイズを設定すると、論理ドメインに割り当てられるメモリブロックが分散（フラグメント化）されにくくなり、ビルディングブロックの切り離しのときにメモリ配置を考慮しやすくなります。

以下にコマンドの実行例を示します。

```
# ldm add-domain root-dom1
# ldm add-domain guest1
# ldm set-core 16 root-dom1
# ldm set-core 32 guest1
# ldm set-memory 32G root-dom1
# ldm set-memory 64G guest1
```

注－Oracle VM Server for SPARC 3.2よりも古いバージョンのOracle VM Server for SPARCの場合、物理パーティションの動的再構成を実施できるようにするためには、各論理ドメインに割り当てるメモリサイズは、（CPUコア数×256 MB）の倍数に設定してください。

---

d. **ルートドメインにルートコンプレックスを割り当てます。**

次の例では、追加されたSPARC M10-4Sのルートコンプレックスを、ルートドメイン（root-dom1）に割り当てています。

ldm list-ioコマンドを実行し、追加するルートコンプレックスを確認します。

NAMEが「/BB1」で始まるデバイスがBB-ID 1上のデバイスで、そのデバイス

のBUSに表示されている文字列がルートコンプレックスを示しています。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9
PCIE10                                      BUS    PCIE10
PCIE11                                      BUS    PCIE11
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13
PCIE14                                      BUS    PCIE14
PCIE15                                      BUS    PCIE15
....
/BB1/CMUL/NET0                              PCIE   PCIE8
/BB1/CMUL/SASHBA                            PCIE   PCIE8
/BB1/PCI0                                   PCIE   PCIE9
/BB1/PCI3                                   PCIE   PCIE10
/BB1/PCI4                                   PCIE   PCIE10
/BB1/PCI7                                   PCIE   PCIE11
/BB1/PCI8                                   PCIE   PCIE11
/BB1/CMUL/NET2                              PCIE   PCIE12
/BB1/PCI1                                   PCIE   PCIE13
/BB1/PCI2                                   PCIE   PCIE13
/BB1/PCI5                                   PCIE   PCIE14
/BB1/PCI6                                   PCIE   PCIE14
/BB1/PCI9                                   PCIE   PCIE15
/BB1/PCI10                                  PCIE   PCIE15
```

ldm add-ioコマンドを実行し、ルートドメイン（root-dom1）にルートコンプレックスを追加し、ldm list-ioコマンドを実行して確認します。

次の例では、root-dom1には、上記で表示されたオンボードデバイス（/BB1/CMUL/NET0、/BB1/CMUL/SASHBA、/BB1/CMUL/NET2）のルートコンプレックス（PCIE8とPCIE12）以外のBB1のルートコンプレックス（PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15）を追加しています。

```
# ldm add-io PCIE9 root-dom1
# ldm add-io PCIE10 root-dom1
....
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
```

```
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8    root-dom1  IOV
PCIE9                                       BUS    PCIE9
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
PCIE14                                      BUS    PCIE14   root-dom1  IOV
PCIE15                                      BUS    PCIE15   root-dom1  IOV
```

e. **ルートドメインのコンソールデバイスを割り当てて起動します。**

ldm set-vconsoleコマンドでルートドメイン（root-dom1）にコンソールデバイスを割り当て、ldm bind-domain、ldm start-domainコマンドで起動します。

```
# ldm set-vconsole port=5001 vcc0 root-dom1
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# telnet localhost 5001
....
```

f. **ルートドメインにOracle Solarisをインストールします。**

起動したルートドメインに、Oracle Solarisをインストールします。Oracle Solarisのインストールの詳細については、Oracle Solarisの関連ドキュメントを参照してください。

g. **ルートドメインの仮想サービスに割り当てる物理デバイスを確認します。**

Oracle Solarisをインストールしたあと、ルートドメインにログインして、formatコマンドを実行し、仮想ディスクサービスに割り当てるデバイス名を確認します。このとき、ldm list-io -lで実行すると、PCIeエンドポイントに対応するデバイスパスが表示されるので、formatコマンドで表示されるディスクのデバイスパス名と比較することで、ディスクが接続されているパスが確認できます。

```
# ldm list-io -l
NAME                                        TYPE   BUS      DOMAIN   STATUS
----                                        ----   ---      ------   ------
....
/BB1/PCI0                                   PCIE   PCIE9    root-dom1OCC
[pci@8900/pci@4/pci@0/pci@0]
/BB1/PCI3                                   PCIE   PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@0]
/BB1/PCI4                                   PCIE   PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@8]....
# telnet localhost 5000
....
```

```
root-dom1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
       0. c3t500000E01BDA70B2d0 <FUJITSU-MBB2147RC-3703 cyl 14087 alt 2 hd 24
sec 848>
          / pci@8900/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w500000e01bda70b2,0
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8a00/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
Specify disk (enter its number): ^C
```

続けて、dladm show-phys -Lコマンドを実行すると、ルートドメインに割り当てられたイーサネットのインターフェース名と物理位置が確認できます。

```
root-dom0# dladm show-phys -L
LINK              DEVICE       LOC
net0              igb0         /BB1/PCI0
net1              igb1         /BB1/PCI0
net2              igb2         /BB1/PCI0
net3              igb3         /BB1/PCI0
```

h. **ルートドメインに仮想I/Oサービスを割り当てます。**

制御ドメインに戻り、ルートドメイン(root-dom1)に仮想I/Oサービスを追加します。I/Oデバイスの詳細な設定については、Oracle社のドキュメント『Oracle VM Server for SPARC 管理ガイド』を参照してください。

次の例では、追加されたルートドメイン（root-dom1）に、仮想ディスクサービス（vds1）と仮想スイッチネットワークインターフェース（vsw10、vsw11）を追加しています。

```
# ldm add-vdiskserver vds1 root-dom1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol0@
vds1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol1@
vds1
# ldm add-vswitch net-dev=net1 vsw10 root-dom1
# ldm add-vswitch net-dev=net2 vsw11 root-dom1
```

i. **ゲストドメインに仮想I/Oデバイスを割り当てます。**

追加した仮想I/Oサービスを使った、仮想I/Oデバイスを各ゲストドメインに追加します。

次の例では、既存のゲストドメイン（guest0）にroot-dom1側の仮想I/Oサービスを使った仮想I/Oデバイス（vdisk10、vnet10）を追加し、新たに追加するゲストドメイン（guest1）には、root-dom0とroot-dom1の仮想I/Oサービスを使った仮想 I/Oデバイス（vdisk1、vdisk11、vnet1、vnet11）、仮想コンソール（vcons）を追加しています。

```
# ldm add-vdisk vdisk10 vol0@vds1 guest0
# ldm add-vnet vnet10 vsw10 guest0
# ldm add-vdisk vdisk1 vol1@vds0 guest1
# ldm add-vdisk vdisk11 vol11@vds1 guest1
# ldm add-vnet vnet1 vsw1 guest1
# ldm add-vnet vnet11 vsw11 guest1
# ldm set-vconsole port=5101 guest1
```

j. **追加したゲストドメインを起動します。**

ldm bind-domain、ldm start-domainコマンドを実行して、追加したゲストドメイン（guest1）を起動します。

```
# ldm bind-domain guest1
# ldm start-domain guest1
LDom guest1 started
```

k. **追加したゲストドメインにOracle Solarisをインストールします。**

追加したゲストドメインにOracle Solarisをインストールします。Oracle Solarisのインストールの詳細は、Oracle Solarisの関連ドキュメントを参照してください。

l. **各ゲストドメインのI/Oを冗長構成にします。**

各ゲストドメインに追加した、仮想I/Oデバイスを冗長構成に変更します。冗長構成の設定方法については、各冗長構成のソフトウェアの関連ドキュメントを参照してください。

m. **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、追加されたSPARC M10-4Sのオンボードデバイスを制御ドメインに割り当てて冗長構成にする例を示します。制御ドメインを遅延再構成モードに移行します。

```
# 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.
```

制御ドメインに、追加したSPARC M10-4Sのルートコンプレックス（PCIE8、PCIE12）を追加してOracle Solarisを再起動します。

```
# ldm add-io PCIE8 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 add-io PCIE12 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.
------------------------------------------------------------------------------
```

```
# shutdown -i6 -g0 -y
....
```

Oracle Solaris再起動後、システムボリュームを冗長構成にします。次の例では、formatコマンドでディスクを確認し、ZFSでミラー構成にしています。冗長構成の設定方法については、各冗長構成のソフトウェアの関連ドキュメントを参照してください。

```
# format
Searching for disks...done


AVAILABLE DISK SELECTIONS:
       0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0

errors: No known data errors
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting
```

zpool statusコマンドで、ミラー構成が組み込まれたことを確認します。このとき、zpool statusコマンドを実行し、同期処理（resilver）が完了したかどうかを確認してください。次の例では、同期処理中を表しています。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:

        NAME                          STATE     READ WRITE CKSUM
```

```
        rpool                       DEGRADED     0     0     0
          mirror-0                  DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0 ONLINE       0     0     0
            c3t50000393A803B13Ed0s0 DEGRADED     0     0     0  (resilvering)

errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          mirror-0                  ONLINE       0     0     0
            c2t50000393E802CCE2d0s0 ONLINE       0     0     0
            c3t50000393A803B13Ed0s0 ONLINE       0     0     0

errors: No known data errors
```

n. **各論理ドメインのメモリ配置を確認して調整します。**

ldm list-devices -a memoryを実行して、メモリブロックの配置を確認します。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom1
    0x720000000000      32G             guest1
    0x740000000000      32G             guest1
    0x760000800000      1272M           _sys_
    0x760050000000      16G             primary
    0x760450000000      10G             primary
    0x7606d0000000      4864M
    0x780000000000      32G             guest0
    0x7a0000000000      32G             guest0
    0x7c0000000000      256M            primary
    0x7c0010000000      3584M           root-dom0
    0x7c00f0000000      24832M          primary
    0x7c0700000000      4G              root-dom0
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      25088M          root-dom0
    0x7e06a0000000      3584M           primary
    0x7e0780000000      2G              primary
```

論理ドメインに割り当てられているメモリブロックや、未割り当てのメモリブロックが小さく分割されている場合、論理ドメインをアンバインドしてからバインドしなおすことで、分割されたメモリブロックを大きな連続領域に

まとめなおすことができます。これにより、あとでSPARC M10-4Sの追加や削除がしやすい状態になります。

手順の詳細は、「A.2　XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがない論理ドメインの構成の場合」の「A.2.2　物理パーティションの構築手順例」にある手順11を参照してください。

o. **論理ドメイン構成情報を保存します。**

ldm list-domainコマンドを実行し、構成した論理ドメインが起動されていることを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS   VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART   64    56G      64%   2h 54m
guest0           active     -n----  5100   64    64G      42%   2h 54m
guest1           active     -n----  5101   64    64G      11%   2h 54m
root-dom0        active     -n--v-  5000   32    32G      20%   2h 54m
root-dom1        active     -n--v-  5001   32    32G      18%   2h 54m
```

ldm add-spconfigコマンドを実行し、論理ドメイン構成情報をXSCFに保存します。

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4   [current]
```

A.4.3　増設手順（PCIeバスを動的に割り当て可能な場合）

PCIeバスを動的に割り当て可能な環境（XCP 2240以降、かつ、Oracle VM Server for SPARC 3.2以降で、ルートドメインがOracle Solaris 11.2 SRU11.2.8.4.0以降）で、「図 A-5　1BBから2BBへ増設後の構成例」に記載される構成例に合わせ、1BB構成のSPARC M10-4Sを、動的再構成により2BB構成に増設する手順は次のとおりです。

ここでの説明は、SPARC M12でも同様です。

1. **マスタXSCFにログインします。**
   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。

   スタンバイ状態のXSCFの場合は、マスタXSCFへログインしなおしてください。

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

2. **showsscpコマンドを実行し、サービスプロセッサ間通信プロトコル（SSCP）のIPアドレスがデフォルト値か、ユーザーの設定した値か確認します。**

```
XSCF> showsscp
```

IPアドレスがデフォルト値で、増設するSPARC M10-4SのIPアドレスもデフォルト値にする場合、次の手順へ進みます。

ユーザー値を設定する場合、setsscpコマンドでIPアドレスを設定し、applynetworkコマンドで増設するSPARC M10-4SのSSCPのIPアドレスを反映させて確認します。その後、rebootxscfコマンドを実行し、設定を完了させてから次の手順へ進みます。手順の詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.6　ネットワーク設定を適用する」を参照してください。

3. **SPARC M10-4Sを増設します。**
   a. **addfruコマンドを実行し、メッセージに従ってSPARC M10-4Sを増設します。**

注ーSSCPのIPアドレスが設定されていない場合、addfruコマンドはエラーとなります。

注ーaddfruコマンドを実行すると、増設するSPARC M10-4Sのファームウェアの版数は、マスタXSCFが動作しているSPARC M10-4Sのファームウェアの版数に自動的に合わされます。この例では、増設されるSPARC M10-4S（BB-ID 1）のファームウェアの版数は、マスタXSCFが動作するSPARC M10-4S（BB-ID 0）のファームウェアに、自動的に版数が合わされます。

次の例では、BB#1が増設されています。

```
XSCF> addfru
--------------------------------------------------------------------------------
Maintenance/Addition Menu
Please select the chassis including added FRU.
No. FRU                  Status
--- -------------------- --------------
 1  /BB#0                Normal
 2  /BB#1                Unmount
 3  /BB#2                Unmount
 4  /BB#3                Unmount
--------------------------------------------------------------------------------
Select [1-16|c:cancel] :2
Maintenance/Addition Menu
Please select the BB or a type of FRU to be added.
1. BB itself
2. PSU (Power Supply Unit)
--------------------------------------------------------------------------------
Select [1,2|c:cancel] :1
Maintenance/Addition Menu
Please select a FRU to be added.
No. FRU                  Status
--- -------------------- --------------
 1  /BB#1                Unmount
Select [1|b:back] :1
```

b. **増設するSPARC M10-4Sをラックに搭載します。**

前述のaddfruコマンド実行中に、「After the added device is connected with the system, please turn on the breaker of the BB#1.」のメッセージが表示されたら、増設するSPARC M10-4Sをラックに搭載します。

搭載手順は、『SPARC M10-4S インストレーションガイド』の「3.4.1 SPARC M10-4Sをラックに搭載する」を参照してください。

c. **増設するSPARC M10-4Sの識別ID（BB-ID）を設定します。**

『SPARC M10-4S インストレーションガイド』の「4.1　筐体の識別ID（BB-ID）を設定する」を参照してください。

d. **クロスバーケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4Sのクロスバーケーブルを接続します。接続ケーブルのルート図とケーブル一覧表は、『SPARC M10-4S インストレーションガイド』の「付録 B　ビルディングブロック構成のケーブル接続資料」を参照してください。筐体間直結で接続する場合は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

e. **XSCF BB制御ケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4SをXSCF BB制御ケーブルで接続します。XSCF BB制御ケーブルのルート図とケーブル一覧表は、『SPARC M10-4S インストレーションガイド』の「付録 B　ビルディングブロック構成のケーブル接続資料」を参照してください。筐体間直結で接続する場合は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

f. **XSCF DUAL制御ケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4SをXSCF DUAL制御ケーブルで接続します。詳細は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

g. **シリアルケーブルやLANケーブルを接続します。**

増設するSPARC M10-4SのXSCFシリアルポートへシリアルケーブルを接続します。また、XSCF-LAN、GbEポート、PCIeカードの各LANポートへケーブルを接続します。詳細は、『SPARC M10-4S インストレーションガイド』の「5.1 SPARC M10-4Sにケーブルを接続する」を参照してください。

h. **増設するSPARC M10-4Sの入力電源を接続します。**

増設するSPARC M10-4Sの電源コードを入力電源に接続します。

i. **addfruコマンドの入力画面から「f」を入力します。**

手順a.で実行しているaddfruコマンドの入力画面で「f」を入力し、SPARC M10-4Sの増設処理を実施します。

```
2) Please select[f:finish] :f

Waiting for BB#1 to enter install state.
[This operation may take up to 20 minute(s)] (progress scale
 reported in seconds)
0.....  30.... done
```

```
Waiting for BB#1 to enter ready state.
[This operation may take up to 45 minute(s)] (progress scale
 reported in seconds)
0.....  30.....  60... done

Do you want to start to diagnose BB#1? [s:start|c:cancel] :
```

j. **増設するSPARC M10-4Sの診断をスキップし、addfruコマンドを終了させます。**

マスタXSCFにあるaddfruコマンドの入力画面で「c」を入力し、増設するSPARC M10-4Sの診断をスキップします。「The addition of BB#1 has completed.」が出力されたら「f」を入力し、最後に「c」を入力して、addfruコマンドを終了させます。

```
Do you want to start to diagnose BB#1? [s:start|c:cancel] :c

Diagnostic tests are about to be skipped.
Running diagnostic tests are strongly recommended before using BB#1.
Are you sure you want to skip testing? [y:yes|n:no] :y

-------------------------------------------------------------------------------
Maintenance/Addition Menu Status of the added FRU.

FRU                  Status
------------------- --------------
/BB#1                Normal
-------------------------------------------------------------------------------
[Warning:007]
Running diagnostic tests on BB#1 is strongly recommended after addfru has
completed.
The addition of BB#1 has completed.[f:finish] :f

-------------------------------------------------------------------------------
Maintenance/Addition Menu
Please select the chassis including added FRU.

