

Sun Cluster のシステム管理 (Solaris OS 版)



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 821-1031-10
2009年11月, Revision A

Sun Microsystems, Inc. (以下米国 Sun Microsystems 社とします) は、本書に記述されている製品に含まれる技術に関連する知的財産権を所有します。特に、この知的財産権はひとつかそれ以上の米国における特許、あるいは米国およびその他の国において申請中の特許を含んでいることがあります。それらに限定されるものではありません。

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 システムに基づいていることがあります。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびに他の国における登録商標です。フォント技術を含む第三者のソフトウェアは、著作権により保護されており、提供者からライセンスを受けているものです。

U.S. Government Rights Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

この配布には、第三者によって開発された素材を含んでいることがあります。

本製品に含まれる HG-MinchoL、HG-MinchoL-Sun、HG-PMinchoL-Sun、HG-GothicB、HG-GothicB-Sun、および HG-PGothicB-Sun は、株式会社リコーがリョービマジクス株式会社からライセンス供与されたタイプフェースマスタをもとに作成されたものです。HeiseiMin-W3H は、株式会社リコーが財団法人日本規格協会からライセンス供与されたタイプフェースマスタをもとに作成されたものです。フォントとして無断複製することは禁止されています。

Sun、Sun Microsystems、Sun のロゴマーク、Solaris のロゴマーク、Java Coffee Cup のロゴマーク、docs.sun.com Solstice DiskSuite、Solaris Volume Manager、Sun Enterprise SyMON、JumpStart、Sun Management Center、OpenBoot、Java および Solaris は、米国およびその他の国における米国 Sun Microsystems 社の商標、登録商標もしくは、サービスマークです。

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。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャに基づくものです。

OPENLOOK、OpenBoot、JLE は、サン・マイクロシステムズ株式会社の登録商標です。

Wnn は、京都大学、株式会社アステック、オムロン株式会社で共同開発されたソフトウェアです。

Wnn8 は、オムロン株式会社、オムロンソフトウェア株式会社で共同開発されたソフトウェアです。Copyright(C) OMRON Co., Ltd. 1995-2000. All Rights Reserved. Copyright(C) OMRON SOFTWARE Co., Ltd. 1995-2009 All Rights Reserved.

「ATOK for Solaris」は、株式会社ジャストシステムの著作物であり、「ATOK for Solaris」にかかる著作権、その他の権利は株式会社ジャストシステムおよび各権利者に帰属します。

「ATOK」および「推測変換」は、株式会社ジャストシステムの登録商標です。

「ATOK for Solaris」に添付するフェイスマーク辞書は、株式会社ビレッジセンターの許諾のもと、同社が発行する『インターネット・パソコン通信フェイスマークガイド』に添付のものを使用しています。

「ATOK for Solaris」に含まれる郵便番号辞書 (7 桁/5 桁) は日本郵政公社が公開したデータを元に制作された物です (一部データの加工を行なっています)。

Unicode は、Unicode, Inc. の商標です。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザインタフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは、OPEN LOOK のグラフィカル・ユーザインタフェースを実装するか、またはその他の方法で米国 Sun Microsystems 社との書面によるライセンス契約を遵守する、米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

本書で言及されている製品や含まれている情報は、米国輸出規制法で規制されるものであり、その他の国の輸出入に関する法律の対象となる場合があります。核、ミサイル、化学あるいは生物兵器、原子力の海洋輸送手段への使用は、直接および間接を問わず厳しく禁止されています。米国が禁輸の対象としている国や、限定はされませんが、取引禁止顧客や特別指定国民のリストを含む米国輸出排除リストで指定されているものへの輸出および再輸出は厳しく禁止されています。

本書は、「現状のまま」をベースとして提供され、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、なんらの保証も行われぬものとします。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

原典: Sun Cluster System Administration Guide for Solaris OS

Part No: 820-7358

Revision A

目次

はじめに	11
1 Sun Cluster の管理の概要	15
Sun Cluster の管理の概要	16
ゾーンクラスタに関する作業	16
Solaris オペレーティングシステム (Solaris OS) の機能制限	17
管理ツール	18
グラフィカルユーザインタフェース	18
コマンド行インタフェース	18
クラスタ管理の準備	20
Sun Cluster ハードウェア構成の記録	20
管理コンソールの使用	20
クラスタのバックアップ	21
クラスタ管理の開始	22
▼ クラスタに遠隔ログインする	24
▼ クラスタコンソールに安全に接続する	25
▼ クラスタ構成ユーティリティにアクセスする	26
▼ Sun Cluster のパッチ情報を表示する	27
▼ Sun Cluster のリリース情報とバージョン情報を表示する	27
▼ 構成されているリソースタイプ、リソースグループ、リソースを表示する	29
▼ クラスタコンポーネントの状態を確認する	31
▼ パブリックネットワークの状態を確認する	34
▼ クラスタ構成を表示する	35
▼ 基本的なクラスタ構成を検証する	45
▼ グローバルマウントポイントを確認する	48
▼ Sun Cluster のコマンドログの内容を表示する	49

2 Sun Cluster と RBAC	53
RBAC の設定と Sun Cluster での使用	53
Sun Cluster RBAC の権限プロファイル	54
Sun Cluster 管理権限プロファイルによる RBAC 役割の作成と割り当て	55
▼管理役割ツールを使用して役割を作成する方法	55
▼コマンド行から役割を作成する方法	57
ユーザーの RBAC プロパティの変更	59
▼ユーザーアカウントツールを使用してユーザーの RBAC プロパティを変更する 方法	59
▼コマンド行からユーザーの RBAC プロパティを変更する方法	60
3 クラスタの停止と起動	61
クラスタの停止と起動の概要	61
▼クラスタを停止する	63
▼クラスタを起動する	65
▼クラスタを再起動する	68
クラスタ内の1つのノードの停止と起動	72
▼ノードを停止する	73
▼ノードを起動する	77
▼ノードを再起動する	79
▼非クラスタモードでノードを起動する	83
満杯の /var ファイルシステムを修復する	87
▼満杯の /var ファイルシステムを修復する	87
4 データ複製のアプローチ	89
データ複製についての理解	90
サポートされるデータ複製方式	91
クラスタ内でのストレージベースのデータ複製の使用	92
クラスタ内でストレージベースのデータ複製を使用する際の要件と制限	94
クラスタ内でストレージベースのデータ複製を使用する際の手動回復に関する懸 念事項	95
ストレージベースのデータ複製を使用する際のベストプラクティス	95

5	グローバルデバイス、ディスクパス監視、およびクラスタファイルシステムの管理	97
	グローバルデバイスとグローバルな名前空間の管理の概要	97
	Solaris ボリュームマネージャーのグローバルデバイスのアクセス権	98
	グローバルデバイスでの動的再構成	98
	Veritas Volume Manager による管理に関する注意事項	99
	ストレージベースの複製されたデバイスの管理	101
	Hitachi TrueCopy で複製されたデバイスの管理	101
	EMC Symmetrix Remote Data Facility で複製したデバイスの管理	113
	クラスタファイルシステムの管理の概要	127
	クラスタファイルシステムの制限事項	127
	VxFS サポートについてのガイドライン	128
	デバイスグループの管理	129
	▼ グローバルデバイス名前空間を更新する	132
	グローバルデバイス名前空間を移行する	133
	▼ 専用パーティションから lofi デバイスにグローバルデバイス名前空間を移行する	133
	▼ lofi デバイスから専用パーティションにグローバルデバイス名前空間を移行する	134
	デバイスグループを追加および登録する	136
	▼ デバイスグループを追加および登録する (Solaris ボリュームマネージャー)	136
	▼ デバイスグループ (raw ディスク) を追加および登録する	138
	▼ 複製デバイスグループ (ZFS) の追加と登録方法	139
	▼ ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)	141
	デバイスグループ名を維持する	141
	デバイスグループを削除して登録を解除する (Solaris ボリュームマネージャー)	142
	▼ すべてのデバイスグループからノードを削除する	142
	▼ デバイスグループからノードを削除する (Solaris ボリュームマネージャー)	143
	▼ 1つのクラスタ内に4つ以上のディスクセットを作成する	145
	▼ ディスクをカプセル化する際に新しいディスクグループを作成する (VERITAS Volume Manager)	147
	▼ 新しいボリュームを既存のデバイスグループに追加する (Veritas Volume Manager)	149
	▼ 既存のディスクグループをデバイスグループに変換する (Veritas Volume Manager)	150

▼ デバイスグループに新しいマイナー番号を割り当てる (VERITAS Volume Manager)	150
▼ ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)	151
▼ ディスクグループの構成変更を登録する (VERITAS Volume Manager)	154
▼ ローカルディスクグループをデバイスグループに変換する (VxVM)	155
▼ デバイスグループをローカルディスクグループに変換する (VxVM)	156
▼ デバイスグループからボリュームを削除する (Veritas Volume Manager)	157
▼ デバイスグループを削除して登録を解除する (Veritas Volume Manager)	158
▼ デバイスグループにノードを追加する (VERITAS Volume Manager)	159
▼ デバイスグループからノードを削除する (Veritas Volume Manager)	161
▼ raw ディスクデバイスグループからノードを削除する	163
▼ デバイスグループのプロパティを変更する	165
▼ デバイスグループの二次ノードの希望数を設定する	166
▼ デバイスグループ構成の一覧を表示する	170
▼ デバイスグループの主ノードを切り替える	171
▼ デバイスグループを保守状態にする	172
ストレージデバイス用の SCSI プロトコル設定の管理	174
▼ すべてのストレージデバイスのデフォルトのグローバルな SCSI プロトコル設定 を表示する	175
▼ 単一ストレージデバイスの SCSI プロトコルを表示する	176
▼ すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル 設定を変更する	176
▼ 単一ストレージデバイスのフェンシングプロトコルを変更する	178
クラスタファイルシステムの管理	180
▼ クラスタファイルシステムを追加する	180
▼ クラスタファイルシステムを削除する	185
▼ クラスタ内のグローバルマウントを確認する	187
ディスクパス監視の管理	187
▼ ディスクパスを監視する	188
▼ ディスクパスの監視を解除する方法	190
▼ 障害のあるディスクパスを表示する	191
▼ ディスクパスの状態エラーを解決する	192
▼ ファイルからディスクパスを監視する	192
▼ 監視しているすべての共有ディスクパスが失敗したときのノードの自動再起動を 有効にする	195
▼ すべての監視共有ディスクパスが失敗した場合にノードの自動再起動を無効にす	

る	195
6 定足数の管理	197
定足数デバイスの管理	197
定足数デバイスへの動的再構成	199
定足数デバイスの追加	200
定足数デバイスの削除または交換	213
定足数デバイスの保守	217
Sun Cluster 定足数サーバーの管理	225
定足数サーバー構成ファイルの概要	225
Sun Cluster 定足数サーバー ソフトウェアの起動と停止	226
▼定足数サーバーを起動する	226
▼定足数サーバーを停止する	227
定足数サーバーに関する情報の表示	228
期限切れの定足数サーバークラスタ情報のクリーンアップ	229
7 クラスタインターコネクトとパブリックネットワークの管理	233
クラスタインターコネクトの管理	233
クラスタインターコネクトでの動的再構成	235
▼クラスタインターコネクトの状態を確認する	235
▼クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートス イッチを追加する	236
▼クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートス イッチを削除する	239
▼クラスタトランスポートケーブルを有効にする	243
▼クラスタトランスポートケーブルを無効にする	244
▼トランスポートアダプタのインスタンス番号を確認する	246
▼既存のクラスタのプライベートネットワークアドレスまたはアドレス範囲を変更 する	247
パブリックネットワークの管理	250
クラスタでIP ネットワークマルチパスグループを管理する	250
パブリックネットワークインタフェースでの動的再構成	251
8 ノードの追加と削除	255
クラスタへのノードの追加	255

▼ ノードを認証ノードリストに追加する	256
グローバルクラスタ内の非投票ノード (ゾーン) の作成	259
クラスタからのノードの削除	261
▼ ゾーンクラスタからノードを削除する	263
▼ クラスタソフトウェア構成からノードを削除する	263
▼ グローバルクラスタから非投票ノード (ゾーン) を削除する	267
▼ 2 ノード接続より大きなクラスタでアレイと単一ノード間の接続を削除する ..	267
▼ エラーメッセージを修正する	270
9 クラスタの管理	271
クラスタの管理の概要	271
▼ クラスタ名を変更する	272
▼ ノード ID をノード名にマップする	273
▼ 新しいクラスタノード認証で作業する	274
▼ クラスタの時刻をリセットする	276
▼ SPARC: ノードで OpenBoot PROM (OBP) を表示する	278
▼ ノードのプライベートホスト名を変更する	279
▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を追加する	282
▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を変更する	283
▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を削除する	284
▼ ノードを保守状態にする	285
▼ ノードを保守状態から戻す	287
ゾーンクラスタ管理タスクの実行	289
▼ ゾーンクラスタを削除する	290
▼ ゾーンクラスタからファイルシステムを削除する	291
▼ ゾーンクラスタからストレージデバイスを削除する	293
▼ Sun Cluster ソフトウェアをクラスタノードからアンインストールする	295
ノードのアンインストールに伴う問題の解決	297
Sun Cluster SNMP イベント MIB の作成、設定、および管理	298
障害追跡	305
▼ 非クラスタモードで起動したノードから Solaris Volume Manager メタセットを取 得する	305
10 CPU 使用率の制御の構成	309
CPU 制御の概要	309

シナリオの選択	310
公平配分スケジューラ	311
CPU 制御の構成	311
▼ SPARC: Solaris 9 OS で CPU 使用率を制御する	311
▼ グローバルクラスタの投票ノードで CPU 使用率を制御する	313
▼ デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する	315
▼ 専用のプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する	318
11 Sun Cluster ソフトウェアとファームウェアのパッチ	323
Sun Cluster へのパッチの適用の概要	323
Sun Cluster パッチの適用に関する注意事項	324
Sun Cluster ソフトウェアへのパッチの適用	325
▼ 再起動パッチを適用する (ノード)	325
▼ 再起動パッチを適用する (クラスタ)	330
▼ 非再起動 Sun Cluster パッチを適用する	334
▼ フェイルオーバーノード構成をしている場合に、シングルユーザーモードでパッチを適用する	335
Sun Cluster パッチの変更	339
12 クラスタのバックアップと復元	341
クラスタのバックアップ	341
▼ バックアップするファイルシステム名を確認する	342
▼ 完全なバックアップに必要なテープ数を決定する	342
▼ ルート(/) ファイルシステムをバックアップする	343
▼ ミラーのオンラインバックアップを実行する (Solaris ボリュームマネージャー)	346
▼ ボリュームのオンラインバックアップの実行 (VERITAS Volume Manager)	349
▼ クラスタ構成をバックアップする	353
クラスタファイルの復元の作業マップ	354
▼ 個々のファイルを対話形式で復元する (Solaris ボリュームマネージャー)	355
▼ ルート(/)ファイルシステムを復元する (/)	355
▼ Solstice DiskSuite メタデバイスまたは Solaris ボリュームマネージャー上に存在していたルート (/) ファイルシステムを復元する	357
▼ カプセル化されていないルート (/) ファイルシステムを復元する (Veritas Volume	

Manager)	363
▼カプセル化されたルート (/) ファイルシステムを復元する (VERITAS Volume Manager)	365
13 グラフィカルユーザーインターフェースによる Sun Cluster の管理	371
Sun Cluster Manager の概要	371
SPARC: Sun Management Center の概要	372
Sun Cluster Manager の構成	373
RBAC の役割の設定	373
▼共通エージェントコンテナを使用して、サービスまたは管理エージェントのポート番号を変更する	373
▼Sun Cluster Manager のサーバーアドレスを変更する	374
▼共通エージェントコンテナのセキュリティー鍵を再生成する	375
Sun Cluster Manager ソフトウェアの起動	376
▼Sun Cluster Manager を起動する	376
A 例	381
Sun StorEdge Availability Suite または Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成	381
クラスタにおける Sun StorageTek Availability Suite ソフトウェアの理解	382
クラスタ間でホストベースのデータ複製を構成するためのガイドライン	385
作業マップ: データ複製の構成例	391
クラスタの接続とインストール	392
デバイスグループとリソースグループの構成例	394
データ複製の有効化例	408
データ複製の実行例	413
フェイルオーバーの管理方法の例	420
B Sun Cluster オブジェクト指向コマンド	423
オブジェクト指向コマンド名および別名	423
オブジェクト指向コマンドセットの概要	424
索引	435

はじめに

『Sun Cluster のシステム管理 (Solaris OS 版)』このマニュアルでは、SPARC® および x86 ベースのシステムで Sun™ Cluster 構成を管理する手順について説明します。

注 - この Sun Cluster リリースでは、SPARC および x86 系列のプロセッサアーキテクチャ (UltraSPARC、SPARC64、AMD64、および Intel 64) を使用するシステムをサポートします。このドキュメントでは、x86 とは 64 ビット x86 互換製品の広範囲なファミリーを指します。このドキュメントの情報では、特に明示されている場合以外はすべてのプラットフォームに関係します。

このマニュアルは、Sun のソフトウェアとハードウェアについて幅広い知識を持っている上級システム管理者を対象としています。販売活動のガイドとしては使用しないでください。

このマニュアルで説明されている作業手順を行うには、Solaris™ オペレーティングシステムに関する知識と、Sun Cluster と共に使用するボリューム管理ソフトウェアに関する知識が必要です。

UNIX コマンド

このマニュアルには、Sun Cluster 構成の管理に固有なコマンドに関する情報が記載されています。このマニュアルでは、基本的な UNIX® コマンドや手順に関するすべての情報は説明されていない場合があります。

これらの情報については、次を参照してください。

- Solaris ソフトウェアのオンラインマニュアル
- システムに付属するその他のソフトウェアマニュアル
- Solaris オペレーティングシステム (Solaris OS) のマニュアルページ

表記上の規則

このマニュアルでは、次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ます。

表P-1 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 machine_name% you have mail.
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	machine_name% su Password:
<i>aabbcc123</i>	プレースホルダー: 実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、rm <i>filename</i> と入力します。
<i>AaBbCc123</i>	参照する書名を示します。	『ユーザーズガイド』の第6章を参照ください。 キャッシュはローカルに保存されているコピーです。 ファイルを保存しないでください。 注: 強調表示されたいくつかの項目はオンラインで太字で表示されます。

コマンド例のシェルプロンプト

次の表は、Solaris OS に含まれるシェルのデフォルトの UNIX システムプロンプトとスーパーユーザーのプロンプトを示しています。コマンド例で表示されるデフォルトのシステムプロンプトは、Solaris のリリースによって異なることに注意してください。

表P-2 シェルプロンプトについて

シェル	プロンプト
Bash シェル、Korn シェル、および Bourne シェル	\$

表 P-2 シェルプロンプトについて (続き)

シェル	プロンプト	
スーパーユーザーの Bash シェル、Korn シェル、および Bourne シェル	C シェル	machine_
C シェルのスーパーユーザー	machine_name#	
#		

関連マニュアル

関連する Sun Cluster トピックについての情報は、以下の表に示すマニュアルを参照してください。Sun Cluster のマニュアルは、すべて <http://docs.sun.com> で入手できます。

項目	マニュアル
概要	『Sun Cluster Overview for Solaris OS』
Concept	『Sun Cluster Concepts Guide for Solaris OS』
ハードウェアの設計と管理	『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』 各ハードウェア管理ガイド
ソフトウェアのインストール	『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』
データサービスのインストールと管理	『Sun Cluster Data Services Planning and Administration Guide for Solaris OS』 各データサービスガイド
データサービスの開発	『Sun Cluster Data Services Developer's Guide for Solaris OS』
システム管理	『Sun Cluster のシステム管理 (Solaris OS 版)』
エラーメッセージ	『Sun Cluster Error Messages Guide for Solaris OS』
コマンドと関数のリファレンス	『Sun Cluster Reference Manual for Solaris OS』

Sun Cluster のマニュアルの完全なリストについては、使用しているリリースの Sun Cluster ソフトウェアのリリースノートを <http://docs.sun.com> で参照してください。

マニュアル、サポート、およびトレーニング

Sun の Web サイトでは、次のサービスに関する情報も提供しています。

- ドキュメント (<http://jp.sun.com/documentation/>)
- サポート (<http://jp.sun.com/support/>)
- トレーニング (<http://jp.sun.com/training/>)

コメントをお寄せください

弊社では、ドキュメントの改善に努力しており、お客様からのコメントおよびご提案をお受けしております。コメントを投稿するには、<http://docs.sun.com> にアクセスして「フィードバック」をクリックします。

問い合わせについて

Sun Cluster のインストールまたは使用で問題が発生した場合は、ご購入先にお問い合わせください。ご購入先には次の情報をお知らせください。

- 名前と電子メールアドレス
- 会社名、住所、および電話番号
- システムのモデルとシリアル番号
- オペレーティングシステムのバージョン番号 (例: Solaris 9)
- Sun Cluster のリリース番号 (例: Sun Cluster 3.2)

次のコマンドを使用し、システムに関して、サービスプロバイダに必要な情報を収集してください。

コマンド	機能
<code>prtconf -v</code>	システムメモリのサイズと周辺デバイス情報を表示します
<code>psrinfo -v</code>	プロセッサの情報を表示する
<code>showrev -p</code>	インストールされているパッチを報告する
<code>SPARC:prtdiag -v</code>	システム診断情報を表示する
<code>/usr/cluster/bin/clnode show-rev</code>	Sun Cluster のリリースとパッケージのバージョン情報を表示する

上記の情報にあわせて、`/var/adm/messages` ファイルの内容もご購入先にお知らせください。

Sun Cluster の管理の概要

この章では、グローバルクラスタとゾーンクラスタの管理に関する次の情報と、Sun™ Cluster 管理ツールの使用手順について説明します。

- 16 ページの「Sun Cluster の管理の概要」
- 17 ページの「Solaris オペレーティングシステム (Solaris OS) の機能制限」
- 18 ページの「管理ツール」
- 20 ページの「クラスタ管理の準備」
- 22 ページの「クラスタ管理の開始」

Solaris™ 10 OS の場合、グローバルクラスタは、1 つ以上のグローバルクラスタ投票ノードのみで構成されますが、オプションで 0 個以上のグローバルクラスタ非投票ノードを含むこともできます。グローバルクラスタには、オプションで、Solaris 8 OS、Solaris 9 OS、LINUX OS、またはネイティブブランドの非大域ゾーン(ノードではなく高可用性コンテナ)をリソースとして含めることもできます。ゾーンクラスタには、グローバルクラスタが必須です。ゾーンクラスタの概要については、『[Sun Cluster Concepts Guide for Solaris OS](#)』を参照してください。

Solaris 10 OS の場合、ゾーンクラスタは、クラスタブランドの 1 つ以上の投票ノードのみで構成されます。ゾーンクラスタは、グローバルクラスタに依存しており、したがって、グローバルクラスタを必要とします。グローバルクラスタはゾーンクラスタを含みません。ゾーンクラスタは、グローバルクラスタなしでは構成できません。ゾーンクラスタが同じマシン上に含むことができるのは、多い場合でも 1 つのゾーンクラスタのみです。ゾーンクラスタノードは、同じマシン上のグローバルクラスタ投票ノードが動作している場合のみ動作します。あるマシンのグローバルクラスタ投票ノードで障害が発生すると、同じマシン上のすべてのゾーンクラスタノードも動作しなくなります。

Sun Cluster の管理の概要

Sun Cluster の高可用性環境によって、重要なアプリケーションの可用性がエンドユーザーに対して保証されます。システム管理者の業務は、Sun Cluster 構成の安定した動作を保証することです。

管理作業を始める前に、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』と『Sun Cluster Concepts Guide for Solaris OS』に記載されている計画情報をよく理解しておいてください。ゾーンクラスタの作成手順については、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の「ゾーンクラスタの設定」を参照してください。Sun Cluster の管理は、次の作業ごとに各マニュアルにまとめられています。

- グローバルクラスタまたはゾーンクラスタを定期的に (または毎日) 管理および維持するための標準的な作業。これらの作業は、このマニュアルで説明されています。
- インストール、構成、属性の変更などのデータサービス作業。これらの作業は、『Sun Cluster Data Services Planning and Administration Guide for Solaris OS』で説明されています。
- 記憶装置やネットワークハードウェアの追加や保守などのサービス作業。これらの作業は、『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』で説明されています。

一般に、Sun Cluster の管理作業はクラスタの稼働中に実行できます。クラスタからノードを取り外す必要がある場合、あるいはノードを停止する必要がある場合でも、残りのノードがクラスタを稼働している間に作業を行うことができます。Solaris 10 では、明記していないかぎり、Sun Cluster の管理作業はグローバルクラスタの投票ノードで行うべきです。クラスタ全体を停止する必要がある手順については、ダウンタイムのスケジュールを通常の業務時間外に設定してシステムへの影響を最小限に抑えてください。クラスタまたはクラスタノードを停止する予定があるときは、あらかじめユーザーに通知しておいてください。

ゾーンクラスタに関する作業

Sun Cluster の 2 つの管理コマンド (`cluster` および `clnode`) は、ゾーンクラスタでも実行できます。ただし、このコマンドの対象は、コマンドが発行されたゾーンクラスタに限定されます。たとえば、グローバルクラスタの投票ノードで `cluster` コマンドを使用すると、投票グローバルクラスタおよびすべてのゾーンクラスタに関するすべての情報が得られます。`cluster` コマンドをゾーンクラスタで使用すると、そのゾーンクラスタのみの情報が得られます。

`clzonecluster` コマンドを投票ノードで使用すると、グローバルクラスタ内のすべてのゾーンクラスタが対象になります。ゾーンクラスタコマンドはゾーンクラスタ上のすべてのノードを対象とします (コマンド発行時に停止していたノードも対象になります)。

ゾーンクラスタは、リソースグループマネージャ (Resource Group Manager、RGM) の制御下にあるリソースの委任管理をサポートしています。そのため、ゾーンクラスタの管理者は、クラスタ境界にまたがるゾーンクラスタ依存関係を表示できます (ただし、変更はできません)。クラスタ境界にまたがるゾーンクラスタ依存関係を作成し、変更し、削除できるのは、投票ノード内の管理者のみです。

次の表に、ゾーンクラスタで実行する主な管理作業を示します。

- ゾーンクラスタの作成 - `clzonecluster configure` コマンドを使用して、新しいゾーンクラスタを作成します。手順については、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「[ゾーンクラスタの設定](#)」を参照してください。
- ゾーンクラスタの起動と再起動 - [第 3 章「クラスタの停止と起動](#)」を参照してください。
- ゾーンクラスタへのノードの追加 - [第 8 章「ノードの追加と削除](#)」を参照してください。
- ゾーンクラスタからのノードの削除 - [263 ページの「ゾーンクラスタからノードを削除する](#)」を参照してください。
- ゾーンクラスタの構成の表示 - [35 ページの「クラスタ構成を表示する](#)」を参照してください。
- ゾーンクラスタの構成の検証 - [45 ページの「基本的なクラスタ構成を検証する](#)」を参照してください。
- ゾーンクラスタの停止 - [第 3 章「クラスタの停止と起動](#)」を参照してください。

Solaris オペレーティングシステム (Solaris OS) の機能制限

Solaris 10 Management Facility (SMF) 管理インタフェースを使用して、次の Sun Cluster サービスを有効または無効にしないでください。

表 1-1 Sun Cluster サービス

Sun Cluster サービス	FMRI
<code>pnm</code>	<code>svc:/system/cluster/pnm:default</code>
<code>cl_event</code>	<code>svc:/system/cluster/cl_event:default</code>
<code>cl_eventlog</code>	<code>svc:/system/cluster/cl_eventlog:default</code>
<code>rpc_pmf</code>	<code>svc:/system/cluster/rpc_pmf:default</code>
<code>rpc_fed</code>	<code>svc:/system/cluster/rpc_fed:default</code>
<code>rgm</code>	<code>svc:/system/cluster/rgm:default</code>

表 1-1 Sun Cluster サービス (続き)

Sun Cluster サービス	FMRI
scdpm	svc:/system/cluster/scdpm:default
cl_ccra	svc:/system/cluster/cl_ccra:default
scsymon_srv	svc:/system/cluster/scsymon_srv:default
spm	svc:/system/cluster/spm:default
cl_svc_cluster_milestone	svc:/system/cluster/cl_svc_cluster_milestone:default
cl_svc_enable	svc:/system/cluster/cl_svc_enable:default
network-multipathing	svc:/system/cluster/network-multipathing

管理ツール

Sun Cluster 構成で管理作業を行うときは、グラフィカルユーザーインターフェース (Graphical User Interface、GUI) またはコマンド行を使用できます。次のセクションでは、GUI とコマンド行の管理ツールの概要を示します。

グラフィカルユーザーインターフェース

Sun Cluster ソフトウェアは、GUI ツールをサポートしています。これらのツールを使えば、クラスタに対してさまざまな管理作業を行うことができます。GUI ツールの 1 つに Sun Cluster Manager があります。また、SPARC ベースのシステムで Sun Cluster ソフトウェアを使用している場合は、Sun Management Center があります。Sun Cluster Manager と Sun Management Center の詳細と構成手順については、[第 13 章「グラフィカルユーザーインターフェースによる Sun Cluster の管理」](#) を参照してください。Sun Cluster Manager に固有の使い方については、GUI のオンラインヘルプを参照してください。

コマンド行インターフェース

Sun Cluster のほとんどの管理作業は、`clsetup(1CL)` ユーティリティを使用して対話形式で実行できます。可能なかぎり、本書の管理手順は `clsetup` ユーティリティを使用します。

`clsetup` ユーティリティを使用すると、「メイン」メニュー内の以下の項目を管理できます。

- 定足数 (quorum)
- リソースグループ

- データサービス
- クラスタインターコネクト
- デバイスグループとボリューム
- プライベートホスト名
- 新規ノード
- そのほかのクラスタタスク

Sun Cluster 構成を管理するために使用するそのほかのコマンドを次のリストに示します。詳細は、マニュアルページを参照してください。

<code>ccp(1M)</code>	クラスタへの遠隔コンソールアクセスを開始します。
<code>if_mpadm(1M)</code>	IP ネットワークマルチパスグループ内のあるアダプタから別のアダプタに IP アドレスを切り換えます。
<code>claccess(1CL)</code>	ノードを追加するために Sun Cluster アクセスポリシーを管理します。
<code>cldevice(1CL)</code>	Sun Cluster デバイスを管理します。
<code>cldevicegroup(1CL)</code>	Sun Cluster デバイスグループを管理します。
<code>clinterconnect(1CL)</code>	Sun Cluster インターコネクトを管理します。
<code>clnasdevice(1CL)</code>	Sun Cluster 構成の NAS デバイスへのアクセスを管理します。
<code>clnode(1CL)</code>	Sun Cluster ノードを管理します。
<code>clquorum(1CL)</code>	Sun Cluster 定足数を管理します。
<code>clreslogicalhostname(1CL)</code>	論理ホスト名のために Sun Cluster リソースを管理します。
<code>clresource(1CL)</code>	Sun Cluster データサービスのリソースを管理します。
<code>clresourcegroup(1CL)</code>	Sun Cluster データサービスのリソースを管理します。
<code>clresourcetype(1CL)</code>	Sun Cluster データサービスのリソースを管理します。
<code>clressharedaddress(1CL)</code>	共有アドレスのために Sun Cluster リソースを管理します。
<code>clsetup(1CL)</code>	Sun Cluster 構成を対話形式で構成します。
<code>clsnmphost(1CL)</code>	Sun Cluster SNMP ホストを管理します。
<code>clsnmpmib(1CL)</code>	Sun Cluster SNMP MIB を管理します。
<code>clsnmpuser(1CL)</code>	Sun Cluster SNMP ユーザーを管理します。
<code>cltelemetryattribute(1CL)</code>	システムリソース監視を構成します。

<code>cluster(1CL)</code>	Sun Cluster 構成のグローバル構成とグローバルステータスを管理します。
<code>clvxdm(1CL)</code>	Veritas Volume Manager (VxVM) を Sun Cluster ノード上で初期化し、状況に応じてルートディスクをカプセル化します。
<code>clzonecluster(1CL)</code>	ゾーンクラスタの作成と変更を行います。

さらに、コマンドを使用して Sun Cluster 構成のボリューム管理ソフトウェアを管理することもできます。使用するコマンドは、クラスタで使用している特定のボリュームマネージャー (Solstice DiskSuite™ ソフトウェア、Veritas Volume Manager、または Solaris ボリュームマネージャー) によって異なります。

注 - Sun Cluster ソフトウェアにはマニュアルに記載されていないコマンドがあり、ソフトウェア自体または Sun エンジニアが開発、診断、またはサポート目的でのみ使用します。マニュアルに記載されていないコマンドをエンドユーザーが使用することは、Sun サポート契約には含まれていません。これらのコマンドが不具合の回避方法としてドキュメントに記載されているか、または特定の状況に対する指示として Sun サポート担当者により提供されている場合を除きます。

クラスタ管理の準備

ここでは、クラスタの管理を準備する方法について説明します。

Sun Cluster ハードウェア構成の記録

Sun Cluster ハードウェア構成は時とともに変化していくので、サイトに固有なハードウェアの特徴は記録しておきます。クラスタを変更または更新するときには、このハードウェアの記録を参照することで管理作業を少なくすることができます。また、さまざまなクラスタ構成要素間のケーブルや接続部にラベルを付けておくと、管理作業が簡単になります。

また、元のクラスタ構成とその後の変更を記録しておく、サン以外のサービスプロバイダがクラスタをサービスする時間を節約できます。

管理コンソールの使用

「管理コンソール」として専用のワークステーション、または管理ネットワークを介して接続されたワークステーションを使用すると動作中のクラスタを管理できます。通常は、Cluster Control Panel (CCP) と、グラフィカルユーザーイン

ターフェース(GUI) ツールを管理コンソールにインストールして実行します。CCPの詳細は、24 ページの「クラスタに遠隔ログインする」を参照してください。Sun Management Center 用の Cluster Control Panel モジュールと Sun Cluster Manager GUI ツールをインストールする方法については、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』を参照してください。

管理コンソールはクラスタノードではありません。管理コンソールは、パブリックネットワークまたはネットワークベースの端末集配信装置(コンセントレータ)を通じてクラスタノードに遠隔アクセスするために使用します。

SPARC クラスタが Sun Enterprise™ 10000 サーバーで構成されている場合、管理コンソールからシステムサービスプロセッサ (SSP) にログインする必要があります。netcon コマンドを使用して接続する。netcon が Sun Enterprise 10000 ドメインと接続する場合デフォルトは、ネットワークインタフェースを経由する方法を使用します。ネットワークにアクセスできない場合は、-f オプションを使用するか、通常の netcon セッション中に ~* を入力し、netcon を「排他モード」で使用できます。どちらの解決方法でも、ネットワークにアクセスできなくなった場合には、シリアルインタフェースに切り換えることができます。

Sun Cluster には、専用の管理コンソールは必要ありませんが、専用コンソールを使用すると、次の利点が得られます。

- コンソールと管理ツールを同じマシンにまとめることで、クラスタ管理を一元化できます。
- システム管理者や保守担当者がすみやかに問題を解決できるようになる可能性があります。

クラスタのバックアップ

ご使用のクラスタを定期的にバックアップしてください。Sun Cluster ソフトウェアは高可用性環境を備えており、データのミラー化されたコピーを記憶装置に保存していますが、これが定期的なバックアップの代わりになるとは考えないでください。Sun Cluster 構成は、複数の障害に耐えることができますが、ユーザーやプログラムのエラー、または致命的な障害から保護する機能を備えていません。したがって、データ損失に対する保護のために、バックアップ手順を用意しておいてください。

次の情報もバックアップしてください。

- すべてのファイルシステムのパーティション
- DBMS データサービスを実行している場合は、すべてのデータベースのデータ
- すべてのクラスタディスクのディスクパーティション情報

- md.tab ファイル (ボリュームマネージャーとして Solstice DiskSuite Solaris Volume Manager を使用している場合)

クラスタ管理の開始

表 1-2 クラスタ管理の開始点を示します。

注-グローバルクラスタ投票ノードからのみ実行する Sun Cluster コマンドは、ゾーンクラスタには使用できません。各種ゾーンでのコマンドの有効な使用方法については、Sun Cluster の該当するマニュアルページを参照してください。

表 1-2 Sun Cluster 3.2 管理ツール

作業	ツール	参照先
クラスタへの遠隔ログイン	ccp コマンドを使用してクラスタコントロールパネル (CCP) を起動します。次に、cconsole、crlogin、cssh、ctelnet の中からいずれかのアイコンを選択します。	24 ページの「クラスタに遠隔ログインする」 25 ページの「クラスタコントロールに安全に接続する」
対話形式でのクラスタの構成	clzonecluster(1CL) ユーティリティーか clsetup(1CL) ユーティリティーを起動します。	26 ページの「クラスタ構成ユーティリティーにアクセスする」
Sun Cluster のリリース番号とバージョン情報の表示	clnode(1CL) コマンドと show-rev --v -node サブコマンドおよびオプションを使用します。	27 ページの「Sun Cluster のリリース情報とバージョン情報を表示する」
インストールされているリソース、リソースグループ、リソースタイプの表示	リソース情報を表示するには、以下に示すコマンドを使用します。 <ul style="list-style-type: none"> ■ clresource(1CL) ■ clresourcegroup(1CL) ■ clresourcetype(1CL) 	29 ページの「構成されているリソースタイプ、リソースグループ、リソースを表示する」
クラスタコンポーネントをグラフィカルに監視	Sun Cluster Manager を使用します。	オンラインヘルプを参照

表 1-2 Sun Cluster 3.2 管理ツール (続き)

作業	ツール	参照先
いくつかのクラスタコンポーネントをグラフィカルに管理	Sun Management Center 用の Sun Cluster Manager または Sun Cluster モジュールを使用します (Sun Management Center は、SPARC ベースシステム上の Sun Cluster でのみ利用可)。	Sun Cluster Manager の場合は、オンラインヘルプを参照 Sun Management Center の場合は、Sun Management Center のマニュアルを参照
クラスタコンポーネントの状態を確認します。	<code>cluster(1CL)</code> コマンドを <code>status</code> サブコマンドとともに使用します。	31 ページの「クラスタコンポーネントの状態を確認する」
パブリックネットワーク上の IP ネットワークマルチパスグループの状態確認	グローバルクラスタの場合は、 <code>-m</code> オプションを指定して、 <code>clnode(1CL) status</code> コマンドを使用します。 ゾーンクラスタの場合は、 <code>clzonecluster(1CL) show</code> コマンドを使用します。	34 ページの「パブリックネットワークの状態を確認する」
クラスタ構成を表示します。	グローバルクラスタの場合は、 <code>cluster(1CL)</code> コマンドを <code>show</code> サブコマンドとともに使用します。 ゾーンクラスタの場合は、 <code>clzonecluster(1CL)</code> コマンドを <code>show</code> サブコマンドとともに使用します。	35 ページの「クラスタ構成を表示する」
グローバルマウントポイントの確認またはクラスタ構成の検証	グローバルクラスタの場合は、 <code>cluster(1CL)</code> クラスタ(1CL) コマンドを <code>check</code> サブコマンドとともに使用します。 ゾーンクラスタの場合は、 <code>clzonecluster(1CL) verify</code> コマンドを使用します。	45 ページの「基本的なクラスタ構成を検証する」
Sun Cluster のコマンドログの内容の参照	<code>/var/cluster/logs/commandlog</code> ファイルを確認します。	49 ページの「Sun Cluster のコマンドログの内容を表示する」
Sun Cluster のシステムメッセージの参照	<code>/var/adm/messages</code> ファイルを確認します。	『Solaris のシステム管理(上級編)』の「システムメッセージの表示」
Solstice DiskSuite の状態の監視	<code>metastat</code> コマンドを使用します。	Solaris ボリュームマネージャーのマニュアル

表 1-2 Sun Cluster 3.2 管理ツール (続き)

作業	ツール	参照先
Solaris ボリュームマネージャの状態を監視する (Solaris 9 または Solaris 10 が動作している場合)	metastat コマンドを使用します。	『Solaris ボリュームマネージャの管理』

▼ クラスタに遠隔ログインする

Cluster Control Panel (CCP) には、`cconsole`、`crlogin`、`cssh`、および `ctelnet` ツール用の起動パッドが用意されています。これらのツールはすべて、指定した一連のノードとの多重ウィンドウ接続を起動するものです。共通ウィンドウへの入力、これら各ホストウィンドウに送信されます。その結果、クラスタのすべてのノード上でコマンドを同時に実行できます。共通ウィンドウへの入力はホストウィンドウすべてに送信されるので、クラスタのすべてのノード上でコマンドを同時に実行できます。

`cconsole`、`crlogin`、`cssh`、`ctelnet` セッションは、コマンド行から開始することもできます。

デフォルトでは、`cconsole` ユーティリティーは `telnet` 接続を使用してノードコンソールに接続します。代わりにコンソールへの Secure Shell 接続を確立するには、`cconsole` ウィンドウの「オプション (Options)」メニューの「SSH の使用 (Use SSH)」チェックボックスをオンにします。または、`ccp` または `cconsole` コマンドを実行するときに `-s` オプションを指定します。

[ccp\(1M\)](#) と [cconsole\(1M\)](#) のマニュアルページを参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

始める前に クラスタコントロールパネル (CCP) を起動する前に、次の条件を満たしていることを確認します。

- `SUNWccon` パッケージを管理コンソール上にインストールします。
- 管理コンソールの `PATH` 変数に、Sun Cluster ツールのディレクトリ `/opt/SUNWcluster/bin` と `/usr/cluster/bin` が含まれることを確認します。ツールのディレクトリには、`$CLUSTER_HOME` 環境変数を設定することで別の場所を指定できます。

- 端末集配信装置を使用している場合は、clusters ファイル、serialports ファイル、nsswitch.conf ファイルを構成します。これらのファイルは、/etc 内ファイルまたは NIS/NIS+ データベースのどちらでもかまいません。詳細は、clusters(4) と serialports(4) のマニュアルページを参照してください。
- 1 Sun Enterprise 10000 サーバー プラットフォームを使用している場合は、システム サービスプロセッサ (SSP) にログインします。
 - a. netcon コマンドを使用して接続する。
 - b. 接続が完了したなら、Shift+@ キーを入力してコンソールのロックを解除し、書き込み権を取得します。
 - 2 管理コンソールから CCP 起動パッドを起動します。


```
phys-schost# ccp clustername
```

 CCP 起動パッドが表示されます。
 - 3 クラスタとの遠隔セッションを開始するには、CCP 起動パッドの cconsole アイコン、crlogin アイコン、cssh アイコン、または ctelnet アイコンをクリックします。

▼ クラスタコンソールに安全に接続する

クラスタノードのコンソールへの Secure Shell 接続を確立するには、次の手順を実行します。

始める前に 端末集配信装置を使用している場合は、clusters ファイル、serialports ファイル、nsswitch.conf ファイルを構成します。これらのファイルは、/etc 内ファイルまたは NIS/NIS+ データベースのどちらでもかまいません。

注 serialports ファイルで、各コンソールアクセスデバイスへの安全な接続に使用するポート番号を割り当てます。Secure Shell 接続のデフォルトのポート番号は 22 です。

詳細は、clusters(4) と serialports(4) のマニュアルページを参照してください。

- 1 管理コンソールでスーパーユーザーになります。
- 2 保護されたモードで cconsole ユーティリティを起動します。


```
# cconsole -s [-l username] [-p ssh-port]
```

 -s Secure Shell 接続を有効にします。

- l *username* 遠隔接続のユーザー名を指定します。-l オプションが指定されていない場合、cconsole ユーティリティーを起動したユーザー名が使用されます。
- p *ssh-port* 使用する Secure Shell ポート番号を指定します。-p オプションが指定されていない場合、デフォルトのポート番号 22 が安全な接続に使用されます。

▼ クラスタ構成ユーティリティーにアクセスする

clsetup ユーティリティーを使用すると、グローバルクラスタの定足数、リソースグループ、クラスタトランスポート、プライベートホスト名、デバイスグループ、新しいノードのオプションを対話形式で構成できます。clzonecluster ユーティリティーは、同様な構成作業をゾーンクラスタに対して実行します。詳細は、clsetup(1CL) と clzonecluster(1CL) のマニュアルページを参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 グローバルクラスタのアクティブなメンバーノード上でスーパーユーザーになります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 2 構成ユーティリティーを起動します。

```
phys-schost# clsetup
```

- グローバルクラスタの場合は、clsetup コマンドでユーティリティーを起動します。

```
phys-schost# clsetup
```

メインメニューが表示されます。

- ゾーンクラスタの場合は、clzonecluster コマンドでユーティリティーを起動します。この例のゾーンクラスタは *sczone* です。

```
phys-schost# clzonecluster configure sczone
```

ユーティリティーで実行可能な操作は、次のオプションで確認できます。

```
clzc:sczone> ?
```

- 3 使用する構成をメニューから選択します。画面に表示される指示に従って、作業を完了します。詳細は、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「[ゾーンクラスタの設定](#)」に記述されている手順を参照してください。

参照 詳細は、`clsetup` または `clzonecluster` のオンラインヘルプを参照してください。

▼ Sun Cluster のパッチ情報を表示する

この手順を実行するためにスーパーユーザーとしてログインする必要はありません。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

● Sun Cluster のパッチ情報の表示:

```
phys-schost# showrev -p
```

&fmv1 更新リリースは、製品のメインパッチ番号と更新バージョンによって識別されます。

例 1-1 Sun Cluster のパッチ情報の表示

次に、パッチ 110648-05 についての情報を表示した例を示します。

```
phys-schost# showrev -p | grep 110648
Patch: 110648-05 Obsoletes: Requires: Incompatibles: Packages:
```

▼ Sun Cluster のリリース情報とバージョン情報を表示する

この手順を実行するためにスーパーユーザーとしてログインする必要はありません。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- **Sun Cluster のリリース情報およびバージョン情報の表示:**

```
phys-schost# clnode show-rev -v -node
```

このコマンドは、すべての Sun Cluster パッケージについて Sun Cluster のリリース番号とバージョン文字列を表示します。

例 1-2 Sun Cluster のリリース情報およびバージョン情報の表示

次に、すべてのパッケージのクラスタのリリース情報とバージョン情報の例を示します。

```
phys-schost# clnode show-rev
3.2
```

```
phys-schost# clnode show-rev -v
Sun Cluster 3.2 for Solaris 9 sparc

SUNWscr:      3.2.0,REV=2006.02.17.18.11
SUNWscu:      3.2.0,REV=2006.02.17.18.11
SUNWsczu:     3.2.0,REV=2006.02.17.18.11
SUNWscsck:    3.2.0,REV=2006.02.17.18.11
SUNWscnm:     3.2.0,REV=2006.02.17.18.11
SUNWscdev:    3.2.0,REV=2006.02.17.18.11
SUNWscgds:    3.2.0,REV=2006.02.17.18.11
SUNWscman:    3.2.0,REV=2005.10.18.08.42
SUNWscsal:    3.2.0,REV=2006.02.17.18.11
SUNWscsam:    3.2.0,REV=2006.02.17.18.11
SUNWscvm:     3.2.0,REV=2006.02.17.18.11
SUNWmdm:      3.2.0,REV=2006.02.17.18.11
SUNWscmasa:   3.2.0,REV=2006.02.17.18.11
SUNWscmautil: 3.2.0,REV=2006.02.17.18.11
SUNWscmautilr: 3.2.0,REV=2006.02.17.18.11
SUNWjfreechart: 3.2.0,REV=2006.02.17.18.11
SUNWscva:     3.2.0,REV=2006.02.17.18.11
SUNWscspm:    3.2.0,REV=2006.02.17.18.11
SUNWscspmu:   3.2.0,REV=2006.02.17.18.11
SUNWscspsmr:  3.2.0,REV=2006.02.17.18.11
SUNWscderby:  3.2.0,REV=2006.02.17.18.11
SUNWsc telemetry: 3.2.0,REV=2006.02.17.18.11
SUNWscdsm:    3.2.0,REV=2006.02.17.18.11
SUNWcsc:      3.2.0,REV=2006.02.21.10.16
```

```
SUNWcscspm: 3.2.0,REV=2006.02.21.10.16
SUNWcscspmu: 3.2.0,REV=2006.02.21.10.16
SUNWdsc: 3.2.0,REV=2006.02.21.10.09
SUNWdscspm: 3.2.0,REV=2006.02.21.10.09
SUNWdscspmu: 3.2.0,REV=2006.02.21.10.09
SUNWesc: 3.2.0,REV=2006.02.21.10.11
SUNWescspm: 3.2.0,REV=2006.02.21.10.11
SUNWescspmu: 3.2.0,REV=2006.02.21.10.11
SUNWfsc: 3.2.0,REV=2006.02.21.10.06
SUNWfscspm: 3.2.0,REV=2006.02.21.10.06
SUNWfscspmu: 3.2.0,REV=2006.02.21.10.06
SUNWhsc: 3.2.0,REV=2006.02.21.10.20
SUNWhscspm: 3.2.0,REV=2006.02.21.10.20
SUNWhscspmu: 3.2.0,REV=2006.02.21.10.20
SUNWjsc: 3.2.0,REV=2006.02.21.10.22
SUNWjscman: 3.2.0,REV=2006.02.21.10.22
SUNWjscspm: 3.2.0,REV=2006.02.21.10.22
SUNWjscspmu: 3.2.0,REV=2006.02.21.10.22
SUNWksc: 3.2.0,REV=2006.02.21.10.14
SUNWkscspm: 3.2.0,REV=2006.02.21.10.14
SUNWkscspmu: 3.2.0,REV=2006.02.21.10.14
```

▼ 構成されているリソースタイプ、リソースグループ、リソースを表示する

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細は、[第 13 章「グラフィカルユーザーインターフェースによる Sun Cluster の管理」](#)または Sun Cluster Manager のオンラインヘルプを参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