No. FRU                  Status
--- ------------------- --------------
 1   /BB#0                Normal
 2   /BB#1                Normal


-------------------------------------------------------------------------------
Select [1,2|c:cancel] :c
```

4. **増設したSPARC M10-4Sを診断します。**

a. **testsbコマンドを実行し、増設したSPARC M10-4Sの診断テストを実施します。**

testsbコマンドを実行し、診断テストを実施します。増設したSPARC M10-4Sの物理システムボード(PSB)番号を指定し、初期診断と接続I/Oの確認を実施します。

```
XSCF> testsb -v -p -s -y 01-0
Initial diagnosis is about to start, Continue?[y|n] :y
PSB#01-0 power on sequence started.
```

エラーが表示された場合は、『SPARC M10-4S インストレーションガイド』の「A.2.4　診断結果を確認する」を参照してください。

b. **diagxbuコマンドを実行し、クロスバーケーブル診断テストを実施します。**

diagxbuコマンドを実行し、組み込み先の物理パーティションと増設したSPARC M10-4Sのクロスバーケーブルの診断テストを実施します。増設したSPARC M10-4Sの識別ID（BB-ID）、および組み込み先の物理パーティションID（PPAR-ID）を指定します。

```
XSCF> diagxbu -y -b 01 -p 00
XBU diagnosis is about to start, Continue?[y|n] :y
Power on sequence started. [7200sec]
0..... 30..... 60..end
XBU diagnosis started. [7200sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240...../
270.....300.....330.....360.....390.....420.....450.....480.....510.....¥
540.....570.....600.....630.....660.....690.....720.....750.....780.....¥
810.....840.....870.....900.....930...end
completed.
Power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....180.end
completed.
```

c. **showlogs errorコマンドを実行し、エラーが表示されないことを確認します。**

```
XSCF> showlogs error
```

エラーが表示された場合は、『SPARC M10-4S インストレーションガイド』の「A.2.4　診断結果を確認する」を参照してください。

d. **showhardconfコマンドを実行し、増設したSPARC M10-4Sの構成や状態を確認します。**

showhardconfコマンドを実行し、増設したSPARC M10-4SのCPUやメモリなどのハードウェア構成や、各ユニットのStatusがNormalになっていることを確認します。

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Service;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
           + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
```

```
            + Memory_Size:128 GB; Type: A ;
            CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
                + Freq:3.000 GHz; Type:0x10;
                + Core:16; Strand:2;
    :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
    :
```

5. **増設したSPARC M10-4SのXSCFネットワークを設定します。**
   a. **testsbコマンドを実行し、増設したSPARC M10-4Sの診断テストを実施します。**

   増設したSPARC M10-4SのXSCF用のネットワークを設定します。詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.2　イーサネット（XSCF-LAN）のIPアドレスを設定する」と「7.5.3　引き継ぎIPアドレスを設定する」を参照してください。設定後は、applynetworkコマンドで実行し、設定内容を反映させ確認します。その後、rebootxscfコマンドを実行し、設定を完了させてから次の手順へ進みます。手順の詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.6　ネットワーク設定を適用する」を参照してください。

6. **増設したSPARC M10-4Sのメモリのミラーモードを設定します。**
   メモリのミラーモードを使用しない場合は、この手順は不要です。

   メモリのミラーモードの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「14.1　メモリをミラー構成にする」を参照してください。

   a. **増設したSPARC M10-4Sのメモリをミラー構成にする場合、setupfruコマンドを使用し、メモリのミラーモードを設定します。**

   次の例では、SPARC M10-4Sの物理システムボード（PSB 01-0）のすべてのCPUに対してメモリをミラーモードに設定しています。

```
XSCF> setupfru -m y sb 01-0
```

   b. **showfruコマンドを使用し、メモリのミラーモード設定を確認します。**

```
XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb     01-0
   cpu 01-0-0   yes
   cpu 01-0-1   yes
   cpu 01-0-2   yes
   cpu 01-0-3   yes
```

7. **物理パーティション構成情報に増設したSPARC M10-4Sのシステムボードを登録します。**

a. **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                          Running
            00      00-0
```

b. **setpclコマンドを実行し、物理パーティション構成情報にシステムボードを登録します。**

setpclコマンドを実行し、増設したSPARC M10-4Sのシステムボード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）01-0を、物理パーティション0の論理システムボード（LSB）01に対応付けています。

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

8. **CPUコア アクティベーションキーを登録し、CPUコアリソースを割り当てます。**

a. **showcodusageコマンドを実行し、CPUコアリソースの情報を確認します。**

showcodusage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コアリソースがあるかを確認します。

ここでは、システムには128個のCPUコアリソースが実装されており、128個のCPUコア アクティベーションが登録され、64個のCPUコアリソースが使用中であり、現在未使用のCPUコア アクティベーション数が64個であることを示しています。

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-------- ------ --------- ------------- ------
PROC         64       128           128 OK: 64 cores available

Note:
  Please confirm the value of the "In Use" by the ldm command of Oracle VM
Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of logical
domains.
```

注ー登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。CPUコア アクティベーションキーの追加については、『SPARC M12/M10 システム運用・管理ガイド』の「5.3　CPUコアリソースを追加する」を参照してください。

---

b. **showcodコマンドを実行し、物理パーティションのCPUコアリソース情報を確認します。**

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コアリソース情報を確認します。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 64
```

割り当てが不足している場合は、setcodコマンドを実行し、CPUコア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、64個のCPUコアリソースを物理パーティションに追加しています。

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注ーXCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

CPUコアリソースを割り当てた場合は、再度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コアリソース情報を確認します。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

9. **論理ドメインの稼働状況を確認します。**

a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

b. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が

次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

この例では、制御ドメインと1つのルートドメイン、1つのゲストドメインが稼働しています。

すべてのドメインが「active」でOracle Solarisが動作状態、または「inactive」状態であることを確認します。OpenBoot PROM状態またはbound状態のドメインが存在すると、物理パーティションの動的再構成に失敗することがあります。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    28G      0.0%  1h 33m
guest0           active     -n----  5100    64    64G      3.1%  2s
root-dom0        active     -n--v-  5000    32    32G      3.1%  17m
```

10. **増設したSPARC M10-4Sのシステムボード（PSB<BB>）を物理パーティションに組み込みます。**

    a. **XSCFシェルに戻り、showboardsコマンドを実行し、PSBの状態を確認します。**

    showboardsコマンドを実行し、増設したSPARCM10-4SのPSB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 -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
```

    b. **addboardコマンドを実行し、PSBを物理パーティションに組み込みます。**

    addboard -c configureコマンドを実行し、PSBを物理パーティションに組み込みます。

```
XSCF> addboard -c configure -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
 [1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

注ーaddboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

終了値が0の場合、addboardコマンドは正常終了です。

終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

増設したSPARC M10-4SのPSBが「Assigned」状態であり、Pwr欄、Conn欄とConf欄の表示がすべて「y」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

11. **論理ドメインの稼働状況を確認します。**

a. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

b. **ldm list-domainコマンドを実行し、論理ドメインの稼働状況を確認します。**

論理ドメインの稼働状況は、[STATE] と [FLAGS] の組み合わせによって確認します。[STATE] が「active」のとき、 [FLAGS] の文字列の左から2文字目が次の意味を持ちます。

「n」: Oracle Solarisが動作中

「t」: OpenBoot PROMの状態

「-」: その他の状態（[STATE] が「active」以外のときも含む）

この例では、制御ドメインと1つのルートドメイン、1つのゲストドメインが稼働しています。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    28G      0.0%  2h 3m
guest0           active     -n----  5100    64    64G      3.1%  33m
root-dom0        active     -n--v-  5000    32    32G      3.1%  47m
```

12. **既存の論理ドメインを構成変更し、新規に論理ドメインを追加します。**

追加されたSPARC M10-4Sのリソースを、既存の論理ドメインに割り当てたり、新規に論理ドメインを構築して割り当てたります。次の例では、構成例に従って、制御ドメインのCPUコアとメモリを追加し、ゲストドメインguest1、ルートドメインroot-dom1を追加し、各ゲストドメインの仮想I/Oを冗長構成にし、さらに制御ドメインのシステムボリュームを内蔵ハードディスクでミラーリング構成にしています。

a. **ldm list-devicesコマンドを実行し、追加されたハードウェアリソースの状態を確認します。**

```
# ldm list-devices
CORE
    ID      %FREE   CPUSET
    128     100    (256, 257)
    132     100    (264, 265)
    136     100    (272, 273)
    140     100    (280, 281)
....
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G
    0x720000000000      32G
    0x740000000000      32G
    0x760050000000      31488M
....
```

b. **ldm add-coreコマンドとldm add-memoryコマンドを実行し、制御ドメインのCPUコアとメモリリソースを追加します。**

ldm add-core、ldm add-memoryコマンドを実行し、制御ドメインのCPUコアとメモリを追加し、ldm list-domainコマンドで追加されたことを確認します。

```
# ldm add-core 16 primary
# ldm add-memory 56G primary
# ldm list-domain
NAME            STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary         active     -n-cv-  UART    64    56G      0.0%  2h 13m
guest0          active     -n----  5100    64    64G      0.1%  43m
root-dom0       active     -n--v-  5000    32    32G      0.1%  57m
```

c. **新規に論理ドメインを作成し、CPUコアとメモリリソースを割り当てます。**

ここでは、ルートドメイン（root-dom1）と、ゲストドメイン（guest1）を追加し、CPUコア、メモリを割り当てる例を示します。なお、ドメインを構築するときに、ldm set-coreコマンドで先にCPUコア数を設定し、その後、ldm set-memoryコマンドで割り当てるメモリサイズを設定すると、論理ドメインに割り当てられるメモリブロックが分散（フラグメント化）されにくくなり、ビルディングブロックの切り離しのときにメモリ配置を考慮しやすくなります。

以下にコマンドの実行例を示します。

```
# ldm add-domain root-dom1
# ldm add-domain guest1
# ldm set-core 16 root-dom1
# ldm set-core 32 guest1
# ldm set-memory 32G root-dom1
# ldm set-memory 64G guest1
```

注－Oracle VM Server for SPARCのバージョンが3.2よりも古い場合、物理パーティションの動的再構成を実施できるようにするためには、各論理ドメインに割り当てるメモリサイズは、（CPUコア数×256MB）の倍数に設定してください。

---

d. **ルートドメインにルートコンプレックスを割り当てます。**

次の例では、追加されたSPARC M10-4Sのルートコンプレックスを、ルートドメイン（root-dom1）に割り当てています。

ldm list-ioコマンドを実行し、ルートドメイン（root-dom1）に追加するルートコンプレックスを確認します。

NAMEが「/BB1」で始まるデバイスがBB-ID 1上のデバイスで、そのデバイスのBUSに表示されている文字列がルートコンプレックスを示しています。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9
PCIE10                                      BUS    PCIE10
PCIE11                                      BUS    PCIE11
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13
PCIE14                                      BUS    PCIE14
PCIE15                                      BUS    PCIE15
....
/BB1/CMUL/NET0                              PCIE   PCIE8
/BB1/CMUL/SASHBA                            PCIE   PCIE8
/BB1/PCI0                                   PCIE   PCIE9
/BB1/PCI3                                   PCIE   PCIE10
/BB1/PCI4                                   PCIE   PCIE10
/BB1/PCI7                                   PCIE   PCIE11
/BB1/PCI8                                   PCIE   PCIE11
/BB1/CMUL/NET2                              PCIE   PCIE12
/BB1/PCI1                                   PCIE   PCIE13
/BB1/PCI2                                   PCIE   PCIE13
/BB1/PCI5                                   PCIE   PCIE14
```

```
/BB1/PCI6                                       PCIE   PCIE14
/BB1/PCI9                                       PCIE   PCIE15
/BB1/PCI10                                      PCIE   PCIE15
```

ldm add-ioコマンドを実行し、ルートドメイン（root-dom1）にルートコンプレックスを追加し、ldm list-ioコマンドを実行して確認します。

次の例では、root-dom1には、上記で表示されたオンボードデバイス（/BB1/CMUL/NET0、/BB1/CMUL/SASHBA、/BB1/CMUL/NET2）のルートコンプレックス（PCIE8とPCIE12）以外のBB1のルートコンプレックス（PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15）を追加しています。

```
# ldm add-io PCIE9 root-dom1
# ldm add-io PCIE10 root-dom1
....
# ldm list-io
NAME                                            TYPE   BUS      DOMAIN     STATUS
----                                            ----   ---      ------     ------
PCIE0                                           BUS    PCIE0    primary    IOV
PCIE1                                           BUS    PCIE1    root-dom0  IOV
PCIE2                                           BUS    PCIE2    root-dom0  IOV
PCIE3                                           BUS    PCIE3    root-dom0  IOV
PCIE4                                           BUS    PCIE4    primary    IOV
PCIE5                                           BUS    PCIE5    root-dom0  IOV
PCIE6                                           BUS    PCIE6    root-dom0  IOV
PCIE7                                           BUS    PCIE7    root-dom0  IOV
PCIE8                                           BUS    PCIE8
PCIE9                                           BUS    PCIE9    root-dom1  IOV
PCIE10                                          BUS    PCIE10   root-dom1  IOV
PCIE11                                          BUS    PCIE11   root-dom1  IOV
PCIE12                                          BUS    PCIE12
PCIE13                                          BUS    PCIE13   root-dom1  IOV
PCIE14                                          BUS    PCIE14   root-dom1  IOV
PCIE15                                          BUS    PCIE15   root-dom1  IOV
```

e. **ルートドメインのコンソールデバイスを割り当てて起動します。**

ldm set-vconsoleコマンドでルートドメイン（root-dom1）にコンソールデバイスを割り当て、ldm bind-domain、ldm start-domainコマンドで起動します。

```
# ldm set-vconsole port=5001 vcc0 root-dom1
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# telnet localhost 5001
....
```

f. **ルートドメインにOracle Solarisをインストールします。**

起動したルートドメインに、Oracle Solarisをインストールします。Oracle Solarisのインストールの詳細については、Oracle Solarisの関連ドキュメントを参照してください。

g. **ルートドメインの仮想サービスに割り当てる物理デバイスを確認します。**

Oracle Solarisをインストールしたあと、ルートドメインにログインして、formatコマンドを実行し、仮想ディスクサービスに割り当てるデバイス名を確認します。
ルートドメインにログインする前に、制御ドメインでldm list-io -lを実行すると、PCIeエンドポイントに対応するデバイスパスが表示されるので、formatコマンドで表示されるディスクのデバイスパス名と比較することで、ディスクが接続されているパスが確認できます。

```
# ldm list-io -l
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
....
/BB1/PCI0                                 PCIE   PCIE9    root-dom1OCC
[pci@8900/pci@4/pci@0/pci@0]
/BB1/PCI3                                 PCIE   PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@0]
/BB1/PCI4                                 PCIE   PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@8]....
# telnet localhost 5001
....
root-dom1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
       0. c3t500000E01BDA70B2d0 <FUJITSU-MBB2147RC-3703 cyl 14087 alt 2 hd 24
sec 848>
          / pci@8900/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w500000e01bda70b2,0
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8a00/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
Specify disk (enter its number): ^C
```

続けて、dladm show-phys -Lコマンドを実行すると、ルートドメインに割り当てられたイーサネットのインターフェース名と物理位置が確認できます。

```
root-dom1# dladm show-phys -L
LINK              DEVICE       LOC
net0              igb0         /BB1/PCI0
net1              igb1         /BB1/PCI0
net2              igb2         /BB1/PCI0
net3              igb3         /BB1/PCI0
```

h. **ルートドメインに仮想I/Oサービスを割り当てます。**

制御ドメインに戻り、ルートドメイン(root-dom1)に仮想I/Oサービスを追加します。I/Oデバイスの詳細な設定については、Oracle社のドキュメント『Oracle VM Server for SPARC 管理ガイド』を参照してください。

次の例では、追加されたルートドメイン（root-dom1）に、仮想ディスクサービス（vds1）と仮想スイッチネットワークインターフェース（vsw10、vsw11）を追加しています。

```
# ldm add-vdiskserver vds1 root-dom1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol0@
vds1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol1@
vds1
# ldm add-vswitch net-dev=net1 vsw10 root-dom1
# ldm add-vswitch net-dev=net2 vsw11 root-dom1
```

i. **ゲストドメインに仮想I/Oデバイスを割り当てます。**

追加した仮想I/Oサービスを使った、仮想I/Oデバイスを各ゲストドメインに追加します。

次の例では、既存のゲストドメイン（guest0）にroot-dom1側の仮想I/Oサービスを使った仮想I/Oデバイス（vdisk10、vnet10）を追加し、新たに追加するゲストドメイン（guest1）には、root-dom0とroot-dom1の仮想I/Oサービスを使った仮想 I/Oデバイス（vdisk1、vdisk11、vnet1、vnet11）、仮想コンソール（vcons）を追加しています。

```
# ldm add-vdisk vdisk10 vol0@vds1 guest0
# ldm add-vnet vnet10 vsw10 guest0
# ldm add-vdisk vdisk1 vol1@vds0 guest1
# ldm add-vdisk vdisk11 vol1@vds1 guest1
# ldm add-vnet vnet1 vsw1 guest1
# ldm add-vnet vnet11 vsw11 guest1
# ldm set-vconsole port=5101 guest1
```

j. **追加したゲストドメインを起動します。**

ldm bind-domain、ldm start-domainコマンドを実行して、追加したゲストドメイン（guest1）を起動します。

```
# ldm bind-domain guest1
# ldm start-domain guest1
LDom guest1 started
```

k. **追加したゲストドメインにOracle Solarisをインストールします。**

追加したゲストドメインにOracle Solarisをインストールします。Oracle Solarisのインストールの詳細は、Oracle Solarisの関連ドキュメントを参照してください。

l. **各ゲストドメインのI/Oを冗長構成にします。**

各ゲストドメインに追加した、仮想I/Oデバイスを冗長構成に変更します。冗長構成の設定方法については、各冗長構成のソフトウェアの関連ドキュメントを参照してください。

m. **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、追加されたSPARC M10-4Sのオンボードデバイスを制御ドメインに割り当てて冗長構成にする例を示します。

制御ドメインに、追加したSPARC M10-4Sのルートコンプレックス（PCIE8、PCIE12）を追加します。

```
# ldm add-io PCIE8 primary
# ldm add-io PCIE12 primary
```

次に、システムボリュームを冗長構成にします。次の例では、formatコマンドでディスクを確認し、ZFSでミラー構成にしています。冗長構成の設定方法については、各冗長構成のソフトウェアの関連ドキュメントを参照してください。

```
# format
Searching for disks...done


AVAILABLE DISK SELECTIONS:
       0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0

errors: No known data errors
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting
```

zpool statusコマンドで、ミラー構成が組み込まれたことを確認します。このとき、zpool statusコマンドを実行し、同期処理（resilver）が完了したかどうかを確認してください。次の例では、同期処理中を表しています。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:
```

```
        NAME                          STATE     READ WRITE CKSUM
        rpool                         DEGRADED     0     0     0
          mirror-0                    DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   DEGRADED     0     0     0  (resilvering)

errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0

errors: No known data errors
```

n. **各論理ドメインのメモリ配置を確認して調整します。**

ldm list-devices -a memoryを実行して、メモリブロックの配置を確認します。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom1
    0x720000000000      32G             guest1
    0x740000000000      32G             guest1
    0x760000800000      1272M           _sys_
    0x760050000000      16G             primary
    0x760450000000      10G             primary
    0x7606d0000000      4864M
    0x780000000000      32G             guest0
    0x7a0000000000      32G             guest0
    0x7c0000000000      256M            primary
    0x7c0010000000      3584M           root-dom0
    0x7c00f0000000      24832M          primary
    0x7c0700000000      4G              root-dom0
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      25088M          root-dom0
    0x7e06a0000000      3584M           primary
    0x7e0780000000      2G              primary
```

論理ドメインに割り当てられているメモリブロックや、未割り当てのメモリブロックが小さく分割されている場合、論理ドメインをアンバインドしてか

らバインドしなおすことで、分割されたメモリブロックを大きな連続領域にまとめなおすことができます。これにより、あとでSPARC M10-4Sの追加や削除がしやすい状態になります。

手順の詳細は、「A.2　XCP 2220以降から新規導入の場合で、ハードウェアリソースに空きがない論理ドメインの構成の場合」の「A.2.2　物理パーティションの構築手順例」にある手順11を参照してください。

o. **論理ドメイン構成情報を保存します。**

ldm list-domainコマンドを実行し、構成した論理ドメインが起動されていることを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      64%   2h 54m
guest0           active     -n----  5100    64    64G      42%   2h 54m
guest1           active     -n----  5101    64    64G      11%   2h 54m
root-dom0        active     -n--v-  5000    32    32G      20%   2h 54m
root-dom1        active     -n--v-  5001    32    32G      18%   2h 54m
```

ldm add-spconfigコマンドを実行し、論理ドメイン構成情報をXSCFに保存します。

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4   [current]
```

A.5　SPARC64 Xプロセッサだけで構成された物理パーティションにSPARC64 X+プロセッサで構成されたシステムボードを増設する場合

ここでは、SPARC M10で、XCP 2210以前のXCPファームウェアが適用されたSPARC64 Xプロセッサ搭載の物理パーティションに、SPARC64 X+プロセッサ搭載のSPARC M10-4Sを増設するための、環境構築例および手順例を説明します。

A.5.1 構成例

ここでは、SPARC64 Xプロセッサ搭載の1つのSPARC M10-4Sで構成される物理パーティションに、SPARC64 X+プロセッサ搭載のSPARC M10-4Sを増設し、それぞれのSPARC M10-4Sの活性交換が実施できるようにするための構成例を示します。
PPAR DRでSPARC M10-4Sの活性交換を実施できるようにするために、以下の条件を満たす構成になっている必要があります。

- 制御ドメインのシステムボリュームのI/Oデバイスやネットワークは、各SPARC M10-4S上のルートコンプレックス配下のI/Oデバイスを接続して、冗長構成にする必要があります。これはSPARC M10-4Sを削除した場合に、制御ドメインが再起動されることによりI/Oデバイスの構成が変更されても、残りのSPARC M10-4SのI/Oデバイスを使用して、動作を継続できるようにするためです。
- SPARC M10-4Sごとにルートドメインを分ける必要があります。これは、SPARC M10-4Sを削除したことで、ほかのSPARC M10-4Sの物理I/Oデバイスを使用したサービスが提供できなくなってしまうことを回避するためです。
- 業務は仮想I/Oデバイス（vdisk、vnet）で構成されたゲストドメインで運用させる必要があります。これは、SPARC M10-4S削除時にルートドメインを停止して物理I/Oデバイスを切り離せるようにするとともに、業務を行っているゲストドメインには影響を与えないようにするためです。

図 A-6　システム構成例（1BB構成:SPARC64 Xプロセッサ搭載）（増設前）

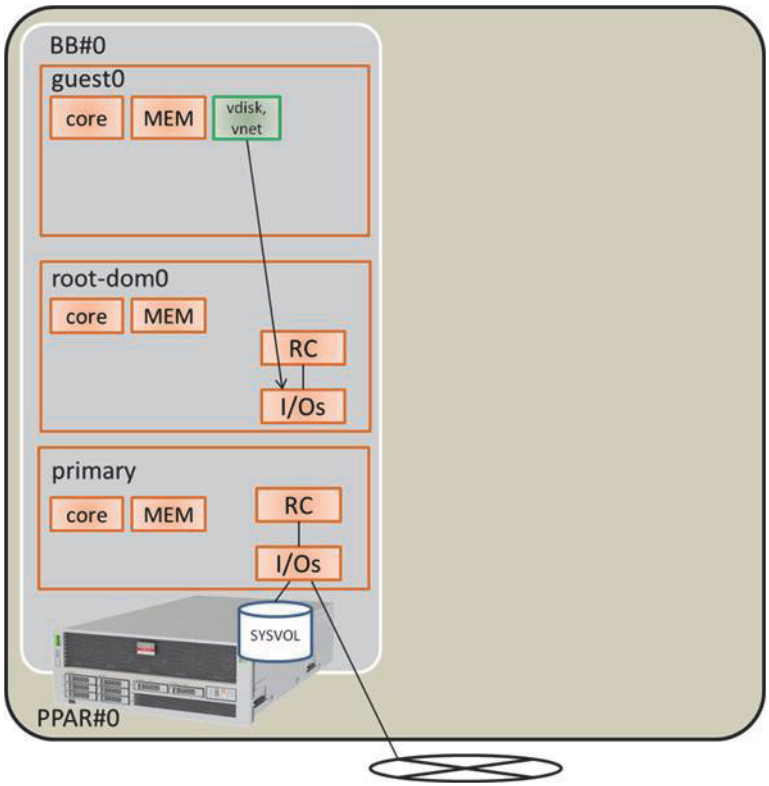


図 A-7　PPAR DR可能なシステム構成例（2BB構成:SPARC64 X+プロセッサ搭載システムボード増設後）

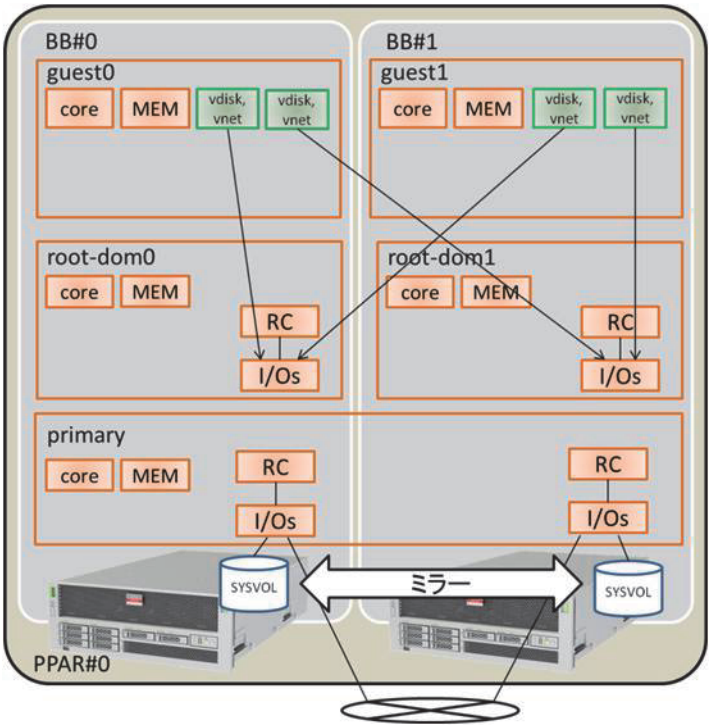


表 A-14　1BBから2BBへ増設する構成例

論理ドメイン	増設前			増設後		
	CPUコア	メモリ	I/O構成	CPUコア	メモリ	I/O構成
制御ドメイン（primary）	16	28 GB	オンボード#0（PCIE0、PCIE4）	32	56 GB	オンボード#0（PCIE0、PCIE4） オンボード#1（PCIE8、PCIE12）
guest0	32	64 GB	vdisk0、vnet0	32	64 GB	vdisk0、vdisk10 vnet0、vnet10
guest1	-	-	-	32	64 GB	vdisk1、vdisk11 vnet1、vnet11
root-dom0	16	32 GB	PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7、vds0、vsw0	16	32 GB	PCIE1、PCIE2、PCIE3、PCIE5、PCIE6、PCIE7、vds0、vsw0
root-dom1	-	-	-	16	32 GB	PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15、vds1、vsw1
未割り当てのリソース	0	約2 GB (*1)	-	0	約4.75 GB(*1)	-

表 A-14　1BBから2BBへ増設する構成例 (続き)

論理ドメイン	増設前			増設後		
	CPUコア	メモリ	I/O構成	CPUコア	メモリ	I/O構成
合計	64	128 GB	-	128	256 GB	-

*1:　各SPARC M10-4S上の約2 GBまたは約1.25 GBのメモリ領域はハイパーバイザのために割り当てられるため、論理ドメインに割り当て可能なメモリリソースは、物理的に搭載されたメモリ容量よりも少なくなります。

A.5.2　構築手順

XCP 2210以前のXCPファームウェアが適用されたSPARC64 Xプロセッサ搭載の物理パーティションに、SPARC64 X+プロセッサ搭載のSPARC M10-4Sを増設するための手順例は、次のとおりです。

1. **論理ドメイン構成情報をXMLファイルに保存します。**

   XCPファームウェアをアップデートしたあとPPAR DR機能を有効にすると、論理ドメイン構成は工場出荷時の状態（factory-default）に戻るため、XSCFに保存した論理ドメイン構成情報が使用できなくなります。そのため、現在の論理ドメイン構成情報をXMLファイルに保存してからXCPファームウェアをアップデートします。アップデート後、XMLファイルから論理ドメイン構成情報を復元することで、論理ドメインの再構築を簡単に実施できます。

   ここでは、XSCFに保存された論理ドメイン構成情報をXMLファイルに保存する手順を説明します。

   なお、XMLファイルに保存される構成情報の詳細は、『Oracle VM Server for SPARC 管理ガイド』を参照してください。

   a. **制御ドメインにログインします。**

   b. **保存する構成情報に切り替えます。**

      制御ドメイン上でldm list-spconfigコマンドを実行し、論理ドメイン構成情報一覧を表示します。[current]と表示されている行が、現在適用されている構成情報を示しています。次の例ではldm-set3が適用されています。

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [current]
```

      [next poweron]のみが表示されている場合は、XSCFと制御ドメインに保存されている論理ドメイン構成情報に差異があるため、ldm add-spconfigコマンドを実行し、現在の論理ドメイン構成情報を別の名前で保存しておきます。

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [next poweron]
```

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

[current] で示されている論理ドメイン構成情報が保存したい構成情報と一致している場合は、手順1.cに進んでください。

異なる場合は、ldm set-spconfigコマンドを実行し、保存したい構成情報を変更してください。

以下の例は、ldm-set1を保存するために、ldm set-spconfigコマンドで構成情報を切り替えています。

```
# ldm set-spconfig ldm-set1
# ldm list-spconfig
factory-default
ldm-set1 [next poweron]
ldm-set2
ldm-set3 [current]
```

XSCF上でpoweroffコマンドおよびpoweronコマンドを実行し、物理パーティション（PPAR）の電源を切断後、再投入します。このとき、論理ドメインの状態を、Oracle Solarisが起動している状態かinactive状態に移行させてから、poweroffコマンドを実行してください。

次の例は、PPAR-ID 0の電源を切断後、再投入しています。

```
XSCF> poweroff -p 0
XSCF> poweron -p 0
```

ldm list-spconfigコマンドを実行し、指定した論理ドメイン構成情報に設定されていることを確認します。次の例では、ldm-set1が現在の構成情報に設定されています。

```
# ldm list-spconfig
factory-default
ldm-set1 [current]
ldm-set2
ldm-set3
```

c. **ldm list-constraintsコマンドを実行し、現在の論理ドメイン構成情報をXMLファイルに保存します。**

ldm list-constraintsコマンドを実行し、現在の論理ドメイン構成情報をXMLファイルに保存します。

以下の例は、ldm-set1.xmlに現在の論理ドメイン構成情報を保存しています。保存したXMLファイルが消失することを防ぐため、別媒体などへバックアップしておきます。

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

d. **XMLファイルに保存されたかどうかを確認します。**

Oracle Solarisのmoreコマンドなどを実行し、XMLファイルに内容が保存されていることを確認します。

```
# more /ldm-set1.xml
<?xml version="1.0"?>
<LDM_interfaceversion="1.3" xmlns:xsi=http://www.w3.org/2001/
XMLSchema-instancce
```

e. **ほかの論理ドメイン構成情報を保存する場合は、手順1.b～1.eを繰り返します。**

ほかに保存したい構成情報がある場合、手順1.b～1.eを繰り返して保存してください。

2. **XSCF設定情報を保存します。**

XSCF設定情報は、USBデバイス、またはネットワーク経由で外部サーバに保存できます。以下に、それぞれの方法を説明します。

- **USBデバイスにXSCF設定情報を保存する場合**

a. **マスタXSCFにログインします。**

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。

注ーシステムがビルディングブロック構成の場合、スタンバイ状態のXSCFにログインしているときは、マスタXSCFへログインしなおしてください。

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

b. **マスタXSCFのUSBデバイスに設定情報を保存します。**

i. **マスタXSCFのXSCFユニットのパネル（背面パネル）にあるUSBポートにUSBデバイスを接続します。**

ii. **XSCF上のローカルなUSBデバイスに対して出力ファイル名を指定して、dumpconfigコマンドを実行します。**

指定したファイル名で、base64エンコーディングテキスト形式で保存されます。

```
XSCF> dumpconfig file:///media/usb_msd/backup-file.txt
operation completed
```

「operation completed」のメッセージが表示されたら、データの転送が正常に終了しています。

iii. **データの転送が終了したら、USBデバイスをUSBポートから取り外しま**

す。

c. **PCのエディタ機能などを使用し、保存した設定ファイルの情報を確認します。**

以下の情報を確認します。

- User-Comments：dumpconfigコマンドに-cオプションを指定したときのコメント
- Created：保存した日時
- Platform：モデル名
- Serial-No：システムのシリアル番号

注ーUSBデバイスに保存されたファイルは、XSCF上では開けません。ダウンロードして確認する必要があります。

```
XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByaXNlAAAAAAAAAFLmZ6gAAPrfADhbdAAAAAIyMTExMjM0MDAzAAAA
...
```

- **ネットワーク経由で外部サーバに設定情報を保存する場合**

a. **マスタXSCFにログインします。**

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。

注ーシステムがビルディングブロック構成の場合、スタンバイ状態のXSCFにログインしているときは、マスタXSCFにログインしなおします。

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

b. **ネットワークを介してターゲットディレクトリを指定し、XSCF設定情報を保存します。**

ターゲットディレクトリおよび出力ファイル名を指定して、dumpconfigコマンドを実行します。指定したファイル名で、base64エンコーディングテキスト形式で保存されます。

user_nameには、保存先のサーバのユーザー名を指定します。

```
XSCF> dumpconfig -u user-name ftp://server/backup/backup-sca-ff2-16.txt
operation completed
```

「operation completed」のメッセージが表示されたら、データの転送が正常に終了しています。

c. **PCのエディタ機能などを使用し、保存した設定ファイルの情報を確認します。**

以下の情報を確認します。

- User-Comments: dumpconfigコマンドに-cオプションを指定したときのコメント
- Created: 保存した日時
- Platform: モデル名
- Serial-No: システムのシリアル番号

```
XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
```

```
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByaXNlAAAAAAAAAFLmZ6gAAPrfADhbdAAAAAIyMTExMjM0MDAzAAAA
...
```

3. **Oracle SolarisとOracle VM Server for SPARCをアップデートします。**
物理パーティションの動的再構成をサポートしているOracle SolarisおよびOracle VM Server for SPARCを入手し、システムに適用します。

a. **制御ドメインのOracle VM Server for SPARCをアップデートします。**

お使いのサーバの最新の『プロダクトノート』を参照して最新のOracle SolarisおよびOracle VM Server for SPARCの版数を確認し、以下の手順でOracle VM Server for SPARCをアップデートします。

i. **制御ドメインに使用する、Oracle VM Server for SPARCを入手します。**

- **制御ドメインがOracle Solaris 10の場合**

  物理パーティションの動的再構成をサポートしているOracle VM Server for SPARCを入手します。入手方法は、お使いのサーバの最新の『プロダクトノート』を参照してください。

- **制御ドメインがOracle Solaris 11の場合**

  SRU11.1.14.0以降を入手します。

ii. **制御ドメインのOracle VM Server for SPARCやOracle Solarisをアップデートした場合は、制御ドメインを再起動してください。**

制御ドメインの再起動によって、I/Oドメインがパニックしたり、ゲストドメインのI/Oが停止したりすることがあるため、これらの論理ドメインがある場合は、事前にshutdownコマンドで停止させるか、制御ドメインからldm stop-domainコマンドを実行して停止させてください。

停止させる必要のある論理ドメインかどうかは、次の方法で確認できます。

- **制御ドメインのからPCIeエンドポイントを割り当てているI/Oドメインの場合**

  ldm list-ioコマンドを実行し、論理ドメインに割り当てているPCIeエンドポイントのルートコンプレックス（BUS）が、primaryに割り当てられているかどうかで判断できます。次の例では、iodom0に割り当てているPCIeエンドポイント「/BB0/PCI3」と「/BB0/PCI4」のバス「PCIE2」がprimary（制御ドメイン）に割り当てられていることがわかります。なお、ここで説明する構成は「A.5.1　構成例」と異なるため、論理ドメインなどを変えて説明しています。

```
# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
```

```
PCIE0                                   BUS    PCIE0    primary  IOV
PCIE1                                   BUS    PCIE1    primary  IOV
PCIE2                                   BUS    PCIE2    primary  IOV
PCIE3                                   BUS    PCIE3    primary  IOV
....
/BB0/CMUL/NET0                          PCIE   PCIE0    primary  OCC
/BB0/CMUL/SASHBA                        PCIE   PCIE0    primary  OCC
/BB0/PCI0                               PCIE   PCIE1    primary  OCC
/BB0/PCI3                               PCIE   PCIE2    iodom0   OCC
/BB0/PCI4                               PCIE   PCIE2    iodom0   OCC
/BB0/PCI7                               PCIE   PCIE3    primary  OCC
/BB0/PCI8                               PCIE   PCIE3    primary  EMP
....
```

- **制御ドメインの仮想サービスを割り当てているゲストドメインの場合**
  ldm list-bindings primaryを実行し、仮想ネットワークスイッチ（VSW）と接続先（PEER）の対応、仮想ディスクサービス（VDS）と仮想ディスクを使用している論理ドメイン（CLIENT）の対応を確認します。次の例の場合、VSWのPEERと、VDSのCLIENTに「guestdom0」が設定されていることが確認できます。

```
# ldm list-bindings primary
....
VSW
    NAME             MAC               NET-DEV   ID   DEVICE     LINKPROP
DEFAULT-VLAN-ID PVID
VID                  MTU   MODE   INTER-VNET-LINK
    vsw0             xx:xx:xx:xx:xx:xx net0      0    switch@0             1
            1                        1500         on
        PEER                        MAC               PVID VID
  MTU   MAXBW      LINKPROP
 INTERVNETLINK
        vnet0@guestdom0             xx:xx:xx:xx:xx:xx 1
  1500
VDS
    NAME             VOLUME         OPTIONS          MPGROUP        DEVICE
....
        CLIENT                      VOLUME
        vdisk0@guestdom0            vol0
```

ldm stop-domainコマンドを実行して、上記で確認した論理ドメインを停止させます。次の例では、ldm stop-domainコマンドでiodom0とguestdom0を停止させ、その後、ldm list-domainコマンドを実行してbound状態になったことを確認しています。

```
# ldm stop-domain guestdom0
LDom guestdom0 stopped
# ldm stop-domain iodom0
LDom iodom0 stopped
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    58G      0.0%  6h 3m
```

```
guestdom0          bound      ------  5100    64    64G
iodom0             bound      ------  5000    32    32G
```

iii. **制御ドメインのOracle VM Server for SPARCをアップデートします。**

- **Oracle Solaris 10の場合**

  古いOracle VM Server for SPARCをアンインストールしてから、新しいOracle VM Server for SPARCをインストールします。詳細は、入手したOracle VM Server for SPARCのREADMEファイルを参照してください。

- **Oracle Solaris 11の場合**

  SRUを適用します。詳細は、入手したSRUのインストール手順書を参照してください。

iv. **shutdownコマンドを実行し、制御ドメインを再起動します。**

```
# shutdown -i6 -g0 -y
....
```

v. **ldm start-domainコマンドを実行し、上記iiで停止させた論理ドメインを起動させます。**

次の例では、ldm start-domainコマンドで「iodom0」と「guestdom0」を起動させたあと、ldm list-domainコマンドでactive状態になったことを確認しています。

```
# ldm start-domain guestdom0
# ldm start-domain iodom0
# ldm list-domain
NAME               STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary            active     -n-cv-  UART    64    58G      0.0%  6h 3m
guestdom0          active     -n----  5100    64    64G
iodom0             active     -n----  5000    32    32G
```

b. **制御ドメイン以外の論理ドメインのOracle Solarisをアップデートします。**

お使いのサーバの最新の『プロダクトノート』を参照して、制御ドメイン以外の論理ドメインのOracle Solarisをアップデートしてください。アップデートの手順は、各アップデート関連の情報を参照してください。

4. **XCPファームウェアをアップデートします。**

SPARC64 Xプロセッサで構成された物理パーティションに、SPARC64 X+プロセッサ搭載のSPARC M10-4Sを増設するためには、SPARC64 Xプロセッサで構成された物理パーティションをXCP 2220以降のXCPファームウェアにアップデートします。

a. **最新XCPファームウェアを入手します。**

i. **XCPファームウェアのプログラムファイルをダウンロードします。**

ウェブサイトから本システムに接続しているPCの任意のフォルダにXCPファームウェアのプログラムファイル（XCPxxxx.tar.gzかXCPxxxx.exe）をダウンロードします。