始める前に スーパーユーザー以外のユーザーがこのサブコマンドを使用するには、RBAC の承認 solaris.cluster.read が必要です。

- クラスタで構成されているリソースタイプ、リソースグループ、リソースを表示します。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
phys-schost# cluster show -t resource, resourcetype, resourcegroup
```

個別のリソース、リソースグループ、およびリソースタイプの詳細については、次のいずれかのコマンドとともに show サブコマンドを使用します。

- 資源
- リソースグループ (resource group)
- resourcetype

例 1-3 構成されているリソースタイプ、リソースグループ、リソースの表示

次に、クラスタ schost に対して構成されているリソースタイプ (RT Name)、リソースグループ (RG Name)、リソース (RS Name) の例を示します。

```
phys-schost# cluster show -t resource, resourcetype, resourcegroup
```

```
=== Registered Resource Types ===
```

```
Resource Type:                SUNW.qfs
  RT_description:              SAM-QFS Agent on SunCluster
  RT_version:                  3.1
  API_version:                 3
  RT_basedir:                  /opt/SUNWsamfs/sc/bin
  Single_instance:            False
  Proxy:                       False
  Init_nodes:                  All potential masters
  Installed_nodes:             <All>
  Failover:                    True
  Pkglist:                     <NULL>
  RT_system:                   False
```

```
=== Resource Groups and Resources ===
```

```
Resource Group:                qfs-rg
  RG_description:              <NULL>
  RG_mode:                     Failover
  RG_state:                    Managed
  Failback:                    False
  Nodelist:                    phys-schost-2 phys-schost-1
```

```
--- Resources for Group qfs-rg ---
```

```
Resource:                      qfs-res
```

```

Type:                               SUNW.qfs
Type_version:                        3.1
Group:                                qfs-rg
R_description:
Resource_project_name:                default
Enabled{phys-schost-2}:               True
Enabled{phys-schost-1}:               True
Monitored{phys-schost-2}:             True
Monitored{phys-schost-1}:             True

```

▼ クラスタコンポーネントの状態を確認する

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

注 - ゾーンクラスタの状態は、`cluster status` コマンドでも表示できます。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

始める前に スーパーユーザー以外のユーザーが `status` サブコマンドを使用するには、RBAC の承認 `solaris.cluster.read` が必要です。

- クラスタコンポーネントの状態を確認します。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
phys-schost# cluster status
```

例 1-4 クラスタコンポーネントの状態確認

次に、`cluster(1CL) status` で返されるクラスタコンポーネントの状態情報の例を示します。

```

phys-schost# cluster status
=== Cluster Nodes ===

--- Node Status ---

```

```

Node Name                               Status
-----                               -
phys-schost-1                           Online
phys-schost-2                           Online

```

=== Cluster Transport Paths ===

```

Endpoint1                               Endpoint2                               Status
-----                               -
phys-schost-1:qfe1                      phys-schost-4:qfe1                      Path online
phys-schost-1:hme1                      phys-schost-4:hme1                      Path online

```

=== Cluster Quorum ===

--- Quorum Votes Summary ---

```

          Needed   Present   Possible
          -
          3         3         4

```

--- Quorum Votes by Node ---

```

Node Name      Present      Possible      Status
-----
phys-schost-1  1             1             Online
phys-schost-2  1             1             Online

```

--- Quorum Votes by Device ---

```

Device Name      Present      Possible      Status
-----
/dev/did/rdisk/d2s2  1           1             Online
/dev/did/rdisk/d8s2  0           1             Offline

```

=== Cluster Device Groups ===

--- Device Group Status ---

```

Device Group Name  Primary      Secondary      Status
-----
schost-2           phys-schost-2  -             Degraded

```

--- Spare, Inactive, and In Transition Nodes ---

Device Group Name	Spare Nodes	Inactive Nodes	In Transition Nodes
schost-2	-	-	-

=== Cluster Resource Groups ===

Group Name	Node Name	Suspended	Status
test-rg	phys-schost-1	No	Offline
	phys-schost-2	No	Online
test-rg	phys-schost-1	No	Offline
	phys-schost-2	No	Error--stop failed
test-rg	phys-schost-1	No	Online
	phys-schost-2	No	Online

=== Cluster Resources ===

Resource Name	Node Name	Status	Message
test_1	phys-schost-1	Offline	Offline
	phys-schost-2	Online	Online
test_1	phys-schost-1	Offline	Offline
	phys-schost-2	Stop failed	Faulted
test_1	phys-schost-1	Online	Online
	phys-schost-2	Online	Online

Device Instance	Node	Status
/dev/did/rdisk/d2	phys-schost-1	Ok
/dev/did/rdisk/d3	phys-schost-1	Ok
	phys-schost-2	Ok
/dev/did/rdisk/d4	phys-schost-1	Ok
	phys-schost-2	Ok
/dev/did/rdisk/d6	phys-schost-2	Ok

```

=== Zone Clusters ===

--- Zone Cluster Status ---

Name          Node Name    Zone HostName  Status   Zone Status
-----
sczone        schost-1     sczone-1       Online   Running
              schost-2     sczone-2       Online   Running

```

▼ パブリックネットワークの状態を確認する

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

IP ネットワークのマルチパスグループのステータスを確認するには、`clnode(1CL)` コマンドと `status` サブコマンドとともに使用します。

始める前に スーパーユーザー以外のユーザーがこのサブコマンドを使用するには、RBAC の承認 `solaris.cluster.read` が必要です。

- クラスタコンポーネントの状態を確認します。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
phys-schost# clnode status -m
```

例 1-5 パブリックネットワークの状態を調べる

次に、`clnode status` コマンドで戻されるクラスタコンポーネントの状態情報の例を示します。

```

% clnode status -m
--- Node IPMP Group Status ---

Node Name          Group Name    Status   Adapter   Status

```

```

-----
phys-schost-1    test-rg    Online    qfel      Online
phys-schost-2    test-rg    Online    qfel      Online

```

▼ クラスタ構成を表示する

この手順は Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

始める前に スーパーユーザー以外のユーザーが status サブコマンドを使用するには、RBAC の承認 solaris.cluster.read が必要です。

- グローバルクラスタまたはゾーンクラスタの構成を表示します。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
% cluster show
```

cluster show コマンドをグローバルクラスタの投票ノードから実行すると、そのクラスタに関する詳細な構成情報が表示されます。また、ゾーンクラスタが構成されていれば、ゾーンクラスタの情報も表示されます。

ゾーンクラスタのみに関する構成情報を表示するには、clzonecluster show コマンドも使用できます。ゾーンクラスタのプロパティには、ゾーンクラスタ名、IP タイプ、自動起動、ゾーンパスなどがあります。show サブコマンドは、ゾーンクラスタの内部で実行され、そのゾーンクラスタのみが対象になります。ゾーンクラスタノードから clzonecluster show コマンドを実行すると、そのゾーンクラスタから認識可能なオブジェクトのみの状態が得られます。

cluster コマンドでより多くの情報を表示するには、冗長オプションを使用します。詳細は、cluster(1CL) のマニュアルページを参照してください。clzonecluster の詳細は、clzonecluster(1CL) のマニュアルページを参照してください。

例 1-6 グローバルクラスタの構成を表示する

次に、グローバルクラスタの構成情報の例を示します。ゾーンクラスタが構成されている場合は、ゾーンクラスタの情報も表示されます。

phys-schost# **cluster show**

=== Cluster ===

```
Cluster Name:                cluster-1
installmode:                 disabled
heartbeat_timeout:          10000
heartbeat_quantum:          1000
private_netaddr:             172.16.0.0
private_netmask:             255.255.248.0
max_nodes:                   64
max_privatenets:             10
global_fencing:              Unknown
Node List:                   phys-schost-1
Node Zones:                   phys_schost-2:za
```

=== Host Access Control ===

```
Cluster name:                clustser-1
Allowed hosts:                phys-schost-1, phys-schost-2:za
Authentication Protocol:      sys
```

=== Cluster Nodes ===

```
Node Name:                   phys-schost-1
Node ID:                      1
Type:                         cluster
Enabled:                      yes
privatehostname:              clusternode1-priv
reboot_on_path_failure:       disabled
globalzoneshares:             3
defaultpsetmin:               1
quorum_vote:                  1
quorum_defaultvote:           1
quorum_resv_key:              0x43CB1E1800000001
Transport Adapter List:       qfe3, hme0
```

--- Transport Adapters for phys-schost-1 ---

```
Transport Adapter:           qfe3
Adapter State:                Enabled
Adapter Transport Type:       dlpi
Adapter Property(device_name): qfe
Adapter Property(device_instance): 3
Adapter Property(lazy_free):  1
Adapter Property(dlpi_heartbeat_timeout): 10000
Adapter Property(dlpi_heartbeat_quantum): 1000
```

```

Adapter Property(nw_bandwidth):      80
Adapter Property(bandwidth):         10
Adapter Property(ip_address):        172.16.1.1
Adapter Property(netmask):           255.255.255.128
Adapter Port Names:                  0
Adapter Port State(0):               Enabled

```

```

Transport Adapter:                   hme0
Adapter State:                       Enabled
Adapter Transport Type:              dlpi
Adapter Property(device_name):       hme
Adapter Property(device_instance):   0
Adapter Property(lazy_free):         0
Adapter Property(dlpi_heartbeat_timeout): 10000
Adapter Property(dlpi_heartbeat_quantum): 1000
Adapter Property(nw_bandwidth):      80
Adapter Property(bandwidth):         10
Adapter Property(ip_address):        172.16.0.129
Adapter Property(netmask):           255.255.255.128
Adapter Port Names:                  0
Adapter Port State(0):               Enabled

```

--- SNMP MIB Configuration on phys-schost-1 ---

```

SNMP MIB Name:                       Event
State:                               Disabled
Protocol:                             SNMPv2

```

--- SNMP Host Configuration on phys-schost-1 ---

--- SNMP User Configuration on phys-schost-1 ---

```

SNMP User Name:                      foo
Authentication Protocol:              MD5
Default User:                         No

```

```

Node Name:                            phys-schost-2:za
Node ID:                              2
Type:                                  cluster
Enabled:                               yes
privatehostname:                      clusternode2-priv
reboot_on_path_failure:               disabled
globalzonestores:                    1
defaultpsetmin:                       2
quorum_vote:                          1
quorum_defaultvote:                   1
quorum_resv_key:                      0x43CB1E1800000002

```

Transport Adapter List: hme0, qfe3

--- Transport Adapters for phys-schost-2 ---

Transport Adapter: hme0
 Adapter State: Enabled
 Adapter Transport Type: dlpi
 Adapter Property(device_name): hme
 Adapter Property(device_instance): 0
 Adapter Property(lazy_free): 0
 Adapter Property(dlpi_heartbeat_timeout): 10000
 Adapter Property(dlpi_heartbeat_quantum): 1000
 Adapter Property(nw_bandwidth): 80
 Adapter Property(bandwidth): 10
 Adapter Property(ip_address): 172.16.0.130
 Adapter Property(netmask): 255.255.255.128
 Adapter Port Names: 0
 Adapter Port State(0): Enabled

Transport Adapter: qfe3
 Adapter State: Enabled
 Adapter Transport Type: dlpi
 Adapter Property(device_name): qfe
 Adapter Property(device_instance): 3
 Adapter Property(lazy_free): 1
 Adapter Property(dlpi_heartbeat_timeout): 10000
 Adapter Property(dlpi_heartbeat_quantum): 1000
 Adapter Property(nw_bandwidth): 80
 Adapter Property(bandwidth): 10
 Adapter Property(ip_address): 172.16.1.2
 Adapter Property(netmask): 255.255.255.128
 Adapter Port Names: 0
 Adapter Port State(0): Enabled

--- SNMP MIB Configuration on phys-schost-2 ---

SNMP MIB Name: Event
 State: Disabled
 Protocol: SNMPv2

--- SNMP Host Configuration on phys-schost-2 ---

--- SNMP User Configuration on phys-schost-2 ---

=== Transport Cables ===

Transport Cable: phys-schost-1:qfe3,switch2@1

```

Cable Endpoint1:          phys-schost-1:qfe3
Cable Endpoint2:          switch2@1
Cable State:              Enabled

Transport Cable:          phys-schost-1:hme0,switch1@1
  Cable Endpoint1:        phys-schost-1:hme0
  Cable Endpoint2:        switch1@1
  Cable State:            Enabled

Transport Cable:          phys-schost-2:hme0,switch1@2
  Cable Endpoint1:        phys-schost-2:hme0
  Cable Endpoint2:        switch1@2
  Cable State:            Enabled

Transport Cable:          phys-schost-2:qfe3,switch2@2
  Cable Endpoint1:        phys-schost-2:qfe3
  Cable Endpoint2:        switch2@2
  Cable State:            Enabled

```

=== Transport Switches ===

```

Transport Switch:         switch2
  Switch State:           Enabled
  Switch Type:            switch
  Switch Port Names:      1 2
  Switch Port State(1):   Enabled
  Switch Port State(2):   Enabled

```

```

Transport Switch:         switch1
  Switch State:           Enabled
  Switch Type:            switch
  Switch Port Names:      1 2
  Switch Port State(1):   Enabled
  Switch Port State(2):   Enabled

```

=== Quorum Devices ===

```

Quorum Device Name:      d3
  Enabled:                yes
  Votes:                  1
  Global Name:            /dev/did/rdisk/d3s2
  Type:                   scsi
  Access Mode:            scsi2
  Hosts (enabled):        phys-schost-1, phys-schost-2

```

```

Quorum Device Name:      qs1

```

```

Enabled:                yes
Votes:                  1
Global Name:            qs1
Type:                   quorum_server
Hosts (enabled):        phys-schost-1, phys-schost-2
Quorum Server Host:     10.11.114.83
Port:                   9000
    
```

=== Device Groups ===

```

Device Group Name:      testdg3
Type:                   SVM
failback:               no
Node List:               phys-schost-1, phys-schost-2
preferenced:            yes
numsecondaries:         1
diskset name:           testdg3
    
```

=== Registered Resource Types ===

```

Resource Type:          SUNW.LogicalHostname:2
RT_description:         Logical Hostname Resource Type
RT_version:             2
API_version:            2
RT_basedir:             /usr/cluster/lib/rgm/rt/hafoip
Single_instance:        False
Proxy:                  False
Init_nodes:             All potential masters
Installed_nodes:        <All>
Failover:                True
Pkglist:                SUNWscu
RT_system:              True
    
```

```

Resource Type:          SUNW.SharedAddress:2
RT_description:         HA Shared Address Resource Type
RT_version:             2
API_version:            2
RT_basedir:             /usr/cluster/lib/rgm/rt/hascip
Single_instance:        False
Proxy:                  False
Init_nodes:             <Unknown>
Installed_nodes:        <All>
Failover:                True
Pkglist:                SUNWscu
RT_system:              True
    
```

```
Resource Type: SUNW.HAStoragePlus:4
  RT_description: HA Storage Plus
  RT_version: 4
  API_version: 2
  RT_basedir: /usr/cluster/lib/rgm/rt/hastorageplus
  Single_instance: False
  Proxy: False
  Init_nodes: All potential masters
  Installed_nodes: <All>
  Failover: False
  Pkglist: SUNWscu
  RT_system: False
```

```
Resource Type: SUNW.haderby
  RT_description: haderby server for Sun Cluster
  RT_version: 1
  API_version: 7
  RT_basedir: /usr/cluster/lib/rgm/rt/haderby
  Single_instance: False
  Proxy: False
  Init_nodes: All potential masters
  Installed_nodes: <All>
  Failover: False
  Pkglist: SUNWscderby
  RT_system: False
```

```
Resource Type: SUNW.sctelemetry
  RT_description: sctelemetry service for Sun Cluster
  RT_version: 1
  API_version: 7
  RT_basedir: /usr/cluster/lib/rgm/rt/sctelemetry
  Single_instance: True
  Proxy: False
  Init_nodes: All potential masters
  Installed_nodes: <All>
  Failover: False
  Pkglist: SUNWsc telemetry
  RT_system: False
```

=== Resource Groups and Resources ===

```
Resource Group: HA_RG
  RG_description: <Null>
  RG_mode: Failover
  RG_state: Managed
  Failback: False
  Nodelist: phys-schost-1 phys-schost-2
```

--- Resources for Group HA_RG ---

```
Resource: HA_R
  Type: SUNW.HAStoragePlus:4
  Type_version: 4
  Group: HA_RG
  R_description:
  Resource_project_name: SCSLM_HA_RG
  Enabled{phys-schost-1}: True
  Enabled{phys-schost-2}: True
  Monitored{phys-schost-1}: True
  Monitored{phys-schost-2}: True
```

```
Resource Group: cl-db-rg
  RG_description: <Null>
  RG_mode: Failover
  RG_state: Managed
  Failback: False
  Nodelist: phys-schost-1 phys-schost-2
```

--- Resources for Group cl-db-rg ---

```
Resource: cl-db-rs
  Type: SUNW.haderby
  Type_version: 1
  Group: cl-db-rg
  R_description:
  Resource_project_name: default
  Enabled{phys-schost-1}: True
  Enabled{phys-schost-2}: True
  Monitored{phys-schost-1}: True
  Monitored{phys-schost-2}: True
```

```
Resource Group: cl-tlmtry-rg
  RG_description: <Null>
  RG_mode: Scalable
  RG_state: Managed
  Failback: False
  Nodelist: phys-schost-1 phys-schost-2
```

--- Resources for Group cl-tlmtry-rg ---

```
Resource: cl-tlmtry-rs
  Type: SUNW.sctelemetry
  Type_version: 1
  Group: cl-tlmtry-rg
```

```

R_description:
Resource_project_name:                default
Enabled{phys-schost-1}:                True
Enabled{phys-schost-2}:                True
Monitored{phys-schost-1}:              True
Monitored{phys-schost-2}:              True

```

=== DID Device Instances ===

```

DID Device Name:                       /dev/did/rdisk/d1
Full Device Path:                       phys-schost-1:/dev/rdsk/c0t2d0
Replication:                             none
default_fencing:                         global

```

```

DID Device Name:                       /dev/did/rdisk/d2
Full Device Path:                       phys-schost-1:/dev/rdsk/c1t0d0
Replication:                             none
default_fencing:                         global

```

```

DID Device Name:                       /dev/did/rdisk/d3
Full Device Path:                       phys-schost-2:/dev/rdsk/c2t1d0
Full Device Path:                       phys-schost-1:/dev/rdsk/c2t1d0
Replication:                             none
default_fencing:                         global

```

```

DID Device Name:                       /dev/did/rdisk/d4
Full Device Path:                       phys-schost-2:/dev/rdsk/c2t2d0
Full Device Path:                       phys-schost-1:/dev/rdsk/c2t2d0
Replication:                             none
default_fencing:                         global

```

```

DID Device Name:                       /dev/did/rdisk/d5
Full Device Path:                       phys-schost-2:/dev/rdsk/c0t2d0
Replication:                             none
default_fencing:                         global

```

```

DID Device Name:                       /dev/did/rdisk/d6
Full Device Path:                       phys-schost-2:/dev/rdsk/c1t0d0
Replication:                             none
default_fencing:                         global

```

=== NAS Devices ===

```

Nas Device:                             nas_filer1
Type:                                     netapp
User ID:                                  root

```

```
Nas Device:          nas2
Type:               netapp
User ID:           llai
```

例 1-7 ゾーンクラスタの構成を表示する

次に、ゾーンクラスタ構成のプロパティの例を示します。

```
% clzonecluster show
=== Zone Clusters ===

Zone Cluster Name:          sczone
  zonename:                 sczone
  zonepath:                 /zones/sczone
  autoboot:                 TRUE
  ip-type:                  shared
  enable_priv_net:         TRUE

--- Solaris Resources for sczone ---

Resource Name:             net
  address:                 172.16.0.1
  physical:                auto

Resource Name:             net
  address:                 172.16.0.2
  physical:                auto

Resource Name:             fs
  dir:                    /gz/db_qfs/CrsHome
  special:                CrsHome
  raw:
  type:                   samfs
  options:                []

Resource Name:             fs
  dir:                    /gz/db_qfs/CrsData
  special:                CrsData
  raw:
  type:                   samfs
  options:                []

Resource Name:             fs
  dir:                    /gz/db_qfs/OraHome
  special:                OraHome
```

```

raw:
type:                                samfs
options:                              []

Resource Name:                        fs
dir:                                  /gz/db_qfs/OraData
special:                              OraData
raw:
type:                                samfs
options:                              []

--- Zone Cluster Nodes for sczone ---

Node Name:                            sczone-1
physical-host:                        sczone-1
hostname:                              lzzone-1

Node Name:                            sczone-2
physical-host:                        sczone-2
hostname:                              lzzone-2

```

▼ 基本的なクラスタ構成を検証する

`cluster(1CL)` コマンドは、`check` コマンドを使用してグローバルクラスタが正しく機能するために必要な基本構成を検証します。チェックにエラーがない場合、`cluster check` はシェルプロンプトに戻ります。チェックにエラーがある場合、`cluster check` が指定したディレクトリかデフォルトの出力ディレクトリにレポートを生成します。`cluster check` を複数のノードに対して実行すると、`cluster check` は、ノードごとのレポートと複数ノードチェックのレポートを生成します。`cluster list-checks` コマンドを使用して、使用可能なすべてのクラスタチェックの一覧を表示させることもできます。

`cluster check` コマンドを詳細モードで `-v` フラグを使用して実行して、進捗情報を表示することができます。

注 - `cluster check` は、デバイス、ボリューム管理コンポーネント、または Sun Cluster 構成を変更するような管理手順を行った後に実行してください。

`clzonecluster(1CL)` コマンドをグローバルクラスタの投票ノードで実行すると、ゾーンクラスタが正しく機能するために必要な構成を検証する一連のチェックが実行されます。すべてのチェックでエラーがなかった場合、`clzonecluster verify`

はシェルプロンプトに戻ります (その場合は、ゾーンクラスタを安全にインストールできません)。エラーがあった場合は、エラーがあったグローバルクラスタノードに関して `clzonecluster verify` から報告があります。 `clzonecluster verify` を複数のノードに対して実行すると、ノードごとのレポートと、複数ノードチェックのレポートが生成されます。ゾーンクラスタ内では、`verify` サブコマンドは指定できません。

- 1 グローバルクラスタのアクティブなメンバーノード上でスーパーユーザーになります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
`phys-schost# su`
- 2 クラスタ構成を検証します。
 - グローバルクラスタの構成を検証します。
`phys-schost# cluster check`
 - ゾーンクラスタの構成を検証して、ゾーンクラスタがインストール可能かどうかを確認します。
`phys-schost# clzonecluster verify zoneclustername`

例 1-8 グローバルクラスタ構成の検証 (エラーがない場合)

次の例は、`cluster check` を詳細モードで `phys-schost-1` および `phys-schost-2` ノードに対して実行し、エラーが発見されなかった場合を示しています。

```
phys-schost# cluster check -v -h phys-schost-1,  
phys-schost-2
```

```
cluster check: Requesting explorer data and node report from phys-schost-1.  
cluster check: Requesting explorer data and node report from phys-schost-2.  
cluster check: phys-schost-1: Explorer finished.  
cluster check: phys-schost-1: Starting single-node checks.  
cluster check: phys-schost-1: Single-node checks finished.  
cluster check: phys-schost-2: Explorer finished.  
cluster check: phys-schost-2: Starting single-node checks.  
cluster check: phys-schost-2: Single-node checks finished.  
cluster check: Starting multi-node checks.  
cluster check: Multi-node checks finished  
#
```

例 1-9 グローバルクラスタ構成の検証(エラーがある場合)

次の例は、suncluster という名前のクラスタのノード phys-schost-2 にマウントポイント /global/phys-schost-1 がないことを示しています。レポートは、出力ディレクトリ /var/cluster/logs/cluster_check/<timestamp> に作成されます。

```
phys-schost# cluster check -v -h phys-schost-1,
phys-schost-2 -o
/var/cluster/logs/cluster_check/Dec5/

cluster check: Requesting explorer data and node report from phys-schost-1.
cluster check: Requesting explorer data and node report from phys-schost-2.
cluster check: phys-schost-1: Explorer finished.
cluster check: phys-schost-1: Starting single-node checks.
cluster check: phys-schost-1: Single-node checks finished.
cluster check: phys-schost-2: Explorer finished.
cluster check: phys-schost-2: Starting single-node checks.
cluster check: phys-schost-2: Single-node checks finished.
cluster check: Starting multi-node checks.
cluster check: Multi-node checks finished.
cluster check: One or more checks failed.
cluster check: The greatest severity of all check failures was 3 (HIGH).
cluster check: Reports are in /var/cluster/logs/cluster_check/<Dec5>.
#
# cat /var/cluster/logs/cluster_check/Dec5/cluster_check-results.suncluster.txt
...
=====
= ANALYSIS DETAILS =
=====
-----
CHECK ID : 3065
SEVERITY : HIGH
FAILURE  : Global filesystem /etc/vfstab entries are not consistent across
all Sun Cluster 3.x nodes.
ANALYSIS : The global filesystem /etc/vfstab entries are not consistent across
all nodes in this cluster.
Analysis indicates:
FileSystem '/global/phys-schost-1' is on 'phys-schost-1' but missing from 'phys-schost-2'.
RECOMMEND: Ensure each node has the correct /etc/vfstab entry for the
filesystem(s) in question.
...
#
```

▼ グローバルマウントポイントを確認する

`cluster(1CL)` コマンドには、クラスタファイルシステムとそのグローバルマウントポイントに構成エラーがないか `/etc/vfstab` ファイルを検査するチェックが含まれています。

注 - `cluster check` は、デバイスやボリューム管理コンポーネントに影響を及ぼすような変更をクラスタ構成に加えたあとで実行してください。

- 1 グローバルクラスタのアクティブなメンバーノード上でスーパーユーザーになります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
% su
```

- 2 グローバルクラスタ構成を検証します。

```
phys-schost# cluster check
```

例 1-10 グローバルマウントポイントの確認

次の例は、`suncluster` という名前のクラスタのノード `phys-schost-2` にマウントポイント `/global/schost-1` がないことを示しています。レポートは、出力ディレクトリ、`/var/cluster/logs/cluster_check/<timestamp>/` に送信されています。

```
phys-schost# cluster check -v1 -h phys-schost-1,phys-schost-2 -o /var/cluster//logs/cluster_check/Dec5/
```

```
cluster check: Requesting explorer data and node report from phys-schost-1.
cluster check: Requesting explorer data and node report from phys-schost-2.
cluster check: phys-schost-1: Explorer finished.
cluster check: phys-schost-1: Starting single-node checks.
cluster check: phys-schost-1: Single-node checks finished.
cluster check: phys-schost-2: Explorer finished.
cluster check: phys-schost-2: Starting single-node checks.
cluster check: phys-schost-2: Single-node checks finished.
cluster check: Starting multi-node checks.
cluster check: Multi-node checks finished.
cluster check: One or more checks failed.
cluster check: The greatest severity of all check failures was 3 (HIGH).
cluster check: Reports are in /var/cluster/logs/cluster_check/Dec5.
#
# cat /var/cluster/logs/cluster_check/Dec5/cluster_check-results.suncluster.txt
```

```
...
```

```
=====
= ANALYSIS DETAILS =
=====
```

```

-----
CHECK ID : 3065
SEVERITY : HIGH
FAILURE : Global filesystem /etc/vfstab entries are not consistent across
all Sun Cluster 3.x nodes.
ANALYSIS : The global filesystem /etc/vfstab entries are not consistent across
all nodes in this cluster.
Analysis indicates:
FileSystem '/global/phys-schost-1' is on 'phys-schost-1' but missing from 'phys-schost-2'.
RECOMMEND: Ensure each node has the correct /etc/vfstab entry for the
filesystem(s) in question.
...
#
# cat /var/cluster/logs/cluster_check/Dec5/cluster_check-results.phys-schost-1.txt

...
=====
= ANALYSIS DETAILS =
=====
-----
CHECK ID : 1398
SEVERITY : HIGH
FAILURE : An unsupported server is being used as a Sun Cluster 3.x node.
ANALYSIS : This server may not been qualified to be used as a Sun Cluster 3.x node.
Only servers that have been qualified with Sun Cluster 3.x are supported as
Sun Cluster 3.x nodes.
RECOMMEND: Because the list of supported servers is always being updated, check with
your Sun Microsystems representative to get the latest information on what servers
are currently supported and only use a server that is supported with Sun Cluster 3.x.
...
#

```

▼ Sun Cluster のコマンドログの内容を表示する

`/var/cluster/logs/commandlog` ASCII テキストファイルには、クラスタ内で実行されている選択済みの Sun Cluster コマンドのレコードが含まれています。コマンドのロギングは、ユーザーがクラスタをセットアップしたときに自動的に開始され、ユーザーがクラスタをシャットダウンしたときに終了します。コマンドは、実行中およびクラスタモードでブートされたすべてのノード上でロギングされます。

クラスタの構成や現在の状態を表示するようなコマンドは、このファイルに記録されません。

次のような、クラスタの現在の状態の設定や変更を行うコマンドは、このファイルに記録されます。

- claccess
- cldevice
- cldevicegroup
- clinterconnect
- clnasdevice
- clnode
- clquorum
- clreslogicalhostname
- clresource
- clresourcegroup
- clresourcetype
- clressharedaddress
- clsetup
- clsnmp host
- clsnmpmib
- clsnmpuser
- cltelemetryattribute
- cluster
- clzonecluster
- sconfig
- scdidadm
- scdpm
- scgdevs
- scrgadm
- scsetup
- scshutdown
- scswitch

commandlog ファイル内のレコードには次の要素を含めることができます。

- 日付とタイムスタンプ
- コマンドの実行元であるホストの名前
- コマンドのプロセス ID
- コマンドを実行したユーザーのログイン名
- ユーザーが実行したコマンド (すべてのオプションとオペランドを含む)

注- すぐに特定し、シェル内でコピー、貼り付け、および実行ができるように、コマンドのオプションは commandlog ファイル内では引用符で囲まれています。

- 実行されたコマンドの終了ステータス

注 - あるコマンドが未知の結果を伴って異常終了した場合、Sun Cluster ソフトウェアは `commandlog` ファイル内には終了ステータスを「表示しません」。

`commandlog` ファイルはデフォルトでは、週に 1 回定期的にアーカイブされません。`commandlog` ファイルのアーカイブポリシーを変更するには、クラスタ内の各ノード上で `crontab` コマンドを使用します。詳細は、[crontab\(1\)](#) のマニュアルページを参照してください。

Sun Cluster ソフトウェアは任意の時点で、アーカイブ済みの `commandlog` ファイルを、クラスタノードごとに最大 8 個保持します。現在の週の `commandlog` ファイルの名前は `commandlog` です。最新の完全な週のファイルの名前は `commandlog.0` です。もっとも古い完全な週のファイルの名前は `commandlog.7` です。

- 一度に 1 つの画面で、現在の週の `commandlog` ファイルの内容を表示します。

```
phys-schost# more /var/cluster/logs/commandlog
```

例 1-11 Sun Cluster のコマンドログの内容の表示

次の例に、`more` コマンドにより表示される `commandlog` ファイルの内容を示します。

```
more -lines10 /var/cluster/logs/commandlog
11/11/2006 09:42:51 phys-schost-1 5222 root START - clsetup
11/11/2006 09:43:36 phys-schost-1 5758 root START - clrg add "app-sa-1"
11/11/2006 09:43:36 phys-schost-1 5758 root END 0
11/11/2006 09:43:36 phys-schost-1 5760 root START - clrg set -y
"RG_description=Department Shared Address RG" "app-sa-1"
11/11/2006 09:43:37 phys-schost-1 5760 root END 0
11/11/2006 09:44:15 phys-schost-1 5810 root START - clrg online "app-sa-1"
11/11/2006 09:44:15 phys-schost-1 5810 root END 0
11/11/2006 09:44:19 phys-schost-1 5222 root END -20988320
12/02/2006 14:37:21 phys-schost-1 5542 jbloggs START - clrg -c -g "app-sa-1"
-y "RG_description=Joe Bloggs Shared Address RG"
12/02/2006 14:37:22 phys-schost-1 5542 jbloggs END 0
```


Sun Cluster と RBAC

この章では、役割に基づくアクセス制御 (RBAC) について Sun Cluster に関連する範囲で説明します。次のトピックについて述べます。

- 53 ページの「RBAC の設定と Sun Cluster での使用」
- 54 ページの「Sun Cluster RBAC の権限プロファイル」
- 55 ページの「Sun Cluster 管理権限プロファイルによる RBAC 役割の作成と割り当て」
- 59 ページの「ユーザーの RBAC プロパティの変更」

RBAC の設定と Sun Cluster での使用

次の表を参考に、RBAC の設定と使用について確認するマニュアルを選んでください。RBAC を作成して、Sun Cluster ソフトウェアで使用するための手順については、この章で後述します。

作業	参照先
RBAC の詳細を調べる	『Solaris のシステム管理 (セキュリティサービス)』の第 8 章「役割と特権の使用 (概要)」
RBAC の設定、要素の管理、RBAC の使用など	『Solaris のシステム管理 (セキュリティサービス)』の第 9 章「役割によるアクセス制御の使用 (手順)」
RBAC の要素とツールの詳細を調べる	『Solaris のシステム管理 (セキュリティサービス)』の第 10 章「役割によるアクセス制御 (参照)」

Sun Cluster RBACの権限プロファイル

Sun Cluster Manager とコマンド行で実行する一部の Sun Cluster コマンドとオプションは、承認のために RBAC を使用します。RBAC の承認を必要とする Sun Cluster のコマンドとオプションは、次の承認レベルを1つ以上必要とします。Sun Cluster RBAC の権限プロファイルは、グローバルクラスタ内の投票ノードおよび非投票ノードの両方に適用されます。

<code>solaris.cluster.read</code>	一覧表示、表示、およびそのほかの読み取り操作の承認。
<code>solaris.cluster.admin</code>	クラスタオブジェクトの状態を変更する承認。
<code>solaris.cluster.modify</code>	クラスタオブジェクトのプロパティを変更する承認。

Sun Cluster コマンドにより必要とされる RBAC の承認の詳細については、コマンドのマニュアルページを参照してください。

RBAC の権限プロファイルには1つ以上の RBAC の承認が含まれます。これらの権限プロファイルをユーザーまたは役割に割り当てることで、Sun Cluster に対するさまざまなレベルのアクセス権をユーザーや役割に与えることができます。次に、Sun Cluster ソフトウェアに含まれる権限プロファイルを示します。

注- 次の表に示す RBAC の権限プロファイルは、以前の Sun Cluster リリースで定義された古い RBAC の承認を引き続きサポートします。

権限プロファイル	含まれる承認	役割に許可されたアクセス権
Sun Cluster コマンド	なし。ただし、 <code>euclid=0</code> を指定して実行される Sun Cluster コマンドのリストが含まれます。	すべての Sun Cluster コマンドの次のサブコマンドを含めて、クラスタを構成および管理するために使用する一部の Sun Cluster コマンドの実行。 <ul style="list-style-type: none"> ■ list ■ show ■ status scha_control(1HA) scha_resource_get(1HA) scha_resource_setstatus(1HA) scha_resourcegroup_get(1HA) scha_resourcetype_get(1HA)
Basic Solaris User	この既存の Solaris 権限プロファイルには、Solaris の承認のほか次のものが含まれません。	

権限プロファイル	含まれる承認	役割に許可されたアクセス権
	<code>solaris.cluster.read</code>	Sun Cluster コマンドの一覧表示、表示、およびその他の読み取り操作の実行、ならびに Sun Cluster Manager GUI へのアクセス。
Cluster Operation	この権限プロファイルは Sun Cluster に固有で、次の承認が含まれています。	
	<code>solaris.cluster.read</code>	Sun Cluster Manager にアクセスするだけでなく、一覧表示、表示、エクスポート、状態、およびその他の読み取り操作の実行。
	<code>solaris.cluster.admin</code>	クラスタオブジェクトの状態の変更。
システム管理者	この既存の Solaris 権限プロファイルには、Cluster Management プロファイルに含まれるものと同じ承認が入っています。	Cluster Management 役割 ID に許可された作業と、その他のシステム管理作業を行えます。
Cluster Management	この権限プロファイルには、Cluster Operation プロファイルに含まれるものと同じ承認のほか、以下の承認が含まれます。	Cluster Operation 役割 ID が実行できるのと同じオペレーションおよびクラスタオブジェクトのプロパティの変更を実行します。
	<code>solaris.cluster.modify</code>	

Sun Cluster 管理権限プロファイルによる RBAC 役割の作成と割り当て

この作業によって、Sun Cluster 管理権限プロファイルを持つ新しい RBAC の役割を作成し、この新しい役割にユーザーを割り当てます。

▼ 管理役割ツールを使用して役割を作成する方法

始める前に 役割を作成するには、Primary Administrator 権利プロファイルが割り当てられている役割になるか、root ユーザーとして実行する必要があります。

- 1 管理役割ツールを起動します。
ユーザーカウントツールを実行するためには、Solaris Management Console を起動する必要があります。これについては、『Solaris のシステム管理(セキュリティサービス)』の「Solaris 管理コンソールで役割を引き受ける方法」を参照してください。「ユーザー」ツールコレクションを開いて、「管理役割 (Administrative Roles)」アイコンをクリックします。
- 2 「管理役割を追加 (Add Administrative)」ウィザードが起動します。
「アクション (Action)」メニューから「管理役割を追加 (Add Administrative Role)」を選択して、「管理役割を追加 (Add Administrative)」ウィザードを起動します。

3 Cluster Management 権限プロファイルが割り当てられる役割を作成します。

「次へ (Next)」および「戻る (Back)」ボタンを使用して、ダイアログボックスを移動します。ただし、すべての必要なフィールドに入力がなされるまで、「次へ (Next)」ボタンはアクティブになりません。最後に、入力したデータを確認するダイアログボックスが表示されます。「戻る (Back)」ボタンを使用して入力を変更するか、「完了 (Finish)」をクリックして新しい役割を保存します。次のリストに、ダイアログボックスのフィールドとボタンの概要を示します。

役割名	役割の短縮名
「フルネーム」	正式名
Description	役割の説明
役割 ID 番号	役割の UID。自動的に増分する
役割のシェル	役割に使用できるプロファイルシェル: Administrator の C シェル、Administrator の Bourne シェル、または Administrator の Korn シェル
役割のメーリングリストを作成	この役割に割り当てられているユーザーのメーリングリストを作成する
有効な権利 / 許可された権利	役割の権利プロファイルの割り当てまたは削除を行う 同一のコマンドを複数回入力しても、エラーにはならない。ただし、権利プロファイルでは、同一のコマンドが複数回発生した場合、最初のコマンドに割り当てられた属性が優先され、後続の同一コマンドはすべて無視される。順番を変更するときは、上矢印または下矢印を使用する
サーバー	ホームディレクトリのサーバー
パス	ホームディレクトリのパス
追加	この役割を引き受けるユーザーを追加する。同じスコープ内でユーザーでなければならない
削除	この役割が割り当てられているユーザーを削除する

注-このプロファイルは、役割に割り当てられるプロファイルリストの先頭に置く必要があります。

- 4 新しく作成した役割に、**Sun Cluster Manager** 機能や **Sun Cluster** コマンドを使用する必要があるユーザーを追加します。

`useradd(1M)` コマンドを使用して、ユーザーアカウントをシステムに追加します。ユーザーのアカウントに役割を割り当てるには、`-P` オプションを使用します。

- 5 「**Finish (完了)**」をクリックします。

- 6 端末ウィンドウを開き、`root` になります。

- 7 ネームサービスキャッシュデーモンを起動して停止します。

新しい役割は、ネームサービスキャッシュデーモンを再起動するまで有効になりません。`root` になったあと、次のテキストを入力します。

```
# /etc/init.d/nscd stop
# /etc/init.d/nscd start
```

▼ コマンド行から役割を作成する方法

- 1 スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になります。

- 2 次のいずれかの役割の作成方法を選択します。

- ローカルスコープの役割を作成する場合は、`roleadd(1M)` コマンドを使用して、新しいローカル役割とその属性を指定します。
- 同じくローカルスコープの役割を作成する場合に、`user_attr(4)` ファイルを編集して、ユーザーに `type=role` を追加することもできます。
この方法は緊急時にのみ使用します。
- ネームサービスの役割を作成する場合は、`smrole(1M)` コマンドを使用して、新しい役割とその属性を指定します。

このコマンドは、スーパーユーザー、またはその他の役割を作成できる役割による認証を必要とします。`smrole` コマンドは、すべてのネームサービスに適用でき、Solaris 管理コンソールサーバーのクライアントとして動作します。

- 3 ネームサービスキャッシュデーモンを起動して停止します。

新しい役割は、ネームサービスキャッシュデーモンを再起動するまで有効になりません。`root` として、次のテキストを入力します。

```
# /etc/init.d/nscd stop
# /etc/init.d/nscd start
```

例 2-1 smrole コマンドを使用してカスタムの Operator 役割を作成する

次のコマンドシーケンスは、smrole コマンドを使用して役割を作成します。この例では、新しい Operator 役割が作成され、標準の Operator 権利プロファイルと Media Restore 権利プロファイルが割り当てられます。

```
% su primaryadmin
# /usr/sadm/bin/smrole add -H myHost -- -c "Custom Operator" -n oper2 -a johnDoe \
-d /export/home/oper2 -F "Backup/Restore Operator" -p "Operator" -p "Media Restore"

Authenticating as user: primaryadmin

Type /? for help, pressing <enter> accepts the default denoted by [ ]
Please enter a string value for: password :: <type primaryadmin password>

Loading Tool: com.sun.admin.usermgr.cli.role.UserMgrRoleCli from myHost
Login to myHost as user primaryadmin was successful.
Download of com.sun.admin.usermgr.cli.role.UserMgrRoleCli from myHost was successful.

Type /? for help, pressing <enter> accepts the default denoted by [ ]
Please enter a string value for: password :: <type oper2 password>

# /etc/init.d/nscd stop
# /etc/init.d/nscd start
```

新しく作成した役割およびその他の役割を表示するには、次のように smrole コマンドに list オプションを指定します。

```
# /usr/sadm/bin/smrole list --
Authenticating as user: primaryadmin

Type /? for help, pressing <enter> accepts the default denoted by [ ]
Please enter a string value for: password :: <type primaryadmin password>

Loading Tool: com.sun.admin.usermgr.cli.role.UserMgrRoleCli from myHost
Login to myHost as user primaryadmin was successful.
Download of com.sun.admin.usermgr.cli.role.UserMgrRoleCli from myHost was successful.
root                0                Super-User
primaryadmin        100              Most powerful role
sysadmin            101              Performs non-security admin tasks
oper2                102              Custom Operator
```

ユーザーのRBACプロパティの変更

ユーザーアカウントツールかコマンド行のいずれかを使用すると、ユーザーのRBACプロパティを変更できます。ユーザーのRBACのプロパティを変更するには、次のいずれかの手順を選択してください。

- 59 ページの「ユーザーアカウントツールを使用してユーザーのRBACプロパティを変更する方法」
- 60 ページの「コマンド行からユーザーのRBACプロパティを変更する方法」

▼ ユーザーアカウントツールを使用してユーザーのRBACプロパティを変更する方法

始める前に ユーザーのプロパティを変更するには、「ユーザー」ツールコレクションをスーパーユーザーとして実行するか、管理者権利プロファイルが割り当てられている役割を持つ必要があります。

- 1 ユーザーアカウントツールを起動します。

ユーザーアカウントツールを実行するためには、Solaris Management Console を起動する必要があります。これについては、『Solaris のシステム管理 (セキュリティサービス)』の「Solaris 管理コンソールで役割を引き受ける方法」を参照してください。「ユーザー」ツールコレクションを開いて、「ユーザーアカウント (User Accounts)」アイコンをクリックします。

ユーザーアカウントツールが起動すると、既存のユーザーアカウントのアイコンが表示区画に表示されます。
- 2 変更する「ユーザーアカウント (User Accounts)」アイコンをクリックして、「アクション (Action)」メニューから「プロパティ (Properties)」を選択するか、ユーザーアカウントのアイコンをダブルクリックします。
- 3 変更するプロパティのダイアログボックスで、適切なタブを次のように選択します。
 - ユーザーに割り当てられた役割を変更するときは、「役割 (Role)」タブをクリックして、変更する役割を「有効な役割 (Available Roles)」または「割り当てられた役割 (Assigned Roles)」列に移動します。
 - ユーザーに割り当てられた権利プロファイルを変更するときは、「権利 (Rights)」タブをクリックして、変更する権利プロファイルを「有効な権利 (Available Rights)」または「許可された権利 (Assigned Rights)」列に移動します。
Available Rights or Assigned Rights.

注-ユーザーに権限プロファイルを直接割り当てることは避けてください。特権付きアプリケーションを実行するときは、ユーザーが役割を引き受けるようにしてください。このようにすると、ユーザーが特権を濫用できなくなります。

▼ コマンド行からユーザーのRBACプロパティを変更する方法

- 1 スーパーユーザーになるか、RBAC承認 `solaris.cluster.modify` を提供する役割になります。
- 2 次のように適切なコマンドを選択します。
 - ローカルスコープに定義されたユーザーに割り当てられている承認、役割、または権利プロファイルを変更する場合は、`usermod(1M)` コマンドを使用します。
 - また同じくローカルスコープに定義されたユーザーに割り当てられている承認、役割、または権利プロファイルを変更する場合は、`user_attr` ファイルを編集することもできます。

この方法は緊急時にのみ使用します。
 - ネームサービスに定義されたユーザーに割り当てられている承認、役割、または権利プロファイルを変更する場合は、`smuser(1M)` コマンドを使用します。

このコマンドは、スーパーユーザー、またはユーザーファイルを変更できる役割による認証を必要とします。`smuser` コマンドは、すべてのネームサービスに適用でき、Solaris 管理コンソールサーバーのクライアントとして動作します。

クラスタの停止と起動

この章では、グローバルクラスタ、ゾーンクラスタ、および個々のノードの停止方法と起動方法について説明します。非大域ゾーンの起動については、『Solaris のシステム管理 (Solaris コンテナ: 資源管理と Solaris ゾーン)』の第 18 章「非大域ゾーンの計画と構成 (手順)」を参照してください。

- 61 ページの「クラスタの停止と起動の概要」
- 72 ページの「クラスタ内の 1 つのノードの停止と起動」
- 87 ページの「満杯の /var ファイルシステムを修復する」

この章の関連手順の詳細な説明については、83 ページの「非クラスタモードでノードを起動する」と表 3-2 を参照してください。

クラスタの停止と起動の概要

Sun Cluster の `cluster(1CL) shutdown` コマンドは、グローバルクラスタサービスを正しい順序で停止し、グローバルクラスタ全体をクリーンに停止します。`cluster shutdown` コマンドは、グローバルクラスタの場所を移動するときに使用できます。また、アプリケーションエラーによってデータが破損した場合に、グローバルクラスタを停止するときにも使用できます。`clzonecluster halt` コマンドは、特定のノード上のゾーンクラスタ、または構成済みのすべてのノード上のゾーンクラスタ全体を停止します (ゾーンクラスタ内で `cluster shutdown` コマンドを使用することもできます)。

この章の手順の `phys-schost#` は、グローバルクラスタプロンプトを表します。`clzonecluster` の対話型シェルプロンプトは `clzc:schost>` です。

注 - グローバルクラスタ全体を正しく停止するには、`cluster shutdown` コマンドを使用します。Solaris の `shutdown` コマンドは `clnode(1CL) evacuate` コマンドとともに使用して、個々のノードを停止します。詳細は、63 ページの「クラスタを停止する」または 72 ページの「クラスタ内の 1 つのノードの停止と起動」を参照してください。

`cluster shutdown` と `clzonecluster halt` コマンドは、それぞれグローバルクラスタまたはゾーンクラスタ内のすべてのノードを停止します。その処理は次のように行われます。

1. 実行中のすべてのリソースグループをオフラインにする。
2. グローバルクラスタまたはゾーンクラスタのすべてのクラスタファイルシステムをマウント解除する。
3. `cluster shutdown` コマンドが、グローバルクラスタまたはゾーンクラスタ上のアクティブなデバイスサービスを停止する。
4. `cluster shutdown` コマンドが `init 0` を実行して、クラスタ上のすべてのノードを OpenBoot™ PROM `ok` プロンプトの状態にする (SPARC ベースのシステムの場合) か、または GRUB メニューの「Press any key to continue」メッセージの状態にする (x86 ベースのシステムの場合)。GRUB メニューの詳細な説明は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。`clzonecluster halt` コマンドが `zoneadm - z zoneclustername halt` コマンドを実行して、ゾーンクラスタのゾーンを停止する (ただし、シャットダウンは行わない)。

注 - 必要であれば、ノードを非クラスタモードで (つまり、ノードがクラスタメンバーシップを取得しないように) 起動できます。非クラスタモードは、クラスタソフトウェアをインストールしたり、特定の管理手順を実行する際に役立ちます。詳細は、83 ページの「非クラスタモードでノードを起動する」を参照してください。

表 3-1 作業リスト: クラスタの停止と起動

作業	参照先
クラスタの停止	63 ページの「クラスタを停止する」
すべてのノードを起動してクラスタを起動 クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクトとの動作中の接続が必要です。	65 ページの「クラスタを起動する」
クラスタの再起動	68 ページの「クラスタを再起動する」

▼ クラスタを停止する

グローバルクラスタ、1つのゾーンクラスタ、またはすべてのゾーンクラスタを停止できます。



注意- グローバルクラスタノードやゾーンクラスタノードを停止する場合に、`send brk` をクラスタコンソール上で使用しないでください。この機能はクラスタ内ではサポートされません。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 グローバルクラスタまたはゾーンクラスタで **Oracle Real Application Clusters (RAC)** を実行している場合は、停止するクラスタ上のデータベースのすべてのインスタンスを停止します。
停止の手順については、Oracle Parallel Server/Real Application Clusters 製品のマニュアルを参照してください。
- 2 クラスタ内の任意のノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 グローバルクラスタ、1つのゾーンクラスタ、またはすべてのゾーンクラスタを停止します。
 - グローバルクラスタを停止します。この操作を行うと、すべてのゾーンクラスタも停止します。

```
phys-schost# cluster shutdown -g0 -y
```
 - 特定のゾーンクラスタを停止します。

```
phys-schost# clzonecluster halt zoneclustername
```
 - すべてのゾーンクラスタを停止します。

```
phys-schost# clzonecluster halt +
```


ゾーンクラスタ内で `cluster shutdown` コマンドを使用して、すべてのゾーンクラスタを停止することもできます。

- 4 SPARC ベースのシステムの場合は、グローバルクラスタまたはゾーンクラスタ上のすべてのノードが `ok` プロンプトの状態になったことを確認します。x86 ベースのシステムの場合は、すべてのノードが **GRUB** メニューの状態になったことを確認します。SPARC ベースのシステムの場合はすべてのノードが `ok` プロンプトになるまで、x86 ベースのシステムの場合はすべてのノードが `Boot Subsystem` の状態になるまで、どのノードの電源も切らないでください。
 - SPARC ベースのシステムの場合はグローバルクラスタノードが `ok` プロンプトの状態にあることを確認し、GRUB ベースの x86 システムの場合は「Press any key to continue」というメッセージの状態にあることを確認します。

```
phys-schost# cluster status -t node
```
 - `status` サブコマンドを使用して、ゾーンクラスタが停止したことを確認します。

```
phys-schost# clzonecluster status
```
- 5 必要であれば、グローバルクラスタのノードの電源を切ります。

例 3-1 ゾーンクラスタの停止

次の例では、`sparse-sczone` というゾーンクラスタを停止しています。

```
phys-schost# clzonecluster halt sparse-sczone
Waiting for zone halt commands to complete on all the nodes of the zone cluster "sparse-sczone"...
Sep  5 19:06:01 schost-4 cl_runtime: NOTICE: Membership : Node 2 of cluster 'sparse-sczone' died.
Sep  5 19:06:01 schost-4 cl_runtime: NOTICE: Membership : Node 4 of cluster 'sparse-sczone' died.
Sep  5 19:06:01 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-sczone' died.
Sep  5 19:06:01 schost-4 cl_runtime: NOTICE: Membership : Node 1 of cluster 'sparse-sczone' died.
phys-schost#
```

例 3-2 SPARC: グローバルクラスタの停止

次に、正常なグローバルクラスタの動作を停止して、すべてのノードを停止し、`ok` プロンプトが表示されたときのコンソールの出力例を示します。ここでは、`-g0` オプションで停止の猶予期間をゼロに設定し、`-y` オプションで、確認プロンプトに対して自動的に `yes` と応答するよう指定しています。停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

```
phys-schost# cluster shutdown -g0 -y
Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
/etc/rc0.d/K05initrgm: Calling scswitch -S (evacuate)
```

```
The system is down.  
syncing file systems... done  
Program terminated  
ok
```

例 3-3 x86: グローバルクラスタの停止

次に、正常なグローバルクラスタの動作を停止して、すべてのノードを停止したときのコンソールの出力例を示します。この例では、すべてのノードで `ok` プロンプトが表示されません。ここでは、`-g 0` オプションで停止の猶予期間をゼロに設定し、`-y` オプションで、確認プロンプトに対して自動的に `yes` と応答するよう指定しています。停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

```
phys-schost# cluster shutdown -g0 -y  
May 2 10:32:57 phys-schost-1 cl_runtime:  
WARNING: CMM: Monitoring disabled.  
root@phys-schost-1#  
INIT: New run level: 0  
The system is coming down. Please wait.  
System services are now being stopped.  
/etc/rc0.d/K05initrgm: Calling scswitch -S (evacuate)  
failfasts already disabled on node 1  
Print services already stopped.  
May 2 10:33:13 phys-schost-1 syslogd: going down on signal 15  
The system is down.  
syncing file systems... done  
Type any key to continue
```

参照 停止したグローバルクラスタまたはゾーンクラスタを再起動するには、[65 ページ](#)の「[クラスタを起動する](#)」を参照してください。

▼ クラスタを起動する

この手順では、ノードが停止されているグローバルクラスタまたはゾーンクラスタを起動する方法について説明します。グローバルクラスタノードに対して、`ok` プロンプト (SPARC システムの場合) または「`Press any key to continue`」メッセージ (GRUB ベースの x86 システムの場合) が表示されています。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

注-ゾーンクラスタを作成するために、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「[ゾーンクラスタの設定](#)」の手順を実行します。

- 1 各ノードをクラスタモードで起動します。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                       |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.

Press enter to boot the selected OS, 'e' to edit the commands before booting, or 'c' for a command-line.

GRUB ベースの起動についての詳細は、『[Solaris のシステム管理 \(基本編\)](#)』の「[GRUB を使用して x86 システムをブートする \(作業マップ\)](#)」を参照してください。

注-クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクトとの動作中の接続が必要です。

- ゾーンクラスタが1つの場合は、ゾーンクラスタ全体を起動できます。

```
phys-schost# clzonecluster boot zoneclustername
```

- ゾーンクラスタが複数ある場合は、すべてのゾーンクラスタを起動できます。zoneclustername の代わりに + を使用してください。

- 2 ノードが問題なく起動し、オンラインであることを確認します。

`cluster(ICL)` ステータスコマンドは、グローバルクラスタノードのステータスを報告します。

```
phys-schost# cluster status -t node
```

`clzonecluster(1CL)` ステータスコマンドをグローバルクラスタノードから実行すると、ゾーンクラスタノードの状態が報告されます。

```
phys-schost# clzonecluster status
```

注 - ノードの `/var` ファイルシステムが満杯になると、そのノード上では Sun Cluster が再起動できなくなる可能性があります。この問題が発生した場合は、[87 ページ](#) の「[満杯の /var ファイルシステムを修復する](#)」を参照してください。

例 3-4 SPARC: グローバルクラスタの起動

次に、ノード `phys-schost-1` を起動してグローバルクラスタに結合させたときのコンソールの出力例を示します。グローバルクラスタ内のほかのノードのコンソールにも同様のメッセージが表示されます。ゾーンクラスタの自動起動プロパティが `true` に設定されている場合は、そのマシン上のグローバルクラスタノードが起動すると、ゾーンクラスタノードも自動的に起動されます。

グローバルクラスタノードが再起動すると、そのマシン上のゾーンクラスタノードがすべて停止します。同じマシン上に、自動起動プロパティが `true` に設定されたゾーンクラスタノードがある場合は、グローバルクラスタノードが再起動するとゾーンクラスタノードも再起動されます。

```
ok boot
Rebooting with command: boot
...
Hostname: phys-schost-1
Booting as part of a cluster
NOTICE: Node phys-schost-1 with votecount = 1 added.
NOTICE: Node phys-schost-2 with votecount = 1 added.
NOTICE: Node phys-schost-3 with votecount = 1 added.
...
NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-2 (incarnation # 937690106) has become reachable.
NOTICE: Node phys-schost-3 (incarnation # 937690290) has become reachable.
NOTICE: cluster has reached quorum.
NOTICE: node phys-schost-1 is up; new incarnation number = 937846227.
NOTICE: node phys-schost-2 is up; new incarnation number = 937690106.
NOTICE: node phys-schost-3 is up; new incarnation number = 937690290.
NOTICE: Cluster members: phys-schost-1 phys-schost-2 phys-schost-3.
...
```

▼ クラスタを再起動する

グローバルクラスタを停止するために `cluster shutdown` コマンドを実行してから、各ノード上で `boot` コマンドを使用してグローバルクラスタを起動します。ゾーンクラスタを停止するために `clzonecluster halt` コマンドを使用してから、`clzonecluster boot` コマンドを使用してゾーンクラスタを起動します。`clzonecluster reboot` コマンドを使用することもできます。詳細は、[cluster\(1CL\)](#)、[boot\(1M\)](#)、および [clzonecluster\(1CL\)](#) のマニュアルページを参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 **Oracle RAC** が動作しているクラスタの場合は、停止するクラスタ上のデータベースのすべてのインスタンスを停止します。

停止の手順については、Oracle Parallel Server/Real Application Clusters 製品のマニュアルを参照してください。

- 2 クラスタ内の任意のノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

- 3 クラスタを停止します。

- グローバルクラスタを停止します。

```
phys-schost# cluster shutdown -g0 -y
```

- ゾーンクラスタがある場合は、グローバルクラスタノードからゾーンクラスタを停止します。

```
phys-schost# clzonecluster halt zoneclustername
```

各ノードが停止します。ゾーンクラスタ内で `cluster shutdown` コマンドを使用して、ゾーンクラスタを停止することもできます。

注-クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクタとの動作中の接続が必要です。

4 各ノードを起動します。

停止中に構成を変更した場合以外は、どのような順序でノードを起動してもかまいません。停止中に構成を変更した場合は、最新の構成情報を持つノードを最初に起動する必要があります。

- SPARC ベースのシステムのグローバルクラスタノードの場合は、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステムのグローバルクラスタノードの場合は、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                       |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

注-クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクトとの動作中の接続が必要です。

GRUB ベースの起動についての詳細は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。

- ゾーンクラスタの場合は、グローバルクラスタの 1 つのノードで次のコマンドを入力して、ゾーンクラスタを起動します。

```
phys-schost# clzonecluster boot zoneclustername
```

クラスタコンポーネントが起動すると、起動されたノードのコンソールにメッセージが表示されます。

5 ノードが問題なく起動し、オンラインであることを確認します。

- `clnode status` コマンドを実行すると、グローバルクラスタ上のノードの状態が報告されます。

```
phys-schost# clnode status
```

- `clzonecluster status` コマンドをグローバルクラスタノード上で実行すると、ゾーンクラスタノードの状態が報告されます。

```
phys-schost# clzonecluster status
```

ゾーンクラスタ内で `cluster status` コマンドを実行して、ノードの状態を確認することもできます。

注- ノードの `/var` ファイルシステムが満杯になると、そのノード上では Sun Cluster が再起動できなくなる可能性があります。この問題が発生した場合は、[87 ページの「満杯の /var ファイルシステムを修復する」](#)を参照してください。

例 3-5 ゾーンクラスタの再起動

次の例は、`sparse-sczone` というゾーンクラスタを停止して起動する方法を示しています。 `clzonecluster reboot` コマンドを使用することもできます。

```
phys-schost# clzonecluster halt sparse-sczone
Waiting for zone halt commands to complete on all the nodes of the zone cluster "sparse-sczone"...
Sep  5 19:17:46 schost-4 cl_runtime: NOTICE: Membership : Node 4 of cluster 'sparse-sczone' died.
Sep  5 19:17:46 schost-4 cl_runtime: NOTICE: Membership : Node 2 of cluster 'sparse-sczone' died.
Sep  5 19:17:46 schost-4 cl_runtime: NOTICE: Membership : Node 1 of cluster 'sparse-sczone' died.
Sep  5 19:17:46 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-sczone' died.
phys-schost#
phys-schost# clzonecluster boot sparse-sczone
Waiting for zone boot commands to complete on all the nodes of the zone cluster "sparse-sczone"...
phys-schost# Sep  5 19:18:23 schost-4 cl_runtime: NOTICE: Membership : Node 1 of cluster
'sparse-sczone' joined.
Sep  5 19:18:23 schost-4 cl_runtime: NOTICE: Membership : Node 2 of cluster 'sparse-sczone' joined.
Sep  5 19:18:23 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-sczone' joined.
Sep  5 19:18:23 schost-4 cl_runtime: NOTICE: Membership : Node 4 of cluster 'sparse-sczone' joined.

phys-schost#
phys-schost# clzonecluster status

=== Zone Clusters ===

--- Zone Cluster Status ---

Name           Node Name   Zone HostName   Status   Zone Status
----           -
sparse-sczone  schost-1   sczone-1        Online   Running
                schost-2   sczone-2        Online   Running
                schost-3   sczone-3        Online   Running
                schost-4   sczone-4        Online   Running

phys-schost#
```

例 3-6 SPARC: グローバルクラスタの再起動

次に、正常なグローバルクラスタの動作を停止して、すべてのノードを停止し、ok プロンプトが表示され、グローバルクラスタが再起動したときのコンソールの出力例を示します。ここでは、`-g 0` オプションで停止の猶予期間をゼロに設定し、`-y` オプションで、確認プロンプトに対して自動的に `yes` と応答するよう指定しています。停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

```
phys-schost# cluster shutdown -g0 -y
Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
...
The system is down.
syncing file systems... done
Program terminated
ok boot
Rebooting with command: boot
...
Hostname: phys-schost-1
Booting as part of a cluster
...
NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-2 (incarnation # 937690106) has become reachable.
NOTICE: Node phys-schost-3 (incarnation # 937690290) has become reachable.
NOTICE: cluster has reached quorum.
...
NOTICE: Cluster members: phys-schost-1 phys-schost-2 phys-schost-3.
...
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...
```

```
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:
```

クラスタ内の1つのノードの停止と起動

グローバルクラスタノード、ゾーンクラスタノード、または非大域ゾーンを停止することができます。ここでは、グローバルクラスタノードとゾーンクラスタノードを停止する手順を説明します。

グローバルクラスタノードを停止するには、`clnode evacuate` コマンドを Solaris の `shutdown` コマンドとともに使用します。`cluster shutdown` コマンドは、グローバルクラスタ全体を停止する場合にのみ使用します。

ゾーンクラスタノードでは、`clzonecluster halt` コマンドをグローバルクラスタで使用して、1つのゾーンクラスタノードまたはゾーンクラスタ全体を停止します。`clnode evacuate` コマンドと `shutdown` コマンドを使用してゾーンクラスタノードを停止することもできます。

非大域ゾーンの停止と起動については、『Solaris のシステム管理 (Solaris コンテナ: 資源管理と Solaris ゾーン)』の第 20 章「非大域ゾーンのインストール、起動、停止、アンインストール、および複製 (手順)」を参照してください。また、`clnode(1CL)`、`shutdown(1M)`、および `clzonecluster(1CL)` も参照してください。

この章の手順の `phys-schost#` は、グローバルクラスタプロンプトを表します。`clzonecluster` の対話型シェルプロンプトは `clzc:schost>` です。

表 3-2 作業マップ: ノードの停止と起動

作業	ツール	参照先
ノードの停止。	グローバルクラスタノードの場合は、 <code>clnode(1CL) evacuate</code> および <code>shutdown</code> を使用。 ゾーンクラスタノードの場合は、 <code>clzonecluster(1CL) halt</code> を使用。	73 ページの「ノードを停止する」

表 3-2 作業マップ: ノードの停止と起動 (続き)

作業	ツール	参照先
ノードの起動. クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクととの動作中の接続が必要です。	グローバルクラスタノードの場合は、 <code>boot</code> または <code>b</code> を使用。 ゾーンクラスタノードの場合は、 <code>clzonecluster(1CL)</code> <code>boot</code> を使用。	77 ページの「ノードを起動する」
クラスタ上のノードをいったん停止してから再起動。 クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクととの動作中の接続が必要です。	グローバルクラスタノードの場合は、 <code>clnode evacuate</code> および <code>shutdown</code> を使用してから、 <code>boot</code> または <code>b</code> を使用。 ゾーンクラスタノードの場合は、 <code>clzonecluster(1CL)</code> <code>reboot</code> を使用。	79 ページの「ノードを再起動する」
ノードがクラスタメンバーシップを取得しないようにノードを起動。	グローバルクラスタノードの場合は、 <code>clnode evacuate</code> および <code>shutdown</code> コマンドを使用してから、 <code>boot -x</code> を使用 (SPARC または x86 の GRUB メニューエントリ編集で)。 基になるグローバルクラスタが非クラスタモードで起動される場合は、ゾーンクラスタノードも自動的に非クラスタモードになります。	83 ページの「非クラスタモードでノードを起動する」

▼ ノードを停止する

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。



注意 - グローバルクラスタやゾーンクラスタ上のノードを停止する場合に、`send brk` をクラスタコンソール上で使用しないでください。この機能はクラスタ内ではサポートされません。

- 1 **Oracle RAC** が動作しているクラスタの場合は、停止するクラスタ上のデータベースのすべてのインスタンスを停止します。

停止の手順については、Oracle Parallel Server/Real Application Clusters 製品のマニュアルを参照してください。

- 2 停止するクラスタノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

- 3 特定のゾーンクラスタメンバーを停止する場合は、手順 **4** ~ **6** をスキップし、グローバルクラスタノードから次のコマンドを実行します。

```
phys-schost# clzonecluster halt -n physical-name zoneclustername
```

特定のゾーンクラスタノードを指定すると、そのノードのみが停止します。halt コマンドは、デフォルトではすべてのノード上のゾーンクラスタを停止します。

- 4 すべてのリソースグループ、リソース、およびデバイスグループを、停止するノードから別のグローバルクラスタノードに切り替えます。

停止するグローバルクラスタノードで、次のようにコマンドを入力します。clnode evacuate コマンドは、非大域ゾーンを含むすべてのリソースグループとデバイスグループを、指定ノードから次に優先されるノードに切り替えます(ゾーンクラスタノード内で clnode evacuate を実行することもできます)。

```
phys-schost# clnode evacuate node
```

node リソースグループとデバイスグループを切り替えるノードを指定します。

- 5 ノードを停止します。

停止するグローバルクラスタノードを指定します。

```
phys-schost# shutdown -g0 -y -i0
```

SPARC ベースのシステムではグローバルクラスタノードが ok プロンプトを表示し、x86 ベースのシステムでは GRUB メニューで「Press any key to continue」というメッセージが表示されていることを確認します。

- 6 必要であればノードの電源を切ります。

例 3-7 SPARC: グローバルクラスタノードの停止

次の例に、ノード `phys-schost-1` が停止した場合のコンソール出力を示します。ここでは、`-g0` オプションで猶予期間をゼロに設定し、`-y` オプションで、確認プロンプトに対して自動的に `yes` と応答するよう指定しています。このノードの停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されません。

```
phys-schost# clnode evacuate nodename
phys-schost# shutdown -g0 -y
Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
Notice: rgmd is being stopped.
Notice: rpc.pmfd is being stopped.
Notice: rpc.fed is being stopped.
umount: /global/.devices/node@1 busy
umount: /global/phys-schost-1 busy
The system is down.
syncing file systems... done
Program terminated
ok
```

例 3-8 x86: グローバルクラスタノードの停止

次の例に、ノード `phys-schost-1` が停止した場合のコンソール出力を示します。ここでは、`-g0` オプションで猶予期間をゼロに設定し、`-y` オプションで、確認プロンプトに対して自動的に `yes` と応答するよう指定しています。このノードの停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されません。

```
phys-schost# clnode evacuate phys-schost-1
phys-schost# shutdown -g0 -y
Shutdown started.   Wed Mar 10 13:47:32 PST 2004

Changing to init state 0 - please wait
Broadcast Message from root (console) on phys-schost-1 Wed Mar 10 13:47:32...
THE SYSTEM phys-schost-1 IS BEING SHUT DOWN NOW ! ! !
Log off now or risk your files being damaged

phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
```

```

/etc/rc0.d/K05initrgm: Calling scswitch -S (evacuate)
failfasts disabled on node 1
Print services already stopped.
Mar 10 13:47:44 phys-schost-1 syslogd: going down on signal 15
umount: /global/.devices/node@2 busy
umount: /global/.devices/node@1 busy
The system is down.
syncing file systems... done
WARNING: CMM: Node being shut down.
Type any key to continue

```

例 3-9 ゾーンクラスタノードの停止

次の例は、`clzonecluster halt` を使用して *sparse-sczone* というゾーンクラスタ上のノードを停止する方法を示しています (ゾーンクラスタノード内で `clnode evacuate` コマンドと `shutdown` コマンドを実行することもできます)。

```
phys-schost# clzonecluster status
```

```
=== Zone Clusters ===
```

```
--- Zone Cluster Status ---
```

Name	Node Name	Zone HostName	Status	Zone Status
sparse-sczone	schost-1	sczone-1	Online	Running
	schost-2	sczone-2	Online	Running
	schost-3	sczone-3	Online	Running
	schost-4	sczone-4	Online	Running

```
phys-schost#
```

```
phys-schost# clzonecluster halt -n schost-4 sparse-sczone
```

```
Waiting for zone halt commands to complete on all the nodes of the zone cluster "sparse-sczone"...
```

```
Sep 5 19:24:00 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-sczone' died.
```

```
phys-host#
```

```
phys-host# clzonecluster status
```

```
=== Zone Clusters ===
```

```
--- Zone Cluster Status ---
```

Name	Node Name	Zone HostName	Status	Zone Status
sparse-sczone	schost-1	sczone-1	Online	Running
	schost-2	sczone-2	Online	Running
	schost-3	sczone-3	Offline	Installed
	schost-4	sczone-4	Online	Running

phys-schost#

参照 停止したグローバルクラスタノードを再起動するには、77ページの「ノードを起動する」を参照してください。

▼ ノードを起動する

グローバルクラスタまたはゾーンクラスタ内のほかのアクティブノードを停止または再起動する場合は、少なくとも起動中のノードが次の状態になるまで待ってください。

- SPARC: Solaris 9 OS を実行している場合は、ログインプロンプトが表示されるのを待ちます。
- Solaris 10 OS を実行している場合は、マルチユーザーサーバーマイルストーンがオンラインになるのを待ちます。

ログインプロンプトが表示されてからでなければ、そのノードは、停止または再起動するクラスタ内の他のノードからサービスを引き継げません。非大域ゾーンの起動については、『Solaris のシステム管理 (Solaris コンテナ: 資源管理と Solaris ゾーン)』の第20章「非大域ゾーンのインストール、起動、停止、アンインストール、および複製(手順)」を参照してください。

注- ノードの起動は、定足数の構成によって変わる場合があります。2ノードのクラスタでは、クラスタの定足数の合計数が3つになるように定足数デバイスを構成する必要があります(各ノードごとに1つと定足数デバイスに1つ)。この場合、最初のノードを停止しても、2つ目のノードは定足数を保持しており、唯一のクラスタメンバーとして動作します。1番目のノードをクラスタノードとしてクラスタに復帰させるには、2番目のノードが稼動中で必要な数のクラスタ定足数(2つ)が存在している必要があります。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

注-クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクタとの動作中の接続が必要です。

- 1 停止したグローバルクラスタノードやゾーンクラスタノードを起動するために、そのノードを起動します。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                       |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

クラスタコンポーネントが起動すると、起動されたノードのコンソールにメッセージが表示されます。

- ゾーンクラスタがある場合は、起動するノードを指定できます。

```
phys-schost# clzonecluster boot -n node zoneclustername
```

- 2 ノードが問題なく起動し、オンラインであることを確認します。

- `cluster status` コマンドを実行すると、グローバルクラスタノードのステータスが報告されます。

```
phys-schost# cluster status -t node
```

- `clzonecluster status` コマンドをグローバルクラスタ上のノードから実行すると、すべてのゾーンクラスタノードのステータスが報告されます。

```
phys-schost# clzonecluster status
```

ホストのノードがクラスタモードで起動される場合は、ゾーンクラスタノードもクラスタモードのみで起動できます。

注- ノードの `/var` ファイルシステムが満杯になると、そのノード上では Sun Cluster が再起動できなくなる可能性があります。この問題が発生した場合は、[87 ページの「満杯の `/var` ファイルシステムを修復する」](#)を参照してください。

例 3-10 SPARC: グローバルクラスタノードの起動

次に、ノード `phys-schost-1` を起動してグローバルクラスタに結合させたときのコンソールの出力例を示します。

```
ok boot
Rebooting with command: boot
...
Hostname: phys-schost-1
Booting as part of a cluster
...
NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:
```

▼ ノードを再起動する

グローバルクラスタまたはゾーンクラスタ内のほかのアクティブノードを停止または再起動するには、少なくとも再起動中のノードが次の状態になるまで待ってください。

- SPARC: Solaris 9 OS を実行している場合は、ログインプロンプトが表示されるのを待ちます。
- Solaris 10 OS を実行している場合は、マルチユーザーサーバーマイルストーンがオンラインになるのを待ちます。

ログインプロンプトが表示されてからでなければ、そのノードは、停止または再起動するクラスタ内の他のノードからサービスを引き継げません。非大域ゾーンの再起動については、『[Solaris のシステム管理 \(Solaris コンテナ: 資源管理と Solaris](#)

ゾーン』の第20章「非大域ゾーンのインストール、起動、停止、アンインストール、および複製(手順)」を参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 グローバルクラスタノードまたはゾーンクラスタノードで **Oracle RAC** を実行している場合は、停止するノード上のデータベースのすべてのインスタンスを停止します。停止の手順については、Oracle Parallel Server/Real Application Clusters 製品のマニュアルを参照してください。
- 2 停止するノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になりますグローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 `clnode evacuate` および `shutdown` コマンドを使用して、グローバルクラスタノードを停止します。グローバルクラスタのノード上で実行する `clzonecluster halt` コマンドで、ゾーンクラスタを停止します。(`clnode evacuate` コマンドと `shutdown` コマンドもゾーンクラスタ内で動作します)。

グローバルクラスタの場合は、停止するノードで次のコマンドを入力します。 `clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定のノード上の大域または非大域ゾーンから、ほかのノード上の次に優先される大域または非大域ゾーンへ、すべてのリソースグループを切り替えます。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# clnode evacuate node
```

```
phys-schost# shutdown -g0 -y -i6
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# clnode evacuate node
```

```
phys-schost# shutdown -g0 -y -i6
```

GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86                               |
```

```
| Solaris failsafe |
| |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

- 停止し、再起動するゾーンクラスタノードを指定します。

```
phys-schost# clzonecluster reboot - node zoneclustername
```

注-クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクタとの動作中の接続が必要です。

- 4 ノードが問題なく起動し、オンラインであることを確認します。

- グローバルクラスタノードがオンラインであることを確認します。

```
phys-schost# cluster status -t node
```

- ゾーンクラスタノードがオンラインであることを確認します。

```
phys-schost# clzonecluster status
```

例 3-11 SPARC: グローバルクラスタノードの再起動

次の例に、ノード `phys-schost-1` が再起動した場合のコンソール出力を示します。このノードの停止時および起動時の通知メッセージは、グローバルクラスタ内のほかのノードのコンソールに表示されます。

```
phys-schost# clnode evacuate phys-schost-1
phys-schost# shutdown -g0 -y -i6
Shutdown started.   Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
```

```
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 6
The system is coming down. Please wait.
System services are now being stopped.
Notice: rgmd is being stopped.
Notice: rpc.pmfd is being stopped.
Notice: rpc.fed is being stopped.
umount: /global/.devices/node@1 busy
umount: /global/phys-schost-1 busy
The system is down.
syncing file systems... done
rebooting...
```

```

Resetting ...
'''
Sun Ultra 1 SBus (UltraSPARC 143MHz), No Keyboard
OpenBoot 3.11, 128 MB memory installed, Serial #5932401.
Ethernet address 8:8:20:99:ab:77, Host ID: 8899ab77.
...
Rebooting with command: boot
...
Hostname: phys-schost-1
Booting as part of a cluster
...
NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
The system is ready.
phys-schost-1 console login:

```

例 3-12 x86: グローバルクラスタノードの再起動

次に、ノード `phys-schost-1` を再起動したときのコンソールの出力例を示します。このノードの停止時および起動時の通知メッセージは、グローバルクラスタ内のほかのノードのコンソールに表示されます。

```

phys-schost# clnode evacuate phys-schost-1
phys-schost # shutdown -y -g0 -i6

GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                       |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
Hostname: phys-schost-1
Booting as part of a cluster
...
NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...

```

```
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:
```

例 3-13 ゾーンクラスタノードの再起動

次の例は、ゾーンクラスタ上のノードを再起動する方法を示しています。

```
phys-schost# clzonecluster reboot -n schost-4 sparse-sczone
Waiting for zone reboot commands to complete on all the nodes of the zone cluster "sparse-sczone"...
Sep  5 19:40:59 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-sczone' died.
phys-schost# Sep  5 19:41:27 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-sczone' joined.
```

```
phys-schost#
phys-schost# clzonecluster status
```

```
=== Zone Clusters ===
```

```
--- Zone Cluster Status ---
```

Name	Node Name	Zone HostName	Status	Zone Status
sparse-sczone	schost-1	sczone-1	Online	Running
	schost-2	sczone-2	Online	Running
	schost-3	sczone-3	Online	Running
	schost-4	sczone-4	Online	Running

```
phys-schost#
```

▼ 非クラスタモードでノードを起動する

グローバルクラスタノードは、非クラスタモードで起動できます(その場合は、ノードがクラスタメンバーシップに参加しません)。非クラスタモードは、クラスタソフトウェアをインストールしたり、ノードにパッチを適用するなどの特定の管理手順を実行する際に役立ちます。ゾーンクラスタノードは、その基になるグローバルクラスタノードの状態と異なる状態では起動できません。グローバルクラスタノードが、非クラスタモードで起動すると、ゾーンクラスタノードも自動的に非クラスタモードになります。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 非クラスタモードで起動するクラスタ上で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 2 `clzonecluster halt` コマンドをグローバルクラスタのノード上で実行して、ゾーンクラスタノードを停止します。`clnode evacuate` および `shutdown` コマンドを使用して、グローバルクラスタノードを停止します。

`clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定のノード上の大域または非大域ゾーンから、ほかのノード上の次に優先される大域または非大域ゾーンへ、すべてのリソースグループを切り替えます。

- 特定のグローバルクラスタを停止します。

```
phys-schost# clnode evacuate node
```

```
phys-schost# shutdown -g0 -y
```

- グローバルクラスタノードから特定のゾーンクラスタノードを停止します。

```
phys-schost# clzonecluster halt -n node zoneclustername
```

ゾーンクラスタ内で `clnode evacuate` コマンドと `shutdown` コマンドを使用することもできます。

- 3 SPARC ベースのシステムではグローバルクラスタノードが `ok` プロンプトを表示し、x86 ベースのシステムでは GRUB メニューで「Press any key to continue」というメッセージが表示されていることを確認します。
- 4 非クラスタモードでグローバルクラスタノードを起動します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -xs
```

- x86 ベースのシステム上で、次のコマンドを実行します。

- a. GRUB メニューで矢印キーを使用して該当する Solaris エントリを選択し、`e` と入力してコマンドを編集します。

GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
```

```
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

GRUB ベースの起動についての詳細は、『[Solaris のシステム管理 \(基本編\)](#)』の「[GRUB を使用して x86 システムをブートする \(作業マップ\)](#)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、e を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.
```

- c. コマンドに -x を追加して、システムが非クラスタモードで起動するように指定します。

```
[ Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits. ]
```

```
grub edit> kernel /platform/i86pc/multiboot -x
```

- d. Enter キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot -x |
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
```

boot sequence, 'c' for a command-line, 'o' to open a new line after ('O' for before) the selected line, 'd' to remove the selected line, or escape to go back to the main menu.-

- e. **b** と入力して、ノードを非クラスタモードで起動します。

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-x` オプションを追加します。

例 3-14 SPARC: 非クラスタモードでグローバルクラスタノードを起動する

次に、ノード `phys-schost-1` を停止し、非クラスタモードで再起動した場合のコンソール出力の例を示します。ここでは、`-g0` オプションで猶予期間をゼロに設定し、`-y` オプションで、確認プロンプトに対して自動的に `yes` と応答するよう指定し、`-i0` で実行レベル 0 (ゼロ) で起動します。このノードの停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

```
phys-schost# clnode evacuate phys-schost-1
phys-schost# cluster shutdown -g0 -y
Shutdown started.   Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:

WARNING: CMM monitoring disabled.
phys-schost-1#
...
rg_name = schost-sa-1 ...
offline node = phys-schost-2 ...
num of node = 0 ...
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
Print services stopped.
syslogd: going down on signal 15
...
The system is down.
syncing file systems... done
WARNING: node phys-schost-1 is being shut down.
Program terminated

ok boot -x
...
Not booting as part of cluster
```

```
...  
The system is ready.  
phys-schost-1 console login:
```

満杯の /var ファイルシステムを修復する

Solaris と Sun Cluster ソフトウェアは、どちらも /var/adm/messages ファイルにエラーメッセージを書き込みます。このため、運用を続けるうちに /var ファイルシステムが満杯になってしまうことがあります。クラスタノードの /var ファイルシステムが満杯になると、そのノード上では Sun Cluster が再起動できなくなる可能性があります。また、そのノードにログインできなくなる可能性もあります。

▼ 満杯の /var ファイルシステムを修復する

/var ファイルシステムが満杯になったことがノードによって報告され、Sun Cluster サービスが引き続き実行されているときは、次の手順で、満杯になったファイルシステムを整理してください。詳細は、『Solaris のシステム管理 (上級編)』の『Solaris のシステム管理 (上級編)』の「システムメッセージの表示」を参照してください。

- 1 満杯の /var ファイルシステムが存在するクラスタノードでスーパーユーザーになります。
- 2 満杯のファイルシステムを整理します。
たとえば、ファイルシステムにある重要ではないファイルを削除します。

データ複製のアプローチ

この章では、Sun Cluster ソフトウェアで使用できるデータ複製技術について説明します。データ複製は、主ストレージデバイスからバックアップデバイス(二次デバイス)へのデータのコピーとして定義されます。主デバイスに障害が発生した場合も、二次デバイスからデータを使用できます。データ複製を使用すると、クラスタの高可用性と耐障害性を確保できます。

Sun Cluster ソフトウェアは、次のデータ複製タイプをサポートします。

- クラスタ間 - 障害回復には、Sun Cluster Geographic Edition を使用します
- クラスタ内 - キャンパスクラスタ内でホストベースのミラーリングの代替として使用します

データ複製を実行するには、複製するオブジェクトと同じ名前のデバイスグループが必要です。1つのデバイスは、一度に1つのデバイスグループにのみ属することができるため、デバイスを有する Sun Cluster デバイスグループをすでに持っている場合、そのグループを削除してからそのデバイスを新しいデバイスグループに追加します。Solaris Volume Manager、Veritas Volume Manager、ZFS、または raw ディスクデバイスグループの作成および管理については、第5章の129ページの「[デバイスグループの管理](#)」を参照してください。

クラスタに最適なサービスを提供する複製アプローチを選択するには、ホストベースとストレージベースのデータ複製を両方とも理解しておく必要があります。障害回復でデータ複製を管理するために使用する Sun Cluster Geographic Edition に関する詳細は、『[Sun Cluster Geographic Edition Overview](#)』を参照してください。

この章で説明する内容は次のとおりです。

- 90 ページの「[データ複製についての理解](#)」
- 92 ページの「[クラスタ内でのストレージベースのデータ複製の使用](#)」

データ複製についての理解

Sun Cluster はデータ複製に対して次のアプローチをサポートしています。

- ホストベースのデータ複製では、ソフトウェアを使用して、地理的に離れたクラスタ間でディスクボリュームをリアルタイムに複製します。リモートミラー複製を使用すると、主クラスタのマスターボリュームのデータを、地理的に離れた二次クラスタのマスターボリュームに複製できます。リモートミラービットマップは、主ディスク上のマスターボリュームと、二次ディスク上のマスターボリュームの差分を追跡します。クラスタ間(およびクラスタとクラスタの外にあるホストとの間)の複製に使用されるホストベースの複製ソフトウェアには、Sun StorageTek Availability Suite 4 や Sun StorEdge Availability Suite 3.2.1 などがあります。
ホストベースのデータ複製は、特別なストレージレイではなくホストリソースを使用するため、比較的安価なデータ複製ソリューションです。Solaris OS を実行する複数のホストが共有ボリュームにデータを書き込むことができるように構成されているデータベース、アプリケーション、またはファイルシステムはサポートされていません(Oracle 9iRAC、Oracle Parallel Server など)。2つのクラスタ間でホストベースのデータ複製を使用する方法の詳細については、『[Sun Cluster Geographic Edition Data Replication Guide for Sun StorageTek Availability Suite](#)』を参照してください。Sun Cluster Geographic Edition を使用しないホストベースの複製の例については、付録 A の381 ページの「[Sun StorEdge Availability Suite または Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成](#)」を参照してください。
- ストレージベースのデータ複製は、ストレージコントローラ上でソフトウェアを使用して、データ複製の作業をクラスタノードからストレージデバイスに移動させます。このソフトウェアはノードの処理能力を一部解放し、クラスタの要求にサービスを提供します。クラスタ内またはクラスタ間でデータを複製できるストレージベースのソフトウェアには、Hitachi TrueCopy、Hitachi Universal Replicator、および EMC SRDF などがあります。ストレージベースのデータ複製は、構内クラスタ構成において特に重要になることがあります。この方法の場合、必要なインフラストラクチャーを簡素化できます。構内クラスタ環境でストレージベースのデータ複製を使用する方法の詳細については、92 ページの「[クラスタ内でのストレージベースのデータ複製の使用](#)」を参照してください。
複数のクラスタ間でのストレージベースの複製と、そのプロセスを自動化する Sun Cluster GeoEdition 製品についての詳細は、『[Sun Cluster Geographic Edition Data Replication Guide for Hitachi TrueCopy and Universal Replicator](#)』と『[Sun Cluster Geographic Edition Data Replication Guide for EMC Symmetrix Remote Data Facility](#)』を参照してください。また、このタイプのクラスタ構成の完全な構成例については、381 ページの「[Sun StorEdge Availability Suite または Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成](#)」も参照してください。

サポートされるデータ複製方式

Sun Cluster ソフトウェアは、クラスタ間またはクラスタ内のデータ複製方式として以下をサポートしています。

1. クラスタ間の複製 - 障害回復目的でのクラスタ間のデータ複製には、ホストベースまたはストレージベースの複製を使用できます。通常、ホストベースの複製とストレージベースの複製のいずれかを選択し、両方の組み合わせは選択しません。Sun Cluster Geographic Edition ソフトウェアで両方の種類の複製を管理できます。

- ホストベースの複製
 - Sun StorageTek Availability Suite 4 (Solaris 10 OS 以降)
 - Sun StorEdge Availability Suite 3.2.1 (Solaris 9 OS)

このマニュアルでは、特に明記していないかぎり、Sun StorageTek Availability Suite ソフトウェアに言及している内容は、Sun StorEdge Availability Suite ソフトウェアにも該当します。

Sun Cluster Geographic Edition ソフトウェアを使用しないでホストベースの複製を使用する場合は、付録 A 「例」の381 ページの「[Sun StorEdge Availability Suite または Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成](#)」を参照してください。

- ストレージベースの複製
 - Hitachi TrueCopy および Hitachi Universal Replicator (Sun Cluster Geographic Edition 経由)
 - EMC Symmetrix Remote Data Facility (SRDF) (Sun Cluster Geographic Edition 経由)

Sun Cluster Geographic Edition ソフトウェアを使用しないでストレージベースの複製を使用する場合は、複製ソフトウェアのマニュアルを参照してください。

2. クラスタ内の複製 - この方式は、ホストベースのミラー化の代替として使用されます。
 - ストレージベースの複製
 - Hitachi TrueCopy および Hitachi Universal Replicator
 - EMC Symmetrix Remote Data Facility (SRDF)
3. アプリケーションベースの複製 - Oracle Data Guard はアプリケーションベースの複製ソフトウェアの例です。この種類のソフトウェアは障害回復目的でのみ使用されます。詳細については、『[Sun Cluster Geographic Edition Data Replication Guide for Oracle Data Guard](#)』を参照してください。

クラスタ内でのストレージベースのデータ複製の使用

ストレージベースのデータ複製は、ストレージデバイスにインストールされているソフトウェアを使用して、クラスタまたは構内クラスタ内の複製を管理します。このようなソフトウェアは、特定のストレージデバイスに固有で、障害回復には使用されません。ストレージベースのデータ複製を構成する際には、ストレージデバイスに付属するマニュアルを参照してください。

使用するソフトウェアに応じて、ストレージベースのデータ複製を使用して自動または手動いずれかのフェイルオーバーを使用できます。Sun Cluster は、Hitachi TrueCopy、Hitachi Universal Replicator、EMC SRDF ソフトウェアによる複製物の手動、自動フェイルオーバーをサポートしています。

この節では、構内クラスタで使用されるストレージベースのデータ複製を説明します。図 4-1 に、2つのストレージレイ間でデータが複製される、2か所に設置されたクラスタ構成の例を示します。この例では、第一の場所に主ストレージレイがあり、これが両方の場所のノードにデータを提供します。また主ストレージレイは、複製するデータを二次ストレージレイに提供します。

注- 図 4-1 は、複製されていないボリューム上に定足数デバイスがあることを示しています。複製されたボリュームを定足数デバイスとして使用することはできません。

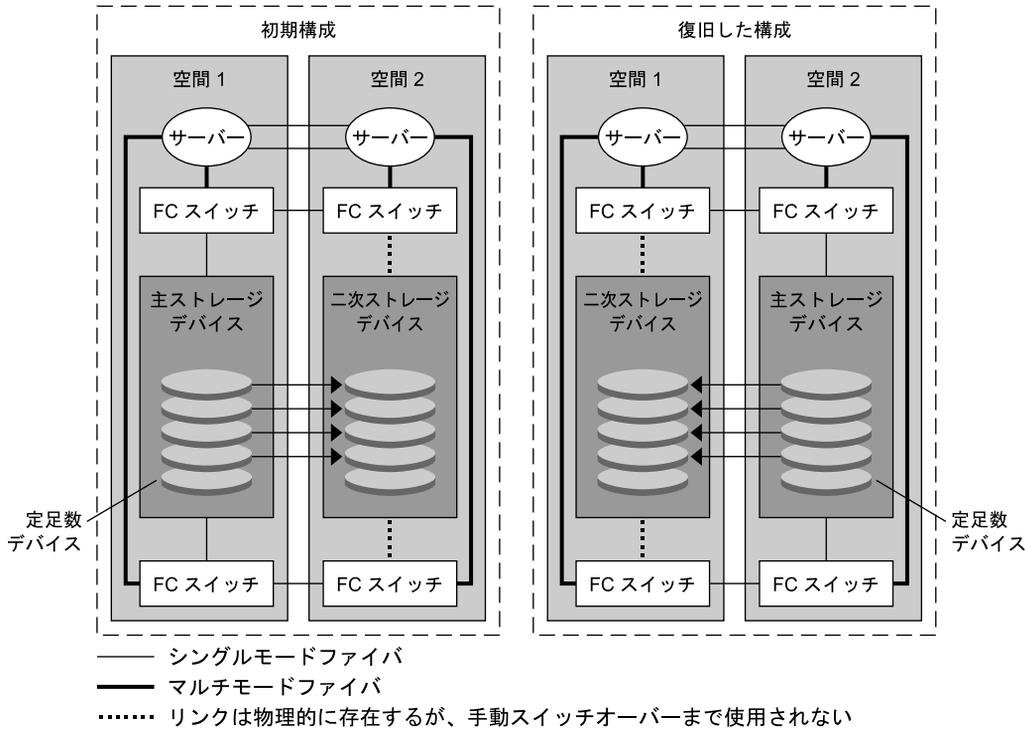


図 4-1 ストレージベースのデータ複製を装備した2ヶ所に設置されたクラスタ構成

使用するアプリケーションの種類によっては、Hitachi TrueCopy または Hitachi Universal Replicator によるストレージベースのデータ複製を Sun Cluster 環境で同期的または非同期に実行できます。構内クラスタで自動フェイルオーバーを実行する場合は、TrueCopy を同期的に使用します。EMC SRDF によるストレージベースの同期複製は、Sun Cluster でサポートされています。EMC SRDF については、非同期複製はサポートされていません。

EMC SRDF のドミノモードまたは適応型コピーモードを使用しないでください。ドミノモードでは、ターゲットが使用可能でないとき、ローカルとターゲット SRDF ボリュームをホストで使用できなくなります。適応型コピーモードは、一般的にデータ移行およびデータセンター移行に使用され、障害回復には推奨されません。

Hitachi TrueCopy または Hitachi Universal Replicator で、データモードまたはステータスモードを使用しないでください。二次ストレージデバイスに不具合が生じた場合、主ストレージへの書き込みに問題が生じる可能性があります。

クラスタ内でストレージベースのデータ複製を使用する際の要件と制限

データの整合性を確保するには、マルチパスおよび適切な RAID パッケージを使用します。次のリストには、ストレージベースのデータ複製を使用するクラスタ構成を実装するための考慮事項が含まれています。

- ノードからノードへの距離は、Sun Cluster Fibre Channel とインターコネクティブラストラクチャーにより制限されます。現在の制限とサポートされる技術の詳細については、Sun のサービスプロバイダにお問い合わせください。
- 複製されたボリュームを、定足数デバイスとして構成しないでください。共有の複製されていないボリュームにある定足数デバイスを見つけるか、定足数サーバーを使用します。
- データの主コピーのみがクラスタノードに認識されるようにします。このようにしないと、ボリュームマネージャーはデータの主コピーと二次コピーの両方に同時にアクセスしようとする場合があります。データコピーの可視性の制御に関しては、ご使用のストレージアレイに付属するマニュアルを参照してください。
- EMC SRDF、Hitachi TrueCopy、Hitachi Universal Replicator では、ユーザーは複製デバイスのグループを決定することができます。各複製デバイスグループには、同じ名前の Sun Cluster デバイスグループが必要です。
- 特定のアプリケーション固有のデータは、非同期データ複製には適さない場合があります。アプリケーションの動作に関する知識を活用して、ストレージデバイス間でアプリケーション固有のデータを複製する最善の方法を決定します。
- クラスタを自動フェイルオーバー用に構成する場合は、同期複製を使用します。複製されたボリュームの自動フェイルオーバー用にクラスタを構成する手順については、[101 ページの「ストレージベースの複製されたデバイスの管理」](#)を参照してください。
- クラスタ内で複製している場合、Oracle Real Application Clusters (RAC) は、SRDF、Hitachi TrueCopy、および Hitachi Universal Replicator でサポートされません。ノードの接続先の複製が現在は主複製ではない場合、そのノードに書き込みアクセス権はありません。クラスタのすべてのノードからの直接書き込みアクセス権が必要なスケラブルアプリケーションは、複製されるデバイスでサポートできません。
- Sun Cluster ソフトウェア用の Veritas Cluster Volume Manager (CVM) および Solaris Volume Manager (SVM) OBAN クラスタはサポートされていません。
- EMC SRDF でドミノモードまたは適応型コピーモードを使用しないでください。詳細は、[92 ページの「クラスタ内でのストレージベースのデータ複製の使用」](#)を参照してください。
- Hitachi TrueCopy または Hitachi Universal Replicator でデータモードまたはステータスモードを使用しないでください。詳細は、[92 ページの「クラスタ内でのストレージベースのデータ複製の使用」](#)を参照してください。

クラスタ内でストレージベースのデータ複製を使用する際の手動回復に関する懸念事項

すべての構内クラスタと同じように、ストレージベースのデータ複製を使用するクラスタは、通常、1つの障害が発生した場合はユーザーの操作は必要ありません。ただし、(図 4-1 に示すように) 手動フェイルオーバーを使用し、主ストレージデバイスを保持する空間が失われた場合、2 ノードクラスタでは問題が発生します。残ったノードは定足数デバイスを予約できず、またクラスタメンバーとして起動できません。このような状況では、クラスタには次の手動操作が必要になります。

1. クラスタメンバーとして起動するよう、Sun のサービスプロバイダが残りのノードを再構成する必要があります。
2. ユーザーまたは Sun のサービスプロバイダが、二次ストレージデバイスの複製されてない方のボリュームを定足数デバイスとして構成する必要があります。
3. 二次ストレージデバイスを主ストレージとして使用できるよう、ユーザーまたは Sun のサービスプロバイダが残りのノードを構成する必要があります。このような再構成には、ボリュームマネージャーボリュームの再構築、データの復元、ストレージボリュームとアプリケーションの関連付けの変更が含まれます。

ストレージベースのデータ複製を使用する際のベストプラクティス

ストレージベースのデータ複製に Hitachi TrueCopy または Hitachi Universal Replicator ソフトウェアを使用するデバイスグループを設定する場合は、次のプラクティスに従ってください。

- 同期複製を使用して、主サイトに障害が発生したときにデータの損失を防ぎます。
- Sun Cluster グローバルデバイスグループと `horcm` 構成ファイルで定義された TrueCopy 複製グループの間に 1 対 1 の関係が存在するようにしてください。これにより、両方のグループが 1 つの単位としてノードからノードへ移動することができます。
- 同一の複製されたデバイスグループ内にグローバルファイルシステムボリュームとフェイルオーバーファイルシステムボリュームを混在させることはできません。制御方法が異なるためです。グローバルファイルシステムはデバイス構成システム (Device Configuration System、DCS) によって制御され、フェイルオーバーファイルシステムボリュームは HAS+ によって制御されます。それぞれの主ノードが異なるノードである可能性があるため、どのノードを複製の主ノードにすべきかについて衝突が発生します。
- すべての RAID マネージャーインスタンスが常に起動され実行中であるべきです。

ストレージベースのデータ複製に EMC SRDF ソフトウェアを使用する場合は、静的デバイスではなく動的デバイスを使用します。静的デバイスでは主複製を変更するのに数分かかり、フェイルオーバー時間に影響を与えることがあります。

グローバルデバイス、ディスクパス監視、およびクラスタファイルシステムの管理

この章では、グローバルデバイス、ディスクパス監視、およびクラスタファイルシステムの管理手順を説明します。

- 97 ページの「グローバルデバイスとグローバルな名前空間の管理の概要」
- 101 ページの「ストレージベースの複製されたデバイスの管理」
- 127 ページの「クラスタファイルシステムの管理の概要」
- 129 ページの「デバイスグループの管理」
- 174 ページの「ストレージデバイス用の SCSI プロトコル設定の管理」
- 180 ページの「クラスタファイルシステムの管理」
- 187 ページの「ディスクパス監視の管理」

この章の関連手順の詳細は、表 5-4 を参照してください。

グローバルデバイス、グローバルな名前空間、デバイスグループ、ディスクパスの監視、およびクラスタファイルシステムに関連する概念については、『[Sun Cluster Concepts Guide for Solaris OS](#)』を参照してください。

グローバルデバイスとグローバルな名前空間の管理の概要

Sun Cluster デバイスグループの管理方法は、クラスタにインストールされているボリューム管理ソフトウェアによって決まります。Solaris ボリュームマネージャーは「クラスタ対応」なので、Solaris ボリュームマネージャーの `metaset(1M)` コマンドを使用してデバイスグループを追加、登録、および削除できます。Veritas Volume Manager (VxVM) を使用している場合、VxVM コマンドを使用してディスクグループを作成し、`clsetup` ユーティリティを使用して、ディスクグループを Sun Cluster のデバイスグループとして登録します。VxVM デバイスグループを削除するには、`clsetup` コマンドと VxVM のコマンドの両方を使用します。

注 - Solaris 10 OS の場合、グローバルデバイスはグローバルクラスタの非投票ノードから直接アクセスすることはできません。

Sun Cluster ソフトウェアは、クラスタ内のディスクデバイスやテープデバイスごとに、raw ディスクデバイスグループを自動的に作成します。ただし、クラスタデバイスグループはグローバルデバイスとしてアクセスされるまでオフラインのままです。デバイスグループやボリューム管理ソフトウェアのディスクグループを管理する際は、グループの主ノードであるクラスタから実行する必要があります。

グローバルな名前空間はインストール中に自動的に設定され、Solaris OS の再起動中に自動的に更新されるため、通常、グローバルデバイス名前空間は管理する必要はありません。ただし、グローバルな名前空間を更新する必要がある場合は、任意のクラスタノードから `cldevice populate` コマンドを実行できます。このコマンドにより、その他のすべてのクラスタノードだけでなく、今後クラスタに結合する可能性があるノードでもグローバルな名前空間を更新できます。

Solaris ボリュームマネージャーのグローバルデバイスのアクセス権

グローバルデバイスのアクセス権に加えた変更は、Solaris ボリュームマネージャーおよびディスクデバイスのクラスタのすべてのノードには自動的に伝達されません。グローバルデバイスのアクセス権を変更する場合は、クラスタ内のすべてのノードで手作業でアクセス権を変更する必要があります。たとえば、グローバルデバイス `/dev/global/dsk/d3s0` のアクセス権を 644 に変更する場合は、クラスタ内のすべてのノード上で次のコマンドを実行します。