お使いのサーバのファームウェアは、以下の方法で入手してください。

- **国内サイト**

  ファームウェアは、SupportDeskを契約されているお客さまは、SupportDesk-Webから入手が可能です。

- **グローバルサイト**

  ファームウェアの最新ファイルの入手方法については、営業担当者にお問い合わせください。以下のファイルが提供されます。

  - ファームウェアプログラムファイル（XSCF Control Package（XCP）ファイル）
  - XSCF拡張MIB（XSCF-SP-MIB）定義ファイル

ii. **ダウンロードしたプログラムファイルのXCP版数を確認します。**

ダウンロードしたXCPファームウェアのプログラムファイルの版数を確認します。XCP版数はファームウェアプログラム（tar.gz形式）のファイル名にある数字4桁を参照し、アップデートするXCPファームウェア版数であることを確認します。例えば、プログラムファイルの名前が「XCP2220.tar.gz」の場合は2220がXCP版数です。

iii. **ダウンロードしたプログラムファイルを解凍します。**

ダウンロードしたXCPファームウェアのプログラムファイルを解凍します。システムにインポートするためのXCPイメージファイルが展開されます。例えば、「XCP2220.tar.gz」を解凍した場合、「BBXCP2220.tar.gz」が展開されます。

b. **現在のXCPファームウェア版数を確認します。**

i. **マスタXSCFにログインします。**

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。

注ーシステムがビルディングブロック構成の場合、スタンバイ状態のXSCFにログインしているときは、マスタXSCFにログインしなおします。

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

ii. **versionコマンドを実行し、現在のシステムのXCP版数を確認します。**

ファームウェアをアップデートする前に、現在のシステムのXCP版数を必ず確認してください。次の例では、-c xcpオプションを付けてversionコマンドを実行し、XCP版数が物理パーティションの動的再構成をサポートしていないXCP 2041であることを確認しています。

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2041
XCP1 (Reserve): 2041
```

c. **XCPファームウェアをアップデートします。**

i. **poweroffコマンドを実行し、すべての物理パーティションを電源切断します。**

```
XSCF> poweroff -a
```

showpparstatusコマンドを実行し、すべての物理パーティションの電源が切断されていることを確認します。

```
XSCF> showpparstatus -a
PPAR-ID         PPAR Status
0               Powered Off
```

ii. **showhardconfコマンドを実行し、マスタXSCF、スタンバイ状態のXSCFの [Status] が「Normal」になっていることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Service;
    + System_Power:Off; System_Phase:Cabinet Power Off;
    Partition#0 PPAR_Status:Powered Off;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
   :
```

iii. **getflashimageコマンドを実行し、XCPイメージファイルをインポートします。**

次の例では、マスタXSCFのXSCFユニットのパネル（背面パネル）にあるUSBポート（MAINTENANCE ONLYと印字）にUSBデバイスを接続して、XCPイメージファイルをインポートしています。

```
XSCF> getflashimage file:///mnt/share/scf-firm/xscf/user/scfadmin/BBXCP2220.tar.gz
  0MB received
  1MB received
  2MB received
...
 86MB received
 87MB received
 88MB received
Download successful: 90539 Kbytes in 58 secs (1562.668 Kbytes/sec) Checking
file...
MD5: 2b89c06548205ce35a8ecb6c2321d999
```

「Download successful: ...」および「MD5: ...」の正常終了のメッセージが表示されたら、XCPイメージファイルのインポートは終了です。

注ーXCPイメージファイルインポート時に「Warning: About to delete existing old versions.」のメッセージが表示されることがあります。これは、インポート済みである古い

XCPイメージファイルを削除するかを確認するメッセージです。「Continue?」が表示されたら、「y」を入力して、インポート処理を継続してください。

注－XCPイメージファイルインポート後に「Error:File is invalid or corrupt」のメッセージが表示された場合は、インポートしたXCPイメージファイルが正しいファイルでなかったことを意味します。XCPイメージファイルが壊れている可能性があるため、再度、正しいXCPイメージファイルを入手し、インポートしてください

---

iv. **getflashimage -lコマンドを実行し、インポートしたXCPイメージファイルの版数を確認します。**

```
XSCF> getflashimage -l
Existing versions:
        Version                 Size  Date
        BBXCP2220.tar.gz    92712351  Thu May 23 15:01:42 JST 2014
```

v. **flashupdate -c checkコマンドを実行し、インポートしたXCPイメージファイルがアップデートに使用できるファイルかどうかを確認します。**

flashupdateコマンドの実行直後にshowresultコマンドを実行し、終了値が0であればアップデートできます。

```
XSCF> flashupdate -c check -m xcp -s 2220
XCP update is started. [3600sec]
0XSCF>
XSCF> showresult
0
XSCF>
```

vi. **flashupdateコマンドを実行し、ファームウェアをアップデートします。**

```
XSCF> flashupdate -c update -m xcp -s 2220
The XSCF will be reset. Continue? [y|n]: y
XCP update is started. [3600sec]
0.....30.....60.....90.....120.....150.....180.....210.....240.....-
270.....300.....330.....360.....390.....420.....450.....480.....510.....|
540.....570.....600.....630.....660.....690.....720.....750.....780.....-
810.....840.....870.....900.....930
:
```

ここでXSCFが再起動され、XSCFのセションが切断されます。この時点では、XCPファームウェアのアップデートはまだ完了していません。

注－ファームウェアの作業時間は、アップデートに45分かかります。システムがビルディングブロック構成となっている場合、アップデートに約60分、アップデート完了後のXSCFの自動切り替えに約10分かかります。

注－アップデートを安全に行うため、XCPファームウェアのアップデートが完了したことを表すメッセージ「XCP update has been completed」を確認されるまで、物理パーティションの電源は操作しないでください。

---

vii. **再度マスタXSCFに接続します。**

注ーシステムがビルディングブロック構成となっている場合、XSCFの再起動直後は、マスタとスタンバイ状態のXSCFが元の状態と逆になっています。例えば、ファームウェアアップデートをBB-ID 0のマスタXSCFで実行した場合にXSCFに再接続すると、BB-ID 1がマスタ、BB-ID 0がスタンバイ状態になっています。

注ー引き継ぎIPアドレスを設定している場合は、引き継ぎIPアドレスで接続すると自動的にマスタXSCFに接続されます。

---

viii. **showbbstatusコマンドを実行し、マスタXSCFにログインしていることを確認します。**

注ーシステムがビルディングブロック構成の場合、スタンバイ状態のXSCFにログインしているときは、マスタXSCFにログインしなおします。

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

ix. **showlogs monitorコマンドを実行し、XCPファームウェアアップデートの完了を確認します。**

「XCP update has been completed」のメッセージが表示されていれば、XCPファームウェアのアップデートが完了しています。

```
XSCF> showlogs monitor
Jan 23 16:42:57 5 M10-4S-0 Event: SCF:XCP update is started (XCP version=2220:
last version=2041)
Jan 23 16:44:52 M10-4S-0 Event: SCF:XSCF update is started (BBID=0, bank=0)
Jan 23 16:45:11  M10-4S-0 Event: SCF:XSCF writing is performed (BBID=0, XSCF
version=02220000)
:
Jan 23 16:54:05 01:50 M10-4S-0 Event: SCF:XSCF bank apply has been completed
(BBID=0, bank=0, XCP version=2220: last version=2041)
:
Jan 23 17:19:57 32:38 M10-4S-1 Event: SCF:CMU update has been completed
(BBID=0)
Jan 23 17:19:59 32:41 M10-4S-1 Even SCF:XCP update has been completed (XCP
version=2220: last version=2041)
```

注ー「XCP update has been completed」のメッセージが表示されていない場合は、まだアップデートが完了していません。再度、showlogs monitorコマンドを実行し、アップデートの完了を確認してください。

---

x. **showhardconfコマンドを実行し、XSCFの [Status] が「Normal」になっていることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Service;
    + System_Power:Off; System_Phase:Cabinet Power Off;
    Partition#0 PPAR_Status:Powered Off;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
   :
```

xi. **versionコマンドを実行し、ファームウェアの版数が新しくなっていることを確認します。**

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
```

注－物理パーティションの電源を投入してファームウェアをアップデートした場合、CMUファームウェアのカレントバンク（Current bank）が、新しくなっています。物理パーティションの電源を停止してファームウェアアップデートした場合、CMUファームウェアのリザーブバンク（Reserve bank）およびカレントバンク（Current bank）の両方が新しくなっています。XCP版数と対応するCMUファームウェアの版数は、最新のプロダクトノートの「これまでのXCPファームウェア版数とサポート情報」を参照してください。

---

5. **物理パーティションの動的再構成機能（PPAR DR機能）を有効にします。**

注－SPARC64 XプロセッサとSPARC64 X+プロセッサの各SPARC M10-4Sを1つの物理パーティションで混在する場合は、setpparmodeコマンドを実行し、CPU動作モードをcompatibleモード（SPARC64 X互換モード）に設定してください。詳細は、「2.6.3 SPARC64 X+プロセッサとSPARC64 Xプロセッサを PPAR内で混在させるための条件」を参照してください。

注－物理パーティションの動的再構成を使用するためには、setpparmodeコマンドを使用してPPAR DRモードを有効に設定する必要があります。ただし、PPAR DRモードを有効に設定すると、論理ドメイン構成情報はfactory-defaultに変更され、既存の論理ドメイン構成情報は使用できなくなります。新たに論理ドメイン構成情報を作成する必要があります。
既存の論理ドメイン構成情報を使用する場合は、PPAR DRモードを無効にしてください。有効に設定し、かつ既存の論理ドメイン構成情報で物理パーティションの入力電源を投入すると、HypervisorでAbortする、Oracle VM Server for SPARCが正常に起動しない、などの現象が発生します。

---

a. **showpparmodeコマンドを実行して、PPAR DRモードの設定を参照します。**

次の例は、物理パーティション番号0（PPAR ID 0）を指定しています。

```
XSCF> showpparmode -p 0
Host-ID                    :00000002
Diagnostic Level           :min
Message Level              :normal
Alive Check                :on
Watchdog Reaction          :reset
Break Signal               :on
Autoboot(Guest Domain)     :on
Elastic Mode               :off
IOreconfigure              :false
PPAR DR(Current)           :-
PPAR DR(Next)              :off
```

b. **CPU動作モードがautoの場合は、setpparmodeコマンドを実行してcompatibleに設定します。**

```
XSCF> setpparmode -p 0 -m cpumode=compatible
```

c. **PPAR DRモードが無効の場合は、setpparmodeコマンドを実行し、有効に設定します。**

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

d. **showpparmodeコマンドを実行して、PPAR DRモードが有効に設定されていることを確認します。**

```
XSCF> showpparmode -p 0
Host-ID                    :00000002
Diagnostic Level           :min
Message Level              :normal
Alive Check                :on
Watchdog Reaction          :reset
Break Signal               :on
Autoboot(Guest Domain)     :on
Elastic Mode               :off
IOreconfigure              :false
PPAR DR(Current)           :-
PPAR DR(Next)              :on
```

6. **showsscpコマンドを実行し、サービスプロセッサ間通信プロトコル（SSCP）のIPアドレスがデフォルト値か、ユーザーの設定した値か確認します。**

```
XSCF> showsscp
```

IPアドレスがデフォルト値で、増設するSPARC M10-4SのIPアドレスもデフォルト値にする場合、次の手順へ進みます。

ユーザー値を設定する場合、setsscpコマンドでIPアドレスを設定し、applynetworkコマンドで増設するSPARC M10-4SのSSCPのIPアドレスを反映さ

せて確認します。その後、rebootxscfコマンドを実行し、設定を完了させてから次の手順へ進みます。手順の詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.6　ネットワーク設定を適用する」を参照してください。

7. **SPARC M10-4Sを増設します。**

   a. **addfruコマンドを実行し、メッセージに従って SPARC M10-4Sを増設します。**

注－SSCPのIPアドレスが設定されていない場合、addfruコマンドはエラーとなります。

注－addfruコマンドを実行すると、増設するSPARC M10-4Sのファームウェアの版数は、マスタXSCFが動作しているSPARC M10-4Sのファームウェアの版数に自動的に合わされます。この例では、増設されるSPARC M10-4S（BB-ID 1）のファームウェアの版数は、マスタXSCFが動作するSPARC M10-4S（BB-ID 0）のファームウェアに、自動的に版数が合わされます。

次の例では、BB#1が増設されています。

```
XSCF> addfru
--------------------------------------------------------------------------------
Maintenance/Addition Menu
Please select the chassis including added FRU.
No. FRU                  Status
--- ------------------- --------------
 1  /BB#0                Normal
 2  /BB#1                Unmount
 3  /BB#2                Unmount
 4  /BB#3                Unmount
--------------------------------------------------------------------------------
Select [1-16|c:cancel] :2
Maintenance/Addition Menu
Please select the BB or a type of FRU to be added.
1. BB itself
2. PSU (Power Supply Unit)
--------------------------------------------------------------------------------
Select [1,2|c:cancel] :1
Maintenance/Addition Menu
Please select a FRU to be added.
No. FRU                  Status
--- ------------------- --------------
 1  /BB#1                Unmount
Select [1|b:back] :1
```

   b. **増設するSPARC M10-4Sをラックに搭載します。**

   前述のaddfruコマンド実行中に、「After the added device is connected with the system, please turn on the breaker of the BB#1.」のメッセージが表示されたら、増設するSPARC M10-4Sをラックに搭載します。

   搭載手順は、『SPARC M10-4S インストレーションガイド』の「3.4.1 SPARC M10-4Sをラックに搭載する」を参照してください。

   c. **増設するSPARC M10-4Sの識別ID（BB-ID）を設定します。**

   『SPARC M10-4S インストレーションガイド』の「4.1　筐体の識別ID

（BB-ID）を設定する」を参照してください。

d. **クロスバーケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4Sのクロスバーケーブルを接続します。接続ケーブルのルート図とケーブル一覧表は、『SPARC M10-4S インストレーションガイド』の「付録 B　ビルディングブロック構成のケーブル接続資料」を参照してください。筐体間直結で接続する場合は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

e. **XSCF BB制御ケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4SをXSCF BB制御ケーブルで接続します。XSCF BB制御ケーブルのルート図とケーブル一覧表は、『SPARC M10-4S インストレーションガイド』の「付録 B　ビルディングブロック構成のケーブル接続資料」を参照してください。筐体間直結で接続する場合は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

f. **XSCF DUAL制御ケーブルを接続します。**

既存のSPARC M10-4Sと増設したSPARC M10-4SをXSCF DUAL制御ケーブルで接続します。詳細は、『SPARC M10-4S インストレーションガイド』の「4.2　ケーブルを接続する（筐体間直結の場合）」を参照してください。

g. **シリアルケーブルやLANケーブルを接続します。**

増設するSPARC M10-4SのXSCFシリアルポートへシリアルケーブルを接続します。また、XSCF-LAN、GbEポート、PCIeカードの各LANポートへケーブルを接続します。詳細は、『SPARC M10-4S インストレーションガイド』の「5.1　SPARC M10-4Sにケーブルを接続する」を参照してください。

h. **増設するSPARC M10-4Sの入力電源を接続します。**

増設するSPARC M10-4Sの電源コードを入力電源に接続します。

i. **addfruコマンドの入力画面から「f」を入力します。**

手順a.で実行しているaddfruコマンドの入力画面で「f」を入力し、SPARC M10-4Sの増設処理を実施します。

```
2) Please select[f:finish] :f

Waiting for BB#1 to enter install state.
[This operation may take up to 20 minute(s)] (progress scale
 reported in seconds)
0.....  30.... done


Waiting for BB#1 to enter ready state.
[This operation may take up to 45 minute(s)] (progress scale
 reported in seconds)
0.....  30.....  60... done

Do you want to start to diagnose BB#1? [s:start|c:cancel] :
```

j. **増設するSPARC M10-4Sの診断をスキップし、addfruコマンドを終了させます。**

マスタXSCFにあるaddfruコマンドの入力画面で「c」を入力し、増設するSPARC M10-4Sの診断をスキップします。「The addition of BB#1 has completed.」が出力されたら「f」を入力し、最後に「c」を入力して、addfruコマンドを終了させます。

```
Do you want to start to diagnose BB#1? [s:start|c:cancel] :c

Diagnostic tests are about to be skipped.
Running diagnostic tests are strongly recommended before using BB#1.
Are you sure you want to skip testing? [y:yes|n:no] :y

--------------------------------------------------------------------------------
Maintenance/Addition Menu Status of the added FRU.

FRU                  Status
------------------- --------------
/BB#1                Normal
--------------------------------------------------------------------------------
[Warning:007]
Running diagnostic tests on BB#1 is strongly recommended after addfru has
completed.
The addition of BB#1 has completed.[f:finish] :f

--------------------------------------------------------------------------------
Maintenance/Addition Menu
Please select the chassis including added FRU.

No. FRU                  Status
--- ------------------- --------------
 1   /BB#0                Normal
 2   /BB#1                Normal


--------------------------------------------------------------------------------
Select [1,2|c:cancel] :c
```

8. **増設したSPARC M10-4Sを診断します。**
   a. **testsbコマンドを実行し、増設したSPARC M10-4Sの診断テストを実施します。**

      testsbコマンドを実行し、診断テストを実施します。増設したSPARC M10-4Sの物理システムボード(PSB)番号を指定し、初期診断と接続I/Oの確認を実施します。

```
XSCF> testsb -v -p -s -y 01-0
Initial diagnosis is about to start, Continue?[y|n] :y
PSB#01-0 power on sequence started.
```

      エラーが表示された場合は、『SPARC M10-4S インストレーションガイド』の「A.2.4　診断結果を確認する」を参照してください。

   b. **diagxbuコマンドに-tオプションを指定し、既存のSPARC M10-4Sと増設したSPARC M10-4Sのクロスバーケーブル診断テストを行います。**

次の例では、既存のSPARC M10-4Sの識別ID 00（BB-ID 00）と、増設したSPARC M10-4Sの識別ID 01（BB-ID 01）を指定し、クロスバーケーブルの接続状態を診断しています。

```
XSCF> diagxbu -y -b 00 -t 01
XBU diagnosis is about to start, Continue?[y|n] :y
Power on sequence started. [7200sec]
0..... 30..... 60..end
XBU diagnosis started. [7200sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240...../
270.....300.....330.....360.....390.....420.....450.....480.....510.....¥
540.....570.....600.....630.....660.....690.....720.....750.....780.....¥
810.....840.....870.....900.....930...end
completed.
Power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....180.end
completed.
```

c. **showlogs errorコマンドを実行し、エラーが表示されないことを確認します。**

```
XSCF> showlogs error
```

エラーが表示された場合は、『SPARC M10-4S インストレーションガイド』の「A.2.4　診断結果を確認する」を参照してください。

d. **showhardconfコマンドを実行し、増設したSPARC M10-4Sの構成や状態を確認します。**

showhardconfコマンドを実行し、増設したSPARC M10-4SのCPUやメモリなどのハードウェア構成や、各ユニットのStatusがNormalになっていることを確認します。

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Service;
    + System_Power:Off; System_Phase:Cabinet Power Off;
    Partition#0 PPAR_Status:Powered Off;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
           + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
            CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
                + Freq:3.000 GHz; Type:0x10;
                + Core:16; Strand:2;
     :
    BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
        + FRU-Part-Number: CA20393-B50X A2                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
```

```
            + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
      :
```

9. **増設したSPARC M10-4SのXSCFネットワークを設定します。**
   増設したSPARC M10-4SのXSCF用のネットワークを設定します。詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.2　イーサネット（XSCF-LAN）のIPアドレスを設定する」と「7.5.3　引き継ぎIPアドレスを設定する」を参照してください。設定後は、applynetworkコマンドで実行し、設定内容を反映させ確認します。その後、rebootxscfコマンドを実行し、設定を完了させてから次の手順へ進みます。手順の詳細は、『SPARC M10-4S インストレーションガイド』の「7.5.6　ネットワーク設定を適用する」を参照してください。

10. **増設したSPARC M10-4Sのメモリのミラーモードを設定します。**
    メモリのミラーモードを使用しない場合は、この手順は不要です。

    メモリのミラーモードの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「14.1　メモリをミラー構成にする」を参照してください。

    a. **増設したSPARC M10-4Sのメモリをミラー構成にする場合、setupfruコマンドを使用し、メモリのミラーモードを設定します。**

    次の例では、SPARC M10-4Sの物理システムボード（PSB 01-0）のすべてのCPUに対してメモリをミラーモードに設定しています。