```
# chmod 644 /dev/global/dsk/d3s0
```

VxVM は、`chmod` コマンドをサポートしません。VxVM でグローバルデバイスのアクセス権を変更する方法については、VxVM の管理者ガイドを参照してください。

グローバルデバイスでの動的再構成

クラスタ内のディスクデバイスやテープデバイス上で動的再構成 (DR) を実行するときには、いくつかの問題を考慮する必要があります。

- Sun Cluster の動的再構成 (DR) のサポートには、Solaris の DR 機能に述べられている必要条件、手順、および制限がすべて適用されます。ただし、オペレーティングシステムの休止操作は除きます。したがって、Sun Cluster ソフトウェアで DR 機能を使用する前に、必ず、Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。

- 主ノードのアクティブなデバイス上では DR 削除操作を実行できません。DR 操作を実行できるのは、主ノードのアクティブでないデバイスか、二次ノードの任意のデバイス上でだけです。
- DR 操作が終了すると、クラスタのデータアクセスが前と同じように続けられます。
- Sun Cluster は、定足数デバイスの使用に影響を与える DR 操作を拒否します。詳細については、199 ページの「定足数デバイスへの動的再構成」を参照してください。



注意-二次ノードに対して DR 操作を行っているときに現在の主ノードに障害が発生すると、クラスタの可用性が損なわれます。新しい二次ノードが提供されるまで、主ノードにはフェイルオーバーする場所がありません。

グローバルデバイス上で DR 操作を実行するには、次の手順をその順番どおりに行います。

表 5-1 作業マップ: ディスクデバイスとテープデバイスでの動的再構成

作業	説明
1. アクティブなデバイスグループに影響するような DR 操作を現在の主ノードに実行する必要がある場合、DR 削除操作をデバイス上で実行する前に、主ノードと二次ノードの切り替えを実行	171 ページの「デバイスグループの主ノードを切り替える」
2. 削除するデバイス上で DR 削除操作を実行	「Solaris 9 on Sun Hardware」コレクションと「Solaris 10 on Sun Hardware」コレクションの『Sun Enterprise 10000 DR Configuration Guide』と『Sun Enterprise 10000 Dynamic Reconfiguration Reference Manual』

Veritas Volume Manager による管理に関する注意事項

- Sun Cluster ソフトウェアで VxVM 名前空間を保持するには、VxVM のディスクグループまたはボリュームの変更を Sun Cluster デバイスグループの構成の変更として登録する必要があります。変更を登録することによって、すべてのクラスタノードを確実に更新できます。名前空間に影響を与える構成の変更の例としては、ボリュームの追加、削除、名前変更があります。また、ボリュームのアクセス権、所有者、グループIDの変更なども名前空間に影響を与えます。

注- ディスクグループを Sun Cluster デバイスグループとしてクラスタに登録した後は、VxVM コマンドを使用して VxVM ディスクグループをインポートまたはデポートしてはいけません。ディスクグループのインポートやデポートが必要な場合は、すべて Sun Cluster ソフトウェアによって処理します。

- 各 VxVM ディスクグループには、クラスタ全体で一意的マイナー番号が与えられています。デフォルトでは、ディスクグループを作成したときに、VxVM によって 1000 の倍数の乱数がディスクグループのベースマイナー番号として選択されます。少数のディスクグループしかないほとんどの構成では、このマイナー番号で十分一意性を保証できます。ただし、新たに作成したディスクグループのマイナー番号が、以前別のクラスタノードにインポートしたディスクグループのマイナー番号と衝突することがあります。この場合、Sun Cluster デバイスグループは登録できません。この問題を解消するには、新しいディスクグループに一意的な値である新しいマイナー番号を付けたうえで、Sun Cluster デバイスグループとして登録してください。
- ミラー化したボリュームを設定している場合、ダーティリージョンログ (DRL) を使用すると、ノードに障害が発生してからボリュームが回復するまでの時間を短縮できます。入出力のスループットが低下することになりますが、DRL の使用を強くお勧めします。
- VxVM は、chmod コマンドをサポートしません。VxVM でグローバルデバイスのアクセス権を変更する方法については、VxVM の管理者ガイドを参照してください。
- Sun Cluster 3.2 ソフトウェアは、同一ノードからの複数パスの VxVM Dynamic Multipathing (DMP) 管理をサポートしていません。
- VxVM を使用して Oracle RAC 用の共有ディスクグループを設定する場合は、『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。Oracle RAC の共有ディスクグループの作成は、ほかのディスクグループの作成とは異なります。Oracle RAC 用の共有ディスクグループをインポートするには、`vx dg -s` を使用する必要があります。Oracle RAC 用の共有ディスクグループをクラスタフレームワークに登録してはいけません。ほかの VxVM ディスクグループを作成する方法については、[141 ページの「ディスクの初期化時に新しいディスクグループを作成 \(Veritas Volume Manager\)」](#)を参照してください。

ストレージベースの複製されたデバイスの管理

ストレージベースの複製によって複製されたデバイスを含めるよう、Sun Cluster デバイスグループを構成することができます。Sun Cluster ソフトウェアは、ストレージベースの複製用に Hitachi TrueCopy および EMC Symmetrix Remote Data Facility ソフトウェアをサポートしています。

Hitachi TrueCopy または EMC Symmetrix Remote Data Facility ソフトウェアでデータを複製する前に、ストレージベースの複製用ソフトウェアのマニュアルによく目を通し、ストレージベースの複製製品と最新のパッチを、使用しているシステムにインストールしておいてください。ストレージベースの複製ソフトウェアのインストールについては、製品のマニュアルを参照してください。

ストレージベースの複製ソフトウェアは、デバイスのペアを複製として構成する際、一方のデバイスを主複製、もう一方のデバイスを二次複製とします。一方のノードのセットに接続されたデバイスが、常に主複製になります。もう一方のノードのセットに接続されたデバイスは、二次複製になります。

Sun Cluster 構成では、複製が属する Sun Cluster デバイスグループが移動されると、常に、主複製が自動的に移動されます。そのため、Sun Cluster 構成下では、主複製を直接移動してはいけません。その代わりに、テイクオーバーは関連する Sun Cluster デバイスグループを移動することによって行うべきです。



注意 - 作成する Sun Cluster デバイスグループ (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) の名前は、複製されたデバイスグループと同じ名前にしてください。

ここでは、次の手順について説明します。

- 101 ページの「Hitachi TrueCopy で複製されたデバイスの管理」
- 113 ページの「EMC Symmetrix Remote Data Facility で複製したデバイスの管理」

Hitachi TrueCopy で複製されたデバイスの管理

次の表に、Hitachi TrueCopy によるストレージベースの複製されたデバイスを設定するために実行する作業を示します。

表 5-2 作業マップ: Hitachi TrueCopy によるストレージベースの複製デバイスの管理

作業	参照先
ストレージデバイスとノードに TrueCopy ソフトウェアをインストールする	Hitachi ストレージデバイスに付属するマニュアルを参照してください。

表 5-2 作業マップ: Hitachi TrueCopy によるストレージベースの複製デバイスの管理 (続き)

作業	参照先
Hitachi 複製グループを構成する	102 ページの「Hitachi TrueCopy 複製グループを構成する」
DID デバイスを構成する	104 ページの「Hitachi TrueCopy を使用して DID デバイスを複製用に構成する」
複製されたグループを登録する	136 ページの「デバイスグループを追加および登録する (Solaris ボリュームマネージャー)」または151 ページの「ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)」
構成を確認する	106 ページの「Hitachi TrueCopy で複製されたグローバルデバイスグループ構成を確認する」

▼ Hitachi TrueCopy 複製グループを構成する

始める前に

まず、主クラスタの共有ディスクに Hitachi TrueCopy デバイスグループを構成します。この構成情報は、Hitachi アレイへのアクセス権を持つ各クラスタノードの `/etc/horcm.conf` ファイルに指定します。 `/etc/horcm.conf` ファイルを構成する方法についての詳細は、『Sun StorEdge SE 9900 V Series Command and Control Interface User and Reference Guide』を参照してください。



注意 - 作成する Sun Cluster デバイスグループ (Solaris Volume Manager、Veritas Volume Manager、ZFS、または raw ディスク) の名前は、複製されたデバイスグループと同じ名前してください。

- 1 ストレージアレイに接続されたすべてのノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 `/etc/services` ファイルに `horcm` エントリを追加します。

```
horcm 9970/udp
```

 新しいエントリのポート番号とプロトコル名を指定します。
- 3 `/etc/horcm.conf` ファイルに **Hitachi TrueCopy** デバイスグループの構成情報を指定します。
 手順については、TrueCopy ソフトウェアに付属するマニュアルを参照してください。
- 4 すべてのノード上で `horcmstart.sh` コマンドを実行することにより、**TrueCopy CCI** デーモンを起動します。

```
# /usr/bin/horcmstart.sh
```

- 5 まだ複製のペアを作成していない場合は、この時点で作成します。
paircreate コマンドを使用して、希望のフェンスレベルを持つ複製のペアを作成します。複製のペアの作成方法の手順については、TrueCopy のマニュアルを参照してください。
- 6 複製されたデバイスを使用して構成された各ノード上で、pairdisplay コマンドを使用することでデータ複製が正しく設定されていることを確認します。ASYNC の fence_level を持つ Hitachi TrueCopy または Hitachi Universal Replicator デバイスグループは、システムの他のデバイスと ctgid を共有することはできません。

```
# pairdisplay -g group-name
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#,P/S,Status,Fence,Seq#,P-LDEV# M
group-name pair1(L) (CL1-C , 0, 9) 54321 58..P-VOL PAIR NEVER ,12345 29 -
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR NEVER ,----- 58 -
```

- 7 すべてのノードが複製グループをマスターできることを確認します。
 - a. どのノードに主複製が含まれ、どのノードに二次複製が含まれているかを判別するには、pairdisplay コマンドを使用します。

```
# pairdisplay -g group-name
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#,P/S,Status,Fence,Seq#,P-LDEV# M
group-name pair1(L) (CL1-C , 0, 9) 54321 58..P-VOL PAIR NEVER ,12345 29 -
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR NEVER ,----- 58 -
```

P-VOL 状態のローカル (L) デバイスを持つノードには主複製が含まれ、S-VOL 状態のローカル (L) デバイスを持つノードには二次複製が含まれます。

- b. 二次ノードをマスターにするには、二次複製が含まれるノード上で horctakeover コマンドを実行します。

```
# horctakeover -g group-name
```

次の手順に進む前に、初期データコピーが完了するのを待ちます。

- c. horctakeover を実行したノードが、この時点で、P-VOL 状態のローカル (L) デバイスを持っていることを確認します。

```
# pairdisplay -g group-name
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#,P/S,Status,Fence,Seq#,P-LDEV# M
group-name pair1(L) (CL1-C , 0, 9) 54321 58..S-VOL PAIR NEVER ,12345 29 -
group-name pair1(R) (CL1-A , 0, 29)12345 29..P-VOL PAIR NEVER ,----- 58 -
```

- d. もとは主複製が含まれていたノード上で horctakeover コマンドを実行します。

```
# horctakeover -g group-name
```

- e. `pairdisplay` コマンドを実行することで、主ノードが元の構成に戻ったことを確認します。

```
# pairdisplay -g group-name
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#,P/S,Status,Fence,Seq#,P-LDEV# M
group-name pair1(L) (CL1-C , 0, 9) 54321 58..P-VOL PAIR NEVER ,12345 29 -
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR NEVER ,----- 58 -
```

次の手順 104 ページの「Hitachi TrueCopy を使用して DID デバイスを複製用に構成する」の手順に従って、複製されたデバイスの構成を続けます。

▼ Hitachi TrueCopy を使用して DID デバイスを複製用に構成する

始める前に 複製されたデバイス用にデバイスグループを構成したあと、複製されたデバイスが使用するデバイス識別子 (DID) ドライバを構成します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 すべてのノード上で `horcm` デーモンが実行中であることを確認します。
実行されていない場合は、次のコマンドでデーモンが起動されます。デーモンがすでに実行されている場合は、システムによりメッセージが表示されます。

```
# /usr/bin/horcmstart.sh
```

- 3 `pairdisplay` コマンドを実行して、どのノードに二次複製が含まれているかを判別します。

```
# pairdisplay -g group-name
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#,P/S,Status,Fence,Seq#,P-LDEV# M
group-name pair1(L) (CL1-C , 0, 9) 54321 58..P-VOL PAIR NEVER ,12345 29 -
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR NEVER ,----- 58 -
```

S-VOL 状態のローカル (L) デバイスを持つノードには二次複製が含まれています。

- 4 (前の手順で判別した) 二次複製を持つノードで、ストレージベースの複製で使用するための DID デバイスを構成します。

このコマンドは、デバイス複製ペアの 2 つの独立した DID インスタンスを、1 つの論理 DID インスタンスに結合します。この 1 つのインスタンスにより、そのデバイスをボリューム管理ソフトウェアで両側から使用できるようになります。



注意-二次複製に複数のノードが接続されている場合、このコマンドは、それらのノードのうちの1つのノード上でのみ実行してください。

```
# cldevice replicate -D primary-replica-nodename -S secondary replica-nodename
```

primary-replica-nodename

主複製が格納されているリモートノードの名前を指定します。

-S

現在のノード以外のソースノードを指定します。

secondary replica-nodename

二次複製が格納されているリモートノードの名前を指定します。

注-デフォルトでは、現在のノードがソースノードです。-s オプションは、別のソースノードを指定するのに使用します。

- 5 DID インスタンスが結合されていることを確認します。

```
# cldevice list -v logical_DID_device
```

- 6 TrueCopy 複製が設定されていることを確認します。

```
# cldevice show logical_DID_device
```

コマンド出力には、TrueCopy が複製タイプであることが示されるはずですが。

- 7 DID の再マッピングによって、すべての複製されたデバイスが正しく結合されなかった場合、手動で個別の複製されたデバイスを結合します。



注意-DID インスタンスを手動で結合する際には、特に注意してください。デバイスの再マッピングが正しくないと、データが破損する可能性があります。

- a. 二次複製が含まれるすべてのノード上で `cldevice combine` コマンドを実行します。

```
# cldevice combine -d destination-instance source-instance
```

-d 主複製に対応するリモート DID インスタンス。

destination-instance

source-instance 二次複製に対応するローカル DID インスタンス。

- b. DID の再マッピングが正しく行われたことを確認します。

```
# cldevice list desination-instance source-instance
```

DID インスタンスの1つは表示されないはずですが。

- 8 すべてのノード上で、すべての結合された DID インスタンスの DID デバイスがアクセス可能であることを確認します。

```
# cldevice list -v
```

次の手順 複製されたデバイスグループの構成を完了するには、次の手順を実行します。

- 136 ページの「デバイスグループを追加および登録する (Solaris ポリリュームマネージャー)」または151 ページの「ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)」

デバイスグループを登録するには、必ず TrueCopy 複製グループと同じ名前を指定します。

- 106 ページの「Hitachi TrueCopy で複製されたグローバルデバイスグループ構成を確認する」

▼ Hitachi TrueCopy で複製されたグローバルデバイスグループ構成を確認する

始める前に グローバルデバイスグループを確認する前に、まずそれを作成します。Solaris Volume Manager、Veritas Volume Manager、ZFS、raw ディスクからデバイスグループを使用することができます。詳細は、次を参照してください。

- 136 ページの「デバイスグループを追加および登録する (Solaris ポリリュームマネージャー)」
- 138 ページの「デバイスグループ (raw ディスク) を追加および登録する」
- 139 ページの「複製デバイスグループ (ZFS) の追加と登録方法」
- 141 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」



注意 - 作成する Sun Cluster デバイスグループ (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) の名前は、複製されたデバイスグループと同じ名前にしてください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 主デバイスグループが、主複製が含まれるノードと同じノードに対応することを確認します。

```
# pairdisplay -g group-name
# cldevicegroup status -n nodename group-name
```
- 2 デバイスグループに複製のプロパティが設定されていることを確認します。

```
# cldevicegroup show -n nodename group-name
```
- 3 デバイスに複製されたプロパティが設定されていることを確認します。

```
# usr/cluster/bin/cldevice status [-s state] [-n node[,?]] [+| [disk-device ]]
```
- 4 試験的にスイッチオーバーを実行して、デバイスグループが正しく構成され、複製がノード間を移動できることを確認します。
 デバイスグループがオフラインのときは、オンラインにします。

```
# cldevicegroup switch -n nodename group-name
```

-n nodename デバイスグループの切り替え先のノード。このノードが新しい主ノードになります。
- 5 次のコマンドの出力を比較することにより、スイッチオーバーが成功したことを確認します。

```
# pairdisplay -g group-name
# cldevicegroup status -n nodename group-name
```

例: Sun Cluster 向けの TrueCopy 複製グループの構成

この例では、クラスタの TrueCopy 複製を設定するのに必要な Sun Cluster 固有の手順を完了します。この例では、すでに次の作業が完了していることが前提となっています。

- Hitachi LUN の設定が完了している
- ストレージデバイスとクラスタノードに TrueCopy ソフトウェアがインストール済みである
- クラスタノード上で複製ペアが構成済みである
 複製ペアの構成手順については、[102 ページの「Hitachi TrueCopy 複製グループを構成する」](#)を参照してください。

この例では、TrueCopy を使用する 3 ノードクラスタを扱います。クラスタは 2 つのリモートサイトにまたがっており、一方のサイトに 2 つのノードがあり、もう一方のサイトに 1 つのノードがあります。各サイトにはそれぞれ Hitachi ストレージデバイスがあります。

次の例に、各ノード上の TrueCopy/etc/horcm.conf 構成ファイルを示します。

例 5-1 ノード 1 上の TrueCopy 構成ファイル

```

HORCM_DEV
#dev_group    dev_name    port#      TargetID    LU#        MU#
VG01          pair1       CL1-A      0           29
VG01          pair2       CL1-A      0           30
VG01          pair3       CL1-A      0           31
HORCM_INST
#dev_group    ip_address  service
VG01          node-3     horcm
    
```

例 5-2 ノード 2 上の TrueCopy 構成ファイル

```

HORCM_DEV
#dev_group    dev_name    port#      TargetID    LU#        MU#
VG01          pair1       CL1-A      0           29
VG01          pair2       CL1-A      0           30
VG01          pair3       CL1-A      0           31
HORCM_INST
#dev_group    ip_address  service
VG01          node-3     horcm
    
```

例 5-3 ノード 3 上の TrueCopy 構成ファイル

```

HORCM_DEV
#dev_group    dev_name    port#      TargetID    LU#        MU#
VG01          pair1       CL1-C      0           09
VG01          pair2       CL1-C      0           10
VG01          pair3       CL1-C      0           11
HORCM_INST
#dev_group    ip_address  service
VG01          node-1     horcm
VG01          node-2     horcm
    
```

上記の例では、3つのLUNが2つのサイト間で複製されます。LUNはすべてVG01という名前の複製グループ内にあります。pairdisplayコマンドを使用すると、この情報が確認され、またノード3には主複製があることが示されます。

例 5-4 ノード 1 上の pairdisplay コマンドの出力

```

# pairdisplay -g VG01
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#.P/S,Status,Fence, Seq#,P-LDEV# M
VG01 pair1(L) (CL1-A , 0, 29)61114 29..S-VOL PAIR NEVER ,----- 58 -
VG01 pair1(R) (CL1-C , 0, 9)20064 58..P-VOL PAIR NEVER ,61114 29 -
VG01 pair2(L) (CL1-A , 0, 30)61114 30..S-VOL PAIR NEVER ,----- 59 -
    
```


例 5-10 使用されているディスクに対応する DID の表示

```
# cldevice list -v
```

```
DID Device Full Device Path
-----
1 node-1:/dev/rdisk/c0t0d0 /dev/did/rdsk/d1
2 node-1:/dev/rdisk/c0t6d0 /dev/did/rdsk/d2
11 node-1:/dev/rdisk/c6t500060E8000000000000E8BA00000020d0 /dev/did/rdsk/d11
11 node-2:/dev/rdisk/c5t500060E8000000000000E8BA00000020d0 /dev/did/rdsk/d11
12 node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Fd0 /dev/did/rdsk/d12
12 node-2:/dev/rdisk/c5t500060E8000000000000E8BA0000001Fd0 /dev/did/rdsk/d12
13 node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Ed0 /dev/did/rdsk/d13
13 node-2:/dev/rdisk/c5t500060E8000000000000E8BA0000001Ed0 /dev/did/rdsk/d13
14 node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Dd0 /dev/did/rdsk/d14
14 node-2:/dev/rdisk/c5t500060E8000000000000E8BA0000001Dd0 /dev/did/rdsk/d14
18 node-3:/dev/rdisk/c0t0d0 /dev/did/rdsk/d18
19 node-3:/dev/rdisk/c0t6d0 /dev/did/rdsk/d19
20 node-3:/dev/rdisk/c5t50060E80000000000004E600000013d0 /dev/did/rdsk/d20
21 node-3:/dev/rdisk/c5t50060E80000000000004E60000003Dd0 /dev/did/rdsk/d21
22 node-3:/dev/rdisk/c5t50060E80000000000004E60000003Cd0 /dev/did/rdsk/d2223
23 node-3:/dev/rdisk/c5t50060E80000000000004E60000003Bd0 /dev/did/rdsk/d23
24 node-3:/dev/rdisk/c5t50060E80000000000004E60000003Ad0 /dev/did/rdsk/d24
```

複製されたデバイスの各ペアの DID インスタンスを結合する場合、`cldevice list` は DID インスタンス 12 を 22、インスタンス 13 を 23、インスタンス 14 を 24 と結合するはずですが、ノード 3 には主複製があるため、ノード 1 またはノード 2 のいずれかから `cldevice -T` コマンドを実行します。インスタンスの結合は常に、二次複製があるノードから行います。このコマンドは 1 つのノードからのみ実行し、両方のノード上では実行しないでください。

次の例に、ノード 1 上でこのコマンドを実行することにより DID インスタンスを結合した場合の出力を示します。

例 5-11 DID インスタンスの結合

```
# cldevice replicate -D node-3
Remapping instances for devices replicated with node-3...
VG01 pair1 L node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Dd0
VG01 pair1 R node-3:/dev/rdisk/c5t50060E80000000000004E600000003Ad0
Combining instance 14 with 24
VG01 pair2 L node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Ed0
VG01 pair2 R node-3:/dev/rdisk/c5t50060E80000000000004E600000003Bd0
Combining instance 13 with 23
VG01 pair3 L node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Fd0
VG01 pair3 R node-3:/dev/rdisk/c5t50060E80000000000004E600000003Cd0
```

例 5-11 DID インスタンスの結合 (続き)

Combining instance 12 with 22

cldevice list の出力を確認すると、両方のサイトの LUN には同じ DID インスタンスがあります。次の例に示すように、同じ DID インスタンスを持っていると、各複製ペアは単一の DID デバイスのように見えます。

例 5-12 結合された DID の表示

```
# cldevice list -v
DID Device Full Device Path
-----
1 node-1:/dev/rdisk/c0t0d0 /dev/did/rdsk/d1
2 node-1:/dev/rdisk/c0t6d0 /dev/did/rdsk/d2
11 node-1:/dev/rdisk/c6t500060E8000000000000EEBA0000020d0 /dev/did/rdsk/d11
11 node-2:/dev/rdisk/c5t500060E8000000000000EEBA0000020d0 /dev/did/rdsk/d11
18 node-3:/dev/rdisk/c0t0d0 /dev/did/rdsk/d18
19 node-3:/dev/rdisk/c0t6d0 /dev/did/rdsk/d19
20 node-3:/dev/rdisk/c5t50060E80000000000004E600000013d0 /dev/did/rdsk/d20
21 node-3:/dev/rdisk/c5t50060E80000000000004E60000003Dd0 /dev/did/rdsk/d21
22 node-1:/dev/rdisk/c6t500060E8000000000000EEBA000001Fd0 /dev/did/rdsk/d1222
22 node-2:/dev/rdisk/c5t500060E8000000000000EEBA000001Fd0 /dev/did/rdsk/d12
22 node-3:/dev/rdisk/c5t50060E8000000000004E60000003Cd0 /dev/did/rdsk/d22
23 node-1:/dev/rdisk/c6t500060E8000000000000EEBA000001Ed0 /dev/did/rdsk/d13
23 node-2:/dev/rdisk/c5t500060E8000000000000EEBA000001Ed0 /dev/did/rdsk/d13
23 node-3:/dev/rdisk/c5t50060E8000000000004E60000003Bd0 /dev/did/rdsk/d23
24 node-1:/dev/rdisk/c6t500060E8000000000000EEBA000001Dd0 /dev/did/rdsk/d24
24 node-2:/dev/rdisk/c5t500060E8000000000000EEBA000001Dd0 /dev/did/rdsk/d24
24 node-3:/dev/rdisk/c5t50060E8000000000004E60000003Ad0 /dev/did/rdsk/d24
```

次に、ボリュームマネージャーデバイスグループを作成します。このコマンドは、主複製があるノード(この例ではノード 3)から実行します。次の例に示すように、デバイスグループには複製グループと同じ名前を指定します。

例 5-13 Solaris ボリュームマネージャーデバイスグループの作成

```
# metaset -s VG01 -ah phys-deneb-3
# metaset -s VG01 -ah phys-deneb-1
# metaset -s VG01 -ah phys-deneb-2
# metaset -s VG01 -a /dev/did/rdsk/d22
# metaset -s VG01 -a /dev/did/rdsk/d23
# metaset -s VG01 -a /dev/did/rdsk/d24
# metaset
Set name = VG01, Set number = 1
```

例 5-13 Solaris ボリュームマネージャーデバイスグループの作成 (続き)

```
Host                Owner
  phys-deneb-3      Yes
  phys-deneb-1
  phys-deneb-2

Drive Dbase
d22  Yes
d23  Yes
d24  Yes
```

この時点で、デバイスグループは使用でき、メタデバイスの作成が可能であり、またデバイスグループは3つのノードのうち任意のノードに移動できます。ただし、スイッチオーバーとフェイルオーバーをより効率的にするため、`cldevicegroup set` を実行して、デバイスグループをクラスタ構成内で複製済みにマークします。

例 5-14 スイッチオーバーとフェイルオーバーの効率化

```
# cldevicegroup sync VG01
# cldevicegroup show VG01
=== Device Groups===
```

```
Device Group Name      VG01
Type:                   SVM
failback:               no
Node List:              phys-deneb-3, phys-deneb-1, phys-deneb-2
preferenced:           yes
numsecondaries:        1
device names:          VG01
Replication type:      truecopy
```

複製グループの構成はこの手順で完了します。構成が正しく行われたことを確認するには、106 ページの「Hitachi TrueCopy で複製されたグローバルデバイスグループ構成を確認する」の手順を実行します。

EMC Symmetrix Remote Data Facility で複製したデバイスの管理

次の表に、EMC Symmetrix Remote Data Facility (SRDF) ストレージベースの複製されたデバイスを設定および管理するために実行する作業を示します。

表 5-3 作業マップ: EMC SRDF ストレージベースの複製されたデバイスの管理

作業	参照先
ストレージデバイスとノードに SRDF ソフトウェアをインストールする	EMC ストレージデバイスに付属するマニュアル。
EMC 複製グループを構成する	114 ページの「EMC SRDF 複製グループを構成する」
DID デバイスを構成する	115 ページの「EMC SRDF を使用して DID デバイスを複製用に構成する」
複製されたグループを登録する	136 ページの「デバイスグループを追加および登録する (Solaris ボリュームマネージャー)」または 151 ページの「ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)」
構成を確認する	117 ページの「EMC SRDF で複製されたグローバルデバイスグループ構成を確認する」
キャンパスクラスタの主ルームが完全に失敗したあとに手動でデータを復元する	124 ページの「主ルームの完全な失敗後に EMC SRDF データを復元する」

▼ EMC SRDF 複製グループを構成する

始める前に

EMC Symmetrix Remote Data Facility (SRDF) 複製グループを構成する前に、すべてのクラスタノードに EMC Solutions Enabler ソフトウェアをインストールしてください。まず、主クラスタの共有ディスクに EMC SRDF デバイスグループを構成します。EMC SRDF デバイスグループを構成する方法についての詳細は、EMC SRDF 製品のマニュアルを参照してください。

EMC SRDF を使用するときは、静的デバイスではなく、動的デバイスを使用します。静的デバイスでは主複製を変更するのに数分かかり、フェイルオーバー時間に影響を与えることがあります。



注意 - 作成する Sun Cluster デバイスグループ (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) の名前は、複製されたデバイスグループと同じ名前にしてください。

- 1 ストレージアレイに接続されたすべてのノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 複製データで構成された各ノードで、シンメトリックデバイス構成を検出します。
この処理には数分かかることがあります。

```
# /usr/symcli/bin/symcfg discover
```

- 3 まだ複製のペアを作成していない場合は、この時点で作成します。
複製のペアを作成するには、`symrdf` コマンドを使用します。複製のペアの作成方法の手順については、SRDF のマニュアルを参照してください。
- 4 複製されたデバイスによって構成された各ノードで、データの複製が正しく設定されていることを確認します。

```
# /usr/symcli/bin/symsg show group-name
```
- 5 デバイスグループのスワップを実行します。
 - a. 主複製と二次複製が同期していることを確認します。

```
# /usr/symcli/bin/symrdf -g group-name verify -synchronized
```
 - b. どのノードに主複製が含まれ、どのノードに二次複製が含まれているかを判別するには、`symsg show` コマンドを使用します。

```
# /usr/symcli/bin/symsg show group-name
```


RDF1 デバイスのノードには主複製が含まれ、RDF2 デバイス状態のノードには二次複製が含まれます。
 - c. 二次複製を有効にします。

```
# /usr/symcli/bin/symrdf -g group-name failover
```
 - d. RDF1 デバイスと RDF2 デバイスをスワップします。

```
# /usr/symcli/bin/symrdf -g group-name swap -refresh R1
```
 - e. 複製ペアを有効にします。

```
# /usr/symcli/bin/symrdf -g group-name establish
```
 - f. 主ノードと二次複製が同期していることを確認します。

```
# /usr/symcli/bin/symrdf -g group-name verify -synchronized
```
- 6 もともと主複製があったノードで上記5つの手順をすべて繰り返します。

次の手順 EMC SRDF で複製されたデバイス用にデバイスグループを構成したあと、複製されたデバイスが使用するデバイス識別子 (DID) ドライバを構成します。

▼ EMC SRDF を使用して DID デバイスを複製用に構成する

この手順では、複製されたデバイスが使用するデバイス識別名 (DID) ドライバを構成します。

始める前に `phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 RDF1 デバイスおよび RDF2 デバイスに対応する DID デバイスを判別します。

```
# /usr/symcli/bin/symsg show group-name
```

注-システムに Solaris デバイスのパッチ全体が表示されない場合は、環境変数 `SYMCLI_FULL_PDEVNAME` を 1 に設定して、`symsg -show` コマンドをもう一度入力します。

- 3 Solaris デバイスに対応する DID デバイスを判別します。

```
# cldevice list -v
```
- 4 一致した DID デバイスのペアごとに、インスタンスを 1 つの複製された DID デバイスにまとめます。RDF2 (二次側) から次のコマンドを実行します。

```
# cldevice combine -t srdf -g replication-device-group \  
-d destination-instance source-instance
```

注-SRDF データ複製デバイスでは、-T オプションはサポートされていません。

<code>-t replication-type</code>	複製タイプを指定します。EMC SRDF の場合、 SRDF を入力します。
<code>-g replication-device-group</code>	<code>symsg show</code> コマンドで表示されるデバイスグループの名前を指定します。
<code>-d destination-instance</code>	RDF1 デバイスに対応する DID インスタンスを指定します。
<code>source-instance</code>	RDF2 デバイスに対応する DID インスタンスを指定します。

注- 誤った DID デバイスを結合した場合は、`scdidadm` コマンドで `-b` オプションを使用して、2つの DID デバイスの結合を取り消します。

```
# scdidadm -b device
```

`-b device` インスタンスを結合したときに `destination_device` に対応していた DID インスタンス。

- 複製デバイスグループの名前が変更された場合は、**Hitachi TrueCopy** および **SRDF** に関する追加の手順が必要です。手順 1-4 が完了したら、該当する追加手順を実行します。

項目	説明
TrueCopy	複製デバイスグループと、対応するグローバルデバイスグループの名前が変更された場合は、 <code>cldevice replicate</code> コマンドを再実行して、複製されたデバイス情報を更新してください。
SRDF	複製デバイスグループと、対応するグローバルデバイスグループの名前が変更された場合は、複製されたデバイス情報を更新してください。それには、まず、 <code>scdidadm -b</code> コマンドを使用して既存の情報を削除します。最後に、 <code>cldevice combine</code> コマンドを使用して、更新された新しいデバイスを作成します。

- DID インスタンスが結合されていることを確認します。

```
# cldevice list -v device
```

- SRDF 複製が設定されていることを確認します。

```
# cldevice show device
```

- すべてのノード上で、すべての結合された DID インスタンスの DID デバイスがアクセス可能であることを確認します。

```
# cldevice list -v
```

次の手順 複製されたデバイスが使用するデバイス識別名 (DID) を構成したら、EMC SRDF で複製されたグローバルデバイスグループ構成を必ず確認してください。

▼ EMC SRDF で複製されたグローバルデバイスグループ構成を確認する

始める前に グローバルデバイスグループを確認する前に、まずそれを作成します。Solaris Volume Manager、Veritas Volume Manager、ZFS、または raw ディスクからデバイスグループを使用することができます。詳細は、次を参照してください。

- 136 ページの「デバイスグループを追加および登録する (Solaris ボリュームマネージャー)」
- 138 ページの「デバイスグループ (raw ディスク) を追加および登録する」
- 139 ページの「複製デバイスグループ (ZFS) の追加と登録方法」
- 141 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」



注意 - 作成する Sun Cluster デバイスグループ (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) の名前は、複製されたデバイスグループと同じ名前にししてください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 主デバイスグループが、主複製が含まれるノードと同じノードに対応することを確認します。

```
# symdg -show group-name
# cldevicegroup status -n nodename group-name
```

- 2 試験的にスイッチオーバーを実行して、デバイスグループが正しく構成され、複製がノード間を移動できることを確認します。

デバイスグループがオフラインのときは、オンラインにします。

```
# cldevicegroup switch -n nodename group-name
```

-n nodename デバイスグループの切り替え先のノード。このノードが新しい主ノードになります。

- 3 次のコマンドの出力を比較することにより、スイッチオーバーが成功したことを確認します。

```
# symdg -show group-name
# cldevicegroup status -n nodename group-name
```

例: Sun Cluster 向けの SRDF 複製グループの構成

この例では、クラスタの SRDF 複製を設定するのに必要な Sun Cluster 固有の手順を完了します。この例では、すでに次の作業が完了していることが前提となっています。

- アレイ間の複製の LUN のペア作成が完了している。

- ストレージデバイスとクラスタノードに SRDF ソフトウェアがインストール済みである。

この例には4ノードクラスタが含まれ、そのうちの2ノードは1つのシンメトリックに接続され、ほかの2ノードはもう1つのシンメトリックに接続されています。SRDF デバイスグループは、dg1 と呼ばれます。

例 5-15 複製ペアの作成

すべてのノードで次のコマンドを実行します。

```
# symcfg discover
! This operation might take up to a few minutes.
# symdev list pd
```

Symmetrix ID: 000187990182

Device Name		Directors			Device		
Sym	Physical	SA	:P DA	:IT Config	Attribute	Sts	Cap (MB)
0067	c5t600604800001879901*	16D:0	02A:C1	RDF2+Mir	N/Grp'd	RW	4315
0068	c5t600604800001879901*	16D:0	16B:C0	RDF1+Mir	N/Grp'd	RW	4315
0069	c5t600604800001879901*	16D:0	01A:C0	RDF1+Mir	N/Grp'd	RW	4315
...							

RDF1 側のすべてのノードで、次のように入力します。

```
# symdg -type RDF1 create dg1
# symld -g dg1 add dev 0067
```

RDF2 側のすべてのノードで、次のように入力します。

```
# symdg -type RDF2 create dg1
# symld -g dg1 add dev 0067
```

例 5-16 データ複製設定の確認

クラスタ内の1つのノードから、次のように入力します。

```
# symdg show dg1

Group Name: dg1
```

例 5-16 データ複製設定の確認 (続き)

```

Group Type                : RDF1      (RDFA)
Device Group in GNS       : No
Valid                     : Yes
Symmetrix ID              : 000187900023
Group Creation Time       : Thu Sep 13 13:21:15 2007
Vendor ID                 : EMC Corp
Application ID            : SYMCLI

```

```

Number of STD Devices in Group : 1
Number of Associated GK's      : 0
Number of Locally-associated BCV's : 0
Number of Locally-associated VDEV's : 0
Number of Remotely-associated BCV's (STD RDF): 0
Number of Remotely-associated BCV's (BCV RDF): 0
Number of Remotely-assoc'd RBCV's (RBCV RDF) : 0

```

Standard (STD) Devices (1):

```

{
-----
LdevName          PdevName          Sym          Cap
                   Dev Att. Sts          (MB)
-----
DEV001            /dev/rdisk/c5t600604800018790002353594D303637d0s2 0067      RW      4315
}

```

Device Group RDF Information

```

...
# symrdf -g dg1 establish

```

Execute an RDF 'Incremental Establish' operation for device group 'dg1' (y/[n]) ? y

An RDF 'Incremental Establish' operation execution is in progress for device group 'dg1'. Please wait...

```

Write Disable device(s) on RA at target (R2).....Done.
Suspend RDF link(s).....Done.
Mark target (R2) devices to refresh from source (R1).....Started.
Device: 0067 ..... Marked.
Mark target (R2) devices to refresh from source (R1).....Done.
Merge device track tables between source and target.....Started.
Device: 0067 ..... Merged.
Merge device track tables between source and target.....Done.
Resume RDF link(s).....Started.
Resume RDF link(s).....Done.

```

例 5-16 データ複製設定の確認 (続き)

The RDF 'Incremental Establish' operation successfully initiated for device group 'dg1'.

```
#
# symrdf -g dg1 query
```

```
Device Group (DG) Name      : dg1
DG's Type                  : RDF2
DG's Symmetrix ID         : 000187990182
```

Target (R2) View				Source (R1) View				MODES
Standard	Logical	Device	Dev	ST	LI	ST		
A	T R1 Inv	E	Tracks	N	K	T R1 Inv	R2 Inv	RDF Pair
				S Dev	E	Tracks	Tracks	MDA STATE
DEV001	0067	WD	0	0 RW	0067	RW	0	0 S.. Synchronized
Total				Total				
MB(s)				0.0	0.0	0.0		0.0

Legend for MODES:

M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
D(omino) : X = Enabled, . = Disabled
A(daptive Copy) : D = Disk Mode, W = WP Mode, . = ACp off

```
#
```

例 5-17 使用されているディスクに対応する DID の表示

RDF1 側と RDF2 側で同じ手順を実行します。

dymdg show dg コマンドの出力の PdevName フィールドの下に DID を表示できます。

RDF1 側で次のように入力します。

```
# symdg show dg1
```

```
Group Name: dg1
```

例 5-17 使用されているディスクに対応する DID の表示 (続き)

```

Group Type                               : RDF1      (RDFA)
...
Standard (STD) Devices (1):
{
-----
LdevName          PdevName          Sym          Cap
Dev  Att.  Sts          (MB)
-----
DEV001            /dev/rdisk/c5t6006048000018790002353594D303637d0s2  0067      RW      4315
}

```

Device Group RDF Information

...

対応する DID を取得するには、次のように入力します。

```

# scdidadm -L | grep c5t6006048000018790002353594D303637d0
217      pmoney1:/dev/rdisk/c5t6006048000018790002353594D303637d0 /dev/did/rdisk/d217
217      pmoney2:/dev/rdisk/c5t6006048000018790002353594D303637d0 /dev/did/rdisk/d217
#

```

対応する DID の一覧を表示するには、次のように入力します。

```
# cldevice show d217
```

```
=== DID Device Instances ===
```

```

DID Device Name:                /dev/did/rdisk/d217
Full Device Path:                pmoney2:/dev/rdisk/c5t6006048000018790002353594D303637d0
Full Device Path:                pmoney1:/dev/rdisk/c5t6006048000018790002353594D303637d0
Replication:                     none
default_fencing:                 global
#

```

#

RDF2 側で次のように入力します。

dymdg show dg コマンドの出力の PdevName フィールドの下に DID を表示できます。

```
# symdg show dg1
```

```
Group Name:  dg1
```

```
Group Type                               : RDF2      (RDFA)
```

例 5-17 使用されているディスクに対応する DID の表示 (続き)

```

...
Standard (STD) Devices (1):
{
-----
LdevName          PdevName          Sym          Cap
Dev  Att.  Sts          (MB)
-----
DEV001            /dev/rdsk/c5t6006048000018799018253594D303637d0s2  0067      WD      4315
}

Device Group RDF Information
...

```

対応する DID を取得するには、次のように入力します。

```

# scdidadm -L | grep c5t6006048000018799018253594D303637d0
108      pmoney4:/dev/rdsk/c5t6006048000018799018253594D303637d0 /dev/did/rdsk/d108
108      pmoney3:/dev/rdsk/c5t6006048000018799018253594D303637d0 /dev/did/rdsk/d108
#

```

対応する DID の一覧を表示するには、次のように入力します。

```

# cldevice show d108

=== DID Device Instances ===

DID Device Name:      /dev/did/rdsk/d108
Full Device Path:     pmoney3:/dev/rdsk/c5t6006048000018799018253594D303637d0
Full Device Path:     pmoney4:/dev/rdsk/c5t6006048000018799018253594D303637d0
Replication:          none
default_fencing:      global

#

```

例 5-18 DID インスタンスの結合

RDF2 側から次のように入力します。

```

# cldevice combine -t srdf -g dg1 -d d217 d108
#

```

例 5-19 結合された DID の表示

クラスタ内の任意のノードから、次のように入力します。

例 5-19 結合された DID の表示 (続き)

```
# cldevice show d217 d108
cldevice: (C727402) Could not locate instance "108".