```
XSCF> setupfru -m y sb 01-0
```

    b. **showfruコマンドを使用し、メモリのミラーモード設定を確認します。**

```
XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb     01-0
   cpu 01-0-0   yes
   cpu 01-0-1   yes
   cpu 01-0-2   yes
   cpu 01-0-3   yes
```

11. **物理パーティション構成情報に増設したSPARC M10-4Sのシステムボードを登録します。**
    a. **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
```

    b. **setpclコマンドを実行し、物理パーティション構成情報にシステムボードを登録します。**

    setpclコマンドを実行し、増設したSPARC M10-4Sのシステムボード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）01-0を、物理パーティション0の論理システムボード（LSB）01に対応付けています。

```
XSCF> setpcl -p 0 -a 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
```

12. **CPUコア アクティベーションキーを登録し、CPUコアリソースを割り当てます。**

a. **showcodusageコマンドを実行し、CPUコアリソースの情報を確認します。**

showcodusage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コアリソースがあるかを確認します。

ここでは、システムには128個のCPUコアリソースが実装されており、128個のCPUコア アクティベーションが登録され、64個のCPUコアリソースが使用中であり、現在未使用のCPUコア アクティベーション数が64個であることを示しています。

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-------- ------ --------- ------------- ------
PROC         64       128           128 OK: 64 cores available

Note:
  Please confirm the value of the "In Use" by the ldm command of Oracle VM
Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of logical
domains.
```

注－登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。CPUコア アクティベーションキーの追加については、『SPARC M12/M10 システム運用・管理ガイド』の「5.3　CPUコアリソースを追加する」を参照してください。

---

b. **showcodコマンドを実行し、物理パーティションのCPUコアリソース情報を確認します。**

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コアリソース情報を確認します。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 64
```

割り当てが不足している場合は、setcodコマンドを実行し、CPUコア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、64個のCPUコアリソースを物理パーティション0に追加しています。

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注－XCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

CPUコアリソースを割り当てた場合は、再度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コアリソース情報を確認します。

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

13. **増設したSPARC M10-4Sのシステムボード（PSB<BB>）を物理パーティションに組み込みます。**
    a. **XSCFシェルに戻り、showboardsコマンドを実行し、PSBの状態を確認します。**

    showboardsコマンドを実行し、増設したSPARCM10-4SのPSB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 -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    n    n    n    Passed  Normal
01-0 SP           Available   n    n    n    Passed  Normal
```

    b. **addboardコマンドを実行し、PSBを物理パーティションに割り当てます。**

    addboard -c assignコマンドを実行し、PSBを物理パーティションに割り当てます。

```
XSCF> addboard -c assign -p 0 01-0
PSB#01-0 will be assigned into PPAR-ID 0. Continue?[y|n] :y
XSCF>
```

注ーaddboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

---

c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

終了値が0の場合、addboardコマンドは正常終了です。

終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

増設したSPARC M10-4SのPSBが「Assigned」状態であり、Pwr欄、Conn欄とConf欄の表示がすべて「n」と表示されてい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
```

14. **物理パーティションの電源を投入します。**

showdomainconfigコマンドを実行し、起動する論理ドメイン構成情報がfactory-defaultであることを確認します。次の例では、物理パーティション番号0（PPAR ID 0）を指定しています。

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) : factory-default
(Next) : factory-default
----------------------------------------------------------------
Index :1
config_name :factory-default
domains :1
date_created:-
----------------------------------------------------------------
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-0811:34:56'
```

```
----------------------------------------------------------------
Index :3
config_name :ldm-set2
domains :20
date_created:'2012-08-0912:43:56
:
```

poweronコマンドを実行し、物理パーティションの電源を投入します。

```
XSCF> poweron -p 0
```

15. **XMLファイルを使用して、論理ドメインを再構築します。**
物理パーティションの起動が完了したら、手順1で保存したXMLファイルから論理ドメインを再構築します。

a. **制御ドメイン上でシステムがfactory-defaultで起動したことを確認します。**

ldm list-spconfigコマンドを実行し、factory-defaultの行に[current]が表示されていることを確認します。

次にコマンドの実行例を示します。

```
# ldm list-spconfig
factory-default  [current]
ldm-set1
ldm-set2
ldm-set3
```

b. **ldm init-systemコマンドでXMLファイルから論理ドメイン構成情報を反映します。**

ldm init-systemコマンドを実行し、保存されているXMLファイルの設定を反映したら、shutdownコマンドを実行し、制御ドメインを再起動します。

```
# ldm init-system -i /ldm-set1.xml
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.
# shutdown -y -g0 -i6
```

c. **Oracle VM Server for SPARCのバージョンが3.2よりも古い場合、物理パーティションの動的再構成を可能にするために、既存の論理ドメインのメモリサイズを調整します。**

ldm list-domainコマンドなどを実行し、XMLファイルから論理ドメインが再構成されたことを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    28G      0.0%  6m
```

```
guest0           inactive   ------          64    64G
root-dom0        inactive   ------          32    32G
```

物理パーティションの動的再構成を実施できるようにするには、各論理ドメインのメモリサイズを「論理ドメインのCPUコア数×256 MBの倍数」に設定する必要があります。

制御ドメイン以外の論理ドメインのメモリサイズを再設定する場合は、ldm set-memoryコマンドでメモリサイズを設定してください。

制御ドメインのメモリサイズを再設定する場合は、最初にldm start-reconfコマンドで遅延再構成モードに移行し、ldm set-coreコマンドで同じコア数を設定してから、ldm set-memoryコマンドで再設定を行い、最後にOracle Solarisを再起動してください。

この例では、制御ドメインのCPUコア数が32のため、32×256 MB＝8192 MBの倍数にする必要があります。

まず、元の設定値（58 GB=59392 MB）に近い値にするために8192 MBの何倍にすればいいかを求めると、59392÷8192=7.25になるため、切り捨てて7にします。その結果、制御ドメインに割り当てなおすメモリサイズは、8192 MB×7=56 GBになります。

以下に、制御ドメインのメモリを56 GBに再設定するコマンドの実行例を示します。

```
# 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 32 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-memory 56G 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.
------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
```

ldm list-domainコマンドを実行し、メモリサイズ（MEMORY）が正しく設定されていることを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    56G      0.0%  6m
guest0           inactive   ------          64    64G
root-dom0        inactive   ------          32    32G
```

ldm bind-domainコマンドを実行し、各論理ドメインをバインドしたあと、ldm start-domainコマンドを実行し、各論理ドメインを起動させます。

以下にコマンドの実行例を示します。

```
# ldm bind-domain root-dom0
# ldm bind-domain guest0
# ldm start-domain root-dom0
LDom root-dom0 started.
# ldm start-domain guest0
LDom guest0 started.
```

ldm list-domainコマンドを実行して、各論理ドメインが起動されたことを確認します。[STATE]が「active」になっていて、[FLAGS]の文字列の左から2番目の文字が「n」になっていることを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    32    28G      0.0%  18m
guest0           active     -n----  5100    64    64G      0.1%  2m
root-dom0        active     -n--v-  5000    32    32G      0.0%  3m
```

16. **新規に論理ドメインを追加します。**

新規に論理ドメインを構築して、追加されたSPARC M10-4Sのリソースを割り当てます。以下の例では、構成例に従って、ゲストドメインguest1、ルートドメインroot-dom1を追加し、各ゲストドメインの仮想I/Oを冗長構成にし、さらに制御ドメインのシステムボリュームを内蔵ハードディスクでミラーリング構成にしています。

a. **ldm list-devicesコマンドを実行し、追加されたハードウェアリソースの状態を確認します。**

ldm list-devicesコマンドを実行すると未割り当てのハードウェアリソースの一覧が表示されるので、ハードウェアリソースが追加されたことを確認します。次の例では、ldm list-devices -a -p coreコマンドの出力から、「free=100」の行数を数えることで、未割り当てのCPUコア数を確認しています。

```
# ldm list-devices -a -p core | grep "free=100" | wc -l
      64
```

メモリについては、prtdiagコマンドで出力されるメモリ情報の[Contains ModulesとBase Address]から、搭載されているSPARC M10-4Sの位置とメモリの物理アドレスの対応を確認します。次に、ldm list-devices memoryコマンドで未割り当てのメモリの物理アドレスとサイズを確認します。

次の例では、BB-ID 1のメモリの物理アドレスの先頭が0x700000000000～0x760000000000であり、この範囲のアドレスがldm list-devices memoryコマンドで表示されていることから、メモリが未割り当ての状態で追加されたことがわかります。

```
# prtdiag
....
======================== Physical Memory Configuration ========================
Segment Table:
----------------------------------------------------------------
Base            Segment  Interleave   Bank     Contains
Address         Size     Factor       Size     Modules
----------------------------------------------------------------
....
0x760000000000 32 GB    4            8 GB     /BB1/CMUL/CMP0/MEM00A
....
0x740000000000 32 GB    4            8 GB     /BB1/CMUL/CMP1/MEM10A
....
0x720000000000 32 GB    4            8 GB     /BB1/CMUU/CMP0/MEM00A
....
0x700000000000 32 GB    4            8 GB     /BB1/CMUU/CMP1/MEM10A
....
# ldm list-devices memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G
    0x720000000000      32G
    0x740000000000      32G
    0x760050000000      31488M
....
```

b. **新規に論理ドメインを作成し、CPUコアとメモリリソースを割り当てます。**

ドメインを構築する場合に、ldm set-coreコマンドで先にCPUコア数を設定し、その後、ldm set-memoryコマンドで割り当てるメモリサイズを設定すると、論理ドメインに割り当てられるメモリブロックが分散（フラグメント化）されにくくなり、SPARC M10-4Sを切り離すときにメモリ配置を考慮しやすくなります。

次の例では、ルートドメイン（root-dom1）と、ゲストドメイン（guest1）を追加し、CPUコア、メモリを割り当てています。

```
# ldm add-domain root-dom1
# ldm add-domain guest1
# ldm set-core 16 root-dom1
# ldm set-core 32 guest1
# ldm set-memory 32G root-dom1
# ldm set-memory 64G guest1
```

注－Oracle VM Server for SPARCのバージョンが3.2よりも古い場合、物理パーティションの動的再構成を実施するには、各論理ドメインに割り当てるメモリサイズを「CPUコア数×256 MBの倍数」に設定してください。

---

c. **ルートドメインにルートコンプレックスを割り当てます。**

追加されたSPARC M10-4Sのルートコンプレックスを、追加したルートドメイン（root-dom1）に割り当てます。ldm list-ioコマンドを実行し、ルートド

メイン（root-dom1）に追加するルートコンプレックスを確認します。
NAMEが「/BB1」で始まるデバイスがBB-ID 1上のデバイスで、そのデバイスのBUSに表示されている文字列がルートコンプレックスを示しています。

```
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8
PCIE9                                       BUS    PCIE9
PCIE10                                      BUS    PCIE10
PCIE11                                      BUS    PCIE11
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13
PCIE14                                      BUS    PCIE14
PCIE15                                      BUS    PCIE15
....
/BB1/CMUL/NET0                              PCIE   PCIE8
/BB1/CMUL/SASHBA                            PCIE   PCIE8
/BB1/PCI0                                   PCIE   PCIE9
/BB1/PCI3                                   PCIE   PCIE10
/BB1/PCI4                                   PCIE   PCIE10
/BB1/PCI7                                   PCIE   PCIE11
/BB1/PCI8                                   PCIE   PCIE11
/BB1/CMUL/NET2                              PCIE   PCIE12
/BB1/PCI1                                   PCIE   PCIE13
/BB1/PCI2                                   PCIE   PCIE13
/BB1/PCI5                                   PCIE   PCIE14
/BB1/PCI6                                   PCIE   PCIE14
/BB1/PCI9                                   PCIE   PCIE15
/BB1/PCI10                                  PCIE   PCIE15
```

ldm add-ioコマンドを実行し、ルートドメイン（root-dom1）に追加するルートコンプレックスを追加し、ldm list-ioコマンドを実行して確認します。

次の例では、root-dom1には、上記で表示されたオンボードデバイス（/BB1/CMUL/NET0, /BB1/CMUL/SASHBA, /BB1/CMUL/NET2）のルートコンプレックス（PCIE8とPCIE12）以外のBB1のルートコンプレックス（PCIE9、PCIE10、PCIE11、PCIE13、PCIE14、PCIE15）を追加しています。

```
# ldm add-io PCIE9 root-dom1
# ldm add-io PCIE10 root-dom1
....
# ldm list-io
NAME                                        TYPE   BUS      DOMAIN     STATUS
----                                        ----   ---      ------     ------
```

```
PCIE0                                       BUS    PCIE0    primary    IOV
PCIE1                                       BUS    PCIE1    root-dom0  IOV
PCIE2                                       BUS    PCIE2    root-dom0  IOV
PCIE3                                       BUS    PCIE3    root-dom0  IOV
PCIE4                                       BUS    PCIE4    primary    IOV
PCIE5                                       BUS    PCIE5    root-dom0  IOV
PCIE6                                       BUS    PCIE6    root-dom0  IOV
PCIE7                                       BUS    PCIE7    root-dom0  IOV
PCIE8                                       BUS    PCIE8    root-dom1  IOV
PCIE9                                       BUS    PCIE9
PCIE10                                      BUS    PCIE10   root-dom1  IOV
PCIE11                                      BUS    PCIE11   root-dom1  IOV
PCIE12                                      BUS    PCIE12
PCIE13                                      BUS    PCIE13   root-dom1  IOV
PCIE14                                      BUS    PCIE14   root-dom1  IOV
PCIE15                                      BUS    PCIE15   root-dom1  IOV
```

d. **ルートドメインにルートコンプレックスを割り当てます。**

ldm set-vconsoleコマンドを実行し、ルートドメイン（root-dom1）にコンソールデバイスを割り当てます。

```
# ldm set-vconsole port=5001 vcc0 root-dom1
```

ldm bind-domain、ldm start-domainコマンドを実行し、設定済みのルートコンプレックスなどの情報をルートドメイン（root-dom1）にバインドしなおしてから、ルートドメインを起動します。

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
```

telnetコマンドを実行し、ルートドメインのコンソールに接続します。

```
# telnet localhost 5001
....
```

e. **ルートドメインにOracle Solarisをインストールします。**

起動したルートドメインに、Oracle Solarisをインストールします。Oracle Solarisのインストールの詳細は、Oracle Solarisの関連ドキュメントを参照してください。

f. **ルートドメインの仮想サービスに割り当てる物理デバイスを確認します。**

Oracle Solarisをインストールしたあと、ルートドメインにログインして、formatコマンドを実行し、仮想ディスクサービスに割り当てるデバイス名を確認します。ルートドメインにログインする前に、制御ドメインでldm list-io -lを実行すると、PCIeエンドポイントに対応するデバイスパスが表示されるので、formatコマンドで表示されるディスクのデバイスパス名と比較することで、ディスクが接続されているパスが確認できます。

```
 # ldm list-io -l
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
....
/BB1/PCI0                                 PCIE   PCIE9    root-dom1OCC
[pci@8900/pci@4/pci@0/pci@0]
/BB1/PCI3                                 PCIE   PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@0]
/BB1/PCI4                                 PCIE   PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@8]....
# telnet localhost 5000
....
root-dom1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
       0. c3t500000E01BDA70B2d0 <FUJITSU-MBB2147RC-3703 cyl 14087 alt 2 hd 24
sec 848>
          / pci@8900/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w500000e01bda70b2,0
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8a00/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
Specify disk (enter its number): ^C
```

続けて、dladm show-phys -Lコマンドを実行すると、ルートドメイン上から見えるイーサネットのインターフェース名と物理位置が確認できます。

```
root-dom0# dladm show-phys -L
LINK              DEVICE       LOC
net0              igb0         /BB1/PCI0
net1              igb1         /BB1/PCI0
net2              igb2         /BB1/PCI0
net3              igb3         /BB1/PCI0
```

g. **ルートドメインに仮想I/Oサービスを割り当てます。**

制御ドメインに戻り、ルートドメイン（root-dom1）に仮想I/Oサービスを追加します。I/Oデバイスの詳細な設定については、Oracle社のドキュメント『Oracle VM Server for SPARC 管理ガイド』を参照してください。

次の例では、追加されたルートドメイン（root-dom1）に、仮想ディスクサービスvds1と仮想スイッチネットワークインターフェース（vsw10、vsw11）を追加しています。

```
# ldm add-vdiskserver vds1 root-dom1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol0@vds1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol1@vds1
# ldm add-vswitch net-dev=net1 vsw10 root-dom1
# ldm add-vswitch net-dev=net2 vsw11 root-dom1
```

h. **ゲストドメインに仮想I/Oデバイスを割り当てます。**

追加した仮想I/Oサービスを使った、仮想I/Oデバイスを各ゲストドメインに追加します。

次の例では、既存のゲストドメイン（guest0）にroot-dom1側の仮想I/Oサービスを使用した仮想I/Oデバイス（vdisk10、vnet10）を追加し、新たに追加するゲストドメイン（guest1）には、root-dom0とroot-dom1の仮想I/Oサービスを使った仮想I/Oデバイス（vdisk1、vdisk11、vnet1、vnet11）、仮想コンソール（vcons）を追加しています。

```
# ldm add-vdisk vdisk10 vol0@vds1 guest0
# ldm add-vnet vnet10 vsw10 guest0
# ldm add-vdisk vdisk1 vol1@vds0 guest1
# ldm add-vdisk vdisk11 vol11@vds1 guest1
# ldm add-vnet vnet1 vsw1 guest1
# ldm add-vnet vnet11 vsw11 guest1
# ldm set-vconsole port=5101 guest1
```

i. **追加したゲストドメインを起動します。**

ldm bind-domain、ldm start-domainコマンドを実行し、追加されたゲストドメイン（guest1）を起動します。

```
# ldm bind-domain guest1
# ldm start-domain guest1
LDom guest1 started
```

j. **追加したゲストドメインにOracle Solarisをインストールします。**

追加したゲストドメインにOracle Solarisをインストールします。Oracle Solarisのインストールの詳細は、Oracle Solarisの関連ドキュメントを参照してください。

k. **各ゲストドメインの仮想I/Oを冗長構成にします。**

ゲストドメイン（guest0）に割り当てた2つの仮想ネットワークインターフェース（vnet）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。

ゲストドメインguest0にログインします。次の例では、ldm list-domainコマンドでguest0のコンソールのポート番号を確認したあと、telnetコマンドでポート番号5100に接続しています。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    16    14G      0.0%  8h 7m
guest0           active     -n----  5100    32    32G      0.0%  20s
guest1           active     -n----  5101    32    32G      0.0%  19s
root-dom0        active     -n--v-  5000    24    24G      0.0%  43s
root-dom1        active     -n--v-  5001    24    24G      0.0%  20s

# telnet localhost 5100
```

```
....
guest0 console login: root
Password:
...
guest0#
```

dladmコマンドを実行し、仮想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、net1）として仮想ネットワークデバイスが参照できています。

```
guest0# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         0      unknown   vnet1
```

ipadm show-ifコマンドを実行して、net0、net1が表示されていないことを確認します。

```
 guest0# 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、net1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。