=== DID Device Instances ===

DID Device Name:                /dev/did/rdsk/d217
Full Device Path:                pmoney1:/dev/rdsk/c5t6006048000018790002353594D303637d0
Full Device Path:                pmoney2:/dev/rdsk/c5t6006048000018790002353594D303637d0
Full Device Path:                pmoney4:/dev/rdsk/c5t6006048000018799018253594D303637d0
Full Device Path:                pmoney3:/dev/rdsk/c5t6006048000018799018253594D303637d0
Replication:                     srdf
default_fencing:                 global

#
```

▼ 主ルールの完全な失敗後に EMC SRDF データを復元する

この手順は、キャンパスクラスタの主ルールが完全に失敗し、主ルールが二次ルールにフェイルオーバーして、主ルールがオンラインに戻ったとき、データ復元を実行します。キャンパスクラスタの主ルールは、主ノードとストレージサイトです。ルールの完全な失敗には、そのルールのホストとストレージ両方の不具合が含まれます。主ルールが失敗した場合、Sun Cluster は自動的に二次ルールにフェイルオーバーし、二次ルールのストレージデバイスを読み書き可能にし、対応するデバイスグループとリソースグループのフェイルオーバーを有効にします。

主ルールがオンラインに戻ったら、二次ルールに書き込まれた SRDF デバイスグループからデータを手動で復元し、データを同期することができます。この手順では、元の二次ルール (この手順では、二次ルールに *phys-campus-2* を使用) からのデータを元の主ルール (*phys-campus-1*) に同期して、SRDF デバイスグループを復元します。また、この手順では、SRDF デバイスグループタイプを、*phys-campus-2* では RDF1 に、*phys-campus-1* では RDF2 に変更します。

始める前に 手動でフェイルオーバーを実行する前に、EMC 複製グループおよび DID デバイスを構成し、EMC 複製グループを登録する必要があります。Solaris ボリュームマネージャー デバイスグループの作成についての詳細は、[136 ページの「デバイスグループを追加および登録する \(Solaris ボリュームマネージャー\)」](#)を参照してください。Veritas Volume Manager デバイスグループの作成については、[147 ページの「ディスクをカプセル化する際に新しいディスクグループを作成する \(VERITAS Volume Manager\)」](#)を参照してください。

注-これらの説明は、主ルームが完全にフェイルオーバーしてからオンラインに戻ったあとに、SRDFデータを手動で復元するための1つの方法を示しています。他の方法については、EMCマニュアルを確認してください。

これらのステップを実行するには、キャンパスクラスタの主ルームにログインしてください。下記の手順では、*dg1*はSRDFデバイスグループ名です。失敗した時点では、この手順の主ルームは*phys-campus-1*で、二次ルームは*phys-campus-2*です。

- 1 キャンパスクラスタの主ルームにログインし、スーパーユーザーになるか、`solaris.cluster.modify RBAC`権限を提供する役割になります。
- 2 主ルームから、`symrdf`コマンドを使用してRDFデバイスの複製ステータスに対するクエリーを実行し、これらのデバイスに関する情報を表示します。

```
phys-campus-1# symrdf -g dg1 query
```

ヒント-`split`状態にあるデバイスグループは同期されません。

- 3 RDFペア状態が`split`で、デバイスグループタイプがRDF1の場合、SRDFデバイスグループのフェイルオーバーを強制実行します。

```
phys-campus-1# symrdf -g dg1 -force failover
```
- 4 RDFデバイスのステータスを表示します。

```
phys-campus-1# symrdf -g dg1 query
```
- 5 フェイルオーバー後、フェイルオーバーしたRDFデバイスのデータをスワップすることができます。

```
phys-campus-1# symrdf -g dg1 swap
```
- 6 RDFデバイスに関する状態および他の情報を検証します。

```
phys-campus-1# symrdf -g dg1 query
```
- 7 主ルームのSRDFデバイスグループを確立します。

```
phys-campus-1# symrdf -g dg1 establish
```
- 8 デバイスグループが同期状態であり、デバイスグループタイプがRDF2であることを確認します。

```
phys-campus-1# symrdf -g dg1 query
```

例 5-20 主サイトフェイルオーバー後に EMC SRDF データを手動で復元する

この例では、キャンパスクラスタの主ルームがフェイルオーバーし、二次ルームが代わりにデータを記録するようになり、主ルームがオンラインに戻ったあとで、EMC SRDF データを手動で復元するために必要な Sun Cluster 固有のステップが提供されています。例では、SRDF デバイスグループは *dg1* と呼ばれ、標準論理デバイスは DEV001 です。失敗した時点では、主ルームは *phys-campus-1* で、二次ルームは *phys-campus-2* です。キャンパスクラスタの主ルーム *phys-campus-1* からステップを実行します。

```
phys-campus-1# symrdf -g dg1 query | grep DEV
DEV001 0012RW 0 0NR 0012RW 2031 0 S.. Split

phys-campus-1# symdg list | grep RDF
dg1 RDF1 Yes 00187990182 1 0 0 0 0

phys-campus-1# symrdf -g dg1 -force failover
...

phys-campus-1# symrdf -g dg1 query | grep DEV
DEV001 0012 WD 0 0 NR 0012 RW 2031 0 S.. Failed Over

phys-campus-1# symdg list | grep RDF
dg1 RDF1 Yes 00187990182 1 0 0 0 0

phys-campus-1# symrdf -g dg1 swap
...

phys-campus-1# symrdf -g dg1 query | grep DEV
DEV001 0012 WD 0 0 NR 0012 RW 0 2031 S.. Suspended

phys-campus-1# symdg list | grep RDF
dg1 RDF2 Yes 000187990182 1 0 0 0 0

phys-campus-1# symrdf -g dg1 establish
...

phys-campus-1# symrdf -g dg1 query | grep DEV
DEV001 0012 WD 0 0 RW 0012 RW 0 0 S.. Synchronized

phys-campus-1# symdg list | grep RDF
dg1 RDF2 Yes 000187990182 1 0 0 0 0
```

クラスタファイルシステムの管理の概要

クラスタファイルシステムを管理するのに特別な Sun Cluster コマンドは必要ありません。クラスタファイルシステムを管理するには、ほかの Solaris ファイルシステムを管理するときと同じように、Solaris の標準のファイルシステムコマンド (mount、newfs など) を使用します。クラスタファイルシステムをマウントするには、mount コマンドに -g オプションを指定します。また、起動時に自動的にマウントすることもできます。クラスタファイルシステムは、グローバルクラスタ内の投票ノードからのみ認識できます。クラスタファイルシステムのデータを非投票ノードからアクセス可能にする必要がある場合は、zoneadm(1M) または HAStoragePlus を使用して非投票ノードにデータをマッピングします。

注-クラスタファイルシステムがファイルを読み取るとき、ファイルシステムはファイルのアクセス時間を更新しません。

クラスタファイルシステムの制限事項

次に、クラスタファイルシステム管理に適用される制限事項を示します。

- 空ではないディレクトリ上では unlink(1M) コマンドはサポートされません。
- lockfs -d コマンドはサポートされません。回避方法として、lockfs -n を使用してください。
- クラスタファイルシステムをマウントし直すとき、directio マウントオプションは指定できません。
- directio ioctl を使用して、directio マウントオプションを単一ファイルに設定することはできません。
- ルートファイルシステムの ZFS がサポートされていますが、重要な例外が1つあります。グローバルデバイスファイルシステムにブートディスクの専用パーティションを使用する場合、ファイルシステムとして UFS のみを使用してください。グローバルデバイスの名前領域には、UFS ファイルシステム上で実行されているプロキシファイルシステム (PxFS) が必要です。ただし、グローバルデバイス名前空間の UFS ファイルシステムは、/var または /home などの、ルート (/) ファイルシステムと他のルートファイルシステム用の ZFS ファイルシステムと共存することができます。また、グローバルデバイス名前空間をホストするために lofi デバイスを使用する場合、ルートファイルシステムに対する ZFS の使用に制限はありません。

VxFS サポートについてのガイドライン

次の VxFS 機能は、Sun Cluster 3.2 クラスタファイルシステムではサポートされていません。ただし、ローカルのファイルシステムではサポートされます。

- クイック入出力
- スナップショット
- 記憶装置チェックポイント
- VxFS 固有のマウントオプション:
 - convosync (Convert O_SYNC)
 - mincache
 - qlog, delaylog, tmplog
- Veritas クラスタファイルシステム (requires VxVM クラスタ機能および Veritas クラスタサーバーが必要)。VxVM クラスタ機能は、x86 ベースのシステム上ではサポートされていません。

キャッシュアドバイザーは使用可能、効果が認められるのは特定のノードのみ

クラスタファイルシステムでサポートされる VxFS のそのほかの機能とオプションは、すべて Sun Cluster 3.2 ソフトウェアでサポートされます。クラスタ構成でサポートされる VxFS オプションの詳細については、VxFS マニュアルを参照してください。

VxFS を使用して高可用性クラスタファイルシステムを作成するための次のガイドラインは、Sun Cluster 3.2 構成に固有のものです。

- VxFS マニュアルの手順に従って VxFS ファイルシステムを作成します。
- 主ノードから VxFS ファイルシステムをマウントおよびマウント解除します。主ノードは、VxFS ファイルシステムが存在するディスクをマスターします。二次ノードから VxFS ファイルシステムをマウントまたはマウント解除すると、失敗することがあります。
- VxFS の管理コマンドはすべて、VxFS クラスタファイルシステムの主ノードから実行します。

VxFS クラスタファイルシステムを管理するための次のガイドラインは、Sun Cluster 3.2 ソフトウェアに固有のものではありません。しかし、これらのガイドラインは UFS クラスタファイルシステムを管理する方法とは異なります。

- VxFS クラスタファイルシステム上にあるファイルは、クラスタ内にある任意のノードから管理できます。例外は `ioctls` で、`ioctls` だけは主ノードから実行する必要があります。管理コマンドが `ioctl` に関連するかどうか分からない場合は、主ノードからコマンドを発行します。

- VxFS クラスタファイルシステムが二次ノードにフェイルオーバーされると、フェイルオーバー時に実行中であったすべての標準システム呼び出し操作は、新しい主ノードで透過的に再実行されます。ただし、フェイルオーバー時に実行していた `ioctl` 関連の操作は失敗します。VxFS クラスタファイルシステムのフェイルオーバーのあとで、このクラスタファイルシステムの状態を調べます。フェイルオーバー以前に古い主ノードから実行された管理コマンドには修正処理が必要になることもあります。詳細については、VxFS のマニュアルを参照してください。

デバイスグループの管理

クラスタの要件の変化により、クラスタ上のデバイスグループの追加、削除、または変更が必要となる場合があります。Sun Cluster には、このような変更を行うために使用できる、`clsetup` と呼ばれる対話型インタフェースがあります。`clsetup` は `cluster` コマンドを生成します。生成されるコマンドについては、各説明の後にある例を参照してください。次の表に、デバイスグループを管理するための作業を示し、またこの節の適切な手順へのリンクを示します。



注意 - ほかのノードが有効なクラスタメンバーであり、それらのノードの少なくとも 1 つがディスクセットを持つ場合は、クラスタの外側で起動されるクラスタノードで `metaset -s setname -f -t` を実行しないでください。

注 - Sun Cluster ソフトウェアは、クラスタ内のディスクデバイスやテープデバイスごとに、`raw` ディスクデバイスグループを自動的に作成します。ただし、クラスタデバイスグループはグローバルデバイスとしてアクセスされるまでオフラインのままです。

表 5-4 作業マップ: デバイスグループの管理

作業	参照先
<code>cldevice populate</code> コマンドを使用することにより、再構成の再起動を行わずにグローバルデバイス名前空間を更新する	132 ページの「グローバルデバイス名前空間を更新する」
既存のグローバルデバイス名前空間を移動する	133 ページの「専用パーティションから <code>lofi</code> デバイスにグローバルデバイス名前空間を移行する」 134 ページの「 <code>lofi</code> デバイスから専用パーティションにグローバルデバイス名前空間を移行する」

表 5-4 作業マップ: デバイスグループの管理 (続き)

作業	参照先
metaset コマンドを使用することにより、Solaris ボリュームマネージャー ディスクセットを追加し、それらをデバイスグループとして登録する	136 ページの「デバイスグループを追加および登録する (Solaris ボリュームマネージャー)」
cldevicegroup コマンドを使用することにより、raw ディスクデバイスグループを追加および登録する	138 ページの「デバイスグループ (raw ディスク) を追加および登録する」
cldevicegroup コマンドを使用することにより、ZFS に名前付きデバイスグループを追加する	139 ページの「複製デバイスグループ (ZFS) の追加と登録方法」
推奨の方法を使用することにより、デバイスグループとして新規ディスクグループを追加および登録する	141 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」
metaset コマンドおよび metaclear コマンドを使用することにより、構成から Solaris ボリュームマネージャー デバイスグループを削除する	142 ページの「デバイスグループを削除して登録を解除する (Solaris ボリュームマネージャー)」
cldevicegroup、metaset、および clsetup コマンドを使用することにより、すべてのデバイスグループからノードを削除する	142 ページの「すべてのデバイスグループからノードを削除する」
metaset コマンドを使用することにより、Solaris ボリュームマネージャー デバイスグループからノードを削除する	143 ページの「デバイスグループからノードを削除する (Solaris ボリュームマネージャー)」

表 5-4 作業マップ: デバイスグループの管理 (続き)

作業	参照先
VxVM コマンドおよび <code>clsetup</code> を使用することにより、Veritas Volume Manager ディスクグループをデバイスグループとして追加する	<p>141 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」</p> <p>147 ページの「ディスクをカプセル化する際に新しいディスクグループを作成する (VERITAS Volume Manager)」</p> <p>149 ページの「新しいボリュームを既存のデバイスグループに追加する (Veritas Volume Manager)」</p> <p>150 ページの「既存のディスクグループをデバイスグループに変換する (Veritas Volume Manager)」</p> <p>150 ページの「デバイスグループに新しいマイナー番号を割り当てる (VERITAS Volume Manager)」</p> <p>151 ページの「ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)」</p> <p>155 ページの「ローカルディスクグループをデバイスグループに変換する (VxVM)」</p> <p>156 ページの「デバイスグループをローカルディスクグループに変換する (VxVM)」</p> <p>154 ページの「ディスクグループの構成変更を登録する (VERITAS Volume Manager)」</p>
<code>clsetup</code> (<code>cldevicegroup</code> コマンドを生成する) コマンドを使用することにより、構成から Veritas Volume Manager デバイスグループを削除する	<p>157 ページの「デバイスグループからボリュームを削除する (Veritas Volume Manager)」</p> <p>158 ページの「デバイスグループを削除して登録を解除する (Veritas Volume Manager)」</p>
<code>clsetup</code> を使用して <code>cldevicegroup</code> を生成することにより、ノードを Veritas Volume Manager デバイスグループに追加する	159 ページの「デバイスグループにノードを追加する (VERITAS Volume Manager)」
<code>clsetup</code> を使用して <code>cldevicegroup</code> を生成することにより、Veritas Volume Manager デバイスグループからノードを削除する	161 ページの「デバイスグループからノードを削除する (Veritas Volume Manager)」
<code>cldevicegroup</code> コマンドを使用することにより、 <code>raw</code> ディスクデバイスグループからノードを削除する	163 ページの「 <code>raw</code> ディスクデバイスグループからノードを削除する」
<code>clsetup</code> を使用して <code>cldevicegroup</code> を生成することにより、デバイスグループのプロパティを変更する	165 ページの「デバイスグループのプロパティを変更する」

表 5-4 作業マップ: デバイスグループの管理 (続き)

作業	参照先
<code>cldevicegroup show</code> コマンドを使用することにより、デバイスグループとプロパティを表示する	170 ページの「デバイスグループ構成の一覧を表示する」
<code>clsetup</code> を使用して <code>cldevicegroup</code> を生成することにより、デバイスグループの二次ノードの希望数を変更する	166 ページの「デバイスグループの二次ノードの希望数を設定する」
<code>cldevicegroup switch</code> コマンドを使用することにより、デバイスグループの主ノードを切り替える	171 ページの「デバイスグループの主ノードを切り替える」
<code>metaset</code> コマンドまたは <code>vxdg</code> コマンドを使用することにより、デバイスグループを保守状態にする	172 ページの「デバイスグループを保守状態にする」

▼ グローバルデバイス名前空間を更新する

新しいグローバルデバイスを追加するときに、`cldevice populate` コマンドを実行して手動でグローバルデバイス名前空間を更新します。

注- コマンドを実行するノードがクラスタのメンバーでない場合は、`cldevice populate` コマンドを実行しても無効です。また、`/global/.devices/node@nodeID` ファイルシステムがマウントされていない場合も、コマンドは無効になります。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 クラスタの各ノードで `devfsadm(1M)` コマンドを実行します。
このコマンドは、すべてのノードで同時に実行できます。
- 3 名前空間を再構成します。

```
# cldevice populate
```
- 4 各ノードで、ディスクセットを作成する前に、`cldevice populate` コマンドが完了していることを確認してください。
ノードの1つで `cldevice` コマンドを実行すると、このコマンドはリモートから自分自身をすべてのノードで呼び出します。`cldevice populate` コマンドが処理を終了したかどうかを確認するには、クラスタの各ノードで次のコマンドを実行します。

```
# ps -ef | grep scgdevs
```

例 5-21 グローバルデバイス名前空間を更新する

次の例に、`cldevice populate` コマンドを正しく実行することにより生成される出力を示します。

```
# devfsadm
cldevice populate
Configuring the /dev/global directory (global devices)...
obtaining access to all attached disks
reservation program successfully exiting
# ps -ef | grep scgdevs
```

グローバルデバイス名前空間を移行する

専用パーティションでグローバルデバイス名前空間を作成するのではなく、ループバックファイルインタフェース (lofi) デバイス上に名前空間を作成することができます。Solaris 10 OS があらかじめインストールされているシステムに Sun Cluster ソフトウェアをインストールする場合、この機能が役立ちます。

注- ルートファイルシステムに ZFS がサポートされていますが、重要な例外が 1 つあります。グローバルデバイスファイルシステムにブートディスクの専用パーティションを使用する場合、ファイルシステムとして UFS のみを使用してください。グローバルデバイス名前空間には、UFS ファイルシステムで動作しているプロキシファイルシステム (PxFS) が必要です。ただし、`/var` または `/home` などの、グローバルデバイス名前空間の UFS ファイルシステムは、ルート (`/`) ファイルシステムや他のルートファイルシステム用の ZFS ファイルシステムと共存することができます。また、lofi デバイスを使用してグローバルデバイス名前空間をホストする場合、ルートファイルシステムに対する ZFS の使用に関する制限はありません。

次の手順は、既存のグローバルデバイス名前空間を専用パーティションから lofi デバイスまたはその逆に移行する方法を説明しています。

- 133 ページの「専用パーティションから lofi デバイスにグローバルデバイス名前空間を移行する」
- 134 ページの「lofi デバイスから専用パーティションにグローバルデバイス名前空間を移行する」

▼ 専用パーティションから lofi デバイスにグローバルデバイス名前空間を移行する

- 1 名前空間の場所を変更したいグローバルクラスタ投票ノードのスーパーユーザーになります。

- 2 `/.globaldevices` という名前のファイルがノードに存在しないことを確認します。ファイルが存在する場合は、削除します。
- 3 `lofi` デバイスを作成します。


```
# mkfile 100m /.globaldevices# lofiadm -a /.globaldevices# \
LOFI_DEV='lofiadm /.globaldevices'# newfs 'echo ${LOFI_DEV} | \
sed -e 's/lofi/rlofi/g'' < /dev/null# lofiadm -d /.globaldevices
```
- 4 `/etc/vfstab` ファイルで、グローバルデバイス名前空間エントリをコメントアウトします。このエントリには、`/global/.devices/node@nodeID` で始まるマウントパスがあります。
- 5 グローバルデバイスパーティション `/global/.devices/node@nodeID` のマウントを解除します。
- 6 `globaldevices` および `scmountdev` SMF サービスを無効にし再度有効にします。


```
# svcadm disable globaldevices# svcadm disable scmountdev# \
svcadm enable scmountdev# svcadm enable globaldevices
```

`lofi` デバイスは現在 `/.globaldevices` に作成され、グローバルデバイスファイルシステムとしてマウントされています。
- 7 パーティションから `lofi` デバイスへ移行したいグローバルデバイス名前空間のある他のノードでもこのステップを繰り返します。
- 8 1つのノードから、グローバルデバイス名前空間を生成します。


```
# /usr/cluster/bin/cldevice populate
```

各ノードで、コマンドが処理を完了したことを確認してから、クラスタに対する以降の操作を実行してください。

```
# ps -ef | grep scgdevs
```

グローバルデバイス名前空間は、現在 `lofi` デバイスにあります。

▼ `lofi` デバイスから専用パーティションにグローバルデバイス名前空間を移行する

- 1 名前空間の場所を変更したいグローバルクラスタ投票ノードのスーパーユーザーになります。
- 2 ノードのローカルディスクで、次の要件を満たす新しいパーティションを作成します。

- サイズが512Mバイト以上
 - UFS ファイルシステムの使用
- 3 グローバルデバイスファイルシステムとしてマウントする新しいパーティションに、`/etc/vfstab` ファイルへのエントリを追加します。
- 現在のノードのノードIDを指定します。
`# /usr/sbin/clinfo -nnode ID`
 - 次の形式を使用して、`/etc/vfstab` ファイルに新しいエントリを作成します。
`blockdevice rawdevice /global/.devices/node@nodeID ufs 2 no global`
- たとえば、使用するパーティションが `/dev/did/rdisk/d5s3` の場合、`/etc/vfstab` ファイルに追加する新しいエントリは、`/dev/did/dsk/d5s3 /dev/did/rdisk/d5s3 /global/.devices/node@3 ufs 2 no global` となります。
- 4 グローバルデバイスパーティション `/global/.devices/node@nodeID` のマウントを解除します。
- 5 `/.globaldevices` ファイルに関連付けられた `lofi` デバイスを削除します。
`# lofiadm -d /.globaldevices`
- 6 `/.globaldevices` ファイルを削除します。
`# rm /.globaldevices`
- 7 `globaldevices` および `scmountdev` SMF サービスを無効にし再度有効にします。
`# svcadm disable globaldevices# svcadm disable scmountdev# \`
`svcadm enable scmountdev# svcadm enable globaldevices`
- パーティションは現在グローバルデバイス名前空間ファイルシステムとしてマウントされています。
- 8 `lofi` デバイスからパーティションへ移行したいグローバルデバイス名前空間のある他のノードでもこのステップを繰り返します。
- 9 クラスタの1つのノードから、`cldevice populate` コマンドを実行してグローバルデバイス名前空間を生成します。
`# /usr/cluster/bin/cldevice populate`
- クラスタのすべてのノードで処理が完了したことを確認してから、ノードに対する作業を実行してください。
`# ps -ef | grep scgdevs`
- グローバルデバイス名前空間は、現在専用パーティションにあります。

デバイスグループを追加および登録する

Solaris Volume Manager、ZFS、Veritas Volume Manager、または raw ディスクのデバイスグループを追加および登録することができます。

▼ デバイスグループを追加および登録する (Solaris ボリュームマネージャー)

`metaset` コマンドを使用して Solaris ボリュームマネージャー ディスクセットを作成し、そのディスクセットを Sun Cluster デバイスグループとして登録します。デバイスグループには、ディスクセットを登録するときにディスクセットに割り当てた名前が自動的に割り当てられます。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。



注意 - 作成する Sun Cluster デバイスグループ (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) の名前は、複製されたデバイスグループと同じ名前にしてください。

- 1 ディスクセットを作成するディスクに接続されたノードのいずれかで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 **SPARC: Solaris 9 のみ:** 構成に必要な **Solstice DiskSuite** メタデバイスや **Solaris** ボリュームマネージャー ボリュームの名前の数を算出し、各ノード上の `/kernel/drv/md.conf` ファイルを変更します。Solaris 10 を実行している場合、この手順は不要です。
『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の「メタデバイス名またはボリューム名とディスクセットの数を算出する」を参照してください。
- 3 **Solaris** ボリュームマネージャー ディスクセットを追加し、このディスクセットをデバイスグループとして **Sun Cluster** に登録します。複数所有者のディスクグループを作成するには、`-M` オプションを使用します。

```
# metaset -s diskset -a -M -h nodelist
```

`-s diskset` 作成するディスクセットを指定します。

- a -h *nodelist* ディスクセットをマスターできるノードの一覧を追加します。
- M ディスクグループを複数所有者として指定します。

注 `-metaset` コマンドを実行して設定した Solstice DiskSuite/Solaris Volume Manager デバイスグループは、そのデバイスグループに含まれるノード数に関わらず、デフォルトで二次ノードになります。デバイスグループが作成されたあと、`clsetup` ユーティリティーを使用することで、二次ノードの希望数を変更できます。ディスクのフェイルオーバーの詳細については、[166 ページの「デバイスグループの二次ノードの希望数を設定する」](#)を参照してください。

- 4 複製されたデバイスグループを構成している場合は、そのデバイスグループの複製プロパティを設定します。

```
# cldevicegroup sync devicegroup
```

- 5 デバイスグループが追加されたことを確認します。
デバイスグループ名は `metaset` に指定したディスクセット名と一致します。

```
# cldevicegroup list
```

- 6 DID マッピングの一覧を表示します。

```
# cldevice show | grep Device
```

- ディスクセットをマスターする(またはマスターする可能性がある)クラスタノードによって共有されているドライブを選択します。
- ディスクセットにドライブを追加する際は、`/dev/did/rdsk/dN` の形式の完全な DID デバイス名を使用してください。

次の例では、DID デバイス `/dev/did/rdsk/d3` のエントリは、ドライブが `phys-schost-1` および `phys-schost-2` によって共有されていることを示しています。

```
=== DID Device Instances ===
DID Device Name:                      /dev/did/rdsk/d1
Full Device Path:                      phys-schost-1:/dev/rdsk/c0t0d0
DID Device Name:                      /dev/did/rdsk/d2
Full Device Path:                      phys-schost-1:/dev/rdsk/c0t6d0
DID Device Name:                      /dev/did/rdsk/d3
Full Device Path:                      phys-schost-1:/dev/rdsk/c1t1d0
Full Device Path:                      phys-schost-2:/dev/rdsk/c1t1d0
...
```

- 7 ディスクセットにドライブを追加します。
完全な DID パス名を使用します。

```
# metaset -s setname -a /dev/did/rdsk/dN
```

- s *setname* デバイスグループ名と同じである、ディスクセット名を指定します。
- a ディスクセットにドライブを追加します。

注 - ディスクセットにドライブを追加するときは、下位デバイス名 (cNtX dY) は使用しないでください。下位レベルデバイス名はローカル名であり、クラスタ全体で一意ではないため、この名前を使用するとディスクセットがスイッチオーバーできなくなる可能性があります。

8 新しいディスクセットとドライブの状態を検査します。

```
# metaset -s setname
```

例 5-22 Solaris ボリュームマネージャー デバイスグループの追加

次の例は、ディスクドライブ /dev/did/rdsk/d1 および /dev/did/rdsk/d2 を持つディスクセットおよびデバイスグループの作成を示し、またデバイスグループが作成されたことを確認しています。

```
# metaset -s dg-schost-1 -a -h phys-schost-1

# cldevicegroup list
dg-schost-1
metaset -s dg-schost-1 -a /dev/did/rdsk/d1 /dev/did/rdsk/d2
```

▼ デバイスグループ (raw ディスク) を追加および登録する

Sun Cluster ソフトウェアでは、他のボリュームマネージャに加え、raw ディスクデバイスグループを使用することができます。Sun Cluster を最初に構成する際、クラスタ内の raw デバイスごとにデバイスグループが自動的に構成されます。この手順を使用して、自動生成されたデバイスグループを Sun Cluster ソフトウェアで使用できるように再構成します。

次の理由のため、raw ディスクタイプの新しいデバイスグループを作成します。

- 複数の DID をデバイスグループに追加したい
- デバイスグループの名前を変更する必要がある
- cldg コマンドの -v オプションを使用せずにデバイスグループのリストを作成したい



注意-複製したデバイスにデバイスグループを作成する場合、作成するデバイスグループ名 (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) は複製したデバイスグループの名前と同じにする必要があります。

- 1 使用する各デバイスを特定し、事前に規定されたデバイスグループの構成を解除します。

次のコマンドは、d7 および d8 に対する定義済みのデバイスグループを除去します。

```
paris-1# cldevicegroup disable dsk/d7 dsk/d8
paris-1# cldevicegroup offline dsk/d7 dsk/d8
paris-1# cldevicegroup delete dsk/d7 dsk/d8
```

- 2 必要なデバイスを含む、新しい raw ディスクデバイスグループを作成します。

次のコマンドは、グローバルデバイスグループ rawdg を作成します。このデバイスグループに d7 および d8 が収められます。

```
paris-1# cldevicegroup create -n phys-paris-1,phys-paris-2 -t rawdisk
-d d7,d8 rawdg
paris-1# /usr/cluster/lib/dcs/cldg show rawdg -d d7 rawdg
paris-1# /usr/cluster/lib/dcs/cldg show rawdg -d d8 rawdg
```

▼ 複製デバイスグループ (ZFS) の追加と登録方法

ZFS を複製するには、名前付きデバイスグループを作成し、zpool に属するディスクをリストする必要があります。デバイスは、一度に 1 つのデバイスグループのみに属することができるため、デバイスを含む Sun Cluster デバイスグループをすでに持っている場合、そのデバイスを新しい ZFS デバイスグループに追加する前にそのグループを削除する必要があります。

作成する Sun Cluster デバイスグループ (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) の名前は、複製されたデバイスグループと同じ名前にしてください。



注意-他社のデータ複製技術による ZFS のフルサポートは保留中です。ZFS サポートに関する最新情報は、最新の Sun Cluster リリースノートを参照してください。

- 1 zpool のデバイスに対応するデフォルトデバイスグループを削除してください。
たとえば、2 つのデバイス /dev/did/dsk/d2 と /dev/did/dsk/d13 を含む mypool と呼ばれる zpool を持っている場合、d2 と d13 と呼ばれる 2 つのデフォルトデバイスグループを削除する必要があります。

```
# cldevicegroup offline dsk/d2 dsk/d13
# cldevicegroup remove dsk/d2 dsk/d13
```

- 2 ステップ 1 で削除したデバイスグループの DID に対応する DID のデバイスグループを作成します。

```
# cldevicegroup create -d d2,d13 -t rawdisk mypool
```

このアクションでは、mypool (zpool と同じ名前) と呼ばれるデバイスグループが作成され、raw デバイス /dev/did/dsk/d2 と /dev/did/dsk/d13 を管理します。

- 3 それらのデバイスを含む zpool を作成します。

```
# zpool create mypool mirror /dev/did/dsk/d2 /dev/did/dsk/d13
```

- 4 リソースグループを作成し、ノードリストに唯一のグローバルゾーンのある複製したデバイス (デバイスグループ内) の移行を管理します。

```
# clrg create -n pnode1,pnode2 migrate_truecopydg-rg
```

- 5 ステップ 4 で作成したリソースグループに hasp-rs リソースを作成し、globaldevicepaths プロパティを raw ディスクのデバイスグループに設定します。このデバイスグループは、ステップ 2 で作成しました。

```
# clr create -t HASStoragePlus -x globaldevicepaths=mypool -g \
migrate_truecopydg-rg hasp2migrate_mypool
```

- 6 アプリケーションリソースグループをローカルゾーンで実行する場合、適切なローカルゾーンを有するノードリストのあるリソースグループを新たに作成してください。ローカルゾーンに対応するグローバルゾーンは、ステップ 4 で作成したリソースグループのノードリストに含まれている必要があります。このリソースからステップ 4 で作成したリソースグループで、rg_affinities プロパティの +++ 値を設定します。

```
# clrg create -n pnode1:zone-1,pnode2:zone-2 -p \
RG_affinities=+++migrate_truecopydg-rg sybase-rg
```

- 7 ステップ 3 で作成した zpool の HASStoragePlus リソース (hasp-rs) を、ステップ 4 または 6 で作成したリソースグループに作成します。resource_dependencies プロパティをステップ 5 で作成した hasp-rs リソースに設定します。

```
# clr create -g sybase-rg -t HASStoragePlus -p zpools=mypool \
-p resource_dependencies=hasp2migrate_mypool \
-p ZpoolsSearchDir=/dev/did/dsk hasp2import_mypool
```

- 8 デバイスグループ名が必要な場合には、この新しいリソースグループ名を使用します。

▼ ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)

注-次の手順は、ディスクを初期化する場合にのみ必要となります。ディスクをカプセル化する場合は、147ページの「ディスクをカプセル化する際に新しいディスクグループを作成する (VERITAS Volume Manager)」の手順を使用してください。

VxVM ディスクグループを追加したら、デバイスグループを登録する必要があります。

VxVM を使用して Oracle RAC 用の共有ディスクグループを設定する場合、『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。

- 1 追加しようとしているディスクグループを構成するディスクに物理的に接続されている任意のクラスタノード上でスーパーユーザーになります。
- 2 VxVM のディスクグループとボリュームを作成します。
ディスクグループとボリュームは任意の方法で作成してください。

注-ミラー化したボリュームを設定している場合、ダーティーリージョンログ (DRL) を使用すると、ノードに障害が発生してからボリュームが回復するまでの時間を短縮できます。ただし、DRL を使用すると I/O スループットが低下することがあります。

この手順を完了する方法については、VERITAS Volume Manager のマニュアルを参照してください。

- 3 VxVM ディスクグループを Sun Cluster デバイスグループとして登録します。
151 ページの「ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)」を参照してください。

Oracle RAC 用の共有ディスクグループをクラスタフレームワークに登録してはいけません。

デバイスグループ名を維持する

デバイスグループに対して様々な管理タスクを実行することができます。

デバイスグループを削除して登録を解除する (Solaris ボリュームマネージャー)

デバイスグループとは、Sun Cluster に登録されている Solaris ボリュームマネージャー ディスクセットのことです。Solaris ボリュームマネージャー デバイスグループを削除するには、`metaclear` と `metaset` コマンドを使用します。これらのコマンドは、Sun Cluster デバイスグループと同じ名前を持つデバイスグループを削除して、ディスクグループの登録を解除します。

ディスクセットを削除する方法については、Solaris ボリュームマネージャーのマニュアルを参照してください。

▼ すべてのデバイスグループからノードを削除する

すべてのデバイスグループの潜在的な主ノードからクラスタノードを削除する場合は、この手順を使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 すべてのデバイスグループの潜在的な主ノードとして削除するノード上で、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 削除するノードがメンバーになっているデバイスグループ (複数可) を確認します。各デバイスグループの `Device group node list` からこのノード名を検索します。

```
# cldevicegroup list -v
```
- 3 [手順 2](#) で特定したデバイスグループの中に、デバイスグループタイプが `SVM` のものがある場合、そのタイプの各デバイスグループに対して [143 ページ](#) の「[デバイスグループからノードを削除する \(Solaris ボリュームマネージャー\)](#)」の手順を実行します。
- 4 [手順 2](#) で特定したデバイスグループの中に、デバイスグループタイプが `VxVM` のものがある場合、そのタイプの各デバイスグループに対して [161 ページ](#) の「[デバイスグループからノードを削除する \(Veritas Volume Manager\)](#)」の手順を実行します。

- 5 削除するノードがメンバーになっている **raw** デバイスディスクグループを特定します。

```
# cldevicegroup list -v
```
- 6 手順5で表示されたデバイスグループの中に、デバイスグループタイプが **Disk** または **Local_Disk** のものがある場合、これらの各デバイスグループに対して、[163 ページ](#) の「**raw** ディスクデバイスグループからノードを削除する」の手順を実行します。
- 7 すべてのデバイスグループの潜在的な主ノードのリストからノードが削除されていることを確認します。
ノードがどのデバイスグループの潜在的な主ノードのリストにも存在しなければ、このコマンドは何も返しません。

```
# cldevicegroup list -v nodename
```

▼ デバイスグループからノードを削除する (Solaris ボリュームマネージャー)

Solaris ボリュームマネージャー デバイスグループの潜在的な主ノードのリストからクラスタノードを削除するには、次の手順を使用します。ノードを削除したいグループデバイスごとに `metaset` コマンドを繰り返します。



注意-ほかのノードが有効なクラスタメンバーであり、それらのノードの少なくとも1つがディスクセットを持つ場合は、クラスタの外側で起動されるクラスタノードで `metaset -s setname -f -t` を実行しないでください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 ノードがまだデバイスグループのメンバーであり、かつ、このデバイスグループが **Solaris** ボリュームマネージャー デバイスグループであることを確認します。
Solaris ボリュームマネージャー のデバイスグループは、デバイスグループタイプが **SDS/SVM** のものです。

```
phys-schost-1% cldevicegroup show devicegroup
```

- 2 どのノードがデバイスグループの現在の主ノードであるかを特定します。

```
# cluster status -t devicegroup
```

- 3 変更したいデバイスグループを所有しているノードでスーパーユーザーになります。

- 4 デバイスグループからこのノードのホスト名を削除します。

```
# metaset -s setname -d -h nodelist
```

-s *setname* デバイスグループの名前を指定します。

-d -h で指定されたノードをデバイスグループから削除します。

-h *nodelist* 削除するノード (複数可) のノード名を指定します。

注-更新が完了するまでに数分間かかることがあります。

コマンドが正常に動作しない場合は、コマンドに -f (force) オプションを追加します。

```
# metaset -s setname -d -f -h nodelist
```

- 5 潜在的な主ノードとしてノードを削除するデバイスグループごとに手順 4 を繰り返します。

- 6 デバイスグループからノードが削除されたことを確認します。
デバイスグループ名は `metaset` に指定したディスクセット名と一致します。

```
phys-schost-1% cldevicegroup list -v devicegroup
```

例 5-23 デバイスグループからノードを削除する (Solaris ボリュームマネージャー)

次に、デバイスグループ構成からホスト名 `phys-schost-2` を削除する例を示します。この例では、指定したデバイスグループから `phys-schost-2` を潜在的な主ノードとして削除します。`cldevicegroup show` コマンドを実行することにより、ノードが削除されていることを確認します。削除したノードが画面に表示されていないことを確認します。

```
[Determine the Solaris ボリュームマネージャー
```

```
  device group for the node:]
```

```
# cldevicegroup show dg-schost-1
```

```
=== Device Groups ===
```

```
Device Group Name:                      dg-schost-1
```

```

Type:                               SVM
failback:                            no
Node List:                            phys-schost-1, phys-schost-2
preferenced:                          yes
numsecondaries:                        1
diskset name:                          dg-schost-1
[Determine which node is the current primary for the device group:]
# cldevicegroup status dg-schost-1
=== Cluster Device Groups ===

--- Device Group Status ---

Device Group Name    Primary          Secondary        Status
-----
dg-schost-1         phys-schost-1   phys-schost-2   Online
[Become superuser on the node that currently owns the device group.]
[Remove the host name from the device group:]
# metaset -s dg-schost-1 -d -h phys-schost-2
[Verify removal of the node:]
phys-schost-1% cldevicegroup list -v dg-schost-1
=== Cluster Device Groups ===

--- Device Group Status ---

Device Group Name    Primary          Secondary        Status
-----
dg-schost-1         phys-schost-1   -                Online

```

▼ 1つのクラスタ内に4つ以上のディスクセットを作成する

Solaris 9 を実行中で、クラスタにディスクセットを4つ以上作成する場合は、ディスクセットを作成する前に次の各手順を行う必要があります。Solaris 10 を実行中の場合は、この手順を実行する必要はありません。初めてディスクセットをインストールする場合や、完全に構成されたクラスタにさらにディスクセットを追加する場合には次の手順に従います。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 md_nsets 変数が十分に大きな値であることを確認します。この値は、クラスタに作成する予定のディスクセットの合計数より大きな値である必要があります。
 - a. クラスタの任意のノードで、/kernel/drv/md.conf ファイルの md_nsets 変数の値を検査します。
 - b. クラスタ内にあるディスクセットの数が md_nsets の既存の値から 1 を引いた値よりも大きい場合、各ノード上で md_nsets の値を増やします。
ディスクセットの最大数は、md_nsets の値から 1 を引いた値です。md_nsets に設定できる最大値は 32 です。
 - c. クラスタの各ノードの /kernel/drv/md.conf ファイルが同じであることを確認します。



注意- このガイドラインに従わないと、重大な Solaris ボリュームマネージャーエラーが発生し、データが失われることがあります。

- d. ノードのどれか 1 つでクラスタを停止します。

```
# cluster shutdown -g0 -y
```

- e. クラスタ内にある各ノードを再起動します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                       |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

- 2 クラスタの各ノードで **devfsadm(1M)** コマンドを実行します。
このコマンドは、すべてのノードで同時に実行できます。
- 3 クラスタのノードの 1 つから **cldevice populate** コマンドを実行します。

- 4 ディスクセットの作成に移る前に、各ノードで `ldevice populate` コマンドが終了しているかを確認します。
- ノードの1つで `cldevice` コマンドを実行すると、このコマンドはリモートから自分自身をすべてのノードで呼び出します。 `cldevice populate` コマンドが処理を終了したかどうかを確認するには、クラスタの各ノードで次のコマンドを実行します。

```
# ps -ef | grep scgdevs
```

▼ ディスクをカプセル化する際に新しいディスクグループを作成する (VERITAS Volume Manager)

注-次の手順は、ディスクをカプセル化する場合にのみ必要となります。ディスクを初期化する場合、141 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」の手順を使用します。

ルート以外のディスクを Sun Cluster デバイスグループに変換するには、そのディスクを VxVM ディスクグループとしてカプセル化してから、そのディスクグループを Sun Cluster デバイスグループとして登録します。

ディスクのカプセル化は、VxVM ディスクグループを初めて作成するときのみサポートされています。VxVM ディスクグループを作成して、Sun Cluster デバイスグループとして登録したあとは、そのディスクグループには、初期化してもよいディスクだけを登録します。

VxVM を使用して Oracle RAC 用の共有ディスクグループを設定する場合、『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

- 2 /etc/vfstab ファイルに、カプセル化されたディスクのファイルシステムのエントリがある場合は、mount at boot オプションを必ず no に設定します。
ディスクがカプセル化されて Sun Cluster デバイスグループとして登録されたあとは、この設定を yes に設定し直します。
- 3 ディスクをカプセル化します。
vxdiskadm のメニューまたはグラフィカルユーザーインターフェースを使用して、ディスクをカプセル化します。VxVM では、2つの空きパーティションのほか、ディスクの始点または終端に未割当てのシリンダが必要です。また、スライス 2 をディスク全体に設定する必要もあります。詳細は、vxdiskadm のマニュアルページを参照してください。
- 4 ノードを停止して再起動します。
clnode evacuate コマンドは、グローバルクラスタ内のすべての非投票ノードを含むすべてのリソースグループとデバイスグループを、指定のノードから次に優先されるノードにスイッチオーバーします。shutdown コマンドを使用して、ノードを停止して再起動します。
clnode evacuate node[,...]
shutdown -g0 -y -i6
- 5 必要であれば、すべてのリソースグループとデバイスグループを元のノードにスイッチバックします。
リソースグループとデバイスグループが、もともと主ノードにフェイルバックするように構成されていた場合、この手順は必要ありません。
cldevicegroup switch -n node devicegroup
clresourcegroup switch -z zone -n node resourcegroup

node ノードの名前。
zone リソースグループをマスターできる node 上の非投票ノードの名前。リソースグループを作成した際に非投票ノードを指定した場合にのみ、zone を指定します。
- 6 VxVM ディスクグループを Sun Cluster デバイスグループとして登録します。
151 ページの「ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)」を参照してください。

Oracle RAC 用の共有ディスクグループをクラスタフレームワークに登録してはいけません。
- 7 手順 2 で mount at boot オプションを no に設定した場合は、yes に戻してください。

▼ 新しいボリュームを既存のデバイスグループに追加する (Veritas Volume Manager)

新しいボリュームを既存の VxVM デバイスグループに追加する場合、次の手順は、オンラインであるデバイスグループの主ノードから実行します。

注- ボリュームを追加したあとで、[154 ページの「ディスクグループの構成変更を登録する \(VERITAS Volume Manager\)」](#) の手順に従って構成変更の内容を登録する必要があります。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.read および solaris.cluster.administer を提供する役割になります。

```
# cldevicegroup status
```
- 2 新しいボリュームを追加するデバイスグループの主ノードを確認します。

```
# cldevicegroup switch -n nodename devicegroup
```

nodename デバイスグループの切り替え先であるノードの名前を指定します。このノードが新しい主ノードになります。

devicegroup 切り替えるデバイスグループを指定します。
- 3 デバイスグループがオフラインである場合、デバイスグループをオンラインにします。

```
# cldevicegroup switch -n nodename devicegroup
```

nodename デバイスグループの切り替え先であるノードの名前を指定します。このノードが新しい主ノードになります。

devicegroup 切り替えるデバイスグループを指定します。
- 4 主ノード(デバイスグループを現在マスターしているノード)から、ディスクグループに VxVM ボリュームを作成します。
VxVM ボリュームの作成方法は、VERITAS Volume Manager のマニュアルを参照してください。
- 5 VxVM ディスクグループに加えた変更を同期化し、グローバルな名前空間を更新します。

```
# cldevicegroup sync
```

154 ページの「ディスクグループの構成変更を登録する (VERITAS Volume Manager)」。

▼ 既存のディスクグループをデバイスグループに変換する (Veritas Volume Manager)

既存の VxVM ディスクグループを Sun Cluster デバイスグループに変換するには、ディスクグループを現在のノードにインポートしてから、そのディスクグループを Sun Cluster デバイスグループとして登録します。

- 1 クラスタ内の任意のノード上でスーパーユーザーになります。
- 2 VxVM ディスクグループを現在のノードにインポートします。

```
# vxdg import diskgroup
```
- 3 VxVM ディスクグループを Sun Cluster デバイスグループとして登録します。
[151 ページの「ディスクグループをデバイスグループとして登録する \(VERITAS Volume Manager\)」](#)を参照してください。

▼ デバイスグループに新しいマイナー番号を割り当てる (VERITAS Volume Manager)

マイナー番号がほかのディスクグループと衝突してデバイスグループの登録が失敗する場合、新しいディスクグループに未使用の新しいマイナー番号を割り当てます。新しいマイナー番号を割り当てた後で、登録手順を再度実行し、ディスクグループを Sun Cluster デバイスグループとして登録します。

- 1 クラスタ内の任意のノード上でスーパーユーザーになります。
- 2 使用中のマイナー番号を確認します。

```
# ls -l /global/.devices/node@nodeid/dev/vx/dsk/*
```
- 3 新しいディスクグループのベースとなるマイナー番号として、使用されていない別の 1000 の倍数を選択します。
- 4 ディスクグループに新しいマイナー番号を割り当てます。

```
# vxdg remminor diskgroup base-minor-number
```

- 5 VxVM ディスクグループを Sun Cluster デバイスグループとして登録します。
151 ページの「ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)」を参照してください。

例 5-24 デバイスグループに新しいマイナー番号を割り当てる

次の例は、マイナー番号 16000 - 16002 と 4000 - 4001 が使用されていることを示しています。ここでは、`vx dg remminor` コマンドを使用して、ベースとなるマイナー番号 5000 を新しいデバイスグループに割り当てています。

```
# ls -l /global/.devices/node@nodeid/dev/vx/dsk/*

/global/.devices/node@nodeid/dev/vx/dsk/dg1
brw----- 1 root    root    56,16000 Oct  7 11:32 dg1v1
brw----- 1 root    root    56,16001 Oct  7 11:32 dg1v2
brw----- 1 root    root    56,16002 Oct  7 11:32 dg1v3

/global/.devices/node@nodeid/dev/vx/dsk/dg2
brw----- 1 root    root    56,4000 Oct  7 11:32 dg2v1
brw----- 1 root    root    56,4001 Oct  7 11:32 dg2v2
# vx dg remminor dg3 5000
```

▼ ディスクグループをデバイスグループとして登録する (VERITAS Volume Manager)

この手順では `clsetup` ユーティリティを使用して、関連付けられた VxVM ディスクグループを Sun Cluster デバイスグループとして登録します。

注-デバイスグループをクラスタに登録したあとは、VxVM コマンドを使用して VxVM ディスクグループをインポートまたはエクスポートしないでください。VxVM ディスクグループやボリュームに変更を加えた場合は、154 ページの「ディスクグループの構成変更を登録する (VERITAS Volume Manager)」の手順に従って、デバイスグループの構成変更を登録してください。この手順によって、グローバルな名前空間が正しい状態になります。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

始める前に VxVM デバイスグループの登録前に、次の必要条件が満たされていることを確認します。

- クラスタ内の任意のノードでのスーパーユーザー特権。
- デバイスグループとして登録する VxVM ディスクグループの名前。
- デバイスグループをマスターするノードの優先順位。
- デバイスグループの二次ノードの希望数。

優先順位を定義する際には、もっとも優先されるノードに障害が発生し、のちにクラスタに戻った場合に、デバイスグループをそのノードにスイッチバックするかどうかも指定します。

ノードの優先順位とフェイルバックのオプションについての詳細は、`cldevicegroup(1CL)`を参照してください。

主ノード以外のクラスタノード (スペア) から二次ノードへの移行ノードの優先順位では通常、デバイスグループの二次ノードのデフォルト数は 1 に設定されます。デフォルトの設定では、主ノードが通常の動作中に複数の二次ノードをチェックすることによって発生する性能の低下を最小限に抑えます。たとえば、4 ノードクラスタでは、デフォルトで、1 つが主ノード、1 つが二次ノード、そして 2 つがスペアノードに構成されます。166 ページの「デバイスグループの二次ノードの希望数を設定する」も参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 `clsetup` ユーティリティーを起動します。
`clsetup`
メインメニューが表示されます。
- 3 VxVM デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプションに対応する数を入力します。
「デバイスグループメニュー」が表示されます。
- 4 VxVM デバイスグループを登録するために、VxVM ディスクグループをデバイスグループとして登録するためのオプションに対応する数を入力します。
指示に従って、Sun Cluster デバイスグループとして登録する VxVM ディスクグループの名前を入力します。

このデバイスグループがストレージベースの複製を使用して複製されている場合、この名前は TrueCopy 複製グループ名と一致する必要があります。

VxVM を使用して Oracle Parallel Server/Oracle RAC 用に共有ディスクグループを設定する場合は、クラスタフレームワークには共有ディスクグループを登録しません。『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。

- 5 デバイスグループを登録しようとしたときに、次のようなエラーが表示された場合は、デバイスグループに新しいマイナー番号を割り当てます。

```
scconf: Failed to add device group - in use
```

デバイスグループに新しいマイナー番号を割り当てるには、150 ページの「[デバイスグループに新しいマイナー番号を割り当てる \(VERITAS Volume Manager\)](#)」の手順を使用してください。この手順によって、既存のデバイスグループが使用しているマイナー番号と衝突しない、新しいマイナー番号を割り当てることができます。

- 6 複製されたデバイスグループを構成している場合は、そのデバイスグループの複製プロパティを設定します。

```
# cldevicegroup sync devicegroup
```

- 7 デバイスグループが登録され、オンラインになったことを確認します。

デバイスグループが適切に登録されている場合、次のコマンドを使用すると、新しいデバイスグループの情報が表示されます。

```
# cldevicegroup status devicegroup
```

注-VxVM ディスクグループ、または、クラスタに登録されているボリュームの構成情報を変更した場合、clsetup を使用してデバイスグループを同期化する必要があります。このような構成変更には、ボリュームの追加や削除、既存ボリュームのグループ、所有者、アクセス権の変更などがあります。構成変更後に登録を行うと、グローバルな名前空間が正しい状態になります。132 ページの「[グローバルデバイス名前空間を更新する](#)」を参照してください。

例 5-25 Veritas Volume Manager デバイスグループの登録

次に、clsetup で VxVM デバイスグループ (dg1) を登録する際に生成される cldevicegroup コマンドの例と、その検証手順を示します。この例では、VxVM ディスクグループとボリュームは以前に作成されたものと想定しています。

```
# clsetup
```

```
# cldevicegroup create -t vxvm -n phys-schost-1,phys-schost-2 -p failback=true dg1
```

```
# cldevicegroup status dg1

=== Cluster Device Groups ===

--- Device Group Status ---

Device Group Name      Primary      Secondary      Status
-----
dg1                    phys-schost-1 phys-schost-2  OnLine
```

参照 VxVM デバイスグループ上にクラスタファイルシステムを作成する場合は、[180 ページ](#)の「[クラスタファイルシステムを追加する](#)」を参照してください。

マイナー番号に問題が発生した場合は、[150 ページ](#)の「[デバイスグループに新しいマイナー番号を割り当てる \(VERITAS Volume Manager\)](#)」を参照してください。

▼ ディスクグループの構成変更を登録する (VERITAS Volume Manager)

VxVM ディスクグループやボリュームの構成情報を変更したときは、Sun Cluster デバイスグループに構成変更を登録する必要があります。この登録によって、グローバルな名前空間が正しい状態になります。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。
- 2 clsetup ユーティリティーを起動します。
clsetup
メインメニューが表示されます。
- 3 VxVM デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプションに対応する数を入力します。
「デバイスグループメニュー」が表示されます。

- 4 構成変更を登録するには、VxVM デバイスグループのボリューム情報を同期化するためのオプションに対応する数を入力します。
指示に従って、構成を変更した VxVM ディスクグループ名を入力します。

例 5-26 Veritas Volume Manager ディスクグループの構成の変更の登録

次に、`clsetup` で VxVM デバイスグループ (`dg1`) を登録する際に生成される `cldevicegroup` コマンドの例を示します。この例では、VxVM ディスクグループとボリュームは以前に作成されたものと想定しています。

```
# clsetup  
  
cldevicegroup sync dg1
```

▼ ローカルディスクグループをデバイスグループに変換する (VxVM)

ローカル VxVM ディスクグループをグローバルにアクセス可能な VxVM デバイスグループに変更するには、次の手順を実行します。

- 1 各クラスタのノードのスーパーユーザーになります。
- 2 `clsetup` ユーティリティーを起動します。
`clsetup`
- 3 `localonly` プロパティの設定を解除します。
 - a. メニュー項目「デバイスグループとボリューム」を選択します。
 - b. メニュー項目「ローカルディスクグループの VxVM ディスクグループへの再設定」を選択します。
 - c. 指示に従い、`localonly` プロパティの設定を解除します。
- 4 ディスクグループをマスターできるノードを指定します。
 - a. `clsetup` ユーティリティーのメインメニューに戻ります。
 - b. メニュー項目「デバイスグループとボリューム」を選択します。

- c. メニュー項目「**VxVM**ディスクグループをデバイスグループとして登録」を選択します。
 - d. 指示に従い、ディスクグループをマスターできるノードを指定します。
 - e. 完了後 `clsetup` ユーティリティーを終了します。
- 5 デバイスグループが構成されていることを確認します。
- ```
phys-schost# cldevicegroup show
```

## ▼ デバイスグループをローカルディスクグループに変換する (VxVM)

VxVM デバイスグループを、Sun Cluster ソフトウェアにより管理されていないローカル VxVM ディスクグループに変更するには、次の手順を実行します。ローカルディスクグループはそのノードリストに複数のノードを持つことができますが、一度に1つのノードによってのみマスターできます。

- 1 各クラスタのノードのスーパーユーザーになります。
- 2 デバイスグループをオフラインにします。
 

```
phys-schost# cldevicegroup offline devicegroup
```
- 3 デバイスグループを登録解除します。
  - a. `clsetup` ユーティリティーを起動します。
 