```
guest0# ipadm create-ip net0
guest0# ipadm create-ip net1
guest0# ipadm show-if
IFNAME     CLASS    STATE    ACTIVE OVER
lo0        loopback ok       yes    --
net0       ip       down     no     --
net1       ip       down     no     --
```

ipadm create-ipmpコマンドを実行し、IPMPインターフェースipmp0を作成したあと、ipadm add-ipmpコマンドを実行し、IPインターフェースnet0、net1をIPMPグループに追加します。

```
guest0# ipadm create-ipmp ipmp0
guest0# ipadm add-ipmp -i net0 -i net1 ipmp0
```

ipadm create-addrコマンドを実行し、IPMPインターフェースipmp0にIPアドレスを割り当てたあと、ipadm show-addrコマンドを実行し、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ています。

```
guest0# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/v4
guest0# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
```

```
ipmp0/v4          static   ok           xxx.xxx.xxx.xxx/24
lo0/v6            static   ok           ::1/128
```

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

```
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0 # ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net1        no      ipmp0       is-----   up        disabled  ok
net0        yes     ipmp0       --mbM--   up        disable
```

ほかのゲストドメイン（例の場合、guest1）も同様に設定を行います。

l. **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。**

追加されたSPARC M10-4Sのオンボードデバイスを制御ドメインに割り当てて冗長構成にします。制御ドメインを遅延再構成モードに移行します。

```
# 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.
```

制御ドメインに、追加したSPARC M10-4Sのルートコンプレックス（PCIE8、PCIE12）を追加したあと、Oracle Solarisを再起動します。

```
# ldm add-io PCIE8 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 add-io PCIE12 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.
--------------------------------------------------------------------------------
# shutdown -i6 -g0 -y
....
```

Oracle Solaris再起動後、システムボリュームを冗長構成にします。冗長構成の設定方法については、各冗長構成のソフトウェアの関連ドキュメントを参照してください。次の例では、formatコマンドでディスクドライブを確認したあと、ZFSでミラー構成にしています。

```
# format
Searching for disks...done


AVAILABLE DISK SELECTIONS:
```

```
       0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:


        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0  ONLINE       0     0     0


errors: No known data errors
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
```

zpool statusコマンドを実行し、ミラー構成が組み込まれたことを確認します。同期処理（resilver）が完了したかどうかはzpool statusコマンドで表示される状態で確認できます。次の例は、同期処理中を示しています。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:

        NAME                        STATE     READ WRITE CKSUM
        rpool                       DEGRADED     0     0     0
          mirror-0                  DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0 ONLINE       0     0     0
            c3t50000393A803B13Ed0s0 DEGRADED     0     0     0  (resilvering)

errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0

errors: No known data errors
```

m. **各論理ドメインのメモリ配置を確認して調整します。**

ldm list-devices -a memoryを実行し、メモリブロックの配置を確認します。

```
# ldm list-devices -a memory
MEMORY
    PA                  SIZE            BOUND
    0x700000000000      32G             root-dom1
    0x720000000000      32G             guest1
    0x740000000000      32G             guest1
    0x760000800000      1272M           _sys_
    0x760050000000      16G             primary
    0x760450000000      10G             primary
    0x7606d0000000      4864M
    0x780000000000      32G             guest0
    0x7a0000000000      32G             guest0
    0x7c0000000000      256M            primary
    0x7c0010000000      3584M           root-dom0
    0x7c00f0000000      24832M          primary
    0x7c0700000000      4G              root-dom0
    0x7e0000800000      1272M           _sys_
    0x7e0050000000      512M            _sys_
    0x7e0070000000      256M            _sys_
    0x7e0080000000      25088M          root-dom0
    0x7e06a0000000      3584M           primary
    0x7e0780000000      2G              primary
```

論理ドメインに割り当てられているメモリブロックや、未割り当てのメモリブロックが小さく分割されている場合、論理ドメインをアンバインドしてからバインドしなおすことで、分割されたメモリブロックを大きな連続領域にまとめなおすことができます。これにより、あとでSPARC M10-4Sを追加したり削除したりやすい状態にできます。手順の詳細は、「A.2.2　物理パーティションの構築手順例」の手順11-jを参照してください。

n. **論理ドメイン構成情報を保存します。**

ldm list-domainコマンドを実行し、構成した論理ドメインが起動されていることを確認します。

```
# ldm list-domain
NAME             STATE      FLAGS   CONS   VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART   64    56G      64%   2h 54m
guest0           active     -n----  5100   64    64G      42%   2h 54m
guest1           active     -n----  5101   64    64G      11%   2h 54m
root-dom0        active     -n--v-  5000   32    32G      20%   2h 54m
root-dom1        active     -n--v-  5001   32    32G      18%   2h 54m
```

ldm add-spconfigコマンドを実行し、論理ドメイン構成情報をXSCFに保存します。

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

o. **復旧させる論理ドメイン構成情報が複数ある場合は、現在の構成を factory-defaultに戻してから、手順15以降を繰り返します。**

現在の構成をfactory-default構成に戻す手順の例を示します。XSCFからshowdomainconfigコマンドを実行して、起動する論理ドメイン構成情報を確認します。

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) :ldm-set1
(Next) :ldm-set1
----------------------------------------------------------------
Index :1
config_name :factory-default
domains :1
date_created:-
----------------------------------------------------------------
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-0811:34:56'
----------------------------------------------------------------
....
```

setdomainconfigコマンドで、物理パーティション番号0（PPAR ID 0）にfactory-defaultのIndex1を指定します。

```
XSCF> setdomainconfig -p 0 -i 1
```

poweroff、poweronコマンドを実行して、物理パーティションを再起動します。起動が完了したら、手順15以降を繰り返してください。

```
XSCF> poweroff -p 0
...
XSCF> poweron -p 0
...
```

A.6 制御ドメインのみの構成でシステムボードを活性交換する場合（Oracle VM Server for SPARC 3.2以降）

Oracle VM Server for SPARC 3.2以降では、動的なPCIeバスの割り当てが可能になるため、制御ドメインおよびルートドメインにおいて遅延再構成をせずに、PCIeルートコンプレックスを追加または削除できます。
したがって、制御ドメインのみの構成（factory-default構成）の場合、Oracle Solarisを停止または再起動せずに、システムボードを追加、削除できるようになります。
ここでは、2つのSPARC M10-4Sでfactory-default構成の物理パーティションにおいて、片方のSPARC M10-4Sを切り離す手順の例を説明します。なお、factory-default構成の場合、すべてのCPUコアとメモリが制御ドメインに割り当てられます。ここでの説明は、SPARC M12でも同様です。

注ー制御ドメインまたはルートドメインがOracle Solaris 10の場合、動的にPCIeバスを追加および削除できないため、この構成は利用できません。

A.6.1 構成例

ここでは2つのSPARC M10-4S、すなわち2つのシステムボード（PSB<BB>）を使った（2BB構成）制御ドメインの構成例を示します。
物理パーティションの動的再構成を実施できるようにするために、制御ドメインのシステムボリュームディスクやネットワークインターフェースは、各SPARC M10-4Sの物理I/Oを接続して冗長構成にする必要があります。これは、SPARC M10-4Sを削除した場合に、残りのSPARC M10-4Sの物理I/Oを使って動作を継続できるようにするためです。

次の図は、上記の条件を満たした構成例の概略図です。なお、各論理ドメインのI/O構成は簡略化しています。

図 A-8　2BB構成ですべてのリソースを割り当てた構成例

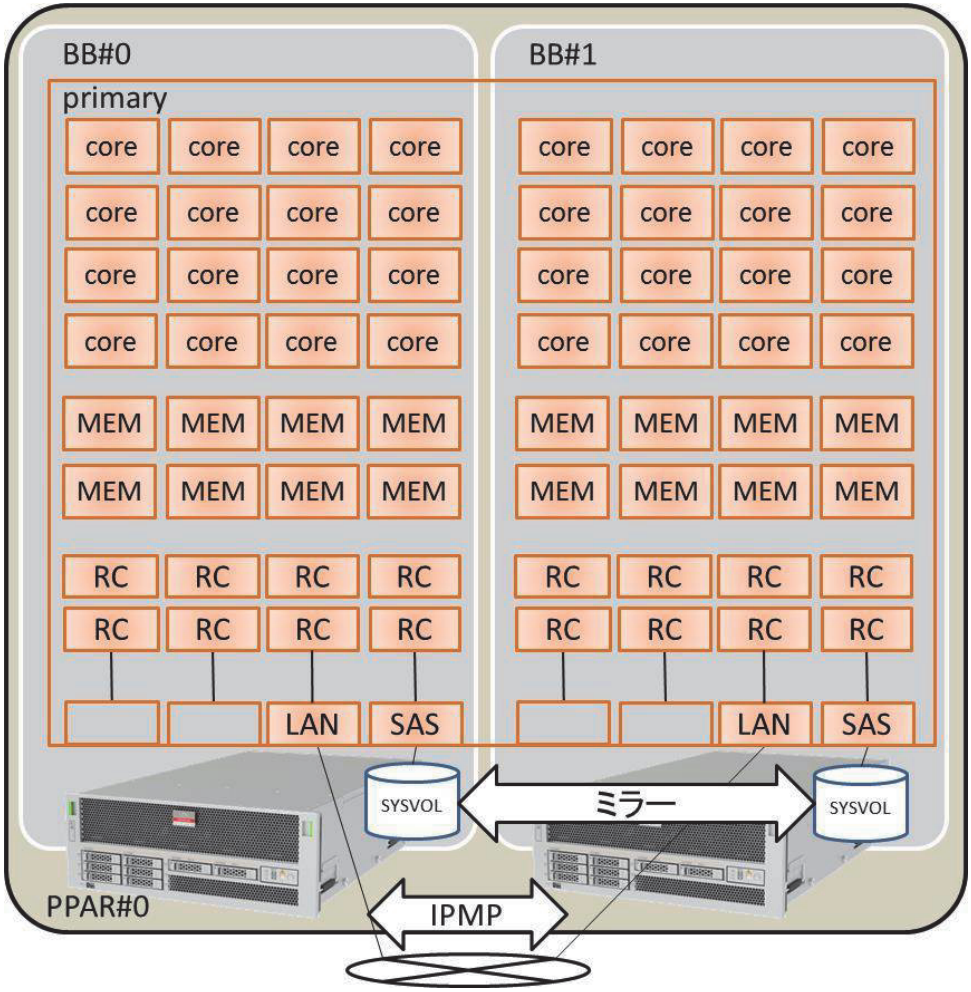


図 A-8では、2BB構成ですべてのCPUコア、メモリ、物理I/Oが制御ドメインに割り当てられている構成の例を示しています。したがって、物理パーティションの動的再構成で切り離されるSPARC M10-4Sに搭載されていたのCPUコアとメモリは、SPARC M10-4Sの切り離しの際に移動できずに削除されることになります。

A.6.2　物理パーティションの構築手順例

1. **マスタXSCFにログインします。**
   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の場合は、マスタXSCFへログインしなおしてください。

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

2. **メモリのミラーモードを設定します。**
   a. **showfruコマンドを実行し、SPARC M10-4Sのシステムボード（PSB<BB>）のメモリのミラーモード設定を確認します。**

      次の例では、物理システムボード（PSB 00-0）のデバイス設定情報を表示しています。

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
    cpu 00-0-0 no
    cpu 00-0-1 no
    cpu 00-0-2 no
    cpu 00-0-3 no
```

   b. **メモリのミラーモードを使用する場合、setupfruコマンドを使用し、メモリのミラーモードを設定します。**

      メモリのミラーモードを使用しない場合は、この手順は不要です。

      メモリのミラーモードの詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「14.1.1　メモリのミラー構成の概要」を参照してください。

      次の例では、システムボード（PSB 00-0）のすべてのCPUに対してメモリをミラーモードに設定しています。

```
XSCF> setupfru -m y sb 00-0
```

      showfruコマンドを実行し、メモリのミラーモード設定を確認します。

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
    cpu 00-0-0 yes
    cpu 00-0-1 yes
    cpu 00-0-2 yes
    cpu 00-0-3 yes
```

3. **物理パーティション構成情報を作成します。**
   a. **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
```

   b. **setpclコマンドを実行し、物理パーティション構成情報にシステムボードを登録します。**

      setpclコマンドを実行し、SPARC M10-4Sのシステムボードを組み込み先の物

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。

```
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。**

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                         Running
            00      00-0
            01      01-0
```

コンフィグレーションポリシー、IO無効化オプション（No-IO）、メモリ無効化オプション（No-Mem）の詳細を表示するには、-vオプションを指定します。

```
XSCF> showpcl -v -p 0
PPAR-ID   LSB    PSB    Status   No-Mem   No-IO    Cfg-policy
00                      Running
                                                  System
          00     00-0            False    False
          01     01-0            False    False
```

コンフィグレーションポリシー、I/O無効化オプション（No-IO）、メモリ無効化オプション（No-Mem）の設定を変更する場合は、setpclコマンドを使用します。

setpclコマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

4. **物理パーティションにシステムボード（PSB<BB>）を割り当てます。**

a. **showboards -aコマンドを実行し、PSBの状態を確認します。**

showboards -aコマンドを実行し、それぞれのPSB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、PSB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、PSB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PSBの状態を確認します。

この例では、それぞれのPSB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、それぞれのPSBの [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
```

5. **CPUコア アクティベーションキーを登録し、CPUコアリソースを割り当てます。**

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。

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ません。

```
XSCF> showcodactivation
Index Description Count
------- ----------- ------
```

注ー登録されているCPUコア アクティベーション数が、使用するCPUコア数を満たしていない場合は、CPUコア アクティベーションを購入し、CPUコア アクティベーションキーを追加してください。

---

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。**

CPUコア アクティベーションキーを追加する方法の詳細は、『SPARC M12/M10 システム運用・管理ガイド』の「5.3　CPUコアリソースを追加する」を参照してください。

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber:10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtxgvZPsrtYguqiNUieB+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            2
      1 PROC            2
      2 PROC            2
      3 PROC            2

--- 中略 ---

     62 PROC            2
     63 PROC            2
```

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コア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、128個のCPUコアリソース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 -c set 128
PROC Permits assigned for PPAR 0 : 0 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

注－XCP 2250以前のXSCFファームウェアでは、-c add、-c delete、および-c setオプションはサポートされていません。setcodコマンドのオプションを以下のように指定して、対話形式により追加および削除を実施してください。
XSCF> **setcod -s cpu**

---

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コア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128個のCPUコアリソースが、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: 128
```

6. **resetdateoffsetコマンドを実行し、XSCFが管理する時刻の差分をリセットします。**

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
```

7. **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                    :00000002
Diagnostic Level           :min
Message Level              :normal
Alive Check                :on
Watchdog Reaction          :reset
Break Signal               :on
Autoboot(Guest Domain)     :on
Elastic Mode               :off
IOreconfigure              :false
PPAR DR(Current)           :-
PPAR DR(Next)              :on
```

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。

setpparmodeコマンドの詳細は、『SPARC M12/M10 XSCFリファレンスマニュアル』を参照してください。

```
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
```

8. **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> poweron -p 0
```

9. **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
```

10. **Oracle SolarisとOracle VM Server for SPARCのインストールします。**
制御ドメインにOracle SolarisとOracle VM Server for SPARCをインストールします。

物理パーティションの動的再構成に必要なOracle Solarisの版数や条件などについては、表 1-3および表 1-4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/en/）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 11.2システムのインストール』

- Oracle VM Server for SPARC

  『Oracle VM Server for SPARC 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」

11. **制御ドメインの物理I/O構成を冗長構成にします。**

    a. **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。**

    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、ZFSでミラーリング構成にするコマンドの例を説明します。詳細は、Oracle社のホームページ（https://docs.oracle.com/en/）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 11.2でのZFSファイルシステムの管理』の「ミラー化ルートプールを構成する方法（SPARCまたはx86/VTOC）」

    その他の冗長構成ソフトウェアをご利用になる場合は、対象ソフトウェアのマニュアルを参照してください。

    zpool statusコマンドを実行して、ルートプールの状態を確認します。

    次の例では、デフォルトのルートプール（rpool）にc2t50000393E802CCE2d0s0が割り当てられていることを示しています。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
config:
       NAME                        STATE     READ WRITE CKSUM
       rpool                       ONLINE       0     0     0
           c2t50000393E802CCE2d0s0 ONLINE       0     0     0
errors: No known data errors
```

    formatコマンドを実行して、追加可能なディスクを確認します。
    次の例では、もう1つのディスクとしてc3t50000393A803B13Ed0が存在することがわかります。

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
       1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
          /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
```

```
          /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
```

zpool attachコマンドを実行して、rpoolに2本目のディスクを追加してミラー構成にします。

次の例では、zpool attachコマンドでc3t50000393A803B13Ed0s0を追加して、zpool statusコマンドで同期処理（resilver）の状況を確認しています。status、actionを見ると、同期処理中であることがわかるので、同期処理が完了するまで、定期的にzpool statusコマンドを実行して状況を確認します。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered. The pool will
continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Wed Jan 29 21:35:39 2014
  3.93G scanned out of 70.6G at 71.9M/s, 0h15m to go 3.90G resilvered, 5.56%
done
config:
       NAME                        STATE    READ WRITE CKSUM
       rpool                       DEGRADED    0     0     0
         mirror-0                  DEGRADED    0     0     0
           c2t50000393E802CCE2d0s0 ONLINE      0     0     0
           c3t50000393A803B13Ed0s0 DEGRADED    0     0     0
 (resilvering)
```

同期処理が完了すると以下のように、 [state] が「ONLINE」となります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h10m with 0 errors on Wed Jan 29 21:45:42 2014
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          mirror-0                  ONLINE       0     0     0
            c2t50000393E802CCE2d0s0  ONLINE       0     0     0
            c3t50000393A803B13Ed0s0  ONLINE       0     0     0
errors: No known data errors
```

b. **制御ドメインのネットワークを冗長構成にします。**

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、オンボードのネットワークインターフェースがnet0〜net7として参照できていることを示しています。

```
# dladm show-phys -L
LINK        DEVICE      LOC
net0        igb0        /BB0/CMUL
net1        igb1        /BB0/CMUL
net2        igb6        /BB0/CMUU
net3        igb7        /BB0/CMUU
net4        igb14       /BB1/CMUL
net5        igb15       /BB1/CMUL
net6        igb12       /BB1/CMUU
net7        igb13       /BB1/CMUU
```

ここでは、BB0側のネットワークインターフェースnet0と、BB1側のネットワークインターフェースnet4を、IPMPで冗長構成にする例を説明します。
まず、ipadm show-ifコマンドを実行して、net0（BB0側）と net4（BB1側）が表示されていないことを確認します。

```
# 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コマンドを実行して、作成できたことを確認します。

```
# ipadm create-ip net0
# ipadm create-ip net4
# 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。

```
# ipadm create-ipmp ipmp0
# 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コマンドを実行して、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。

```
# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/v4
# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
ipmp0/v4          static   ok           xx.xx.xx.xx/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コマンドを実行して、IPMP構成を確認します。

```
# ipadm set-ifprop -p standby=on -m ip net4
# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net4        no      ipmp0       is-----   up        disabled  ok
net0        yes     ipmp0       --mbM--   up        disabled  ok
```

A.6.3 活性交換手順例

ここでは、動的なPCIeバスの割り当てが利用可能な環境（Oracle VM Server for SPARC 3.2以降）で、「図 A-8　2BB構成ですべてのリソースを割り当てた構成例」に記載された2BB構成システムに対して、PPAR DRを使用して、BB#01を活性交換する手順例を記載します。

ここでの説明は、SPARC M12でも同様です。

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

1. **マスタXSCFにログインします。**
   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の場合は、マスタXSCFへログインしなおしてください。

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

2. **showhardconfコマンドを実行し、交換するSPARC M10-4SのXSCFの [Status] が「Normal」であることを確認します。**