```
phys-schost# clsetup
```
  - b. メニュー項目「デバイスグループとボリューム」を選択します。
  - c. メニュー項目「**VxVM**デバイスグループを登録を解除」を選択します。
  - d. 指示に従い、**Sun Cluster** ソフトウェアから登録を解除する **VxVM** ディスクグループを指定します。
  - e. `clsetup` ユーティリティーを終了します。
- 4 ディスクグループが **Sun Cluster** ソフトウェアに登録されていないことを確認します。
 

```
phys-schost# cldevicegroup status
```

コマンド出力には、登録を解除されたデバイスグループは表示されなくなるはずで  
す。

- 5 ディスクグループをインポートします。

```
phys-schost# vxdg import diskgroup
```

- 6 ディスクグループの localonly プロパティを設定します。

- a. clsetup ユーティリティを起動します。

```
phys-schost# clsetup
```

- b. メニュー項目「デバイスグループとボリューム」を選択します。

- c. メニュー項目「VxVM ディスクグループのローカルディスクグループとしての設  
定」を選択します。

- d. 指示に従い、localonly プロパティを設定し、ディスクグループを排他的にマ  
スターする1つのノードを指定します。

- e. 完了後 clsetup ユーティリティを終了します。

- 7 ディスクグループが正しくローカルディスクグループとして構成されていることを  
確認します。

```
phys-schost# vxdg list diskgroup
```

## ▼ デバイスグループからボリュームを削除する (Veritas Volume Manager)

---

注-デバイスグループからボリュームを削除したあとは、[154 ページの「ディスクグループの構成変更を登録する \(VERITAS Volume Manager\)」](#)の手順に従って、デバイスグループに構成の変更を登録する必要があります。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手  
順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマン  
ドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一で  
す。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指  
向コマンド」](#)を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBACの承認 solaris.cluster.read および solaris.cluster.modify を提供する役割になります。
- 2 デバイスグループの主ノードと状態を確認します。  

```
cldevicegroup status devicegroup
```
- 3 デバイスグループがオフラインのときは、オンラインにします。  

```
cldevicegroup online devicegroup
```
- 4 主ノード(デバイスグループを現在マスターしているノード)から、ディスクグループの VxVM ボリュームを削除します。  

```
vxedit -g diskgroup -rf rm volume
```

-g *diskgroup* ボリュームが含まれる VxVM ディスクグループを指定します。

-rf *rm volume* 指定したボリュームを削除します。-r オプションは、処理を再帰的に繰り返す指定です。-f オプションは、有効に設定されているボリュームを削除する場合に必要です。
- 5 clsetup ユーティリティを使用して、デバイスグループの構成変更を登録し、グローバルな名前空間を更新します。  
[154 ページの「ディスクグループの構成変更を登録する \(VERITAS Volume Manager\)」](#)を参照してください。

## ▼ デバイスグループを削除して登録を解除する (Veritas Volume Manager)

Sun Cluster デバイスグループを削除すると、対応する VxVM ディスクグループはエクスポートされます(消去されるわけではない)。ただし、VxVM ディスクグループが引き続き存在していても、再登録しないかぎりクラスタで使用することはできません。

次の手順では、clsetup ユーティリティを使用して、VxVM ディスクグループを削除し、Sun Cluster デバイスグループから登録を解除します。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。
- 2 デバイスグループをオフラインにします。  
# `cldevicegroup offline devicegroup`
- 3 `clsetup` ユーティリティーを起動します。  
# `clsetup`  
メインメニューが表示されます。
- 4 VxVM デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプションに対応する数を入力します。  
「デバイスグループメニュー」が表示されます。
- 5 VxVM ディスクグループを登録解除するには、VxVM デバイスグループを登録解除するためのオプションに対応する数を入力します。  
指示に従って、登録を解除する VxVM ディスクグループの名前を入力します。

#### 例 5-27 Veritas Volume Manager デバイスグループの削除および登録の解除

次に、VxVM デバイスグループ `dg1` をオフラインにして、デバイスグループの削除および登録解除の際に `clsetup` により生成される `cldevicegroup` コマンドの例を示します。

```
cldevicegroup offline dg1
clsetup

cldevicegroup delete dg1
```

## ▼ デバイスグループにノードを追加する (VERITAS Volume Manager)

この手順では、`clsetup` ユーティリティーを使用してデバイスグループにノードを追加します。

VxVM デバイスグループにノードを追加するには以下が必要です。

- クラスタ内のノードでのスーパーユーザー特権
- ノードの追加先の VxVM デバイスグループの名前
- 追加するノードの名前またはノード ID

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.read および solaris.cluster.modify を提供する役割になります。
- 2 clsetup ユーティリティーを起動します。  
# clsetup  
メインメニューが表示されます。
- 3 VxVM デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプションに対応する数を入力します。  
「デバイスグループメニュー」が表示されます。
- 4 VxVM デバイスグループにノードを追加するには、VxVM デバイスグループへノードを追加するためのオプションに対応する数を入力します。  
指示に従って、デバイスグループ名とノード名を入力します。
- 5 ノードが追加されたことを確認します。  
次のコマンドを実行し、表示される新しいディスクのデバイスグループ情報を確認します。  
# cldevicegroup show devicegroup

#### 例 5-28 Veritas Volume Manager デバイスグループへのノードの追加

次に、clsetup でノード (phys-schost-3) を VxVM デバイスグループ (dg1) に追加する際に生成される scconf コマンドと、その検証手順の例を示します。

```
clsetup

cldevicegroup add-node -n phys-schost-3 dg1

cldevicegroup show dg1

=== Device Groups ===

Device Group Name: dg1
Type: VxVM
```

```

failback: yes
Node List: phys-schost-1, phys-schost-3
preferenced: no
numsecondaries: 1
diskgroup names: dg1

```

## ▼ デバイスグループからノードを削除する (Veritas Volume Manager)

Veritas Volume Manager (VxVM) デバイスグループ (ディスクグループ) の潜在的な主ノードリストからクラスタノードを削除する場合は、この手順を使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 ノードがまだグループのメンバーであり、かつ、そのグループが **VxVM** デバイスグループであることを確認します。  
デバイスグループタイプ VxVM は VxVM デバイスグループを示します。  
`phys-schost-1% cldevicegroup show devicegroup`
- 2 現在のクラスタメンバーノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。
- 3 `clsetup` ユーティリティを起動します。  
`# clsetup`  
メインメニューが表示されます。
- 4 デバイスグループを再構成する場合は、デバイスグループおよびボリュームのオプションに対応する数を入力します。
- 5 **VxVM** デバイスグループからノードを削除する場合は、**VxVM** デバイスグループからノードを削除するためのオプションに対応する数を入力します。  
プロンプトに従って、デバイスグループからクラスタノードを削除します。次の情報を入力するよう求められます。
  - VxVM のデバイスグループ
  - ノード名

- 6 ノードが VxVM デバイスグループ (複数可) から削除されていることを確認します。

```
cldevicegroup show devicegroup
```

### 例 5-29 デバイスグループからノードを削除する (VxVM)

この例では、dg1 という VxVM のデバイスグループから phys-schost-1 というノードを削除します。

```
[Determine the VxVM device group for the node:]
```

```
cldevicegroup show dg1
```

```
=== Device Groups ===
```

```
Device Group Name: dg1
Type: VXVM
failback: no
Node List: phys-schost-1, phys-schost-2
preferenced: no
numsecondaries: 1
diskgroup names: dg1
```

```
[Become superuser and start the clsetup utility:]
```

```
clsetup
```

```
Select Device groups and volumes>Remove a node from a VxVM device group.
```

Answer the questions when prompted.

You will need the following information.

| Name:                  | Example:      |
|------------------------|---------------|
| VxVM device group name | dg1           |
| node names             | phys-schost-1 |

```
[Verify that the cldevicegroup command executed properly:]
```

```
cldevicegroup remove-node -n phys-schost-1 dg1
```

Command completed successfully.

Dismiss the clsetup Device Groups Menu and Main Menu.

```
[Verify that the node was removed:]
```

```
cldevicegroup show dg1
```

```
=== Device Groups ===
```

```
Device Group Name: dg1
Type: VXVM
failback: no
Node List: phys-schost-2
preferenced: no
numsecondaries: 1
```

device names:

dg1

## ▼ raw ディスクデバイスグループからノードを削除する

raw ディスクデバイスグループの潜在的な主ノードリストからクラスタノードを削除する場合は、この手順を使用します。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタ内のノード、ただし削除するノード以外のノード上で、スーパーユーザーになるか、RBAC の承認 solaris.cluster.read および solaris.cluster.modify を提供する役割になります。
- 2 削除されるノードに接続されたデバイスグループを特定し、どれが raw ディスクデバイスグループであるかを判別します。

```
cldevicegroup show -n nodename -t rawdisk +
```

- 3 すべての Local\_Disk raw ディスクデバイスグループの localonly プロパティを無効にします。

```
cldevicegroup set -p localonly=false devicegroup
```

localonly プロパティについての詳細は、cldevicegroup(1CL) のマニュアルページを参照してください。

- 4 削除するノードに接続されているすべての raw ディスクデバイスグループの localonly プロパティが無効になっていることを確認します。  
デバイスグループタイプ Disk は、この raw ディスクデバイスグループの localonly プロパティが無効になっていることを表します。

```
cldevicegroup show -n nodename -t rawdisk -v +
```

- 5 手順 2 で特定されたすべての raw ディスクデバイスグループからノードを削除します。

この手順は、削除するノードに接続されている raw ディスクデバイスグループごとに行う必要があります。

```
cldevicegroup remove-node -n nodename devicegroup
```

## 例 5-30 raw デバイスグループからノードを削除する

この例では、raw ディスクデバイスグループからノード (phys-schost-2) を削除します。すべてのコマンドは、クラスタの別のノード (phys-schost-1) から実行します。

[Identify the device groups connected to the node being removed, and determine which are raw-disk device groups:]

```
phys-schost-1# cldevicegroup show -n phys-schost-2 -t rawdisk -v +
```

```
Device Group Name: dsk/d4
Type: Disk
failback: false
Node List: phys-schost-2
preferenced: false
localonly: false
autogen true
numsecondaries: 1
device names: phys-schost-2
```

```
Device Group Name: dsk/d2
Type: VxVM
failback: true
Node List: pbrave2
preferenced: false
localonly: false
autogen true
numsecondaries: 1
diskgroup name: vxdg1
```

```
Device Group Name: dsk/d1
Type: SVM
failback: false
Node List: pbrave1, pbrave2
preferenced: true
localonly: false
autogen true
numsecondaries: 1
diskset name: ms1
```

```
(dsk/d4) Device group node list: phys-schost-2
(dsk/d2) Device group node list: phys-schost-1, phys-schost-2
(dsk/d1) Device group node list: phys-schost-1, phys-schost-2
```

[Disable the localonly flag for each local disk on the node:]

```
phys-schost-1# cldevicegroup set -p localonly=false dsk/d4
```

[Verify that the localonly flag is disabled:]

```
phys-schost-1# cldevicegroup show -n phys-schost-2 -t rawdisk +
```

```
(dsk/d4) Device group type: Disk
(dsk/d8) Device group type: Local_Disk
```

[Remove the node from all raw-disk device groups:]

```
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d4
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d2
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d1
```

## ▼ デバイスグループのプロパティを変更する

デバイスグループの主所有権を確立する方法は、`preferenced` という所有権設定属性の設定に基づきます。この属性を設定していない場合は、ほかで所有されていないデバイスグループの主所有者が、そのグループ内のディスクへのアクセスを試みる最初のノードになります。一方、この属性を設定してある場合は、ノードが所有権の確立を試みる優先順位を指定する必要があります。

`preferenced` 属性を無効にすると、`failback` 属性も自動的に無効に設定されます。ただし、`preferenced` 属性を有効または再有効にする場合は、`failback` 属性を有効にするか無効にするかを選択できます。

`preferenced` 属性を有効または再有効にした場合は、主所有権の設定一覧でノードの順序を確立し直す必要があります。

次の手順では、`clsetup` を使用し、Solaris ポリリュームマネージャーまたは VxVM デバイスグループの、`preferenced` 属性と `failback` 属性を設定または設定解除します。

始める前に この手順を実行するには、属性値を変更するデバイスグループの名前が必要です。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。

- 2 `clsetup` ユーティリティを起動します。

```
clsetup
```

メインメニューが表示されます。

- 3 デバイスグループを使用して作業するには、デバイスグループおよびポリリュームのオプションに対応する数を入力します。

「デバイスグループメニュー」が表示されます。

- 4 デバイスグループの重要なプロパティを変更するには、**VxVM** または **Solaris** ボリュームマネージャーデバイスグループの重要なプロパティを変更するためのオプションに対応する数を入力します。  
「デバイスグループのプロパティ変更メニュー」が表示されます。
- 5 デバイスグループのプロパティを変更するには、**preference**、**failback** などのプロパティを変更するオプションに対応する数を入力します。  
指示に従って、デバイスグループの **preferenced** および **failback** オプションを設定します。
- 6 デバイスグループの属性が変更されたことを確認します。  
次のコマンドを実行し、表示されるデバイスグループ情報を確認します。  
`# cldevicegroup show -v devicegroup`

### 例 5-31 デバイスグループのプロパティの変更

次に、`clsetup` でデバイスグループ (`dg-schost-1`) の属性値を設定したときに生成される `cldevicegroup` コマンドの例を示します。

```
cldevicegroup set -p preferenced=true -p failback=true -p numsecondaries=1 \
-p nodelist=phys-schost-1,phys-schost-2 dg-schost-1
cldevicegroup show dg-schost-1
```

=== Device Groups ===

```
Device Group Name: dg-schost-1
Type: SVM
failback: yes
Node List: phys-schost-1, phys-schost-2
preferenced: yes
numsecondaries: 1
diskset names: dg-schost-1
```

## ▼ デバイスグループの二次ノードの希望数を設定する

`numsecondaries` プロパティは、主ノードに障害が発生した場合にグループをマスターできる、デバイスグループ内のノード数を指定します。デバイスサービスの二次ノードのデフォルト数は1です。この値には、1からデバイスグループ内で動作している主ノード以外のプロバイダノード数までの任意の整数を設定できます。

この設定は、クラスタの性能と可用性のバランスをとるための重要な要因になります。たとえば、二次ノードの希望数を増やすと、クラスタ内で同時に複数の障害が発生した場合でも、デバイスグループが生き残る可能性が増えます。しかし、二次ノード数を増やすと、通常の動作中の性能が一様に下がります。通常、二次ノード数を減らすと、性能が上がりますが、可用性が下がります。しかし、二次ノード数を増やしても、必ずしも、当該のファイルシステムまたはデバイスグループの可用性が上がるわけではありません。詳細については、『[Sun Cluster Concepts Guide for Solaris OS](#)』の第3章「[Key Concepts for System Administrators and Application Developers](#)」を参照してください。

`numsecondaries` プロパティを変更すると、二次ノードの実数と希望数の間に整合性がない場合、二次ノードはデバイスグループに追加されるか、またはデバイスグループから削除されます。

この手順では、`clsetup` ユーティリティを使用して、すべてのタイプのデバイスグループの `numsecondaries` プロパティを設定します。デバイスグループを構成する際のデバイスグループのオプションの詳細については、`cldevicegroup(1CL)` を参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。
- 2 `clsetup` ユーティリティを起動します。  

```
clsetup
```

メインメニューが表示されます。
- 3 デバイスグループを使用して作業するため、「デバイスグループとボリューム」というラベルのオプションを選択します。  
「デバイスグループメニュー」が表示されます。
- 4 デバイスグループの重要なプロパティを変更するには、「デバイスグループのキープロパティを変更」というラベルのオプションを選択します。  
「デバイスグループのプロパティ変更メニュー」が表示されます。

- 5 二次ノードの希望数を変更するには、numsecondaries プロパティを変更するためのオプションに対応する数を入力します。

指示に従って、デバイスグループに構成したい二次ノードの希望数を入力します。対応する cldevicegroup コマンドが実行され、ログが出力され、ユーティリティーは前のメニューに戻ります。

- 6 デバイスグループの構成を検証します。

```
cldevicegroup show dg-schost-1
```

```
=== Device Groups ===
```

```
Device Group Name: dg-schost-1
Type: VxVm This might also be SDS or Local_Disk.
failback: yes
Node List: phys-schost-1, phys-schost-2 phys-schost-3
preferenced: yes
numsecondaries: 1
diskgroup names: dg-schost-1
```

---

注-VxVM ディスクグループ、または、クラスタに登録されているボリュームの構成情報を変更した場合、clsetup を使用してデバイスグループを再登録する必要があります。このような構成変更には、ボリュームの追加や削除、既存ボリュームのグループ、所有者、アクセス権の変更などがあります。構成変更後に登録を行うと、グローバルな名前空間が正しい状態になります。[132 ページの「グローバルデバイス名前空間を更新する」](#) を参照してください。

---

- 7 デバイスグループの属性が変更されたことを確認します。

次のコマンドを実行して、表示されるデバイスグループ情報を確認します。

```
cldevicegroup show -v devicegroup
```

### 例 5-32 二次ノードの希望数の変更 (Solaris ボリュームマネージャー)

次に、デバイスグループ (dg-schost-1) の二次ノードの希望数を構成するときに、clsetup によって生成される cldevicegroup コマンドの例を示します。この例では、ディスクグループとボリュームは以前に作成されているものと想定していません。

```
cldevicegroup set -p numsecondaries=1 dg-schost-1
```

```
cldevicegroup show -v dg-schost-1
```

```
=== Device Groups ===
```

```
Device Group Name: dg-schost-1
Type: SVM
```

```

failback: yes
Node List: phys-schost-1, phys-schost-2
preferenced: yes
numsecondaries: 1
diskset names: dg-schost-1

```

### 例 5-33 二次ノードの希望数の設定 (Veritas Volume Manager)

次に、デバイスグループ (dg-schost-1) の二次ノードの希望数を 2 に設定するとき、`clsetup` によって生成される `cldevicegroup` コマンドの例を示します。デバイスグループを作成したあとで二次ノードの希望数を変更する方法については、[166 ページの「デバイスグループの二次ノードの希望数を設定する」](#)を参照してください。

```

cldevicegroup set -p numsecondaries=2 dg-schost-1

cldevicegroup show dg-schost-1
=== Device Groups ===

Device Group Name: dg-schost-1
Type: VxVM
failback: yes
Node List: phys-schost-1, phys-schost-2
preferenced: yes
numsecondaries: 1
diskgroup names: dg-schost-1

```

### 例 5-34 二次ノードの希望数のデフォルト値への設定

次に、ヌル文字列値を使用して、二次ノードのデフォルト数を構成する例を示します。デバイスグループは、デフォルト値が変更されても、デフォルト値を使用するように構成されます。

```

cldevicegroup set -p numsecondaries= dg-schost-1
cldevicegroup show -v dg-schost-1

=== Device Groups ===

Device Group Name: dg-schost-1
Type: SVM
failback: yes
Node List: phys-schost-1, phys-schost-2 phys-schost-3
preferenced: yes
numsecondaries: 1
diskset names: dg-schost-1

```

## ▼ デバイスグループ構成の一覧を表示する

構成の一覧を表示するために、スーパーユーザーになる必要はありません。ただし、`solaris.cluster.read` の権限は必要です。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 次に示されている方法のどれかを選択してください。

|                                               |                                                                              |
|-----------------------------------------------|------------------------------------------------------------------------------|
| Sun Cluster Manager GUI                       | 詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。                              |
| <code>cldevicegroup show</code>               | <code>cldevicegroup show</code> を使用して、クラスタ内のすべてのデバイスグループの構成を一覧表示します。         |
| <code>cldevicegroup show devicegroup</code>   | <code>cldevicegroup show devicegroup</code> を使用して、1つのデバイスグループの構成を一覧表示します。    |
| <code>cldevicegroup status devicegroup</code> | <code>cldevicegroup status devicegroup</code> を使用して、1つのデバイスグループのステータスを判別します。 |
| <code>cldevicegroup status +</code>           | <code>cldevicegroup status +</code> を使用して、クラスタ内のすべてのデバイスグループのステータスを判別します。    |

詳細情報を表示するには、上記のコマンドと `-v` オプションを使用します。

### 例 5-35 すべてのデバイスグループのステータスの一覧表示

```
cldevicegroup status +

=== Cluster Device Groups ===

--- Device Group Status ---

Device Group Name Primary Secondary Status

dg-schost-1 phys-schost-2 phys-schost-1 Online
dg-schost-2 phys-schost-1 -- Offline
```

```
dg-schost-3 phys-schost-3 phy-shost-2 Online
```

### 例 5-36 特定のデバイスグループの構成の一覧表示

```
cldevicegroup show dg-schost-1
```

```
=== Device Groups ===
```

```
Device Group Name: dg-schost-1
Type: SVM
failback: yes
Node List: phys-schost-2, phys-schost-3
preferenced: yes
numsecondaries: 1
diskset names: dg-schost-1
```

## ▼ デバイスグループの主ノードを切り替える

次の手順は、アクティブでないデバイスグループを起動する(オンラインにする)ときにも使用できます。

Sun Cluster Manager GUI を使用すると、アクティブでないデバイスグループをオンラインにしたり、デバイスグループの主ノードを切り替えることができます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供するプロファイルを使用します。
- 2 `cldevicegroup switch` を使用して、デバイスグループの主ノードを切り替えます。  

```
cldevicegroup switch -n nodename devicegroup
```

`-n nodename` 切り替え先のノードの名前を指定します。このノードが新しい主ノードになります。

`devicegroup` 切り替えるデバイスグループを指定します。

- 3 デバイスグループが新しい主ノードに切り替わったことを確認します。  
デバイスグループが適切に登録されている場合、次のコマンドを使用すると、新しいデバイスグループの情報が表示されます。

```
cldevice status devicegroup
```

### 例 5-37 デバイスグループの主ノードの切り替え

次に、デバイスグループの主ノードを切り替えて変更結果を確認する例を示します。

```
cldevicegroup switch -n phys-schost-1 dg-schost-1
```

```
cldevicegroup status dg-schost-1
```

```
=== Cluster Device Groups ===
```

```
--- Device Group Status ---
```

| Device Group Name | Primary       | Secondary     | Status |
|-------------------|---------------|---------------|--------|
| dg-schost-1       | phys-schost-1 | phys-schost-2 | Online |

## ▼ デバイスグループを保守状態にする

デバイスグループを保守状態にすることによって、デバイスのいずれかにアクセスされたときに、デバイスグループが自動的にオンラインになることを防ぎます。デバイスグループを保守状態にする必要があるのは、修理手順において、修理が終わるまで、すべての入出力活動を停止する必要がある場合などです。また、デバイスグループを保守状態にすることによって、別のノード上のディスクセットまたはディスクグループを修復していても、当該ノード上のデバイスグループはオンラインにならないため、データの損失を防ぎます。

---

注-デバイスグループを保守状態にする前に、そのデバイスへのすべてのアクセスを停止し、依存するすべてのファイルシステムをマウント解除する必要があります。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 デバイスグループを保守状態にします。
  - a. デバイスグループが有効である場合は、デバイスグループを無効にします。  
`# cldevicegroup disable devicegroup`
  - b. デバイスグループをオフラインにします。  
`# cldevicegroup offline devicegroup`
- 2 修理手順を実行するときに、ディスクセットまたはディスクグループの所有権が必要な場合は、ディスクセットまたはディスクグループを手動でインポートします。  
Solaris ボリュームマネージャー の場合:  
`# metaset -C take -f -s diskset`



注意 - Solaris ボリュームマネージャー ディスクセットの所有権を取得する場合、デバイスグループが保守状態にあるときは、`metaset -C take` コマンドを使用する必要があります。`metaset -t` を使用すると、所有権の取得作業の一部として、デバイスグループがオンラインになります。VxVM ディスクグループをインポートする場合、ディスクグループをインポートするときは、`-t` フラグを使用する必要があります。`-t` フラグを使用することで、当該ノードが再起動した場合に、ディスクグループが自動的にインポートされることを防ぎます。

Veritas Volume Manager の場合:

```
vxdg -t import disk-group-name
```

- 3 必要な修理手順を実行します。
- 4 ディスクセットまたはディスクグループの所有権を解放します。



注意 - デバイスグループを保守状態から戻す前に、ディスクセットまたはディスクグループの所有権を解放する必要があります。所有権を解放しないと、データが失われる可能性があります。

- Solaris ボリュームマネージャー の場合:

```
metaset -C release -s diskset
```

- Veritas Volume Manager の場合:

```
vxdg deport diskgroupname
```

- 5 デバイスグループをオンラインにします。  
`# cldevicegroup online devicegroup`  
`# cldevicegroup enable devicegroup`

### 例 5-38 デバイスグループを保守状態にする

次に、デバイスグループ dg-schost-1 を保守状態にし、保守状態からデバイスグループを削除する方法の例を示します。

```
[Place the device group in maintenance state.]
cldevicegroup disable dg-schost-1
cldevicegroup offline dg-schost-1
[If needed, manually import the disk set or disk group.]
For Solaris ボリュームマネージャー:
 # metaset -C take -f -s dg-schost-1
For Veritas Volume Manager:
 # vxvg -t import dg1

[Complete all necessary repair procedures.]

[Release ownership.]
For Solaris ボリュームマネージャー:
 # metaset -C release -s dg-schost-1
For Veritas Volume Manager:
 # vxvg deport dg1

[Bring the device group online.]
cldevicegroup online dg-schost-1
cldevicegroup enable dg-schost-1
```

## ストレージデバイス用の SCSI プロトコル設定の管理

Sun Cluster ソフトウェアをインストールすると、自動的に、すべてのストレージデバイスに SCSI リザベーションが割り当てられます。次の手順に従って、複数のデバイスの設定を確認し、必要に応じてデバイスの設定を上書きします。

- 175 ページの「すべてのストレージデバイスのデフォルトのグローバルな SCSI プロトコル設定を表示する」
- 176 ページの「単一ストレージデバイスの SCSI プロトコルを表示する」
- 176 ページの「すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定を変更する」
- 178 ページの「単一ストレージデバイスのフェンシングプロトコルを変更する」

## ▼ すべてのストレージデバイスのデフォルトのグローバルな SCSI プロトコル設定を表示する

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 スーパーユーザーになるか、RBAC の承認 solaris.cluster.read を提供する役割になります。
- 2 任意のノードから、現在のグローバルなデフォルト SCSI プロトコル設定を表示します。

```
cluster show -t global
```

詳細は、[cluster\(1CL\)](#) のマニュアルページを参照してください。

### 例 5-39 すべてのストレージデバイスのデフォルトのグローバルな SCSI プロトコル設定の表示

次の例に、クラスタ上のすべてのストレージデバイスの SCSI プロトコル設定を示します。

```
cluster show -t global
```

```
=== Cluster ===
```

```
Cluster Name: racerxx
installmode: disabled
heartbeat_timeout: 10000
heartbeat_quantum: 1000
private_netaddr: 172.16.0.0
private_netmask: 255.255.248.0
max_nodes: 64
max_privatenets: 10
global_fencing: pathcount
Node List: phys-racerxx-1, phys-racerxx-2
```

## ▼ 単一ストレージデバイスの SCSI プロトコルを表示する

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 スーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` を提供する役割になります。
- 2 任意のノードから、ストレージデバイスの SCSI プロトコル設定を表示します。

```
cldevice show device
```

`device` デバイスパスの名前またはデバイス名。

詳細は、[cldevice\(ICL\)](#) のマニュアルページを参照してください。

### 例 5-40 単一デバイスの SCSI プロトコルの表示

次の例に、デバイス `/dev/rdisk/c4t8d0` の SCSI プロトコルを示します。

```
cldevice show /dev/rdisk/c4t8d0
```

```
=== DID Device Instances ===
```

|                   |                          |
|-------------------|--------------------------|
| DID Device Name:  | /dev/did/rdsk/d3         |
| Full Device Path: | phappy1:/dev/rdsk/c4t8d0 |
| Full Device Path: | phappy2:/dev/rdsk/c4t8d0 |
| Replication:      | none                     |
| default_fencing:  | global                   |

## ▼ すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定を変更する

フェンシングは、クラスタに接続されているすべてのストレージデバイスに対して、グローバルにオンまたはオフに設定できます。あるストレージデバイスのデ

フォルトのフェンシングが `pathcount`、`prefer3`、または `nofencing` に設定されている場合、そのデバイスのデフォルトのフェンシング設定は、グローバル設定よりも優先されます。ストレージデバイスのデフォルトのフェンシング設定が `global` に設定されている場合、ストレージデバイスはグローバル設定を使用します。たとえば、ストレージデバイスのデフォルト設定が `pathcount` である場合、ここでの手順を使用してグローバルな SCSI プロトコル設定を `prefer3` に変更しても、設定は変更されません。単一デバイスのデフォルト設定を変更するには、178 ページの「[単一ストレージデバイスのフェンシングプロトコルを変更する](#)」の手順を使用します。



注意 - フェンシングを誤ってオフに設定すると、アプリケーションのフェイルオーバー時にデータの破損する可能性が生じやすくなります。フェンシングをオフに設定する場合は、そのような状況でもデータが破損しないかどうか十分に検査してください。共有ストレージデバイスが SCSI プロトコルをサポートしていない場合や、クラスタのストレージへのアクセスをクラスタ外のホストに対して許可する場合は、フェンシングをオフに設定できません。

定足数デバイスのデフォルトのフェンシング設定を変更するには、デバイスの構成を解除し、フェンシング設定を変更して、定足数デバイスを再構成します。フェンシングをオフに設定したあとで、定足数デバイスを含むデバイスについては定期的にオンに戻す場合は、定足数サーバーサービスを利用して定足数を構成することを検討してください (そうすることで、定足数の動作を中断せずに済みます)。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 定足数デバイスではないすべてのストレージデバイスのフェンシングプロトコルを設定します。

```
cluster set -p global_fencing={pathcount | prefer3 | nofencing | nofencing-noscrub}
```

`-p global_fencing`                   すべての共有デバイスの現在のグローバルなデフォルトフェンシングアルゴリズムを設定します。

`prefer3`                               パスが2より多いデバイスに対して SCSI-3 プロトコルを使用します。

|                   |                                                                                                                                                 |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| pathcount         | 共有デバイスに接続されている DID パスの数でフェンシングプロトコルを決定します。pathcount 設定は、定足数デバイスで使用されます。                                                                         |
| nofencing         | フェンシングをオフに設定します (すべてのストレージデバイスについてフェンシングステータスを設定します)。                                                                                           |
| nofencing-noscrub | ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスが消去され、クラスタの外側にあるシステムからストレージへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大な問題があるストレージデバイスに対してのみ使用してください。 |

**例 5-41** すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定の設定

次の例では、クラスタ上のすべてのストレージデバイスのフェンシングプロトコルを、SCSI-3 プロトコルに設定します。

```
cluster set -p global_fencing=prefer3
```

## ▼ 単一ストレージデバイスのフェンシングプロトコルを変更する

フェンシングプロトコルは、1つのストレージデバイスに対して設定することもできます。

---

注- 定足数デバイスのデフォルトのフェンシング設定を変更するには、デバイスの構成を解除し、フェンシング設定を変更して、定足数デバイスを再構成します。フェンシングをオフに設定したあとで、定足数デバイスを含むデバイスについては定期的にオンに戻す場合は、定足数サーバーサービスを利用して定足数を構成することを検討してください (そうすることで、定足数の動作を中断せずに済みます)。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。



注意-フェンシングを誤ってオフに設定すると、アプリケーションのフェイルオーバー時にデータの破損する可能性が生じやすくなります。フェンシングをオフに設定する場合は、そのような状況でもデータが破損しないかどうか十分に検査してください。共有ストレージデバイスが SCSI プロトコルをサポートしていない場合や、クラスタのストレージへのアクセスをクラスタ外のホストに対して許可する場合は、フェンシングをオフに設定できません。

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 ストレージデバイスのフェンシングプロトコルを設定します。

```
cldevice set -p default_fencing ={pathcount | \
scsi3 | global | nofencing | nofencing-noscrub} device
```

|                    |                                                                                                                                                     |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| -p default_fencing | デバイスの default_fencing プロパティを変更します。                                                                                                                  |
| pathcount          | 共有デバイスに接続されている DID パスの数でフェンシングプロトコルを決定します。                                                                                                          |
| scsi3              | SCSI-3 プロトコルを使用します。                                                                                                                                 |
| 大域 (global)        | グローバルなデフォルトのフェンシング設定を使用します。global 設定は、定足数デバイス以外のデバイスで使用されます。<br><br>指定された DID インスタンスのフェンシングステータスを設定することで、フェンシングをオフに設定します。                           |
| nofencing-noscrub  | ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスが消去され、クラスタの外側にあるシステムからストレージデバイスへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大な問題があるストレージデバイスに対してのみ使用してください。 |
| device             | デバイスパスの名前またはデバイス名を指定します。                                                                                                                            |

詳細は、`cluster(1CL)` のマニュアルページを参照してください。

### 例 5-42 単一デバイスのフェンシングプロトコルの設定

次の例では、(デバイス番号で指定される)デバイス d5 を SCSI-3 プロトコルに設定します。

```
cldevice set -p default_fencing=prefer3 d5
```

次の例では、d11 デバイスのデフォルトフェンシングをオフに設定します。

```
#cldevice set -p default_fencing=nofencing d11
```

## クラスタファイルシステムの管理

クラスタファイルシステムは、クラスタのどのノードからでも読み取りやアクセスが可能なグローバルなファイルシステムです。

表 5-5 作業リスト:クラスタファイルシステムの管理

| 作業                                                  | 参照先                            |
|-----------------------------------------------------|--------------------------------|
| クラスタファイルシステムを Sun Cluster の最初のインストール後に追加する          | 180 ページの「クラスタファイルシステムを追加する」    |
| クラスタファイルシステムを削除する                                   | 185 ページの「クラスタファイルシステムを削除する」    |
| クラスタ内のグローバルマウントポイントをチェックして、ノード間の一貫性が保たれているかどうかを確認する | 187 ページの「クラスタ内のグローバルマウントを確認する」 |

### ▼ クラスタファイルシステムを追加する

次の作業は、Sun Cluster の初期インストール後に作成するクラスタファイルシステムごとに実行します。



注意-必ず、正しいディスクデバイス名を指定してください。クラスタファイルシステムを作成すると、ディスク上のデータはすべて消去されます。デバイス名を誤って指定すると、本来消去する必要のないデータを失うことになります。

クラスタファイルシステムを追加する前に、次の必要条件が満たされていることを確認します。

- クラスタ内のノード上でスーパーユーザー特権が確立されていること。

- ボリュームマネージャーソフトウェアがクラスタ上にインストールおよび構成されていること。
- クラスタファイルシステムの作成先がデバイスグループ (Solaris ボリュームマネージャー デバイスグループまたは VxVM デバイスグループ)、またはブロックディスクスライスであること。

Sun Cluster Manager を使用してデータサービスをインストールした場合は、クラスタファイルシステムがすでに自動的に作成されています (十分な共有ディスクが存在する場合)。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

## 1 クラスタ内にある任意のノード上でスーパーユーザーになります。

Solaris 10 OS 上で、グローバル以外のゾーンがクラスタで構成されている場合、グローバルゾーンからこの手順を実行してください。

---

ヒント-ファイルシステムを迅速に作成するには、ファイルシステムを作成するグローバルデバイスの現在の主ノードでスーパーユーザーになります。

---

## 2 ファイルシステムを作成する。



**Caution**-ファイルシステムを作成するとき、ディスク上のデータは破壊されます。必ず、正しいディスクデバイス名を指定してください。間違ったデバイス名を指定した場合、削除するつもりのないデータが削除されてしまいます。

---

- **UFS** ファイルシステムの場合は、`newfs(1M)` コマンドを使用します。

`phys-schost# newfs raw-disk-device`

下の表に、引数 `raw-disk-device` の名前例を挙げます。命名規約はボリューム管理ソフトウェアごとに異なるので注意してください。

| ボリューム管理ソフトウェア       | ディスクデバイス名の例                       | 説明                                         |
|---------------------|-----------------------------------|--------------------------------------------|
| Solaris ボリュームマネージャー | <code>/dev/md/nfs/rdisk/d1</code> | nfs ディスクセット内の raw ディスクデバイス <code>d1</code> |

| ボリューム管理ソフトウェア          | ディスクデバイス名の例               | 説明                             |
|------------------------|---------------------------|--------------------------------|
| Veritas Volume Manager | /dev/vx/rdisk/oradg/vol01 | oradg ディスクセット内の raw デバイス vol01 |
| なし                     | /dev/global/rdisk/d1s3    | raw ディスクデバイス d1s3              |

- **Veritas File System (VxFS)** ファイルシステムの場合、**VxFS** ドキュメントに記載されている手順を実行してください。

- 3 クラスタ内の各ノードで、クラスタファイルシステムのマウントポイントのディレクトリを作成します。

そのノードからはクラスタファイルシステムにアクセスしない場合でも、マウントポイントはノードごとに必要です。

ヒント-管理を行いやすくするには、マウントポイントを `/global/device-group/` ディレクトリに作成します。この場所を使用すると、グローバルに利用できるクラスタファイルシステムとローカルファイルシステムを区別しやすくなります。

```
phys-schost# mkdir -p /global/device-group/mountpoint/
```

*device-group* デバイスが含まれるデバイスグループ名に対応するディレクトリ名を指定します。

*mountpoint* クラスタファイルシステムのマウント先のディレクトリ名を指定します。

- 4 クラスタ内の各ノードで、マウントポイント用の `/etc/vfstab` ファイルにエントリを追加します。

詳細については、`vfstab(4)` のマニュアルページを参照してください。

注-クラスタに非大域ゾーンが設定されている場合は、大域ゾーンのクラスタファイルシステムを必ず大域ゾーンのルートディレクトリのパスにマウントしてください。

- a. 各エントリで、使用する種類のファイルシステムに必要なマウントオプションを指定します。

---

注 - Solaris ボリュームマネージャートランザクションボリュームには、`logging` マウントオプションは使用しないでください。トランザクションボリュームでは、独自のロギングが実行されます。

その他、Solaris ボリュームマネージャートランザクションボリュームロギングが Solaris 10 OS から削除されます。Solaris UFS ロギングは、より低い管理条件とオーバーヘッドで、同様の機能を高いパフォーマンスで提供します。

---

- b. クラスタファイルシステムを自動的にマウントするには、`mount at boot` フィールドを `yes` に設定します。
- c. 各クラスタファイルシステムで、`/etc/vfstab` エントリの情報が各ノードで同じになるようにします。
- d. 各ノードの `/etc/vfstab` ファイルのエントリに、デバイスが同じ順序で表示されることを確認します。
- e. ファイルシステムの起動順の依存関係を検査します。

たとえば、`phys-schost-1` がディスクデバイス `d0` を `/global/oracle/` にマウントし、`phys-schost-2` がディスクデバイス `d1` を `/global/oracle/logs/` にマウントすると仮定します。この構成では、`phys-schost-1` が起動され、`/global/oracle/` がマウントされたあとにのみ、`phys-schost-2` を起動し、`/global/oracle/logs/` をマウントできます。

- 5 クラスタの任意のノード上で、設定確認ユーティリティを実行します。

```
phys-schost# cluster check -k vfstab
```

設定確認ユーティリティは、マウントポイントが存在することを確認します。また、`/etc/vfstab` ファイルのエントリが、クラスタのすべてのノードで正しいことを確認します。エラーが発生していない場合は、何も戻されません。

詳細は、`cluster(1CL)` のマニュアルページを参照してください。

- 6 クラスタファイルシステムをマウントします。

```
phys-schost# mount /global/device-group/mountpoint/
```

- **UFS** の場合は、クラスタ内の任意のノードから、クラスタファイルシステムをマウントします。
- **VxFS** の場合、`device-group` の現在のマスタからクラスタファイルシステムをマウントして、ファイルシステムを確実にマウントします。

さらに、VxFS ファイルシステムを正しく確実にマウント解除するには、`device-group` の現在のマスタからファイルシステムをマウント解除します。

---

注 - VxFS クラスタファイルシステムを Sun Cluster 環境で管理するには、VxFS クラスタファイルシステムがマウントされている主ノードだけから管理コマンドを実行します。

---

- 7 クラスタ内にある各ノード上で、クラスタファイルシステムがマウントされていることを確認します。

df コマンドまたは mount コマンドのいずれかを使用し、マウントされたファイルシステムの一覧を表示します。詳細は、[df\(1M\)](#) マニュアルページまたは [mount\(1M\)](#) マニュアルページを参照してください。

Solaris 10 OS の場合、クラスタファイルシステムは大域ゾーンおよび非大域ゾーンの両方からアクセスできます。

### 例 5-43 UFS クラスタファイルシステムの作成

次に、Solaris ボリュームマネージャボリューム /dev/md/oracle/rdisk/d1 上に、UFS クラスタファイルシステムを作成する例を示します。各ノードの `vfstab` ファイルにクラスタファイルシステムのエントリが追加されます。次に、1つのノードから `cluster check` コマンドを実行します。設定確認プロセスが正しく終了すると、1つのノードからクラスタファイルシステムがマウントされ、全ノードで確認されます。

```
phys-schost# newfs /dev/md/oracle/rdisk/d1
...
phys-schost# mkdir -p /global/oracle/d1
phys-schost# vi /etc/vfstab
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
/dev/md/oracle/dsk/d1 /dev/md/oracle/rdisk/d1 /global/oracle/d1 ufs 2 yes global,logging
...
phys-schost# cluster check -k vfstab
phys-schost# mount /global/oracle/d1
phys-schost# mount
...
/global/oracle/d1 on /dev/md/oracle/dsk/d1 read/write/setuid/global/logging/largefiles
on Sun Oct 3 08:56:16 2005
```

## ▼ クラスタファイルシステムを削除する

クラスタファイルシステムを削除するには、単に、そのクラスタファイルシステムのマウントを解除します。データも削除する場合は、配下のディスクデバイス(またはメタデバイスかボリューム)をシステムから削除します。

---

注 - クラスタファイルシステムは、`cluster shutdown` を実行してクラスタ全体を停止したときに、システム停止処理の一環として自動的にマウント解除されません。`shutdown` を実行して単独でノードを停止したときはマウント解除されません。なお、停止するノードが、ディスクに接続されている唯一のノードの場合は、そのディスク上のクラスタファイルシステムにアクセスしようとするエラーが発生します。

---

クラスタファイルシステムをマウント解除する前に、次の必要条件が満たされていることを確認します。

- クラスタ内のノード上でスーパーユーザー特権が確立されていること。
- ファイルシステムが使用中でないこと。ファイルシステムが使用中と見なされるのは、ユーザーがファイルシステム内のディレクトリにアクセスしている場合や、プログラムがファイルシステム内のファイルを開いている場合です。ユーザーやプログラムは、クラスタ内のどのノードでもアクセスできます。

1 クラスタ内にある任意のノード上でスーパーユーザーになります。

2 マウントされているクラスタファイルシステムを確認します。

```
mount -v
```

3 各ノードで、クラスタファイルシステムを使用中の全プロセスの一覧を表示し、停止するプロセスを判断します。

```
fuser -c [-u] mountpoint
```

-c                   ファイルシステムのマウントポイントとなっているファイルと、マウントされているファイルシステム内のファイルがすべて表示されます。

-u                   (任意)各プロセス ID のユーザーログイン名を表示します。

*mountpoint*       プロセスを停止するクラスタファイルシステムの名前を指定します。

- 4 各ノードで、クラスタファイルシステムのプロセスをすべて停止します。  
プロセスは任意の方法で停止できます。必要であれば、次のコマンドを使用して、クラスタファイルシステムに関するプロセスを強制終了してください。  
**# fuser -c -k mountpoint**  
クラスタファイルシステムを使用している各ノードに SIGKILL が送信されます。
- 5 各ノードで、ファイルシステムを使用しているプロセスがないことを確認します。  
**# fuser -c mountpoint**
- 6 1つのノードからファイルシステムをマウント解除します。  
**# umount mountpoint**  
  
*mountpoint*            マウント解除するクラスタファイルシステムの名前を指定します。クラスタファイルシステムがマウントされているディレクトリの名前や、ファイルシステムのデバイス名パスを指定できます。
- 7 (任意) /etc/vfstab ファイルを編集して、削除するクラスタファイルシステムのエントリーを削除します。  
この手順は、/etc/vfstab ファイルにこのクラスタファイルシステムのエントリーがある各クラスタノードで実行してください。
- 8 (任意) ディスクデバイス group/metadevice/volume/plex を削除します。  
詳細については、ボリューム管理ソフトウェアのマニュアルを参照してください。

#### 例 5-44 クラスタファイルシステムの削除

次に、Solaris ボリュームマネージャー メタデバイスまたはボリューム /dev/md/oracle/rdisk/d1 にマウントされた UFS クラスタファイルシステムを削除する例を示します。

```
mount -v
...
/global/oracle/d1 on /dev/md/oracle/dsk/d1 read/write/setuid/global/logging/largefiles
fuser -c /global/oracle/d1
/global/oracle/d1: 4006c
fuser -c -k /global/oracle/d1
/global/oracle/d1: 4006c
fuser -c /global/oracle/d1
/global/oracle/d1:
umount /global/oracle/d1
```

(On each node, remove the highlighted entry:)

```
vi /etc/vfstab
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
/dev/md/oracle/dsk/d1 /dev/md/oracle/rdisk/d1 /global/oracle/d1 ufs 2 yes global,logging
```

[Save and exit.]

クラスタファイルシステム上のデータを削除するには、配下のデバイスを削除します。詳細については、ボリューム管理ソフトウェアのマニュアルを参照してください。

## ▼ クラスタ内のグローバルマウントを確認する

`cluster(1CL)/etc/vfstab` ファイルのユーティリティはクラスタファイルシステムのエントリ構文を検証します。エラーが発生していない場合は、何も戻されません。

---

注-クラスタファイルシステムの削除など、デバイスやボリューム管理コンポーネントに影響を及ぼすような変更をクラスタ構成に加えたあとに `cluster check` コマンドを実行します。

---

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 クラスタのグローバルマウントを確認します。

```
cluster check -k vfstab
```

## ディスクパス監視の管理

ディスクパス監視 (DPM) の管理コマンドを使用すれば、二次ディスクパス障害の通知を受け取ることができます。この節では、ディスクパスの監視に必要な管理作業を行うための手順を説明します。ディスクパス監視デーモンの概念については、『[Sun Cluster Concepts Guide for Solaris OS](#)』の第3章「[Key Concepts for System Administrators and Application Developers](#)」を参照してください。 `scdpmd` コマンドオプションと関連するコマンドについては、`cldevice(1CL)` マニュアルページを参照してください。 `scdpmd` デーモンの調整に関する詳細は、`scdpmd.conf(4)` マニュアルページを参照してください。デーモンがレポートするログエラーに関しては、`syslogd(1M)` マニュアルページも参照してください。

注-cldevice コマンドを使ってノードに入出力デバイスを追加すると、監視を行っていた監視リストにディスクバスが自動的に追加されます。Sun Cluster コマンドを使ってノードからデバイスを削除すると、ディスクバスは自動的に監視から除外されます。

表 5-6 作業マップ: ディスクバス監視の管理

| 作業                                                                                                     | 参照先                                                                                                    |
|--------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| ディスクバスを監視します。                                                                                          | 188 ページの「ディスクバスを監視する」                                                                                  |
| ディスクバスの監視を解除します。                                                                                       | 190 ページの「ディスクバスの監視を解除する方法」                                                                             |
| あるノードに対する障害のあるディスクバスのステータスを表示します。                                                                      | 191 ページの「障害のあるディスクバスを表示する」                                                                             |
| ファイルからディスクバスを監視します。                                                                                    | 192 ページの「ファイルからディスクバスを監視する」                                                                            |
| 監視しているすべての共有ディスクバスが失敗したときのノードの自動再起動を有効化または無効化します。                                                      | 195 ページの「監視しているすべての共有ディスクバスが失敗したときのノードの自動再起動を有効にする」<br>195 ページの「すべての監視共有ディスクバスが失敗した場合にノードの自動再起動を無効にする」 |
| 不正なディスクバス状態を解決します。起動時に監視対象の DID デバイスを利用できず、DID インスタンスが DID ドライバにアップロードされない場合、不正なディスクバス状態が報告されることがあります。 | 192 ページの「ディスクバスの状態エラーを解決する」                                                                            |

cldevice コマンドを実行する以下のセクションの手順にはディスクバス引数が含まれます。ディスクバス引数はノード名とディスク名からなります。ただし、ノード名は必須ではありません。指定しないと、all が使用されます。

## ▼ ディスクバスを監視する

この作業は、クラスタのディスクバスを監視するときに行います。



注意 - DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。
- 2 ディスクパスを監視します。  
# cldevice monitor -n node disk
- 3 ディスクパスが監視されているか確認します。  
# cldevice status device

#### 例 5-45 単一ノードのディスクパスを監視

次の例では、単一ノードから schost-1:/dev/did/rdisk/d1 ディスクパスを監視します。ディスク /dev/did/dsk/d1 へのパスを監視するのは、ノード schost-1 上の DPM デーモンだけです。

```
cldevice monitor -n schost-1 /dev/did/dsk/d1
cldevice status d1
```

| Device Instance   | Node          | Status |
|-------------------|---------------|--------|
| /dev/did/rdisk/d1 | phys-schost-1 | Ok     |

#### 例 5-46 すべてのノードのディスクパスを監視

次の例では、すべてのノードから schost-1:/dev/did/dsk/d1 ディスクパスを監視します。DPM は、/dev/did/dsk/d1 が有効なパスであるすべてのノードで起動されます。

```
cldevice monitor /dev/did/dsk/d1
cldevice status /dev/did/dsk/d1
```

| Device Instance   | Node          | Status |
|-------------------|---------------|--------|
| /dev/did/rdisk/d1 | phys-schost-1 | Ok     |

### 例 5-47 CCR からディスク構成を読み直す

次の例では、デーモンが CCR からディスク構成を読み直し、監視されているディスクバスをそのステータスとともに出力します。

```
cldevice monitor +
cldevice status
Device Instance Node Status

/dev/did/rdisk/d1 schost-1 Ok
/dev/did/rdisk/d2 schost-1 Ok
/dev/did/rdisk/d3 schost-1 Ok
 schost-2 Ok
/dev/did/rdisk/d4 schost-1 Ok
 schost-2 Ok
/dev/did/rdisk/d5 schost-1 Ok
 schost-2 Ok
/dev/did/rdisk/d6 schost-1 Ok
 schost-2 Ok
/dev/did/rdisk/d7 schost-2 Ok
/dev/did/rdisk/d8 schost-2 Ok
```

## ▼ ディスクバスの監視を解除する方法

ディスクバスの監視を解除する場合は、この手順を使用します。



注意-DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。
- 2 監視を解除するディスクバスの状態を調べます。

```
cldevice status device
```

- 3 各ノードで、適切なディスクパスの監視を解除します。

```
cldevice unmonitor -n node disk
```

#### 例 5-48 ディスクパスの監視解除

次の例では、schost-2:/dev/did/rdisk/d1 ディスクパスの監視を解除し、クラスタ全体のディスクパスの一覧とそのステータスを出力します。

```
cldevice unmonitor -n schost2 /dev/did/rdisk/d1
cldevice status -n schost2 /dev/did/rdisk/d1
```

| Device Instance   | Node     | Status      |
|-------------------|----------|-------------|
| -----             | ----     | -----       |
| /dev/did/rdisk/d1 | schost-2 | Unmonitored |

## ▼ 障害のあるディスクパスを表示する

クラスタに障害のあるディスクパスを表示する場合は、次の手順を使用します。



注意 - DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 全クラスタ内の障害のあるディスクパスを表示します。

```
cldevice status -s fail
```

#### 例 5-49 障害のあるディスクパスを表示する

次の例では、全クラスタ内の障害のあるディスクパスを表示します。

```
cldevice status -s fail
```

| Device Instance | Node          | Status |
|-----------------|---------------|--------|
| -----           | ----          | -----  |
| dev/did/dsk/d4  | phys-schost-1 | fail   |

## ▼ ディスクパスの状態エラーを解決する

次のイベントが発生すると、DPM が障害の発生したパスがオンラインになっても、そのパスの状態を更新しない可能性があります。

- 監視対象パスの障害によって、ノードが再起動する。
- 再起動したノードがオンラインに戻るまで、監視対象の DID パスの下のデバイスがオンラインに戻らない。

起動時に監視対象の DID デバイスを利用できず、このため DID インスタンスが DID ドライバにアップロードされないため、不正なディスクパス状態が報告されます。このような状態が発生する場合は、手動で DID 情報を更新します。

- 1 1つのノードからグローバルデバイス名前空間を更新します。

```
cldevice populate
```

- 2 次の手順に進む前に、各ノードでコマンド処理が完了していることを確認します。このコマンドは、1つのノードからのみ実行されても、リモートからすべてのノードで実行されます。コマンドが処理を終了したかどうかを確認するには、クラスタの各ノードで次のコマンドを実行します。

```
ps -ef | grep scgdevs
```

- 3 DPM ポーリングタイムフレーム内で障害の発生したディスクパスの状態が OK になっていることを確認します。

```
cldevice status disk-device
```

| Device Instance | Node          | Status |
|-----------------|---------------|--------|
| -----           | ----          | -----  |
| dev/did/dsk/dN  | phys-schost-1 | Ok     |

## ▼ ファイルからディスクパスを監視する

ファイルを使ってディスクパスを監視したり、その監視を解除する場合は、次の手順を使用します。

ファイルを使用してクラスタ構成を変更するには、まず現在の構成をエクスポートします。このエクスポート操作により XML ファイルが作成されます。このファイルは、変更する構成項目を設定するために修正できます。この手順では、このプロセス全体を説明します。



注意 - DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

- 2 デバイス構成を XML ファイルにエクスポートします。

```
cldevice export -o configurationfile
```

```
-o configurationfile XML ファイルのファイル名を指定します。
```

- 3 デバイスパスが監視されるよう、構成ファイルを変更します。  
監視するデバイスパスを検索し、`monitored` 属性を `true` に設定します。

- 4 デバイスパスを監視します。

```
cldevice monitor -i configurationfile
```

```
-i configurationfile 変更された XML ファイルのファイル名を指定します。
```

- 5 この時点でデバイスパスが監視されていることを確認します。

```
cldevice status
```

#### 例 5-50 ファイルからディスクパスを監視する

次の例では、ノード `phys-schost-2` とデバイス `d3` の間のデバイスパスが、XML ファイルを使用することによって監視されています。

最初に、現在のクラスタ構成をエクスポートします。

```
cldevice export -o deviceconfig
```

deviceconfig XML ファイルは、phys-schost-2 と d3 の間のバスが現在は監視されていないことを示しています。

```
<?xml version="1.0"?>
<!DOCTYPE cluster SYSTEM "/usr/cluster/lib/xml/cluster.dtd">
<cluster name="brave_clus">
.
.
.
 <deviceList readonly="true">
 <device name="d3" ctd="c1t8d0">
 <devicePath nodeRef="phys-schost-1" monitored="true"/>
 <devicePath nodeRef="phys-schost-2" monitored="false"/>
 </device>
 </deviceList>
</cluster>
```

そのバスを監視するには、次のように、監視される attribute を true に設定します。

```
<?xml version="1.0"?>
<!DOCTYPE cluster SYSTEM "/usr/cluster/lib/xml/cluster.dtd">
<cluster name="brave_clus">
.
.
.
 <deviceList readonly="true">
 <device name="d3" ctd="c1t8d0">
 <devicePath nodeRef="phys-schost-1" monitored="true"/>
 <devicePath nodeRef="phys-schost-2" monitored="true"/>
 </device>
 </deviceList>
</cluster>
```

cldevice コマンドを使用して、ファイルを読み込み、監視を有効にします。

```
cldevice monitor -i deviceconfig
```

cldevice コマンドを使用して、この時点でデバイスが監視されていることを確認します。

```
cldevice status
```

**参照** クラスタ構成のエクスポート、および結果の XML ファイルを使用したクラスタ構成の設定の詳細は、[cluster\(1CL\)](#) および [clconfiguration\(5CL\)](#) のマニュアルページを参照してください。

## ▼ 監視しているすべての共有ディスクパスが失敗したときのノードの自動再起動を有効にする

この機能を有効にすると、次の条件が満たされる場合、ノードは自動的に再起動します。

- ノード上ですべての監視対象の共有ディスクパスが失敗した。
- 少なくとも1つのディスクがクラスタ内の異なるノードからアクセス可能である。

ノードが再起動すると、そのノード上でマスターされているすべてのリソースグループとデバイスグループが別のノード上で再起動します。

ノードが自動再起動したあと、ノード上のすべての監視対象共有ディスクパスがアクセス不能のままである場合、そのノードは再び自動再起動しません。しかし、ノードが再起動したが失敗したあとに、利用可能になったディスクパスがある場合、そのノードは再び自動再起動します。

`reboot_on_path_failure` プロパティを有効にすると、ローカルディスクパスの状態は、ノードの再起動が必要かどうか決定するときには考慮されません。監視された共有ディスクのみが影響を受けます。

- 1 クラスタ内の任意のノードで、スーパーユーザーになるか、`solaris.cluster.modify RBAC` の承認を提供する役割になります。
- 2 クラスタのすべてのノードに対して、監視共有ディスクパスがすべて失敗したときの、ノードの自動再起動を有効にします。

```
clnode set -p reboot_on_path_failure=enabled +
```

## ▼ すべての監視共有ディスクパスが失敗した場合にノードの自動再起動を無効にする

この機能を無効にすると、あるノード上のすべての監視共有ディスクパスに障害が発生しても、ノードは自動的に再起動しません。

- 1 クラスタ内の任意のノードで、スーパーユーザーになるか、`solaris.cluster.modify RBAC` の承認を提供する役割になります。
- 2 クラスタ内のすべてのノードに対して、ノードへのすべての監視共有ディスクパスに障害が発生した場合の、ノードの自動再起動を無効にします。

```
clnode set -p reboot_on_path_failure=disabled +
```



## 定足数の管理

---

この章では、Sun Cluster および Sun Cluster 定足数サーバー内の定足数デバイスの管理手順について説明します。定足数の概念については、『[Sun Cluster Concepts Guide for Solaris OS](#)』の「[Quorum and Quorum Devices](#)」を参照してください。

- 197 ページの「[定足数デバイスの管理](#)」
- 225 ページの「[Sun Cluster 定足数サーバーの管理](#)」

### 定足数デバイスの管理

定足数デバイスとは、複数のノードによって共有される共有ストレージデバイスまたは定足数サーバーで、定足数を確立するために使用される票を構成します。このセクションでは、定足数デバイスを管理するための手順について説明します。

`clquorum(1CL)` コマンドを使用すると、定足数デバイスの管理手順をすべて実行できます。また、`clsetup(1CL)` 対話型ユーティリティーや Sun Cluster Manager GUI を使用しても、いくつかの管理手順を実行できます。このセクションの管理手順は、可能な限り `clsetup` ユーティリティーを使用して説明してあります。GUI を使用して定足数手順を実行する方法については、Sun Cluster Manager のオンラインヘルプを参照してください。定足数デバイスを使用して作業する際は、次のガイドラインに注意してください。

- Solaris 10 OS を実行している場合、すべての定足数コマンドはグローバルクラスタの投票ノードで実行する必要があります。
- `clquorum` コマンドが中断または失敗すると、定足数の構成情報は、クラスタ構成データベースで矛盾することになります。このような矛盾が発生した場合は、このコマンドを再度実行するか、`clquorum reset` コマンドを実行して定足数構成をリセットします。
- クラスタの可用性を最高にするには、定足数デバイスによる合計の投票数が、ノードによる合計の投票数よりも少なくなるようにします。少なくなければ、すべてのノードが機能していても、すべての定足数デバイスを使用できない場合、そのノードはクラスタを形成できません。

- 現在定足数デバイスとして構成されているディスクは、Solaris ZFS ストレージプールには追加しないでください。構成済みの定足数デバイスを Solaris ZFS ストレージプールに追加すると、ディスクは EFI ディスクとしてラベルが変更され、また定足数構成情報が失われ、ディスクはクラスタへの定足数投票を提供しなくなります。ディスクがストレージプール内に入ると、そのディスクは定足数デバイスとして構成できます。または、ディスクの定足数デバイス構成を解除し、ディスクをストレージプールに追加した後に、そのディスクを定足数デバイスとして再構成することができます。

注-clsetup コマンドは、ほかの Sun Cluster コマンドに対する対話型インタフェースです。clsetup の実行時、このコマンドは適切な固有のコマンドを生成します。今回の場合は、clquorum コマンドです。これらのコマンドは、各説明の後にある例の中で示しています。

定足数構成を表示するには、clquorum show を使用します。clquorum list コマンドは、クラスタ内の定足数デバイスの名前を表示します。clquorum status コマンドは、状態と投票数の情報を提供します。

このセクションで示す例は、主に3ノードクラスタです。

表 6-1 作業リスト:定足数の管理

作業	説明
clsetup(ICL) を使用することで、定足数デバイスをクラスタに追加する	200 ページの「定足数デバイスの追加」
clsetup を使用する (clquorum を生成することにより、クラスタから定足数デバイスを削除する)	213 ページの「定足数デバイスを削除する」
clsetup を使用する (clquorum を生成することにより、クラスタから最後の定足数デバイスを削除する)	215 ページの「クラスタから最後の定足数デバイスを削除する」
追加と削除の手順を使用することで、クラスタ内の定足数デバイスを交換する	216 ページの「定足数デバイスを交換する」
追加と削除の手順を使用することで、定足数デバイスのリストを変更する	217 ページの「定足数デバイスのノードリストを変更する」

表 6-1 作業リスト:定足数の管理 (続き)

作業	説明
clsetup を使用する (clquorum を生成する) ことにより、定足数デバイスを保守状態にする  (保守状態にある場合、定足数デバイスは定足数確立の投票に参加しません。)	220 ページの「定足数デバイスを保守状態にする」
clsetup を使用する (clquorum を生成する) ことにより、定足数構成をデフォルト状態にリセットする	221 ページの「定足数デバイスを保守状態から戻す」
clquorum(1CL) コマンドを使用することで、定足数デバイスと投票数を一覧表示する	223 ページの「クラスタ構成を一覧表示する」

## 定足数デバイスへの動的再構成

クラスタ内の定足数デバイス上で動的再構成 (DR) を実行するときには、いくつかの問題を考慮する必要があります。

- Solaris の DR 機能の説明で述べられているすべての必要条件、手順、制限は Sun Cluster の DR サポートにも適用されます (オペレーティングシステムでの休止操作をしない場合を除く)。したがって、Sun Cluster ソフトウェアで DR 機能を使用する前に、必ず、Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- Sun Cluster は、定足数デバイス用に構成されたインタフェースが存在する場合 DR 削除操作を実行できません。
- DR 操作がアクティブなデバイスに影響する場合、Sun Cluster はその操作を拒否して、その操作から影響を受けるデバイスを識別します。

定足数デバイスを削除するには、次の手順をその順番どおりに行います。

表 6-2 作業マップ:定足数デバイスへの動的再構成

作業	説明
1. 削除する定足数デバイスと交換する、新しい定足数デバイスを有効に設定	200 ページの「定足数デバイスの追加」
2. 削除する定足数デバイスを無効に設定	213 ページの「定足数デバイスを削除する」

表 6-2 作業マップ: 定足数デバイスへの動的再構成 (続き)

作業	説明
3. 削除する定足数デバイス上で DR 削除操作を実行	「Solaris 9 on Sun Hardware」コレクションと「Solaris 10 on Sun Hardware」コレクションの『Sun Enterprise 10000 DR 構成マニュアル』と『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』

## 定足数デバイスの追加

この節では、定足数デバイスを追加する手順について説明します。クラスタのすべてのノードがオンラインか確認してから、新しい定足数デバイスを追加します。クラスタに必要な定足数投票数を確認する方法、推奨される定足数構成、障害回避などについては、『Sun Cluster Concepts Guide for Solaris OS』の「Quorum and Quorum Devices」を参照してください。



注意 - 現在定足数デバイスとして構成されているディスクは、Solaris ZFS ストレージプールには追加しないでください。構成済みの定足数デバイスを Solaris ZFS ストレージプールに追加すると、ディスクは EFI ディスクとしてラベルが変更され、また定足数構成情報が失われ、ディスクはクラスタへの定足数投票を提供しなくなります。ディスクがストレージプール内に入ると、そのディスクは定足数デバイスとして構成できます。ディスクの定足数デバイス構成を解除し、ディスクをストレージプールに追加したあとに、そのディスクを定足数デバイスとして再構成することもできます。

Sun Cluster ソフトウェアは、次の種類の定足数デバイスをサポートしています。

- 直接接続共有-ディスク (SCSI または Serial Attached Technology Attachment (SATA) デバイスの場合)
- Sun NAS
- Network Appliance (NetApp) NAS
- Sun Cluster 定足数サーバー

これらのデバイスを追加する方法については、次の節で説明しています。

- 201 ページの「共有ディスク定足数デバイスを追加する」
- 206 ページの「Network Appliance ネットワーク接続ストレージ (NAS) 定足数デバイスを追加する」
- 209 ページの「定足数サーバー定足数をデバイスとして追加する」

---

注-複製されたディスクを定足数デバイスとして構成することはできません。複製されたディスクを定足数デバイスとして追加しようとすると、次のエラーメッセージが表示され、コマンドはエラーコードとともに終了します。

*Disk-name* is a replicated device. Replicated devices cannot be configured as quorum devices.

---

共有ディスク定足数デバイスは、Sun Cluster ソフトウェアがサポートする任意の接続済みストレージデバイスです。共有ディスクは、クラスタの複数のノードに接続されます。フェンシングをオンに設定すると、デュアルポートのディスクを定足数デバイスとして構成して、SCSI-2またはSCSI-3(デフォルトはSCSI-2)を使用できます。フェンシングがオンに設定され、共有デバイスが3つ以上のノードに接続されている場合は、SCSI-3 プロトコル(2ノードを超える場合のデフォルトのプロトコル)を使用する定足数デバイスとして共有ディスクを構成できます。SCSI オーバーライドフラグを使用すると、デュアルポートの共有ディスクでSCSI-3 プロトコルを使用するように Sun Cluster ソフトウェアに対して指示できます。

共有ディスクのフェンシングをオフに設定した場合は、ソフトウェア定足数プロトコルを使用する定足数デバイスとしてディスクを構成できます。これは、そのディスクがSCSI-2とSCSI-3のどちらのプロトコルをサポートしている場合でも有効です。ソフトウェア定足数は、Sun Microsystemsが開発したプロトコルで、SCSI Persistent Group Reservations (PGR) の形式をエミュレートします。



---

注意-使用するディスクがSCSI (SATA など) をサポートしていない場合は、SCSI フェンシングをオフにするようにしてください。

---

定足数デバイスには、ユーザーデータを持つディスクまたはデバイスグループのメンバーであるディスクを使用できます。共有ディスクがある定足数サブシステムで使用されているプロトコルは、`cluster show` コマンドの出力の、共有ディスクの `access-mode` 値で確認します。

これらの作業は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

次の手順で使用されるコマンドについては、`clsetup(1CL)` および `clquorum(1CL)` のマニュアルページを参照してください。

## ▼ 共有ディスク定足数デバイスを追加する

Sun Cluster ソフトウェアでは、共有ディスク (SCSI と SATA の両方) デバイスを定足数デバイスとして使用できます。SATA デバイスはSCSI 予約をサポートしていないため、その種類のディスクを定足数デバイスとして構成するには、SCSI 予約フェンシングフラグをオフに設定し、ソフトウェア定足数プロトコルを使用します。

この手順を実行するには、ノードが共有するデバイス ID (Device Identifier、DID) によりディスクドライブを確認します。cldevice show コマンドを使用して、DID 名の一覧を参照します。詳細は、[cldevice\(1CL\)](#) のマニュアルページを参照してください。クラスタのすべてのノードがオンラインか確認してから、新しい定足数デバイスを追加します。

次の手順を実行して、SCSI または SATA デバイスを構成します。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。
- 2 clsetup ユーティリティーを起動します。  
# clsetup  
clsetup のメインメニューが表示されます。
- 3 定足数のオプションに対応する番号を入力します。  
「定足数メニュー」が表示されます。
- 4 定足数デバイスを追加するためのオプションに対応する番号を入力し、追加する定足数デバイスを確認するメッセージ clsetup ユーティリティーのプロンプトが表示されたら「yes」と入力します。  
追加する定足数デバイスの種類を確認するメッセージが表示されます。
- 5 共有ディスク定足数デバイスのオプションに対応する番号を入力します。  
どのグローバルデバイスを使用するかを確認するメッセージが表示されます。
- 6 使用しているグローバルデバイスを入力します。  
指定したグローバルデバイスに新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。
- 7 「yes」と入力し、新しい定足数デバイスの追加を続行します。  
新しい定足数デバイスが正常に追加されると、clsetup ユーティリティーではその旨のメッセージが表示されます。

- 8 定足数デバイスが追加されていることを確認します。

```
clquorum list -v
```

### 例 6-1 共有ディスク定足数デバイスの追加

次の例は、共有ディスク定足数デバイスを追加する際に `clsetup` によって生成される `clquorum` コマンドと、検証ステップを示しています。

Become superuser or assume a role that provides `solaris.cluster.modify` RBAC authorization on any cluster node.

```
[Start the clsetup utility:]
clsetup
[Select Quorum>Add a quorum device]
[Answer the questions when prompted.]
[You will need the following information.]
 [Information: Example:]
 [Directly attached shared disk shared_disk]
 [Global device d20]

[Verify that the clquorum command was completed successfully:]
clquorum add d20

 Command completed successfully.
[Quit the clsetup Quorum Menu and Main Menu.]
[Verify that the quorum device is added:]
clquorum list -v
```

```
Quorum Type

d20 shared_disk
scphyshost-1 node
scphyshost-2 node
```

### ▼ Sun NAS 定足数デバイスを追加する

この手順を実行するには、ノードが共有するデバイス ID (Device Identifier、DID) によりディスクドライブを確認します。 `cldevice show` コマンドを使用して、DID 名の一覧を参照します。詳細は、[cldevice\(1CL\)](#) のマニュアルページを参照してください。クラスタのすべてのノードがオンラインか確認してから、新しい定足数デバイスを追加します。

---

注 - Sun Cluster は、2 ノードクラスタ構成でのみ Sun NAS 定足数デバイスをサポートしています。

---

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

1 Sun NAS GUI を使用して、Sun NAS ファイラ上で iSCSI デバイスを設定します。

a. 約 50 M バイトのサイズのファイルボリュームを作成します。

```
File Volume Operations -> Create File Volume
```

b. 各ノードで、iSCSI アクセスリストを作成します。

```
iSCSI Configuration -> Configure Access List
```

i. クラスタの名前を iSCSI アクセスリスト名として使用します。

ii. 各クラスタノードのイニシエータノード名をアクセスリストに追加します。CHAP および IQN は不要です。

c. iSCSI LUN を構成します。

```
iSCSI Configuration -> Configure iSCSI LUN
```

バッキングファイルボリュームの名前を LUN の名前として使用できます。各ノードのアクセスリストを LUN に追加します。

2 各クラスタノードで、iSCSI LUN を検出して、iSCSI アクセスリストを静的構成に設定します。

```
iscsiadm modify discovery -s enable
```

```
iscsiadm list discovery
```

```
Discovery:
```

```
Static: enabled
```

```
Send Targets: disabled
```

```
iSNS: disabled
```

```
iscsiadm add static-config iqn.LUNName,IPAddress_of_NASDevice
```

```
devfsadm -i iscsi
```

```
cldevice refresh
```

3 1つのクラスタノードから DID を iSCSI LUN 用に構成します。

```
/usr/cluster/bin/scgdevs
```

4 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。

- 5 `clsetup`ユーティリティを起動します。  
# `clsetup`  
`clsetup`のメインメニューが表示されます。
- 6 定足数のオプションに対応する番号を入力します。  
「定足数メニュー」が表示されます。
- 7 定足数デバイスを追加するためのオプションに対応する番号を入力し、追加する定足数デバイスを確認するメッセージ `clsetup`ユーティリティのプロンプトが表示されたら「**yes**」と入力します。  
追加する定足数デバイスの種類を確認するメッセージが表示されます。
- 8 共有ディスク定足数デバイスのオプションに対応する番号を入力します。  
どのグローバルデバイスを使用するかを確認するメッセージが表示されます。
- 9 使用しているグローバルデバイスを入力します。  
指定したグローバルデバイスに新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。
- 10 「**yes**」と入力し、新しい定足数デバイスの追加を続行します。  
新しい定足数デバイスが正常に追加されると、`clsetup`ユーティリティではその旨のメッセージが表示されます。
- 11 定足数デバイスが追加されていることを確認します。  
# `clquorum list -v`

## 例6-2 Sun NAS 定足数デバイスの追加

次の例は、Sun NAS 定足数デバイスを追加する際に `clsetup` によって生成される `clquorum` コマンドと、検証ステップを示しています。

```
Add an iSCSI device on the Sun NAS filer.
Use the Sun NAS GUI to create a file volume that is approximately 50mb in size.
File Volume Operations -> Create File Volume
For each node, create an iSCSI access list.
iSCSI Configuration -> Configure Access List
Add the initiator node name of each cluster node to the access list.
*** Need GUI or command syntax for this step. ***
Configure the iSCSI LUN
iSCSI Configuration -> Configure iSCSI LUN
On each of the cluster nodes, discover the iSCSI LUN and set the iSCSI access list to static configuration.
iscsiadm modify discovery -s enable
iscsiadm list discovery
```

Discovery:

```
Static: disable
Send Targets: enables
iSNS: disabled
```

iscsiadm add status-config

```
iqn.1986-03.com.sun0-1:000e0c66efe8.4604DE16.thinqorum,10.11.160.20
```

devsadm -i iscsi

From one cluster node, configure the DID devices for the iSCSI LUN.

```
/usr/cluster/bin/scgdevs
```

```
/usr/cluster/bin/scgdevs
```

Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.

[Start the clsetup utility:]

```
clsetup
```

[Select Quorum>Add a quorum device]

[Answer the questions when prompted.]

[You will need the following information.]

```
[Information: Example:]
[Directly attached shared disk device shared_disk]
[Global device d20]
```

[Verify that the clquorum command was completed successfully:]

```
clquorum add d20
```

Command completed successfully.

[Quit the clsetup Quorum Menu and Main Menu.]

[Verify that the quorum device is added:]

```
clquorum list -v
```

Quorum	Type
-----	----
d20	shared_disk
scphyshost-1	node
scphyshost-2	node

## ▼ Network Appliance ネットワーク接続ストレージ (NAS) 定足数デバイスを追加する

Network Appliance (NetApp) ネットワーク接続ストレージ (Network-Attached Storage、NAS) デバイスを定足数デバイスとして使用する場合は、次の要件があります。

- NetApp の iSCSI ライセンスをインストールする必要があります。
- クラスタ化されたファイラを定足数デバイスとして使用する場合は、そのファイラに iSCSI LUN を構成する必要があります。
- 時間の同期をとるために NTP を使用するには、NetApp NAS ユニットの構成する必要があります。

- クラスタ化されたファイラに選択されているNTPサーバーのうち少なくとも1つは、Sun Cluster ノードのNTPサーバーでなければなりません。
- クラスタを起動する場合は、常にクラスタノードを起動する前にNASデバイスを起動する必要があります。  
誤った順序でデバイスを起動すると、ノードは定足数デバイスを検出できません。このような状況でノードが停止した場合、クラスタはサービスに対応できなくなる可能性があります。サービスの中断が発生した場合は、クラスタ全体を起動し直すか、NetApp NAS 定足数デバイスを削除して追加し直す必要があります。
- クラスタは、各NASデバイスを単一の定足数デバイスにしか使用できません。  
定足数デバイスがさらに必要な場合は、ほかの共有ストレージを構成できません。同じNASデバイスを使用するほかのクラスタは、そのデバイスの別のLUNをそれらの定足数デバイスとして使用できます。

Sun Cluster 環境で NetApp NAS ストレージデバイスをインストールする方法については、Sun Cluster のマニュアル『[Sun Cluster 3.1 - 3.2 With Network-Attached Storage Devices Manual for Solaris OS](#)』。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

- 1 すべての **Sun Cluster** ノードがオンライン状態であり、クラスタ化された **NetApp** ファイラと通信が行えることを確認します。
- 2 クラスタの任意のノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.modify` を提供する役割になります。
- 3 `clsetup` ユーティリティーを起動します。  
# `clsetup`  
`clsetup` のメインメニューが表示されます。
- 4 定足数のオプションに対応する番号を入力します。  
「定足数メニュー」が表示されます。
- 5 定足数デバイスを追加するためのオプションに対応する番号を入力します。定足数デバイスを追加することを確認するには、「**yes**」と入力します。  
追加する定足数デバイスの種類を確認するメッセージが表示されます。

- 6 **netapp\_nas** 定足数デバイスのオプションに対応する番号を入力します。**netapp\_nas** 定足数デバイスを追加することを確認するには、「**yes**」と入力します。  
新しい定足数デバイスの名前を入力するように、**clsetup** ユーティリティのプロンプトが表示されます。
- 7 追加する定足数デバイスの名前を入力します。  
定足数デバイスの名前は任意に選択できます。この名前は、今後の管理コマンドの処理だけに使用されるものです。  
新しい定足数デバイスのファイラの名前を入力するように、**clsetup** ユーティリティのプロンプトが表示されます。
- 8 新しい定足数デバイスのファイラの名前を入力します。  
この名前には、ネットワークアクセスが可能なファイラ名またはファイラのアドレスを指定してください。  
ファイラの LUN ID を指定するように、**clsetup** ユーティリティのプロンプトが表示されます。
- 9 ファイラの定足数デバイス LUN の ID を入力します。  
ファイラに新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。
- 10 「**yes**」と入力し、新しい定足数デバイスの追加を続行します。  
新しい定足数デバイスが正常に追加されると、**clsetup** ユーティリティではその旨のメッセージが表示されます。
- 11 定足数デバイスが追加されていることを確認します。  
`# clquorum list -v`

### 例 6-3 NetApp NAS 定足数デバイスの追加

次の例は、NetApp NAS 定足数デバイスを追加する際に **clsetup** によって生成される **clquorum** コマンドを示しています。またこの例では検証ステップも示します。

**Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.**

```
[Start the clsetup utility:]
clsetup
[Select Quorum>Add a quorum device]
[Answer the questions when prompted.]
[You will need the following information.]
[Information: Example:]
[Quorum Device Netapp_nas quorum device]
[Name: qd1]
```

```
[Filer: nas1.sun.com]
[LUN ID: 0]
```

```
[Verify that the clquorum command was completed successfully:]
clquorum add -t netapp_nas -p filer=nas1.sun.com,-p lun_id=0 qd1
```

Command completed successfully.

```
[Quit the clsetup Quorum Menu and Main Menu.]
```

```
[Verify that the quorum device is added:]
```

```
clquorum list -v
```

```
Quorum Type

qd1 netapp_nas
scphyshost-1 node
scphyshost-2 node
```

## ▼ 定足数サーバー定足数をデバイスとして追加する

始める前に Sun Cluster 定足数サーバーを定足数デバイスとして追加するには、Sun Cluster 定足数サーバーソフトウェアがホストマシン上にインストールされ、定足数サーバーが起動され実行中である必要があります。定足数サーバーのインストールについては、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「[定足数サーバーソフトウェアをインストールして構成する](#)」を参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。
- 2 すべての Sun Cluster ノードがオンライン状態であり、Sun Cluster 定足数サーバーと通信が行えることを確認します。
  - a. クラスタノードに直接接続されているネットワークスイッチが次の基準のいずれかを満たすことを確認します。
    - スイッチは RSTP (Rapid Spanning Tree Protocol) をサポートしています。
    - スイッチ上で高速ポートモードが有効になっています。

クラスタノードと定足数サーバー間ですぐに通信できるようにするには、これらの機能の 1 つが必要です。この通信がスイッチによって大幅に遅延すると、クラスタはこの通信の中断を定足数デバイスが失われたものと解釈します。

- b. パブリックネットワークで可変長サブネット化(CIDR (Classless Inter-Domain Routing) と呼ばれる)を使用している場合は、各ノードで次のファイルを変更します。

クラスフルサブネットを使用する場合は、これらの手順を実行する必要はありません。

- i. `/etc/inet/netmasks` ファイルにクラスタが使用する各パブリックサブネットのエントリを追加します。

パブリックネットワークの IP アドレスとネットマスクを含むエントリの例を次に示します。

```
10.11.30.0 255.255.255.0
```

- ii. それぞれの `/etc/hostname.adapter` ファイルに `netmask + broadcast +` を追加します。

```
nodename netmask + broadcast +
```

- c. クラスタ内の各ノード上で、定足数サーバーのホスト名を `/etc/inet/hosts` ファイルまたは `/etc/inet/ipnodes` ファイルに追加します。

次のように、ホスト名とアドレスのマッピングをファイルに追加します。

```
ipaddress qshost1
```

```
ipaddress 定足数サーバーが実行中であるコンピュータの IP アドレス。
```

```
qshost1 定足数サーバーが実行中であるコンピュータのホスト名。
```

- d. ネームサービスを使用する場合、定足数サーバーホストの名前とアドレスの対応付けをネームサービスデータベースに追加します。

- 3 `clsetup` ユーティリティーを起動します。

```
clsetup
```

`clsetup` のメインメニューが表示されます。

- 4 定足数のオプションに対応する番号を入力します。

「定足数メニュー」が表示されます。

- 5 定足数デバイスを追加するためのオプションに対応する番号を入力します。定足数デバイスを追加することを確認するには、「**yes**」と入力します。

追加する定足数デバイスの種類を確認するメッセージが表示されます。

- 6 **quorum\_server** 定足数デバイスのオプションに対応する番号を入力します。 **quorum\_server** 定足数デバイスを追加することを確認するには、「**yes**」と入力します。  
新しい定足数デバイスの名前を入力するように、**clsetup** ユーティリティーのプロンプトが表示されます。
- 7 追加する定足数デバイスの名前を入力します。  
定足数デバイスの名前は任意に選択できます。この名前は、今後の管理コマンドの処理だけに使用されるものです。  
新しい定足数デバイスのファイラの名前を入力するように、**clsetup** ユーティリティーのプロンプトが表示されます。
- 8 定足数サーバーのホストの名前を入力します。  
この名前で、定足数サーバーが動作するマシンの IP アドレス、またはネットワーク上のマシンのホスト名を指定します。  
ホストの IPv4 または IPv6 構成に応じて、マシンの IP アドレスを `/etc/hosts` ファイル、`/etc/inet/ipnodes` ファイル、またはその両方で指定します。

---

注-指定したマシンはすべてのクラスタノードから到達可能で、定足数サーバーをマシン上で実行してある必要があります。

---

**clsetup** ユーティリティーは、定足数サーバーのポート番号を入力するようメッセージを表示します。

- 9 クラスタノードとやり取りする際に定足数サーバーが使用するポート番号を入力します。  
新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。
- 10 「**yes**」と入力し、新しい定足数デバイスの追加を続行します。  
新しい定足数デバイスが正常に追加されると、**clsetup** ユーティリティーではその旨のメッセージが表示されます。
- 11 定足数デバイスが追加されていることを確認します。

```
clquorum list -v
```

#### 例 6-4 定足数サーバー定足数デバイスの追加

次の例は、定足数サーバー定足数デバイスを追加する際に **clsetup** によって生成される **clquorum** コマンドを示しています。またこの例では検証ステップも示します。

Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.

[Start the clsetup utility:]

**# clsetup**

[Select Quorum > Add a quorum device]

[Answer the questions when prompted.]

[You will need the following information.]

```
[Information: Example:]
[Quorum Device quorum_server quorum device]
[Name: qd1]
[Host Machine Name: 10.11.124.84]
[Port Number: 9001]
```

[Verify that the clquorum command was completed successfully:]

**clquorum add -t quorum\_server -p qshost=10.11.124.84, -p port=9001 qd1**

Command completed successfully.

[Quit the clsetup Quorum Menu and Main Menu.]

[Verify that the quorum device is added:]

**# clquorum list -v**

```
Quorum Type

qd1 quorum_server
scphyshost-1 node
scphyshost-2 node
```

**# clquorum status**

=== Cluster Quorum ===

-- Quorum Votes Summary --

Needed	Present	Possible
-----	-----	-----
3	5	5

-- Quorum Votes by Node --

Node Name	Present	Possible	Status
-----	-----	-----	-----
phys-schost-1	1	1	Online
phys-schost-2	1	1	Online

-- Quorum Votes by Device --

Device Name	Present	Possible	Status
-----	-----	-----	-----

qd1	1	1	Online
d3s2	1	1	Online
d4s2	1	1	Online

## 定足数デバイスの削除または交換

この節では、定足数デバイスを削除または交換するための次の手順を説明します。

- 213 ページの「定足数デバイスを削除する」
- 215 ページの「クラスタから最後の定足数デバイスを削除する」
- 216 ページの「定足数デバイスを交換する」

### ▼ 定足数デバイスを削除する

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

定足数デバイスを削除すると、そのデバイスは定足数確立の投票に参加できなくなります。2 ノードクラスタでは、定足数デバイスが少なくとも1つは構成されている必要があります。構成されているデバイスが、クラスタの最後の定足数デバイスの場合は、**clquorum(1CL)** はデバイスを構成から削除できません。ノードを削除する場合は、そのノードに接続されている定足数デバイスをすべて削除してください。

---

注-削除するデバイスがクラスタの最後の定足数デバイスの場合は、[215 ページの「クラスタから最後の定足数デバイスを削除する」](#)の手順を参照してください。

---

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 削除する定足数デバイスを判別します。  
# `clquorum list -v`
- 3 `clsetup(1CL)` ユーティリティーを実行します。  
# `clsetup`

メインメニューが表示されます。

- 4 定足数のオプションに対応する番号を入力します。
- 5 定足数デバイスを削除するオプションに対応する番号を入力します。  
削除プロセス中に表示される質問に答えます。
- 6 `clsetup`を終了します。
- 7 定足数デバイスが削除されたことを確認します。

```
clquorum list -v
```

### 例 6-5 定足数デバイスの削除

次に、2つ以上の定足数デバイスが構成されているクラスタから定足数デバイスを削除する例を示します。

Become superuser or assume a role that provides `solaris.cluster.modify` RBAC authorization on any cluster node.

```
[Determine the quorum device to be removed:]
clquorum list -v
[Start the clsetup utility:]
clsetup
[Select Quorum>Remove a quorum device]
[Answer the questions when prompted.]
Quit the clsetup Quorum Menu and Main Menu.]
[Verify that the quorum device is removed:]
clquorum list -v
```

```
Quorum Type

scphyshost-1 node
scphyshost-2 node
scphyshost-3 node
```

**注意事項** 定足数サーバー定足数デバイスの削除中に、クラスタと定足数サーバーホストの間の通信が失われた場合、定足数サーバーホストに関する無効な構成情報をクリーンアップする必要があります。このクリーンアップの実行に関する説明は、[229 ページ](#)の「[期限切れの定足数サーバークラスタ情報のクリーンアップ](#)」を参照してください。

## ▼ クラスタから最後の定足数デバイスを削除する

この手順では、クラスタから最後の定足数デバイスを削除します。この手順は、2 ノードクラスタがあり、最後の定足数デバイスを削除する場合にのみ使用します。いずれかのノードに障害が発生した場合は、`-F` オプションを使用して、デバイスを削除します。通常、不具合が起きたデバイスをまず削除し、代替りの定足数デバイスを追加します。

定足数デバイスを追加する処理では、ノードが再構成されるため、障害のあった定足数デバイスに影響が及び、マシンでパニックが発生します。`F` (強制) オプションを使用すると、マシンでパニックを発生させることなく、障害があった定足数デバイスを削除できます。`clquorum(1CL)` コマンドでは、構成からデバイスを削除することができます。不具合が発生した定足数デバイスを削除したあと、`clquorum add` コマンドで新しいデバイスを追加することができます。[200 ページの「定足数デバイスの追加」](#)を参照してください。

削除するデバイスが2 ノードクラスタの最後のデバイスでない場合は、[213 ページの「定足数デバイスを削除する」](#)の手順に従ってください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 `clquorum` コマンドを使用して定足数デバイスを削除します。定足数デバイスに障害が発生した場合は、`-F` (強制) オプションを使用して、そのデバイスを削除します。

```
clquorum remove -F qd1
```

---

注-また、削除するノードを保持状態とし、定足数デバイスを `clquorum removequorum` コマンドを使用して削除することができます。`clsetup(1CL)` クラスタ管理メニューは、クラスタがインストールモードのときは使用できません。詳細は、[285 ページの「ノードを保守状態にする」](#)を参照してください。

---

- 3 定足数デバイスが削除されたことを確認します。

```
clquorum list -v
```

## 例 6-6 最後の定足数デバイスの削除

この例では、クラスタを保持モードにし、クラスタ構成で最後の定足数デバイスを削除する方法を示しています。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.]
[Place the cluster in install mode:]
cluster set -p installmode=enabled
[Remove the quorum device:]
clquorum remove d3
[Verify that the quorum device has been removed:]
clquorum list -v
Quorum Type

scphyshost-1 node
scphyshost-2 node
scphyshost-3 node
```

### ▼ 定足数デバイスを交換する

この作業は、既存の定足数デバイスをほかの定足数デバイスに交換する場合に行います。定足数デバイスは、類似したデバイスタイプに交換することも(例: NAS デバイスをほかの NAS デバイスに置き換える)、あるいは類似点がないデバイスに交換することも(例: NAS デバイスを共有ディスクに置き換える)こともできます。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

#### 1 新しい定足数デバイスを構成します。

最初に、古いデバイスの代わりに、新しい定足数デバイスを構成に追加する必要があります。クラスタに新しい定足数デバイスを追加する方法は、[200 ページの「定足数デバイスの追加」](#)を参照してください。

#### 2 定足数デバイスとして交換するデバイスを削除します。

構成から古い定足数デバイスを削除する方法は、[213 ページの「定足数デバイスを削除する」](#)を参照してください。

- 3 定足数デバイスが障害が発生したディスクである場合は、ディスクを取り替えます。  
『[Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS](#)』で、使用しているディスク装置のハードウェア作業を参照してください。

## 定足数デバイスの保守

この節では、定足数デバイスを保守するための次の手順を説明します。

- 217 ページの「定足数デバイスのノードリストを変更する」
- 220 ページの「定足数デバイスを保守状態にする」
- 221 ページの「定足数デバイスを保守状態から戻す」
- 223 ページの「クラスタ構成を一覧表示する」
- 224 ページの「定足数デバイスを修復する」

### ▼ 定足数デバイスのノードリストを変更する

`clsetup(1CL)` ユーティリティーを使用すると、既存の定足数デバイスのノードリストにノードを追加したり、ノードリストからノードを削除したりできます。定足数デバイスのノードリストを変更するには、定足数デバイスを削除し、削除した定足数デバイスへのノードの物理的な接続を変更して、定足数デバイスをクラスタ構成に追加し直す必要があります。定足数デバイスを追加すると、`clquorum(1CL)` は自動的に、ディスクが接続されているすべてのノードについて、ノードからディスクへのパスを構成します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.modify` を提供する役割になります。

- 2 変更したい定足数デバイスの名前を判別します。

```
clquorum list -v
```

- 3 `clsetup` ユーティリティーを起動します。

```
clsetup
```

メインメニューが表示されます。

- 4 定足数オプションに対応する番号を入力します。  
「定足数メニュー」が表示されます。
- 5 定足数デバイスを削除するオプションに対応する番号を入力します。  
画面の指示に従います。削除するディスクの名前を問い合わせられます。
- 6 定足数デバイスへのノード接続を追加または削除します。
- 7 定足数デバイスを追加するオプションに対応する番号を入力します。  
画面の指示に従います。定足数デバイスとして使用するディスクの名前を問い合わせられます。
- 8 定足数デバイスが追加されていることを確認します。  
**# clquorum list -v**

### 例 6-7 定足数デバイスノードリストの変更

次の例に、`clsetup` ユーティリティを使用して、定足数デバイスのノードリストにノードを追加したり、ノードリストからノードを削除する方法を示します。この例では、定足数デバイスの名前は `d2` であり、この手順の最終目的は別のノードを定足数デバイスのノードリストに追加することです。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any node in the cluster.]
```

```
[Determine the quorum device name:]
```

```
clquorum list -v
Quorum Type

d2 shared_disk
sc-phys-schost-1 node
sc-phys-schost-2 node
sc-phys-schost-3 node
```

```
[Start the clsetup utility:]
```

```
clsetup
```

```
[Type the number that corresponds with the quorum option.]
```

```
.
```

```
[Type the number that corresponds with the option to remove a quorum device.]
```

```
.
```

```
[Answer the questions when prompted.]
```

```
[You will need the following information:]
```

```

Information: Example:
Quorum Device Name: d2

```

[Verify that the clquorum command completed successfully:]

```

clquorum remove d2
 Command completed successfully.

```

[Verify that the quorum device was removed.]

```

clquorum list -v
Quorum Type

sc-phys-schost-1 node
sc-phys-schost-2 node
sc-phys-schost-3 node

```

[Type the number that corresponds with the Quorum option.]

.

[Type the number that corresponds with the option to add a quorum device.]

.

[Answer the questions when prompted.]

[You will need the following information:]

```

Information Example:
quorum device name d2

```

[Verify that the clquorum command was completed successfully:]

```

clquorum add d2
 Command completed successfully.

```

Quit the clsetup utility.

[Verify that the correct nodes have paths to the quorum device.  
In this example, note that phys-schost-3 has been added to the  
enabled hosts list.]

```

clquorum show d2 | grep Hosts
=== Quorum Devices ===

Quorum Device Name: d2
 Hosts (enabled): phys-schost-1, phys-schost-2, phys-schost-3

```

[Verify that the modified quorum device is online.]