```
XSCF> showhardconf
SPARC M10-4S;
    + Serial: 2081230011; Operator_Panel_Switch:Locked;
    + System_Power:On; System_Phase:Cabinet Power On;
    Partition#0 PPAR_Status:Running;
    BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
        + FRU-Part-Number: CA07361-D202 A1                        ;
        + Power_Supply_System: ;
        + Memory_Size:256 GB;
        CMUL Status:Normal; Ver:0101h; Serial:PP1236052K  ;
           + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
            + Memory_Size:128 GB; Type: A ;
            CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
                + Freq:3.000 GHz; Type:0x10;
                + Core:16; Strand:2;
```

```
    :
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
    + FRU-Part-Number: CA20393-B50X A2                        ;
    + Power_Supply_System: ;
    + Memory_Size:256 GB;
    CMUL Status:Normal; Ver:0101h; Serial:PP123406CB  ;
        + FRU-Part-Number:CA07361-D941 C4   /7060911              ;
        + Memory_Size:128 GB; Type: A ;
    :
```

注ー活性交換するSPARC M10-4SのXSCFが故障している場合、PPAR DRによる活性交換はできません。
活性交換するSPARC M10-4Sが属する物理パーティションを停止し、交換するSPARC M10-4Sの入力電源を切断した保守作業が必要です。

---

3. **showbbstatusコマンドを実行し、交換するSPARC M10-4SのXSCFがマスタでないことを確認します。**

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

交換するSPARC M10-4SがマスタXSCFの場合、switchscfコマンドを実行し、XSCFを切り替えます。

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

注ーXSCFの切り替え、およびXSCFの再起動が完了したことを確認したあとに、SPARC M10-4Sの切り離し操作を行ってください。

---

4. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

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

5. **制御ドメインのシステムボリュームや物理I/Oデバイスの冗長構成を切り離します。**
制御ドメインで使用している、交換対象（BB#01）のSPARC M10-4Sの物理I/Oデバイスを切り離します。冗長構成を解除する手順の詳細は、各冗長化ソフトウェアのドキュメントを参照してください。

a. **制御ドメインのシステムボリュームの冗長化を解除します。**

次の例では制御ドメインのシステムボリュームのZFSミラー機能を解除する手順を示します。

制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認しま

す。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

zpool detachコマンドを実行し、ディスクをミラー構成から切り離します。

```
# zpool detach rpool c3t50000393A803B13Ed0
```

zpool statusコマンドで、ミラー構成が解除されたことを確認します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を解除したり、デバイスの利用を停止したりしてください。冗長構成の解除方法やデバイスの利用停止方法については、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

b. **制御ドメインのネットワークの冗長構成を解除します。**

ipmpstat -i コマンドを実行し、IPMPを構成しているネットワークインターフェースの構成情報を確認します。

```
# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net4        no      ipmp0       is-----   up        disabled  ok
```

if_mpadm -dコマンドを実行し、IPMPのグループからnet4を切り離して、ipmpstat -iコマンドを実行し、切り離されたことを確認します。次の例の場合、STATEがofflineになったことを確認します。

```
# if_mpadm -d net4
# ipmpstat -i
INTERFACE   ACTIVE  GROUP       FLAGS     LINK      PROBE     STATE
net0        yes     ipmp0       -smbM--   up        disabled  ok
net4        no      ipmp0       -s---d-   up        disabled  offline
```

ipadm delete-ipコマンドを実行して、net4を削除します。

```
# ipadm delete-ip net4
```

6. **SPARC M10-4Sのシステムボード（PSB<BB>）を物理パーティションから切り離します。**

   a. **deleteboard -c disconnectコマンドを実行し、PSBを物理パーティションから切り離します。**

   事前にCPUコア、メモリ、PCIeルートコンプレックスのいずれかのリソースを手動で切り離していない場合は、必ず「-m unbind=resource」オプションを指定してください。このオプションを指定すると、自動的にリソースを削除してから、PSBを切り離します。

```
XSCF> deleteboard -c disconnect -m unbind=resource 01-0
PSB#01-0 will be unconfigured from PPAR immediately.
Continue?[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

   b. **showresultコマンドを実行し、直前に実行したdeleteboardコマンドの終了ステータスを確認します。**

   終了値が0の場合、deleteboardコマンドは正常終了です。

   終了値が0以外の場合、およびdeleteboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.2 deleteboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

   c. **showboardsコマンドを実行し、PSBの状態を確認します。**

   交換するSPARC M10-4SのPSBが「Assigned」状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

7. **replacefruコマンドを実行し、SPARC M10-4Sを交換します。**

```
XSCF> replacefru
```

注ーreplacefruコマンドを使用したSPARC M10-4Sの交換の詳細は、『SPARCM10-4/M10-4Sサービスマニュアル』の「5.8　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムから切り離す」および「6.2　replacefruコマンドでSPARC M10-4/M10-4SのFRUをシステムに組み込む」を参照してください。

---

8. **PSBを物理パーティションに組み込みます。**

   a. **showboardsコマンドを実行し、PSBの状態を確認します。**

   交換したSPARC M10-4SのPSBがAssigned状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されてい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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

   b. **addboard -c configureコマンドを実行し、PSBを物理パーティションに組み込みます。**

   論理ドメイン構成を元の構成に戻すため、-m bind=resourceオプションを指定して、addboard -c configureコマンドを実行します。

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

注ーaddboardコマンド実行中、エラーメッセージが出力された場合は、「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

---

c. **showresultコマンドを実行し、直前に実行したaddboardコマンドの終了ステータスを確認します。**

終了値が0の場合、addboardコマンドは正常終了です。

終了値が0以外の場合、およびaddboardコマンド実行中にエラーメッセージが出力された場合は、異常終了です。エラーメッセージより「C.1.1　addboard」を参照して、エラー内容の特定と対処を実施してください。

```
XSCF> showresult
0
```

d. **showboardsコマンドを実行し、PSBの状態を確認します。**

交換したSPARC M10-4SのPSBが正常に組み込まれたことで、 [Conn] 欄と [Conf] 欄の表示がともに「y」と表示され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    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

9. **制御ドメインのシステムボリュームや物理I/Oデバイスを冗長構成に戻します。**

a. **制御ドメインのシステムボリュームの冗長化を構成します。**

制御ドメインでzpool statusコマンドを実行し、ミラー構成の状態を確認します。

次の例では制御ドメインのシステムボリュームのZFSミラー機能を構成する手順を示します。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
        NAME                        STATE     READ WRITE CKSUM
        rpool                       ONLINE       0     0     0
          c2t50000393E802CCE2d0s0   ONLINE       0     0     0
errors: No known data errors
```

zpool attachコマンドを実行し、ディスクをミラー構成に組み込みます。

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

zpool statusコマンドで、ミラー構成が組み込まれたことを確認します。

なお、同期処理（resilver）が完了したかどうか、zpool statusコマンドで状態を確認してください。

次の例は、同期処理中の表示例です。

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         DEGRADED     0     0     0
          mirror-0                    DEGRADED     0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   DEGRADED     0     0     0  (resilvering)
errors: No known data errors
```

同期処理が完了すると、次のような表示になります。

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c2t50000393E802CCE2d0s0   ONLINE       0     0     0
            c3t50000393A803B13Ed0s0   ONLINE       0     0     0
errors: No known data errors
```

ほかにもBB#01上のデバイスを使用している場合、冗長構成を構築したり、デバイスの利用を再開したりしてください。冗長構成の構築方法やデバイスの利用再開方法は、各冗長構成のソフトウェアおよびOracle Solarisのドキュメントを参照してください。

付録 B

動的再構成使用時の補足情報

ここでは、物理パーティションの動的再構成機能を使用するための補足情報や、問題が発生した場合の対処方法について説明します。

- XSCF再起動またはフェイルオーバー時の留意点
- CPU動作モードに関する補足情報
- その他の留意事項

B.1 XSCF再起動またはフェイルオーバー時の留意点

システムボード（PSB<BB>）の追加または削除により物理パーティションの動的再構成を行っている際に、XSCFの再起動またはフェイルオーバーが発生した場合は、物理パーティションの動的再構成に関する操作が完了していない可能性があります。この場合、以下を実施して、操作が完了していることを確認してください。完了していない場合は、再度、動的再構成を使用して、PSB（BB）の追加または削除を行ってください。

追加の完了確認

1. **XSCF上でshowboardsコマンドを実行し、システムボード（PSB<BB>）の状態を確認します。**
   次の例では、増設したPSB（BB）がAssigned状態であり、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「y」と表示されていることを確認します。

```
XSCF> showboards -p PPAR_ID
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    y    y    y    Passed  Normal
```

2. **consoleコマンドを実行し、制御ドメインのコンソールに接続し、ログインします。**

```
XSCF> console -p PPAR_ID
```

3. **ldm list-devicesコマンドを実行し、追加されたハードウェアリソースの状態を確認します。**
   ldm list-devicesコマンドを実行すると未割り当てのハードウェアリソースの一覧が表示されます。

   ハードウェアリソースが追加されたことを確認します。

```
# ldm list-devices
CORE
    ID       %FREE    CPUSET
    128      100      (256, 257)
    132      100      (264, 265)
    136      100      (272, 273)
    140      100      (280, 281)
....
MEMORY
    PA                     SIZE        BOUND
    0x700000000000         32G
    0x720000000000         32G
    0x740000000000         32G
    0x760050000000         31488M
    ....
```

削除の完了確認

showboardsコマンドを実行し、システムボード（PSB<BB>）の状態を確認します。削除したPSB（BB）がAssigned状態で、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されているか、または、削除したシステムボードが、「SP」(システムボードプール)、「Available」の状態で、[Pwr] 欄、[Conn] 欄と [Conf] 欄の表示がすべて「n」と表示されていることを確認します。

以下は、それぞれの例を示しています。

```
XSCF> showboards -p PPAR_ID
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
01-0 00(01)       Assigned    n    n    n    Passed  Normal
```

```
XSCF> showboards -p PPAR_ID
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test    Fault
---- ------------ ----------- ---- ---- ---- ------- --------
00-0 00(00)       Assigned    y    y    y    Passed  Normal
```

```
01-0 SP          Available   n    n    n    Passed  Normal
```

B.2 CPU動作モードに関する補足情報

setpparmodeコマンドで設定されたCPU動作モードと、物理パーティションを起動するときに使用される論理ドメイン構成情報の整合性がとれない場合は、XSCFによって、論理ドメイン構成情報がfactory-defaultに切り替えられます。
指定されている物理パーティションの構成やCPU動作モードの設定値、論理ドメイン構成情報の組み合わせで、factory-defaultに切り替えられるケースは以下のとおりです。

注－SPARC M12システムでは、CPU動作モードは未サポートです。

表 B-1 factory-defaultに切り替わるCPU構成、CPU動作モード、論理ドメイン構成情報の関係

物理パーティションのCPU構成	CPU動作モードの設定値	論理ドメイン構成情報
すべてSPARC64 X+プロセッサ	compatible	SPARC64 X+機能で動作している状態で作成された論理ドメイン構成情報を、setdomainconfigコマンドまたはldm set-spconfigコマンドで指定
すべてSPARC64 Xプロセッサ	auto、compatibleの両方	
SPARC64 X+プロセッサとSPARC64 Xプロセッサが混在	auto、compatibleの両方	

上記の表から、論理ドメイン構成情報がfactory-defaultに切り替わる要因として想定される操作は、以下のとおりです。

- SPARC64 X+プロセッサだけで構成された物理パーティションのCPU動作モードを、autoモードからcompatibleモードに変更したあと物理パーティションを起動した場合
- SPARC64 X+プロセッサだけで構成された物理パーティションに、SPARC64 Xプロセッサで構成されたシステムボードをaddboard -c assignコマンドで追加したあと物理パーティションを起動した場合

B.3 その他の留意事項

- 物理パーティション（PPAR）を構成するすべてのシステムボード（PSB）に、CPUメモリユニット（上段）（CMUU）が搭載されている場合、動的再構成機能（PPAR DR）を使用して、PPAR構成からいったんPSBを削除し、CMUUの減設を行って、PSBを戻した場合、CMULの一部のI/Oが見えなくなります。必要に応じて、論理ドメインの構成情報を設定しなおしてください。
- deleteboardコマンドを実行して異常終了した場合は、addboardコマンドを実行しないでください。addboardコマンドで「internal error」のエラーが検出され異常

終了します。また、その後の操作に影響する可能性があります。

- 動的再構成機能を使用し、deleteboardコマンドを実行してPSB（BB）を削除する途中で異常終了した場合は、以下の手順を実施してください。
  1. エラーメッセージに従い、異常となった原因を特定します。
  2. 異常となった原因を取り除きます。
  3. 再度、動的再構成機能使用し、deleteboardコマンドを実行してPPAR構成からPSBを削除します。

  再び削除に失敗した場合は、物理パーティション（PPAR）を停止したあと、「6.3.1　システムボードの割り当てを解除する操作例」にある操作を実施するか、「6.3.3　システムボードの割り当て解除を予約する操作例」にある操作を実施してください。

- 動的再構成機能を使用し、addboardコマンドを実行してPSB（BB）を追加する途中で異常終了した場合は、以下の手順、または操作を実施してください。
  1. エラーメッセージに従い、異常となった原因を特定します。
  2. 異常となった原因を取り除きます。
  3. 再度、動的再構成機能を使用し、addboardコマンドを実行してPPAR構成にPSBを追加します。

  または、動的再構成機能を使用し、deleteboardコマンドを実行してPPAR構成からPSBを削除したあと、再度addboardコマンドを実行してPPAR構成にPSBを追加します。

  どちらを実施しても再び追加に失敗した場合は、物理パーティション（PPAR）を停止したあと、「6.2.1　システムボードを割り当てる操作例」にある操作を実施するか、「6.2.3　システムボードの組み込みを予約する操作例」にある操作を実施してください。

- 動的再構成機能を使用し、addboardコマンドによるPSB（BB）の追加、またはdeleteboardコマンドによるPSBの削除を行っているときは、Oracle VM Server for SPARCを停止したり、再起動したりしないでください。実行中のaddboardコマンドやdeleteboardコマンドが異常終了します。
  このとき、Oracle VM Server for SPARCを再起動したあと、再度addboardコマンドやdeleteboardコマンドを実行しても、異常終了することがあります。

  この場合、追加のときは、物理パーティション（PPAR）を停止したあと「6.2.1　システムボードを割り当てる操作例」にある操作を実施するか、「6.2.3　システムボードの組み込みを予約する操作例」にある操作を実施してください。

  削除のときは、物理パーティションを停止したあと、「6.3.1　システムボードの割り当てを解除する操作例」にある操作を実施するか、「6.3.3　システムボードの割り当て解除を予約する操作例」にある操作を実施してください。

- 動的再構成機能を使用し、addboardコマンドによるPSB（BB）の追加、またはdeleteboardコマンドによるPSBの削除を行っているときは、rebootxscf -aコマンドやXSCF Webで、すべてのXSCFの再起動を実行しないでください。この操作を行うとDRの処理が異常終了するだけでなく、動作中の物理パーティションでOracle Solarisがハングアップしたりハードウェア故障が発生したりして物理パーティションが異常停止し、部品交換が必要になる場合があります。

- 動的再構成機能を使用し、内蔵ハードディスクに対する割り当てを削除する場合、下記のメッセージが出力され、削除処理が失敗することがあります。

```
# ldm remove-io /BB0/CMUL/SASHBA primary
The removal of /BB0/CMUL/SASHBA from the domain primary failed.
Error message from svc:/ldoms/agents in domain primary:
ERROR: devices or resources are busy.
```

上記のエラーは、fmdサービスが削除対象のデバイスを使用している場合に発生することがあります。対象デバイスが割り当てられているドメイン上のfmdサービスを停止指示し、サービスが停止状態になったことを確認してから、再度、削除処理を行ってください。また、対象デバイスの割り当てを削除したあとは、fmdサービスを再開してください。

fmdサービスの停止／再開する場合は、以下のコマンドを使用します。

- fmdサービスの停止

```
# svcadm disable svc:/system/fmd:default
```

■ fmdサービスの停止状態の確認

fmdサービスの停止が完了するまでに数分かかることがあります。

停止指示後、STATEがdisabledとなり停止状態になったこと確認してください。

```
# svcs fmd
STATE STIME FMRI
disabled 00:00:00 svc:/system/fmd:default
```

- fmdサービスの再開

```
# svcadm enable svc:/system/fmd:default
```

- SPARC M12-2SのオンボードLANまたは、Dual 10Gigabit Ethernetカード（SP1X7HF1F, SE1X7HE3G）が割り当てられている論理ドメインにおいて、動的再構成機能によってOracle Solarisがサスペンド、レジュームすると、「NIC-8000-1C」のエラーを検出することがあります。このエラーは動作に影響ないため無視してください。
- deleteboardによってPSBを削除する場合、削除されるPSBのLANやI/Oデバイスを事前に冗長構成から切り離すか、使用されていない状態にしてください。
- addboardコマンドによってPSBを追加した場合、追加されたPSBのLANまたはI/Oデバイスを冗長構成に組み込むか、再度使用できるよう設定を行ってください。

付録 C

メッセージの説明と対処

ここでは、ドメイン構築の作業中に出力されるメッセージの説明と対処方法について説明します。

■ コマンドメッセージ

C.1 コマンドメッセージ

ここでは、物理パーティションの構成作業で出力されるメッセージの説明と対処方法をコマンドごとに説明します。

C.1.1 addboard

A hardware error occurred. Please check the error log for details.
［説明］ハードウェア異常によりコマンドが終了しました。

［対処方法］エラーログで被疑部品を調べ、部品を交換してください。

An internal error has occurred. Please contact your system administrator.
［説明］内部エラーが発生したため、システム管理者に連絡をする必要があります。

［対処方法］XSCF内部で使用しているライブラリで、異常、XSCF内部リソース不足などが発生している可能性があります。しばらく待ったあと、再度処理を実行してください。再実行後もエラーが解消されない場合は、弊社保守員に連絡して原因を調査してください。

Another DR operation is in progress. Try again later.
［説明］指定した物理パーティションは別のセッションで動的再構成を実行中であるため操作できません。

［対処方法］別のセッションで動的再構成が実行されています。物理パーティションの状態を確認し、しばらく待ってから再実行してください。

Cannot communicate with Hypervisor or Logical Domains Manager. Please check the domain's state.

［説明］ハイパーバイザまたはLogical Domains Managerとの通信で異常が発生しました。

［対処方法］監視メッセージやコンソールメッセージから論理ドメインのOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

Cannot communicate with XXX. Please check XXX's state.
［説明］スタンバイ状態またはスレーブの XSCF が異常です。

［対処方法］スタンバイ状態またはスレーブのXSCF の異常を解消してください。このエラーは、1台構成のSPARC M12/M10では発生しません。

Configured PSB to Logical Domains (LDoms) Manager.
［説明］Logical Domains Managerへの組み込みが完了しました。

Connected PSB to PPAR.
［説明］物理パーティション構成への組み込みが完了しました。

Control domain is stopped for CoD resource violation.
［説明］CPUコア アクティベーション違反により制御ドメインが停止したため、指定したシステムボードを物理パーティションに組み込むことができませんでした。

［対処方法］CPUコア アクティベーション違反を解消してから再度実行してください。

DR operation canceled by operator.
［説明］ユーザーが動的再構成の中止を指示したため、実行がキャンセルされました。

Initial diagnosis has completed.
［説明］ハードウェア診断が完了しました。

Initial diagnosis started.
［説明］ハードウェア診断を開始しました。

Invalid parameter.
［説明］パラメーター異常です。

［対処方法］引数・オペランドを間違えていないことを確認してください。

Operation has completed.
［説明］組み込み処理が完了しました。

Operation is continued as "no".
［説明］[y|n]の問い合わせに対して正しいオペレーションが行われなかったため、nを選択したときの動作が行われました。

［対処方法］システムボード（PSB）の情報を確認し、組み込みが必要なPSBがある場合は、再度組み込み操作を行ってください。

Operation not supported on this system.
［説明］実行したコマンドが本システムで未サポートです。

［対処方法］モデル・XCP版数を確認して、本コマンドが実行可能であることを調べてください。

Permission denied.
［説明］アクセス権がありません。または、スタンバイ状態のXSCFで、実行可

能なコマンド以外のコマンドが実行され、実行権限異常となっています。

［対処方法］自分のユーザー権限とコマンドに必要な実行権限を比較してください。

PPAR DR function setting is disabled.
［説明］PPAR DR機能が無効に設定されているため、動的再構成を行えません。

［対処方法］PPAR DR機能の設定を確認し、再度実行してください。

PPAR is currently unavailable for DR, because XBU status has failed.
［説明］指定した物理パーティションやシステムボード（PSB）のクロスバーユニット（XBU）、または構成されるクロスバーボックスのクロスバーユニットに異常があるため、動的再構成が実行できません。

［対処方法］監視メッセージやエラーログから該当するクロスバーユニットに故障が発生していないか確認してください。故障していた場合は故障部品を確認して交換を実施してください。

PPAR (PPAR-ID XX) is currently unavailable for DR, because another operation is in progress. Try again later.
［説明］指定した物理パーティションはほかのオペレーションによって実行中のため、動的再構成が実行できません。

［対処方法］別セッションで電源の切断、または接続されているPCIボックスの構成情報を変更するオペレーションが実行されている可能性があります。物理パーティションの状態を確認しながら、しばらく待機したあと、再度実行してください。

PPAR (PPAR-ID XX) is not currently running.
［説明］-c configureオプションを指定したときに、組み込み先の物理パーティションが稼働していなかったため、組み込みを行えませんでした。

［対処方法］物理パーティションの動作状態を確認するか、-c assignオプション指定して実行してください。

PSB#XX-X could not be configured into PPAR-ID XX due to operating system or Logical Domains Manager error.
［説明］論理ドメインのOracle SolarisまたはLogical Domains Managerから動的再構成の処理でエラーが応答されたため、組み込みができませんでした。

［対処方法］監視メッセージやコンソールメッセージからOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

PSB#XX-X encountered a hardware error. See error log for details.
［説明］動的再構成で行っているハードウェア診断でエラーが検出されました。ハードウェアが故障している可能性があります。

［対処方法］監視メッセージやエラーログでハードウェアに故障が発生していないことを確認してください。故障していた場合は故障部品を確認して交換を実施してください。

PSB#XX-X has not been registered in PCL.
［説明］指定したシステムボードが物理パーティション（PPAR）構成情報に登録されていないため、実行できませんでした。

［対処方法］物理パーティション（PPAR）構成情報を登録してから実行してください。

PSB#XX-X is already assigned to another PPAR.
［説明］指定したPSBはほかの物理パーティションに割り当てられているため、指定した物理パーティションに組み込むことができませんでした。

［対処方法］PSBが別の物理パーティションに割り当てられています。PSBの状態を確認してください。

PSB#XX-X is currently unavailable for DR. Try again later.
［説明］指定したPSBはほかのオペレーションによって実行中であるため動的再構成できません。

［対処方法］動的再構成、物理パーティションの起動／停止、または保守作業が別セッションで実行されている可能性があります。PSBの状態を確認し、しばらく待ってから再実行してください。

PSB#XX-X is not installed.
［説明］指定したPSBが搭載されていないため実行できませんでした。

［対処方法］PSBの指定を誤っている可能性があります。システムボードの状態を確認してください。

PSB#XX-X status is FAILED.
［説明］指定したPSBは故障状態です。

［対処方法］監視メッセージやエラーログでハードウェアに故障が発生していないことを確認してください。故障していた場合は故障部品を確認して交換を実施してください。

PSB#XX-X will be assigned to PPAR-ID XX. Continue?[y|n] :
［説明］PSB番号XX-Xを物理パーティションYYに割り当てます。継続しますか。継続する場合は「y」を、中止する場合は「n」を入力してください。

PSB#XX-X will be configured into PPAR-ID XX. Continue?[y|n] :
［説明］PSB番号XX-Xを物理パーティションYYに組み込みます。継続しますか。継続する場合は「y」を、中止する場合は「n」を入力してください。

Start configuring PSB to Logical Domains (LDoms) Manager.
［説明］Logical Domains Managerへの組み込みを開始しました。

Start connecting PSB to PPAR.
［説明］物理パーティション構成への組み込みを開始しました。

The current configuration does not support this operation.
［説明］構成異常によりコマンドが終了しました。

［対処方法］コマンド実行に必要なハードウェアの実装や、機能の設定を見直してください。

The other XSCF is currently unavailable for DR. Try again later.
［説明］スタンバイ状態またはスレーブのXSCFがシャットダウン中のため、動的再構成が実行できませんでした。

［対処方法］スタンバイ状態またはスレーブのXSCFがシャットダウン中である可能性があります。スタンバイ状態またはスレーブのXSCFの状態を確認しながら、再度実行してください。

The system is currently unavailable for DR, because another operation is in progress. Try again later.
［説明］SPARC M12/M10システム全体へのオペレーションが実行中のため、動

的再構成が実行できませんでした。

［対処方法］別セッションでファームウェアのアップデートが実行されている可能性があります。ファームウェアのアップデートが完了したあと、再度実行してください。

Timeout detected during communicate with Logical Domains Manager.
［説明］Logical Domains Managerとの通信でタイムアウトが発生しました。

［対処方法］監視メッセージやコンソールメッセージから論理ドメインのOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

Timeout detected during configuration into PPAR-ID XX.
［説明］組み込み処理が完了せず、タイムアウトしました。

［対処方法］監視メッセージやコンソールメッセージからOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

Timeout detected during self-test of PSB#XX-X.
［説明］動的再構成で行っているハードウェア診断が完了せずタイムアウトしました。ハードウェアが故障している可能性があります。

［対処方法］監視メッセージやエラーログでハードウェアに故障が発生していないことを確認してください。故障していた場合は故障部品を確認して交換を実施してください。

C.1.2 deleteboard

A hardware error occurred. Please check the error log for details.
［説明］ハードウェア異常によりコマンドが終了しました。

［対処方法］エラーログで被疑部品を調べ、部品を交換してください。

An internal error has occurred. Please contact your system administrator.
［説明］内部エラーが発生したため、システム管理者に連絡をする必要があります。

［対処方法］XSCF内部で使用しているライブラリで、異常、XSCF内部リソース不足などが発生している可能性があります。しばらく待ったあと、再度処理を実行してください。再実行後もエラーが解消されない場合は、弊社保守員に連絡して原因を調査してください。

Cannot communicate with Hypervisor or Logical Domains Manager. Please check the domain's state.
［説明］ハイパーバイザまたはLogical Domains Managerとの通信で異常が発生しました。

［対処方法］監視メッセージやコンソールメッセージから論理ドメインのOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

Cannot communicate with XXX. Please check XXX's state.
［説明］スタンバイ状態またはスレーブの XSCF が異常です。

［対処方法］スタンバイ状態またはスレーブのXSCF の異常を解消してください。

このエラーは、1台構成のSPARC M12/M10では発生しません。

Control domain is stopped for CoD resource violation.
［説明］CPUコア アクティベーション違反により制御ドメインが停止したため、指定したPSB（BB）を切り離すことができませんでした。

［対処方法］CPUコア アクティベーション違反を解消してから再度実行してください。

DR operation canceled by operator.
［説明］ユーザーが動的再構成の中止を指示したため、実行がキャンセルされました。

Invalid parameter.
［説明］パラメーター異常です。

［対処方法］引数・オペランドを間違えていないことを確認してください。

Operation has completed.
［説明］切り離し処理が完了しました。

Operation is continued as "no".
［説明］[y|n]の問い合わせに対して正しいオペレーションが行われなかったため、nを選択したときの動作が行われました。

［対処方法］システムボード（PSB）の情報を確認し、切り離しが必要なPSBがある場合は、再度切り離し操作を行ってください。

Operation not supported on this system.
［説明］実行したコマンドが本システムで未サポートです。

［対処方法］モデル・XCP版数を確認して、本コマンドが実行可能であることを調べてください。

Permission denied.
［説明］アクセス権がありません。または、スタンバイ状態のXSCFで、実行可能なコマンド以外のコマンドが実行され、実行権限異常となっています。

［対処方法］自分のユーザー権限とコマンドに必要な実行権限を比較してください。

PPAR DR function setting is disabled.
［説明］PPAR DR機能が無効に設定されているため、動的再構成を行えません。

［対処方法］PPAR DR機能の設定を確認し、再度実行してください。

PPAR is currently unavailable for DR, because XBU status has failed.
［説明］指定した物理パーティションやシステムボード（PSB）のクロスバーユニット（XBU）、または構成されるクロスバーボックスのクロスバーユニットに異常があるため、動的再構成が実行できません。

［対処方法］監視メッセージやエラーログから該当するクロスバーユニットに故障が発生していないか確認してください。故障していた場合は故障部品を確認して交換を実施してください。

PPAR (PPAR-ID XX) is currently unavailable for DR, because another operation is in progress. Try again later.
［説明］指定した物理パーティションはほかのオペレーションによって実行中のため、動的再構成が実行できません。

［対処方法］別セッションで電源の切断または接続されているPCIボックスの構

成情報を変更するオペレーションが実行されている可能性があります。物理パーティションの状態を確認しながら、しばらく待機したあと、再度実行してください。

PPAR (PPAR-ID XX) is not currently running.
［説明］-c unassignオプションまたは-c disconnectオプションを指定したときに、組み込み先の物理パーティションが稼働状態でなかったため、切り離しできませんでした。

［対処方法］物理パーティションの動作状態を確認してください。

PSB power off sequence started.
［説明］PSB（BB）の電源の切断を開始しました。

PSB#XX-X can not be unconfigured because there is Hypervisor global memory.
［説明］指定したPSBはハイパーバイザメモリの移行が必要となるため、切り離しできません。

PSB#XX-X could not be unconfigured from PPAR-ID XX due to Hypervisor error.
［説明］ハイパーバイザから動的再構成の処理でエラーが応答されたため、切り離し処理ができませんでした。

［対処方法］監視メッセージやコンソールメッセージから論理ドメインのOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

PSB#XX-X could not be unconfigured from PPAR-ID XX due to operating system or Logical Domains Manager error.
［説明］論理ドメインのOracle SolarisまたはLogical Domains Managerから動的再構成の処理でエラーが応答されたため、切り離し処理ができませんでした。

［対処方法］監視メッセージやコンソールメッセージから論理ドメインのOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

PSB#XX-X has not been registered in PCL.
［説明］指定したPSBが物理パーティション（PPAR）構成情報に登録されていないため、実行できませんでした。

［対処方法］物理パーティション（PPAR）構成情報を登録してから実行してください。

PSB#XX-X is currently unavailable for DR. Try again later.
［説明］指定したPSBはほかのオペレーションによって実行中であるため動的再構成できません。

［対処方法］動的再構成、物理パーティションの起動／停止、または保守作業が別セッションで実行されている可能性があります。PSBの状態を確認し、しばらく待ってから再実行してください。

PSB#XX-X is not installed.
［説明］指定したPSBが搭載されていないため実行できませんでした。

［対処方法］PSBの指定を誤っている可能性があります。PSBの状態を確認してください。

PSB#XX-X is the last LSB for PPAR-ID XX, and this PPAR is still running. Operation failed.

［説明］指定したPSB（BB）は物理パーティションを構成する最後のPSB（BB）であるため、構成から外すことができませんでした。

［対処方法］物理パーティションを停止してから実行するか、-c reserveオプションを指定して物理パーティションを停止してください。

PSB#XX-X status is FAILED.
［説明］指定したPSB（BB）が故障状態であるため、切り離しできませんでした。

［対処方法］監視メッセージやエラーログでハードウェアに故障が発生していないことを確認してください。故障していた場合は故障部品を確認して交換を実施してください。

PSB#XX-X will be unassigned from PPAR after the PPAR restarts. Continue? [y|n] :
［説明］物理パーティションの再起動後にPSB番号XX-Xを物理パーティションから切り離す予約を行います。継続しますか。

PSB#XX-X will be unassigned from PPAR immediately. Continue?[y|n] :
［説明］PSB番号XX-Xを物理パーティションから切り離します。継続しますか。継続する場合は「y」を、中止する場合は「n」を入力してください。

PSB#XX-X will be unconfigured from PPAR immediately. Continue?[y|n] :
［説明］PSB番号XX-Xを物理パーティションから切り離して割り当て状態にします。継続しますか。継続する場合は「y」を、中止する場合は「n」を入力してください。

Start unconfigure preparation of PSB.
［説明］PSB（BB）の切り離し準備処理を開始しました。

Start unconfiguring PSB from PPAR.
［説明］PSB（BB）の切り離し処理を開始しました。

System may be temporarily suspended, proceed?[y|n] :
［説明］システムは一時的にサスペンド状態となります。継続しますか。継続する場合は「y」を、中止する場合は「n」を入力してください。

The current configuration does not support this operation.
［説明］構成異常によりコマンドが終了しました。

［対処方法］コマンド実行に必要なハードウェアの実装や、機能の設定を見直してください。

The other XSCF is currently unavailable for DR. Try again later.
［説明］スタンバイ状態またはスレーブのXSCFがシャットダウン中のため、動的再構成が実行できませんでした。

［対処方法］スタンバイ状態またはスレーブのXSCFがシャットダウン中である可能性があります。スタンバイ状態またはスレーブのXSCFの状態を確認しながら、再度実行してください。

The system is currently unavailable for DR, because another operation is in progress. Try again later.
［説明］SPARC M12/M10システム全体へのオペレーションが実行中のため、動的再構成が実行できませんでした。

［対処方法］別セッションでファームウェアのアップデートが実行されている可能性があります。ファームウェアのアップデートが完了したあと、再度実行してください。

There is no available action for the following resources.
または

DR operation would require shutting down domains, use force to proceed

［説明］削除しようとしているシステムボード（PSB）の物理I/Oデバイスが論理ドメインに割り当てられている可能性があります。

［対処方法］制御ドメインで、該当する物理I/Oデバイスの割り当て状況をldm list-ioコマンドで確認してください。該当する物理I/Oデバイスが論理ドメインに割り当てられている場合は、ldm remove-ioコマンドで削除してください。

Timeout detected during communicate with Logical Domains Manager.
［説明］Logical Domains Managerとの通信でタイムアウトが発生しました。

［対処方法］監視メッセージやコンソールメッセージから論理ドメインのOracle SolarisまたはLogical Domains Managerで動的再構成が失敗した原因を調べ、要因を取り除いたうえで再度実行してください。

Timeout detected during poweroff.
［説明］動的再構成で行っているシステムボード（PSB）の電源切断処理が完了せずタイムアウトしました。

［対処方法］ハードウェアに故障が発生していないことを確認してください。故障していた場合は故障部品を確認して交換を実施してください。

Timeout detected during unconfiguration of PSB#XX-X.
［説明］動的再構成で行っている切り離し処理が完了せずタイムアウトしました。

［対処方法］監視メッセージやエラーログでハードウェアに故障が発生していないことを確認してください。故障していた場合は故障部品を確認して交換を実施してください。

Unconfigure preparation of PSB has completed.
［説明］PSB（BB）の切り離し準備処理が完了しました。

Unconfigured PSB from PPAR.
［説明］PSB（BB）の切り離し処理が完了しました。

索引

A
addboard，453

C
CPU，23
CPUコア アクティベーションキー情報の確認，84
CPUコア アクティベーションキーの追加，93
CPUコア アクティベーションの設定情報の確認，87
CPUコア アクティベーションの登録，87
CPUコアリソースの使用状況の確認，85
CPUコアリソースの割り当て，93
CPUソケットごとのリソースの使用状況の確認，109
CPUソケット制約の設定，124
CPU動作モード，28，72

D
deleteboard，457

I
I/Oデバイスの使用状況の確認，112
I/Oデバイスの設定，126

L
Logical Domains Manager起動の確認，105

O
Oracle Solarisの条件，38
Oracle Solarisの設定，38

P
PPAR DRモード，65

R
recovery mode，156

S
SPARC64 X+プロセッサ，27，67，71
SPARC64 Xプロセッサ，27，71
SR-IOVの仮想機能の作成・破棄，128

X
XSCFの条件，35
XSCFのステータス管理，29
XSCFの設定，35

か
仮想CPUの設定，116
仮想コンソールの設定，135
仮想ディスクサーバの設定，133
仮想ディスクの設定，134
仮想ネットワークデバイスの設定，132
仮想メモリの設定，121

き
起動デバイスの設定，135

け

ゲストドメインの移行，195，198
ゲストドメインの移行例，198
ゲストドメインの概要，195
ゲストドメインの起動，136
ゲストドメインの構築，154

こ

コマンドメッセージ，453

さ

サービスの確認，106

し

システム運用の条件，29
システム運用の設定，29
システムボード移動の操作例，181
システムボード交換の操作例，188
システムボード削除の操作例，173
システムボード追加の操作例，163
システムボードの削除，98
シャットダウングループの指定，136

す

ステータスの変化，32
スワップ領域，39

せ

制御ドメインコンソール，103
制御ドメインの初期設定，153
制御ドメインへのログイン，152

そ

ソフトウェア条件，24

ち

遅延再構成モードの開始，114

て

デバイス設定情報の確認，82
デバイスの再構成，137
デフォルトサービスの設定，152
デフォルトサービスの設定，115

と

動的再構成，447
動的再構成の条件，53
動的再構成の設定，53
ドメイン構成，1
ドメイン構築，75
ドメイン構築の概要，1

は

ハードウェアリソース，9

ふ

復旧モードの設定，156
復旧モードの有効，137
物理I/Oデバイス，21
物理パーティション，12
物理パーティション構成情報の確認，76
物理パーティション構成情報の設定，91
物理パーティション構成の操作，14
物理パーティション構築，75
物理パーティション構築のながれ，139
物理パーティション再構築のながれ，157
物理パーティションの起動，103
物理パーティションの構成要素，12
物理パーティションの構築と操作，140
物理パーティションのステータス，29
物理パーティションのステータス確認，78
物理パーティションの再構築，157
物理パーティションの停止，103
物理パーティションの動作モードの確認，88
物理パーティションの動作モードの設定，100
物理パーティションの動的再構成，22，72，202
物理パーティションの動的再構成の概要，22
物理パーティションの動的再構成の用途，22
物理パーティションの動的再構成を使用した環境構築・手順例，201
物理パーティション番号を変更する場合の留意点，193

め

メモリミラーの設定，90

メモリミラーモード，38

り

リアルタイムプロセス，40
リソースグループごとに使用状況を確認，109
リソース使用状況の確認，107
リソースのバインド，136
リソース割り当て状況の確認，107

ろ

論理ドメイン構成情報，102
論理ドメイン構成情報の設定，124
論理ドメイン構成情報の表示，89，112
論理ドメイン構築，104
論理ドメイン構築の操作例，151
論理ドメイン構築のながれ，149
論理ドメインとは，17
論理ドメインの構成，223，296
論理ドメインの構成情報の保存，156
論理ドメインの構成の操作，19
論理ドメインの構成要素，17
論理ドメインの構築，41
論理ドメインの構築例，149
論理ドメインの再構成，52
論理ドメインの作成，125
論理ドメインの条件，40
論理ドメインの状態確認，111
論理ドメインのステータス確認，83
論理ドメインの設定，40

わ

割り当て可能な仮想CPU数の確認，106