```

clquorum status d2
=== Cluster Quorum ===

--- Quorum Votes by Device ---

```

Device Name	Present	Possible	Status
-----	-----	-----	-----
d2	1	1	Online

## ▼ 定足数デバイスを保守状態にする

`clquorum(1CL)` コマンドを使用して定足数デバイスを保守状態にします。現在、`clsetup(1CL)` ユーティリティーにこの機能はありません。この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

サービスから定足数デバイスを長時間はずす場合は、その定足数デバイスを保守状態にします。定足数デバイスの定足数投票数 (quorum vote count) はゼロに設定されるため、そのデバイスが稼働中でも定足数確立の投票には参加しません。保守状態でも定足数デバイスの構成情報は保持されます。

---

注-2 ノードクラスタでは、定足数デバイスが少なくとも1つは構成されている必要があります。構成されているデバイスが2ノードクラスタの最後の定足数デバイスの場合、`clquorum` は失敗してデバイスは保守状態になりません。

---

クラスタノードを保守状態にする方法については、[285 ページの「ノードを保守状態にする」](#)を参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

- 2 定足数デバイスを保守状態にします。

```
clquorum disable device
```

`device` 変更するディスクデバイスの DID 名 (d4 など) を指定します。

- 3 定足数デバイスが保守状態にあることを確認します。

保守状態にしたデバイスの出力は、定足数デバイスの投票数 (以下の例の Quorum device votes) がゼロになっていなければなりません。

```
clquorum status device
```

## 例 6-8 定足数デバイスを保守状態にする

次に、定足数デバイスを保守状態にし、結果を検証する例を示します。

```
clquorum disable d20
clquorum status d20

=== Cluster Quorum ===

--- Quorum Votes by Device ---

Device Name Present Possible Status

d20 1 1 Offline
```

参照 定足数デバイスを有効にし直す方法については、221 ページの「定足数デバイスを保守状態から戻す」を参照してください。

ノードを保守状態にする方法については、285 ページの「ノードを保守状態にする」を参照してください。

## ▼ 定足数デバイスを保守状態から戻す

この作業は、定足数デバイスが保守状態にある場合にその状態から定足数デバイスを戻して定足数投票数をデフォルトにリセットするときに実行します。



注意 -globaldev または node オプションのどちらも指定しない場合、定足数投票数はクラスタ全体でリセットされます。

定足数デバイスを構成する場合、Sun Cluster ソフトウェアは定足数デバイスに投票数として  $N-1$  を割り当てます ( $N$  は定足数デバイスに結合された投票の数)。たとえば、2 つのノードに接続された、投票数がゼロ以外の定足数デバイスの投票数は  $1(2-1)$  になります。

- クラスタノードと、そのクラスタノードに関係付けられた定足数デバイスを保守状態から戻す方法については、287 ページの「ノードを保守状態から戻す」を参照してください。
- 定足数投票数の詳細については『Sun Cluster Concepts Guide for Solaris OS』の「About Quorum Vote Counts」を参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

- 2 定足数投票数をリセットします。

```
clquorum enable device
```

*device* リセットする定足数デバイスの DID 名 (d4 など) を指定します。

- 3 ノードが保守状態にあったために定足数投票数をリセットする場合は、このノードを再起動します。

- 4 定足数投票数を確認します。

```
clquorum show +
```

#### 例 6-9 定足数投票数 (定足数デバイス) のリセット

次に、定足数デバイスの投票数をリセットしてデフォルト設定に戻し、結果を検証する例を示します。

```
clquorum enable d20
```

```
clquorum show +
```

```
=== Cluster Nodes ===
```

```
Node Name: phys-schost-2
Node ID: 1
Quorum Vote Count: 1
Reservation Key: 0x43BAC41300000001
```

```
Node Name: phys-schost-3
Node ID: 2
Quorum Vote Count: 1
Reservation Key: 0x43BAC41300000002
```

```
=== Quorum Devices ===
```

```
Quorum Device Name: d3
Enabled: yes
Votes: 1
```

```

Global Name: /dev/did/rdisk/d20s2
Type: shared_disk
Access Mode: scsi2
Hosts (enabled): phys-schost-2, phys-schost-3

```

## ▼ クラスタ構成を一覧表示する

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

定足数構成を一覧表示するには、スーパーユーザーになる必要はありません。RBAC の承認 `solaris.cluster.read` を提供する任意の役割になることができます。

---

注-定足数デバイスに対するノード接続の数を増減させる場合、定足数が自動的に再計算されることはありません。すべての定足数デバイスをいったん削除し、その後それらを構成に追加し直すと、正しい定足数が再設定されます。2 ノードクラスタの場合、定足数デバイスを取り外して、もとの定足数デバイスに戻す前に一時的に新しい定足数デバイスを追加します。次に一時的に追加した定足数デバイスを取り外します。

---

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- `clquorum(1CL)` を使用して、定足数構成を一覧表示します。

```
% clquorum show +
```

### 例 6-10 定足数構成の一覧表示

```
% clquorum show +
```

```
=== Cluster Nodes ===
```

```

Node Name: phys-schost-2
Node ID: 1
Quorum Vote Count: 1
Reservation Key: 0x43BAC41300000001

Node Name: phys-schost-3
Node ID: 2

```

```
Quorum Vote Count: 1
Reservation Key: 0x43BAC41300000002
```

```
=== Quorum Devices ===
```

```
Quorum Device Name: d3
Enabled: yes
Votes: 1
Global Name: /dev/did/rdisk/d20s2
Type: shared_disk
Access Mode: scsi2
Hosts (enabled): phys-schost-2, phys-schost-3
```

## ▼ 定足数デバイスを修復する

この作業は、動作が不正な定足数デバイスを交換する場合に行なってください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

### 1 定足数デバイスとして交換するディスクデバイスを削除します。

---

注-削除するデバイスが最後の定足数デバイスである場合は、必要に応じて初めにほかのディスクを新しい定足数デバイスとして追加してください。この手順により、交換作業中に障害が発生した場合も定足数デバイスが有効になります。新しい定足数デバイスを追加する方法については、[200 ページの「定足数デバイスの追加」](#)を参照してください。

---

定足数デバイスとしてのディスクデバイスを削除する方法については、[213 ページの「定足数デバイスを削除する」](#)を参照してください。

### 2 ディスクデバイスを交換します。

ディスクデバイスを交換する方法については、『[Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS](#)』に記されたディスク装置のハードウェア作業の説明を参照してください。

### 3 交換したディスクを新しい定足数デバイスとして追加します。

ディスクを新しい定足数デバイスとして追加する方法については、[200 ページの「定足数デバイスの追加」](#)を参照してください。

注-手順1で定足数デバイスを別途追加した場合は、デバイスを削除しても安全です。定足数デバイスを削除する方法については、213ページの「定足数デバイスを削除する」を参照してください。

## Sun Cluster 定足数サーバーの管理

QuorumServer は、共有ストレージデバイスではない、定足数デバイスを提供します。このセクションでは、Sun Cluster 定足数サーバーを管理するための次のような手順について説明します。

- 225 ページの「定足数サーバー構成ファイルの概要」
- 226 ページの「Sun Cluster 定足数サーバーソフトウェアの起動と停止」
- 226 ページの「定足数サーバーを起動する」
- 227 ページの「定足数サーバーを停止する」
- 228 ページの「定足数サーバーに関する情報の表示」
- 229 ページの「期限切れの定足数サーバークラスタ情報のクリーンアップ」

Sun Cluster Quorum Server のインストールと構成については、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の「定足数サーバーソフトウェアをインストールして構成する」を参照してください。

### 定足数サーバー構成ファイルの概要

Sun Cluster ソフトウェアをインストールすると、デフォルトの構成ファイルである `/etc/scqsd/scqsd.conf` が作成されます。このファイルには1つのデフォルトの定足数サーバーに関する情報が含まれています。`/etc/scqsd/scqsd.conf` ファイルの各行は、次のような形式になっています。

```
/usr/cluster/lib/sc/scqsd [-d quorumdirectory] [-i instancename] -p port
```

`/usr/cluster/lib/sc/scqsd` Sun Cluster ソフトウェアをインストールした場所へのフルパスです。この値は、`/usr/cluster/lib/sc/scqsd` である必要があります。

`-d quorumdirectory` 定足数サーバーが定足数データを格納できるディレクトリへのパスです。

クラスタ固有の定足数情報を格納するために、定足数サーバープロセスはこのディレクトリに1クラスタにつき1つのファイルを作成します。デフォルトでは、このオプションの値は `/var/scqsd` です。この

	ディレクトリは、ユーザーが構成する各定足数サーバーに対して一意にします。
<code>-i instancename</code>	定足数サーバーインスタンスに対してユーザーが選択する一意の名前です。
<code>-p port</code>	定足数サーバーがクラスタからの要求を待機するポート番号です。デフォルトのポートは9000です。

インスタンス名はオプションです。定足数サーバーに対して名前を指定する場合、その名前はシステム内のすべての定足数サーバー間で一意にします。インスタンス名のオプションを省略した場合は、定足数サーバーが待機するポートにより定足数サーバーを参照します。

## Sun Cluster 定足数サーバー ソフトウェアの起動と停止

次の手順では、Sun Cluster ソフトウェアを起動および停止する方法を説明します。

デフォルトでは、次の手順は、定足数サーバー構成ファイル `/etc/scqsd/scqsd.conf` の内容をカスタマイズしていない場合の、1つのデフォルト定足数サーバーを起動および停止します。デフォルトの定足数サーバーはポート9000上にバインドされ、定足数情報には `/var/scqsd` ディレクトリを使用します。

定足数サーバー構成ファイルをカスタマイズするには、[225 ページの「定足数サーバー構成ファイルの概要」](#)を参照してください。定足数サーバーのソフトウェアのインストールについては、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「[定足数サーバーソフトウェアをインストールして構成する](#)」を参照してください。

### ▼ 定足数サーバーを起動する

- 1 Sun Cluster ソフトウェアを起動するホスト上でスーパーユーザーになります。
- 2 ソフトウェアを起動するには、`clquorumserver start` コマンドを使用します。

```
/usr/cluster/bin/clquorumserver start quorumserver
```

`quorumserver` 定足数サーバーを識別します。定足数サーバーが待機するポート番号を使用できます。構成ファイルでインスタンス名を指定した場合は、代わりにその名前を使用できます。

1台の定足数サーバーを起動するには、インスタンス名とポート番号のいずれかを指定します。複数の定足数サーバーを構成している

場合、すべての定足数サーバーを起動するには、+オペランドを使用します。

#### 例 6-11 すべての構成済み定足数サーバーの起動

次の例では、構成されているすべての定足数サーバーを起動します。

```
/usr/cluster/bin/clquorumserver start +
```

#### 例 6-12 特定の定足数サーバーの起動

次の例では、ポート番号 2000 で待機している定足数サーバーを起動します。

```
/usr/cluster/bin/clquorumserver start 2000
```

## ▼ 定足数サーバーを停止する

1 Sun Cluster ソフトウェアを起動するホスト上でスーパーユーザーになります。

2 ソフトウェアを停止するには、`clquorumserver stop` コマンドを使用します。

```
/usr/cluster/bin/clquorumserver stop [-d] quorumserver
```

`-d` マシンを次回起動したときに、定足数サーバーを起動するかどうかを制御します。`-d` オプションを指定すると、次のマシン起動時に定足数サーバーは起動しません。

`quorumserver` 定足数サーバーを識別します。定足数サーバーが待機するポート番号を使用できます。構成ファイルでインスタンス名を指定した場合は、代わりにその名前を使用できます。

1 台の定足数サーバーを停止するには、インスタンス名とポート番号のいずれかを指定します。複数の定足数サーバーを構成している場合、すべての定足数サーバーを停止するには、+オペランドを使用します。

#### 例 6-13 すべての構成済み定足数サーバーの停止

次の例では、構成されているすべての定足数サーバーを停止します。

```
/usr/cluster/bin/clquorumserver stop +
```

## 例 6-14 特定の定足数サーバーの停止

次の例では、ポート番号 2000 で待機している定足数サーバーを停止します。

```
/usr/cluster/bin/clquorumserver stop 2000
```

## 定足数サーバーに関する情報の表示

定足数サーバーについての構成情報を表示することができます。このコマンドは、定足数サーバーを定足数デバイスとして構成しているすべてのクラスタごとに、対応するクラスタ名、クラスタ ID、予約鍵のリスト、および登録鍵のリストを表示します。

### ▼ 定足数サーバーに関する情報を表示する

- 1 定足数サーバーの情報を表示するホスト上でスーパーユーザーになります。スーパーユーザー以外のユーザーには、`solaris.cluster.read` RBAC (Role-Based Access Control) の承認が必要です。RBAC 権限プロファイルの詳細については、`rbac(5)` のマニュアルページを参照してください。
- 2 `clquorumserver` コマンドを使用することで、定足数サーバーの構成情報を表示します。

```
/usr/cluster/bin/clquorumserver show quorumserver
```

`quorumserver` 1 つまたは複数の定足数サーバーを識別します。インスタンス名またはポート番号で定足数サーバーを指定できます。すべての定足数サーバーの構成情報を表示するには、+ オペランドを使用します。

## 例 6-15 1 つの定足数サーバーの構成の表示

次の例では、ポート 9000 を使用する定足数サーバーの構成情報を表示します。次のコマンドは、定足数サーバーが定足数デバイスとして構成されているすべてのクラスタの情報を表示します。この情報にはクラスタの名前と ID、およびデバイスの予約鍵と登録鍵のリストが含まれます。

次の例では、クラスタ `bastille` の ID が 1、2、3、および 4 であるノードが、定足数サーバー上に鍵を登録しています。また、ノード 4 は定足数デバイスの予約を所有しているため、その鍵は予約リストに表示されます。

```
/usr/cluster/bin/clquorumserver show 9000
```

```
=== Quorum Server on port 9000 ===
```

```

--- Cluster bastille (id 0x439A2EFB) Reservation ---

Node ID: 4
 Reservation key: 0x439a2efb00000004

--- Cluster bastille (id 0x439A2EFB) Registrations ---

Node ID: 1
 Registration key: 0x439a2efb00000001

Node ID: 2
 Registration key: 0x439a2efb00000002

Node ID: 3
 Registration key: 0x439a2efb00000003

Node ID: 4
 Registration key: 0x439a2efb00000004

```

#### 例 6-16 複数の定足数サーバーの構成の表示

次の例では、3つの定足数サーバー qs1、qs2、および qs3 の構成情報を表示します。

```
/usr/cluster/bin/clquorumserver show qs1 qs2 qs3
```

#### 例 6-17 動作しているすべての定足数サーバーの構成の表示

次の例では、動作しているすべての定足数サーバーの構成情報を表示します。

```
/usr/cluster/bin/clquorumserver show +
```

## 期限切れの定足数サーバークラスタ情報のク リーンアップ

quorumserver のタイプの定足数デバイスを削除するには、213 ページの「定足数デバイスを削除する」で説明されているように、clquorum remove コマンドを使用します。通常の動作では、このコマンドは定足数サーバーホストに関する定足数サーバーの情報も削除します。ただし、クラスタが定足数サーバーホストとの通信を失うと、定足数デバイスを削除しても、この情報がクリーンアップされません。

定足数サーバークラスタ情報は、次の状況で無効になります。

- `clquorum remove` コマンドを使用してクラスタ定足数デバイスを削除せずに、クラスタの運用を停止した場合。
- 定足数サーバーホストが停止している間に、`quorum_server` タイプの定足数デバイスをクラスタから削除した場合。



注意-タイプ `quorumserver` の定足数デバイスがまだクラスタから削除されていない場合、この手順を使用して無効な定足数サーバーを削除すると、クラスタ定足数に障害が発生する可能性があります。

## ▼ 定足数サーバーの構成情報をクリーンアップする

始める前に 213 ページの「定足数デバイスを削除する」で説明されているとおりに、定足数サーバーの定足数デバイスを削除します。



注意-クラスタがまだこの定足数サーバーを使用している場合、この手順を実行するとクラスタ定足数に障害が発生します。

- 1 定足数サーバーホストでスーパーユーザーになります。
- 2 `clquorumserver clear` コマンドを使用して、構成ファイルをクリーンアップします。  

```
clquorumserver clear -c clustername -I clusterID quorumserver [-y]
```

`-c clustername` 以前に定足数サーバーを定足数デバイスとして使用していたクラスタの名前です。

クラスタ名を取得するには、クラスタノード上で `cluster show` を実行します。

`-I clusterID` クラスタ ID です。

クラスタ ID は 8 桁の 16 進数です。クラスタ ID を取得するには、クラスタノード上で `cluster show` を実行します。

`quorumserver` 1 つまたは複数の定足数サーバーの識別子です。

定足数サーバーは、ポート番号かインスタンス名で識別できます。ポート番号は、クラスタノードが定足数サーバーと通信するために使用されます。インスタンス名は、定足数サーバーの構成ファイル `/etc/scqsd/scqsd.conf` で指定されます。

`-y` 実行前に確認のプロンプトを表示することなく、`clquorumserver clear` コマンドに、構成ファイルからクラスタ情報をクリーンアップさせます。

期限切れのクラスタ情報を定足数サーバーから削除したいことが確かである場合のみ、このオプションを使用します。

- 3 (省略可能) このサーバーインスタンスでほかに定足数デバイスが構成されていない場合は、定足数サーバーを停止します。

#### 例 6-18 定足数サーバー構成からの期限切れのクラスタ情報のクリーンアップ

次の例は、`sc-cluster` という名前のクラスタについての情報を、ポート 9000 を使用する定足数サーバーから削除します。

```
clquorumserver clear -c sc-cluster -I 0x4308D2CF 9000
```

```
The quorum server to be unconfigured must have been removed from the cluster.
Unconfiguring a valid quorum server could compromise the cluster quorum. Do you
want to continue? (yes or no) y
```



# クラスタインターコネクトとパブリックネットワークの管理

---

この章では、Sun Cluster インターコネクトとパブリックネットワークのソフトウェア上の作業手順について説明します。

クラスタインターコネクトとパブリックネットワークの管理には、ハードウェア上の作業とソフトウェア上の作業が含まれます。通常、初めてクラスタをインストールおよび構成するときには、IP ネットワークマルチパス (IP Network Multipathing) グループを含むクラスタインターコネクトとパブリックネットワークを構成します。あとで、クラスタインターコネクトネットワーク構成を変更する必要が生じた場合は、この章のソフトウェア手順を使用します。クラスタ内に IP Network Multipathing グループを構成する方法については、[250 ページの「パブリックネットワークの管理」](#)の節を参照してください。

この章では、次のトピックの手順について説明します。

- [233 ページの「クラスタインターコネクトの管理」](#)
- [250 ページの「パブリックネットワークの管理」](#)

この章の関連手順の詳細な説明については、[表 7-1](#) および [表 7-3](#) を参照してください。

クラスタインターコネクトとパブリックネットワークの背景情報や概要については、『[Sun Cluster Concepts Guide for Solaris OS](#)』を参照してください。

## クラスタインターコネクトの管理

このセクションでは、クラスタトランスポートアダプタ、クラスタトランスポートケーブルなどのクラスタインターコネクトの再構成手順について説明します。これらの手順では、Sun Cluster ソフトウェアをインストールする必要があります。

通常、`clsetup` ユーティリティを使用すると、クラスタインターコネクトのクラスタトランスポートを管理できます。詳細は、`clsetup(1CL)` のマニュアルページを参

照してください。Solaris 10 OS を実行している場合、すべてのクラスタインターコネクットコマンドはグローバルクラスタの投票ノードで実行する必要があります。

クラスタソフトウェアをインストールする手順については、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』を参照してください。クラスタハードウェアコンポーネントをサービスする手順については、『[Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS](#)』を参照してください。

注-クラスタインターコネクット手順中、通常は、(適切であれば)デフォルトのポート名を選択してもかまいません。デフォルトのポート名は、ケーブルのアダプタ側が接続されているノードの内部ノード ID 番号と同じです。ただし、SCI などの特定の種類のアダプタではデフォルトのポート名は使用できません。

表 7-1 作業リスト:クラスタインターコネクットの管理

作業	参照先
<code>clsetup(ICL)</code> を使用することで、クラスタトランスポートを管理する	26 ページの「クラスタ構成ユーティリティにアクセスする」
<code>clinterconnect status</code> を使用することで、クラスタインターコネクットのステータスを確認する	235 ページの「クラスタインターコネクットの状態を確認する」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブル、トランスポートアダプタ、またはスイッチを追加する	236 ページの「クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを追加する」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブル、トランスポートアダプタ、またはトランスポートスイッチを削除する	239 ページの「クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを削除する」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブルを有効にする	243 ページの「クラスタトランスポートケーブルを有効にする」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブルを無効にする	244 ページの「クラスタトランスポートケーブルを無効にする」
トランスポートアダプタのインスタンス番号の確認	246 ページの「トランスポートアダプタのインスタンス番号を確認する」
IP アドレスまたは既存のクラスタのアドレス範囲の変更	247 ページの「既存のクラスタのプライベートネットワークアドレスまたはアドレス範囲を変更する」

## クラスタインターコネクットでの動的再構成

クラスタインターコネクット上で動的再構成 (DR) を実行するときには、いくつかの問題を考慮する必要があります。

- Sun Cluster の DR 機能の説明で述べられているすべての必要条件、手順、制限は Sun Cluster の DR サポートにも適用されます (オペレーティングシステムでの休止操作をしない場合を除く)。したがって、Sun Cluster ソフトウェアで DR 機能を使用する前に、必ず、Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- Sun Cluster ソフトウェアは、アクティブなプライベートインターコネクットインタフェース上で実行された DR ボード削除操作を拒否します。
- アクティブなクラスタインターコネクットで DR を実行するには、クラスタからアクティブなアダプタを完全に削除する必要があります。scsetup メニューまたは該当する scconf コマンドを使用します。



注意 - Sun Cluster ソフトウェアの個々のクラスタノードには、他のすべてのクラスタノードに対する有効なパスが、少なくとも 1 つは存在していなければなりません。したがって、個々のクラスタノードへの最後のパスをサポートするプライベートインターコネクットインタフェースを無効にしないでください。

パブリックネットワークインタフェース上で DR 操作を実行するときは、次の手順をその順番どおりに行います。

表 7-2 作業マップ:パブリックネットワークインタフェースでの動的再構成

作業	参照先
1. アクティブなインターコネクットからインタフェースを無効にして削除	251 ページの「パブリックネットワークインタフェースでの動的再構成」
2. パブリックネットワークインタフェース上で DR 操作を実行	「Solaris 9 on Sun Hardware」コレクションと「Solaris 10 on Sun Hardware」コレクションの『Sun Enterprise 10000 DR 構成マニュアル』と『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』

### ▼ クラスタインターコネクットの状態を確認する

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

この手順を実行するためにスーパーユーザーとしてログインする必要はありません。

- 1 クラスタインターコネクットの状態を確認します。  
% clinterconnect status
- 2 一般的な状態メッセージについては、以下の表を参照してください。

状態メッセージ	説明および可能な処置
Path online	パスが現在正常に機能しています。処置は必要ありません。
Path waiting	パスが現在初期化中です。処置は必要ありません。
Faulted	パスが機能していません。これは、パスが一時的に待機状態とオンライン状態の間にある状態の可能性があります。再び clinterconnect status を実行してもメッセージが繰り返される場合は、適切な処置を行ってください。

#### 例 7-1 クラスタインターコネクットの状態を確認する

次に、正常に機能しているクラスタインターコネクットの状態の例を示します。

```
% clinterconnect status
-- Cluster Transport Paths --
 Endpoint Endpoint Status
 ----- -
Transport path: phys-schost-1:qfe1 phys-schost-2:qfe1 Path online
Transport path: phys-schost-1:qfe0 phys-schost-2:qfe0 Path online
Transport path: phys-schost-1:qfe1 phys-schost-3:qfe1 Path online
Transport path: phys-schost-1:qfe0 phys-schost-3:qfe0 Path online
Transport path: phys-schost-2:qfe1 phys-schost-3:qfe1 Path online
Transport path: phys-schost-2:qfe0 phys-schost-3:qfe0 Path online
```

### ▼ クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを追加する

クラスタ固有のトランスポートの要件については、『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』の「Interconnect Requirements and Restrictions」を参照してください。

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタトランスポートケーブルが物理的に取り付けられていることを確認します。  
クラスタトランスポートケーブルのインストール手順については、『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』を参照してください。
- 2 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 3 `clsetup` ユーティリティーを起動します。  
`# clsetup`  
メインメニューが表示されます。
- 4 クラスタインターコネクットメニューを表示するためのオプションに対応する番号を入力します。

---

注-SCI アダプタを使用する構成では、この手順の「Add(追加)」部分において表示されるアダプタ接続(ポート名)のデフォルトを受け入れてはいけません。その代わりに、ノードに物理的に(ケーブルで)接続されている、Dolphin スイッチ上のポート名(0、1、2、または3)を指定します。

---

- 5 トランスポートケーブルを追加するためのオプションに対応する番号を入力します。  
指示に従い、必要な情報を入力します。
- 6 トランスポートアダプタをノードに追加するためのオプションに対応する番号を入力します。  
指示に従い、必要な情報を入力します。  
クラスタインターコネクットで次のアダプタのいずれかを使用する予定の場合、関連するエントリを各クラスタノードの `/etc/system` ファイルに追加します。このエントリは、次のシステム再起動後に有効になります。

アダプタ	エントリ
ce	set ce:ce_taskq_disable=1
ipge	set ipge:ipge_taskq_disable=1
ixge	set ixge:ixge_taskq_disable=1

- 7 トランスポートスイッチを追加するためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。

- 8 クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチが追加されたことを確認します。

```
clinterconnect show node:adapter,adapternode
clinterconnect show node:adapter
clinterconnect show node:switch
```

#### 例 7-2 クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチの追加

次の例に、clsetupユーティリティを使用し、トランスポートケーブル、トランスポートアダプタ、トランスポートスイッチをノードに追加する方法を示します。

```
[Ensure that the physical cable is installed.]
[Start the clsetup utility:]
clsetup
[Select Cluster interconnect]

[Select either Add a transport cable,
Add a transport adapter to a node,
or Add a transport switch.]
[Answer the questions when prompted.]
 [You Will Need:]
[Information: Example:]
 node names phys-schost-1
 adapter names qfe2
 switch names hub2
 transport type dlpi
[Verify that the clinterconnect
command completed successfully:]Command completed successfully.
Quit the clsetup Cluster Interconnect Menu and Main Menu.
[Verify that the cable, adapter, and switch are added:]
clinterconnect show phys-schost-1:qfe2,hub2
===Transport Cables===
Transport Cable: phys-schost-1:qfe2@0,hub2
Endpoint1: phys-schost-2:qfe0@0
```

```

Endpoint2: ethernet-1@2 ???? Should this be hub2?
State: Enabled

clinterconnect show phys-schost-1:qfe2
=== Transport Adepters for qfe2
Transport Adapter: qfe2
 Adapter State: Enabled
 Adapter Transport Type: dlpi
 Adapter Property (device_name): ce
 Adapter Property (device_instance): 0
 Adapter Property (lazy_free): 1
 Adapter Property (dlpi_heartbeat_timeout): 10000
 Adpater Property (dlpi_heartbeat_quantum): 1000
 Adapter Property (nw_bandwidth): 80
 Adapter Property (bandwidth): 70
 Adapter Property (ip_address): 172.16.0.129
 Adapter Property (netmask): 255.255.255.128
 Adapter Port Names: 0
 Adapter Port STate (0): Enabled

clinterconnect show phys-schost-1:hub2

```

```

=== Transport Switches ===
Transport Switch: hub2
 Switch State: Enabled
 Switch Type: switch
 Switch Port Names: 1 2
 Switch Port State(1): Enabled
 Switch Port State(2): Enabled

```

次の手順 クラスタトランスポートケーブルのインターコネクットのステータスを確認するには、[235 ページの「クラスタインターコネクットの状態を確認する」](#)を参照してください。

## ▼ クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを削除する

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

次の手順を使用して、クラスタトランスポートケーブル、クラスタトランスポートアダプタ、およびトランスポートスイッチをノード構成から削除します。ケーブル

を無効にした場合、このケーブルの2つのエンドポイントは構成されたままになります。トランスポートケーブルの終端として使用されているアダプタは削除できません。



注意-各クラスタノードには、他のすべてのクラスタノードに対する(機能している)トランスポートパスが少なくとも1つずつ必要です。2つのノードは必ず接続されており、お互いに分離されているノードは存在しません。ケーブルを無効にする前には、必ず、ノードのクラスタインターコネクットの状態を確認してください。状態が冗長な場合、つまり別の接続が使用できる場合だけ、ケーブル接続を無効にします。ノードの最後の機能しているケーブルを無効にすると、そのノードはクラスタメンバーシップから外れます。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 残りのクラスタトランスポートパスの状態を確認します。

```
clinterconnect status
```



注意-2 ノードクラスタのいずれかのノードを削除しようとして「パス障害 (Path faulted)」などのエラーメッセージが表示された場合、この手順を続ける前に問題を調査してください。このような問題は、ノードパスが利用できないことを示しています。残りの動作中のパスを削除すると、このノードはクラスタメンバーシップから外れ、クラスタが再構成されます。

- 3 `clsetup` ユーティリティを起動します。

```
clsetup
```

メインメニューが表示されます。

- 4 クラスタインターコネクットメニューにアクセスするためのオプションに対応する番号を入力します。
- 5 トランスポートケーブルを無効にするためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。

- 6 トランスポートケーブルを削除するためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。

注-物理的にケーブル接続を解除する場合は、ポートと宛先デバイスをつないでいるケーブルを切り離します。

- 7 トランスポートアダプタをノードから削除するためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。

注-アダプタをノードから物理的に取り外す場合のハードウェアサービス手順については、『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』を参照してください。

- 8 トランスポートスイッチを削除するためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。

注-ポートがトランスポートケーブルの終端として使用されている場合、スイッチは削除できません。

- 9 ケーブル、アダプタ、またはスイッチが削除されたことを確認します。

```
clinterconnect show node:adapter,adapternode
clinterconnect show node:adapter
clinterconnect show node:switch
```

ノードからトランスポートケーブルやトランスポートアダプタが削除された場合は、このコマンドの出力には表示されません。

### 例 7-3 トランスポートケーブル、トランスポートアダプタ、トランスポートスイッチの削除

次の例に、`clsetup` コマンドを使用して、トランスポートケーブル、トランスポートアダプタ、またはトランスポートスイッチを削除する方法を示します。

```
[Become superuser on any node in the cluster.]
[Start the utility:]
clsetup
```

```

[Select Cluster interconnect.]
[Select either Remove a transport cable,
Remove a transport adapter to a node,
or Remove a transport switch.]
[Answer the questions when prompted.]
 You Will Need:
 Information Example:
 node names phys-schost-1
 adapter names qfe1
 switch names hub1
[Verify that the clinterconnect
command was completed successfully:]
Command completed successfully.
[Quit the clsetup utility Cluster Interconnect Menu and Main Menu.]
[Verify that the cable, adapter, or switch is removed:]
clinterconnect show phys-schost-1:qfe2,hub2
===Transport Cables ===
Transport Cable: phys-schost-2:qfe2@0,hub2
Cable Endpoint1: phys-schost-2:qfe0@0
Cable Endpoint2: ethernet-1@2 ??? Should this be hub2???
Cable State: Enabled

clinterconnect show phys-schost-1:qfe2
=== Transport Adepters for qfe2
Transport Adapter: qfe2
Adapter State: Enabled
Adapter Transport Type: dlpi
Adapter Property (device_name): ce
Adapter Property (device_instance): 0
Adapter Property (lazy_free): 1
Adapter Property (dlpi_heartbeat_timeout): 10000
Adpater Property (dlpi_heartbeat_quantum): 1000
Adapter Property (nw_bandwidth): 80
Adapter Property (bandwidth): 70
Adapter Property (ip_address): 172.16.0.129
Adapter Property (netmask): 255.255.255.128
Adapter Port Names: 0
Adapter Port State (0): Enabled

clinterconnect show phys-schost-1:hub2
=== Transport Switches ===
Transport Switch: hub2
Switch State: Enabled
Switch Type: switch
Switch Port Names: 1 2
Switch Port State(1): Enabled
Switch Port State(2): Enabled

```

## ▼ クラスタトランスポートケーブルを有効にする

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

このオプションを使用し、既存のクラスタトランスポートケーブルを有効にします。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 **clsetup** ユーティリティを起動します。  
# **clsetup**  
メインメニューが表示されます。
- 3 クラスタインターコネクメニューにアクセスするためのオプションに対応する番号を入力し、**Return** キーを押します。
- 4 トランスポートケーブルを有効にするためのオプションに対応する番号を入力し、**Return** キーを押します。  
プロンプトが表示されたなら、指示に従います。ケーブルのいずれかの終端のノード名およびアダプタ名の両方を入力する必要があります。
- 5 ケーブルが有効になっていることを確認します。  
# **clinterconnect show node:adapter,adapternode**

### 例 7-4 クラスタトランスポートケーブルを有効にする

次の例に、ノード phys-schost-2 にあるアダプタ qfe-1 のクラスタトランスポートケーブルを有効にする方法を示します。

```
[Become superuser on any node.]
[Start the clsetup utility:]
clsetup
[Select Cluster interconnect>Enable a transport cable.]

[Answer the questions when prompted.]
```

```
[You will need the following information.]
 You Will Need:
Information: Example:
node names phys-schost-2
adapter names qfe1
switch names hub1
[Verify that the scinterconnect
command was completed successfully:]

clinterconnect enable phys-schost-2:qfe1

Command completed successfully.
[Quit the clsetup Cluster Interconnect Menu and Main Menu.]
[Verify that the cable is enabled:]
clinterconnect show phys-schost-1:qfe2,hub2
Transport cable: phys-schost-2:qfe1@0 ethernet-1@2 Enabled
Transport cable: phys-schost-3:qfe0@1 ethernet-1@3 Enabled
Transport cable: phys-schost-1:qfe0@0 ethernet-1@1 Enabled
```

## ▼ クラスタトランスポートケーブルを無効にする

この手順は、Sun Cluster Manager GUI を使用しても実行できます。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

クラスタトランスポートケーブルを無効にし、クラスタインターコネクトパスを一時的に停止する必要があることがあります。一時的な停止は、クラスタインターコネクトで発生する問題の解決や、クラスタインターコネクットのハードウェアの交換に便利です。

ケーブルを無効にした場合、このケーブルの2つのエンドポイントは構成されたままになります。トランスポートケーブルの終端として使用されているアダプタは削除できません。



注意 - 各クラスタノードには、他のすべてのクラスタノードに対する (機能している) トランスポートパスが少なくとも1つずつ必要です。2つのノードは必ず接続されており、お互いに分離されているノードは存在しません。ケーブルを無効にする前には、必ず、ノードのクラスタインターコネクットの状態を確認してください。状態が冗長な場合、つまり別の接続が使用できる場合だけ、ケーブル接続を無効にします。ノードの最後の機能しているケーブルを無効にすると、そのノードはクラスタメンバーシップから外れます。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 ケーブルを無効にする前に、クラスタインターコネクットの状態を確認します。

```
clinterconnect status
```



注意-2 ノードクラスタのいずれかのノードを削除しようとして「パス障害 (Path faulted)」などのエラーメッセージが表示された場合、この手順を続ける前に問題を調査してください。このような問題は、ノードパスが利用できないことを示しています。残りの動作中のパスを削除すると、このノードはクラスタメンバーシップから外れ、クラスタが再構成されます。

- 3 clsetup ユーティリティーを起動します。

```
clsetup
```

メインメニューが表示されます。

- 4 クラスタインターコネクットメニューにアクセスするためのオプションに対応する番号を入力し、**Return** キーを押します。
- 5 トランスポートケーブルを無効にするためのオプションに対応する番号を入力し、**Return** キーを押します。

指示に従い、必要な情報を入力します。このクラスタインターコネクットのすべてのコンポーネントは無効になります。ケーブルのいずれかの終端のノード名およびアダプタ名の両方を入力する必要があります。

- 6 ケーブルが無効になっていることを確認します。

```
clinterconnect show node:adapter,adapternode
```

#### 例 7-5 クラスタトランスポートケーブルを無効にする

次の例に、ノード phys-schost-2 にあるアダプタ qfe-1 のクラスタトランスポートケーブルを無効にする方法を示します。

```
[Become superuser on any node.]
[Start the clsetup utility:]
clsetup
[Select Cluster interconnect>Disable a transport cable.]
```

```
[Answer the questions when prompted.]
```

```
[You will need the following information.]
[You Will Need:]
Information: Example:
node names phys-schost-2
adapter names qfe1
switch names hub1
[Verify that the clinterconnect
command was completed successfully:]
Command completed successfully.
[Quit the scsetup Cluster Interconnect Menu and Main Menu.]
[Verify that the cable is disabled:]
clinterconnect show -p phys-schost-1:qfe2,hub2
Transport cable: phys-schost-2:qfe1@0 ethernet-1@2 Disabled
Transport cable: phys-schost-3:qfe0@1 ethernet-1@3 Enabled
Transport cable: phys-schost-1:qfe0@0 ethernet-1@1 Enabled
```

## ▼ トランスポートアダプタのインスタンス番号を確認する

clsetup コマンドを使用して正しいトランスポートアダプタの追加と削除を行うには、トランスポートアダプタのインスタンス番号を確認する必要があります。アダプタ名は、アダプタの種類とアダプタのインスタンス番号を組み合わせたものです。この作業では、SCI-PCI アダプタを例として使用しています。

- 1 スロット番号にもとづき、アダプタの名前を確認してください。  
次の画面は例であり、個々のハードウェアと一致しない可能性があります。

```
prtdiag
...
===== IO Cards =====
 Bus Max
 Bus
IO Port Bus Freq Bus Dev,
Type ID Side Slot MHz Freq Func State Name Model

PCI 8 B 2 33 33 2,0 ok pci11c8,0-pci11c8,d665.11c8.0.0
PCI 8 B 3 33 33 3,0 ok pci11c8,0-pci11c8,d665.11c8.0.0
...
```

- 2 アダプタのパスを使用して、アダプタのインスタンス番号を確認してください。  
次の画面は例であり、個々のハードウェアと一致しない可能性があります。

```
grep sci /etc/path_to_inst
"/pci@1f,400/pci11c8,o@2" 0 "sci"
"/pci@1f,4000.pci11c8,0@4 "sci"
```

- 3 アダプタの名前とスロット番号を使用してアダプタのインスタンス番号を確認してください。

次の画面は例であり、個々のハードウェアと一致しない可能性があります。

```
prtconf
...
pci, instance #0
 pcic8, instance #0
 pcic8, instance #1
...
```

## ▼ 既存のクラスタのプライベートネットワークアドレスまたはアドレス範囲を変更する

プライベートネットワークアドレスまたは使用されるネットワークアドレスの範囲、またはその両方を変更するには、次の手順に従います。

始める前に スーパーユーザーのリモートシェル (rsh(1M)) または Secure Shell (ssh(1)) アクセスが、すべてのクラスタノードで有効になっていることを確認します。

- 1 各クラスタノード上で次のサブステップを実行することで、すべてのクラスタノードを再起動し、非クラスタモードにします。
  - a. 非クラスタモードで起動するクラスタノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。
  - b. `clnode evacuate` および `cluster shutdown` コマンドを使用してノードを停止します。  
`clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定ノード上の投票ノードまたは非投票ノードから、次に優先される投票ノードまたは非投票ノードへ、すべてのリソースグループを切り替えます。  

```
clnode evacuate node
cluster shutdown -g0 -y
```
- 2 1つのノードから、`clsetup` ユーティリティを起動します。  
 非クラスタモードで動作している場合、`clsetup` ユーティリティは非クラスタモード動作のメインメニューを表示します。
- 3 IPアドレス範囲を変更するためのオプションに対応する番号を入力し、**Return** キーを押します。  
`clsetup` ユーティリティは現在のプライベートネットワーク構成を表示し、この構成を変更するかどうかを尋ねます。

- 4 プライベートネットワーク IP アドレスか IP アドレス範囲のいずれかを変更するには、「yes」と入力し、**Return** キーを押します。

clsetup ユーティリティーはデフォルトのプライベートネットワーク IP アドレスである 172.16.0.0 を表示し、このデフォルトをそのまま使用してもよいかどうかを尋ねます。

- 5 プライベートネットワーク IP アドレスを変更するか、そのまま使用します。

- デフォルトのプライベートネットワーク IP アドレスをそのまま使用し、IP アドレス範囲の変更に進むには、「yes」と入力し、**Return** キーを押します。

clsetup ユーティリティーは、デフォルトのネットマスクをそのまま使用してもよいかどうかを尋ねます。次の手順に進み、応答を入力します。

- デフォルトのプライベートネットワーク IP アドレスを変更するには、次のサブステップを実行します。

- a. clsetup ユーティリティーの、デフォルトのアドレスをそのまま使用してもよいかどうかに関する質問に対しては「no」と入力し、**Return** キーを押します。

clsetup ユーティリティーは、新しいプライベートネットワーク IP アドレスを入力するプロンプトを表示します。

- b. 新しい IP アドレスを入力し、**Return** キーを押します。

clsetup ユーティリティーはデフォルトのネットマスクを表示し、デフォルトのネットマスクをそのまま使用してもよいかどうかを尋ねます。

- 6 デフォルトのプライベートネットワーク IP アドレス範囲を変更するか、そのまま使用します。

Solaris 9 OS では、デフォルトのネットマスクは 255.255.248.0 です。このデフォルトの IP アドレス範囲は、クラスタ内で最大 64 のノードと最大 10 のプライベートネットワークをサポートします。Solaris 10 OS では、デフォルトのネットマスクは 255.255.240.0 です。このデフォルトの IP アドレス範囲は、クラスタ内で最大 64 のノード、最大 12 のゾーンクラスタ、最大 10 のプライベートネットワークをサポートします。

- デフォルトの IP アドレス範囲をそのまま使用するには、「yes」と入力して、**Return** キーを押します。

続いて、次の手順に進みます。

- IP アドレス範囲を変更するには、次のサブステップを実行します。
  - a. `clsetup`ユーティリティーの、デフォルトのアドレス範囲をそのまま使用してもよいかどうかに関する質問に対しては「no」と入力し、**Return** キーを押します。  
デフォルトのネットマスクを拒否する場合、`clsetup`ユーティリティーは、ユーザーがクラスタ内で構成する予定のノードとプライベートネットワークの数、および Solaris 10 OS 上のゾーンクラスタの数を入力するプロンプトを出します。
  - b. ユーザーがクラスタ内で構成する予定のノードとプライベートネットワークの数、および **Solaris 10 OS** 上のゾーンクラスタの数を入力します。  
これらの数から、`clsetup`ユーティリティーは2つの推奨ネットマスクを計算します。
    - 最初のネットマスクは、指定したノードとプライベートネットワークの数、および Solaris 10 OS 上のゾーンクラスタの数をサポートする最小限のネットマスクです。
    - 2番目のネットマスクは、将来の成長を見越して、指定したノードとプライベートネットワークの数、および Solaris 10 OS 上のゾーンクラスタの数の2倍の数をサポートします。
  - c. 計算されたネットマスクのいずれかを指定するか、ノードとプライベートネットワークの予定数、および **Solaris 10 OS** 上のゾーンクラスタの予定数をサポートする別のネットマスクを指定します。
- 7 更新の継続に関する `clsetup`ユーティリティーの質問に対しては、「yes」と入力します。
- 8 完了後 `clsetup`ユーティリティーを終了します。
- 9 各クラスタノードに対して次のサブステップを実行することで、各クラスタノードを再起動し、クラスタモードに戻します。
  - a. ノードを起動します。
    - SPARC ベースのシステム上で、次のコマンドを実行します。  
`ok boot`
    - x86 ベースのシステム上で、次のコマンドを実行します。  
GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。  
  
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)  
+-----+

```

| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.

```

- 10 ノードが問題なく起動し、オンラインであることを確認します。

```
cluster status -t node
```

## パブリックネットワークの管理

Sun Cluster 3.1 4/04、Sun Cluster 3.1 8/05、Sun Cluster 3.2、および Sun Cluster 3.2 2/08 は、パブリックネットワークの IP (Internet Protocol) ネットワークマルチパスの Solaris ソフトウェア実装をサポートします。IP ネットワークマルチパスの基本的な管理は、クラスタ環境でも非クラスタ環境でも同じです。マルチパスの管理については、適切な Solaris OS のマニュアルを参照してください。ただし、Sun Cluster 環境で IP ネットワークマルチパスを管理する前には、以下のガイドラインを熟読してください。

### クラスタで IP ネットワークマルチパスグループを管理する

IP ネットワークマルチパス手順をクラスタ上で実行する前に、次のガイドラインについて考慮してください。

- 各パブリックネットワークアダプタは、マルチパスグループに属している必要があります。
- `local-mac-address?` 変数には、Ethernet アダプタの値として `true` が指定されていなければなりません。
- 次に示すタイプのマルチパスグループ内に存在するアダプタごとにテスト IP アドレスを設定する必要があります。
  - Solaris 9 または Solaris 10 OS で稼働しているクラスタ内のすべてのマルチアダプタマルチパスグループSolaris 9 または Solaris 10 OS 上のシングルアダプタマルチパスグループは、テスト IP アドレスを必要としません。
- 同一マルチパスグループ内のすべてのアダプタ用のテスト IP アドレスは、単一の IP サブネットに属する必要があります。
- テスト IP アドレスは高可用性でないため、通常のアプリケーションが使用しないようにします。

- マルチパスグループの命名に制限はありません。しかし、リソースグループを構成するとき、`netiflist`には、任意のマルチパス名にノードID番号またはノード名が続くものを指定します。たとえば、マルチパスグループの名前が`sc_ipmp0`であるとき、ノードIDが1である`phys-schost-1`というノード上にアダプタが存在する場合、`netiflist`には`sc_ipmp0@1`または`sc_ipmp0@phys-schost-1`のどちらかを指定してもかまいません。
- あらかじめIPアドレスをグループ内の削除する予定のアダプタから代替アダプタにスイッチオーバーせずに、IPネットワークマルチパスグループのアダプタを構成解除(`unplumb`する)または停止しないようにします(つまり、`if_mpadm(1M)`コマンドを使用)。
- 個々のマルチパスグループから削除する前に、アダプタを別のサブネットに配線しないようにします。
- 論理アダプタ操作は、マルチパスグループで監視中の場合でもアダプタに対して行うことができます。
- クラスタ内の各ノードについて、最低1つのパブリックネットワーク接続を維持しなければなりません。クラスタは、パブリックネットワーク接続がないとアクセスできません。
- クラスタ上のIPネットワークマルチパスグループの状態を表示するには、`clinterconnect status`コマンドを使用します。

IPネットワークマルチパスの詳細については、Solaris OS システム管理マニュアルセットの適切なマニュアルを参照してください。

表 7-3 作業リスト:パブリックネットワークの管理

Solaris オペレーティングシステムリリース	参照先
SPARC: Solaris 9 オペレーティングシステム	『Solaris のシステム管理 (IP サービス)』の「IP ネットワークマルチパス (トピック)」
Solaris 10 オペレーティングシステム	『Solaris のシステム管理 (IP サービス)』の「IP ネットワークマルチパス (トピック)」

クラスタソフトウェアをインストールする手順については、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』を参照してください。パブリックネットワークハードウェアコンポーネントをサービスする手順については、『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』を参照してください。

## パブリックネットワークインタフェースでの動的再構成

クラスタ内のパブリックネットワークインタフェース上で動的再構成 (DR) を実行するときには、いくつかの問題を考える必要があります。

- Sun Cluster の DR 機能の説明で述べられているすべての必要条件、手順、制限は Sun Cluster の DR サポートにも適用されます (オペレーティングシステムでの休止操作をしない場合を除く)。したがって、Sun Cluster ソフトウェアで DR 機能を使用する前に、必ず、Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- DR ボード削除操作は、パブリックネットワークインタフェースがアクティブでないときだけ成功します。アクティブなパブリックネットワークインタフェースを削除する前に、`if_mpadm(1M)` コマンドを使用して、削除するアダプタからマルチパスグループ内の別のアダプタに IP アドレスを切り替えます。
- アクティブなネットワークインタフェースを適切に無効にせずにパブリックネットワークインタフェースカードを削除しようとした場合、Sun Cluster はその操作を拒否して、その操作から影響を受けるインタフェースを識別します。



注意-2つのアダプタを持つマルチパスグループの場合、無効にしたネットワークアダプタ上で DR 削除操作を実行している間に残りのネットワークアダプタに障害が発生すると、可用性に影響が生じます。これは、DR 操作の間は、残りのネットワークアダプタのフェイルオーバー先が存在しないためです。

パブリックネットワークインタフェース上で DR 操作を実行するときは、次の手順をその順番どおりに行います。

表 7-4 作業マップ:パブリックネットワークインタフェースでの動的再構成

作業	参照先
1. <code>if_mpadm</code> を使用して、IP アドレスをマルチパスグループ内の削除する予定のアダプタから代替アダプタへの切り換えを実行	<a href="#">if_mpadm(1M)</a> のマニュアルページ 適切な SolarisOS のマニュアル: Solaris 9: 『Solaris のシステム管理 (IP サービス)』の「IP ネットワークマルチパス (トピック)」 Solaris 10: 『Solaris のシステム管理 (IP サービス)』のパート VI 「IPMP」
2. <code>ifconfig</code> コマンドを使用して、マルチパスグループからアダプタを削除	適切な Solaris のマニュアル: Solaris 9: 『Solaris のシステム管理 (IP サービス)』の「IP ネットワークマルチパス (トピック)」 <a href="#">ifconfig(1m)</a> のマニュアルページ Solaris 10: 『Solaris のシステム管理 (IP サービス)』のパート VI 「IPMP」

表 7-4 作業マップ:パブリックネットワークインタフェースでの動的再構成 (続き)

作業	参照先
3. パブリックネットワークインタフェース上でDR操作を実行	「Solaris 9 on Sun Hardware」コレクションと「Solaris 10 on Sun Hardware」コレクションの『 <a href="#">Sun Enterprise 10000 DR Configuration Guide</a> 』と『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』



# ノードの追加と削除

---

この章では、クラスタへのノードの追加とノードの削除方法を説明します。

- 255 ページの「クラスタへのノードの追加」
- 261 ページの「クラスタからのノードの削除」

クラスタのメンテナンスに関する情報は、第9章「クラスタの管理」を参照してください。

## クラスタへのノードの追加

このセクションでは、グローバルクラスタまたはゾーンクラスタにノードを追加する方法を説明します。新しいゾーンクラスタノードは、そのゾーンクラスタをホストするグローバルクラスタのノード上に作成できますが、それは、グローバルクラスタノードが、そのゾーンクラスタのノードをまだホストしていない場合に限られます。グローバルクラスタ上の既存の非投票ノードを、ゾーンクラスタノードに変換することはできません。

この章での `phys-schost#` は、グローバルクラスタのプロンプトを表します。 `clzonecluster` の対話型シェルプロンプトは `clzc:schost>` です。

次の表に、ノードを既存のクラスタに追加するときに行う作業を示します。作業は、示されている順に実行してください。

表 8-1 作業マップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加

作業	参照先
ホストアダプタのノードへの取り付けと、既存のクラスタインターコネクトが新しいノードをサポートできることの確認	『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』

表 8-1 作業マップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加 (続き)

作業	参照先
共有記憶装置の追加	『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』
追加ノードのクラスタの準備	『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の「追加のグローバルクラスタノード用にクラスタを準備する」
clsetup を使用した、承認済みノードリストへのノードの追加	256 ページの「ノードを認証ノードリストに追加する」
新しいクラスタノードへのソフトウェアのインストールと構成	『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の第 2 章「グローバルクラスタノードへのソフトウェアのインストール」
クラスタが Sun Cluster Geographic Edition のパートナーシップで構成されている場合、構成内のアクティブな参加メンバーとして新しいノードを構成する	『Sun Cluster Geographic Edition System Administration Guide』の「How to Add a New Node to a Cluster in a Partnership」

## ▼ ノードを認証ノードリストに追加する

Solaris ホストまたは仮想マシンを既存のグローバルクラスタまたはゾーンクラスタに追加する前に、プライベートクラスタインターコネクタへの物理的な接続が正しく動作しているかどうかも含め、必要なハードウェアがすべてノードに正しくインストールおよび構成されていることを確認してください。

ハードウェアのインストールについては、『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』または各サーバーに付属するハードウェアマニュアルを参照してください。

この手順によって、マシンは自分自身をクラスタ内にインストールします。つまり、自分のノード名を当該クラスタの認証ノードリストに追加します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 現在のグローバルクラスタメンバーで、現在のクラスタメンバー上のスーパーユーザーになります。次の手順は、グローバルクラスタのノードから実行します。

- 2 表8-1の作業マップに記載されている必要なハードウェアのインストールと構成作業をすべて正しく完了していることを確認します。

- 3 `clsetup`ユーティリティを起動します。

```
phys-schost# clsetup
```

メインメニューが表示されます。

---

注-ノードをゾーンクラスタに追加するために、`clzonecluster`ユーティリティを使用します。ゾーンをゾーンクラスタに手動で追加する方法は、手順9を参照してください。

---

- 4 新規ノードメニューを表示するためのオプションに対応する番号を入力し、**Return**キーを押します。

- 5 承認済みリストを変更するためのオプションに対応する番号を入力し、**Return**キーを押します。自分自身を追加できるマシンの名前を指定します。

指示に従って、ノードの名前をクラスタに追加します。追加するノードの名前が問い合わせられます。

- 6 作業が正常に行われたことを確認します。

作業が正常に行われた場合、`clsetup`ユーティリティは「コマンドが正常に完了しました」というメッセージを表示します。

- 7 新しいマシンがクラスタに追加されないようにするために、新しいマシンを追加する要求を無視するようクラスタに指示する、オプションに対応する番号を入力します。**Return**キーを押します。

`clsetup`のプロンプトに従います。このオプションを設定すると、クラスタは、自分自身をクラスタに追加しようとする新しいマシンからのパブリックネットワーク経由の要求をすべて無視します。

- 8 `clsetup`ユーティリティを終了します。

- 9 ノードをゾーンクラスタに手動で追加するには、**Solaris**ホストおよび仮想ノード名を指定してください。また、各ノードでパブリックネットワーク通信に使用するネットワークリソースも指定してください。次の例では、ゾーン名は `sczone` で、`bge0`は両方のマシンのパブリックネットワークアダプタです。

```
clzc:sczone>add node
clzc:sczone:node>set physical-host=phys-cluster-1
clzc:sczone:node>set hostname=hostname1
clzc:sczone:node>add net
clzc:sczone:node:net>set address=hostname1
clzc:sczone:node:net>set physical=bge0
```

```
clzc:sczone:node:net>end
clzc:sczone:node>end
clzc:sczone>add node
clzc:sczone:node>set physical-host=phys-cluster-2
clzc:sczone:node>set hostname=hostname2
clzc:sczone:node>add net
clzc:sczone:node:net>set address=hostname2
clzc:sczone:node:net>set physical=bge0
clzc:sczone:node:net>end
clzc:sczone:node>end
```

ノードを構成する手順の詳細は、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「[ゾーンクラスタの設定](#)」を参照してください。

- 10 新しいクラスタノード上でソフトウェアをインストールして構成します。  
scinstall または JumpStart™ ソフトウェアのいずれかを使用して、新しいノードのインストールと構成を完了します。詳細は、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』を参照してください。

#### 例 8-1 認証ノードリストへのグローバルクラスタノードの追加

次に、ノード phys-schost-3 を既存のクラスタの認証ノードリストに追加する例を示します。

```
[Become superuser and execute the clsetup utility.]
phys-schost# clsetup
[Select New nodes>Specify the name of a machine which may add itself.]
[Answer the questions when prompted.]
[Verify that the sconfg command completed successfully.]

claccess allow -h phys-schost-3

 Command completed successfully.
[Select Prevent any new machines from being added to the cluster.]
[Quit the clsetup New Nodes Menu and Main Menu.]
[Install the cluster software.]
```

参照 [clsetup\(1CL\)](#)

クラスタノードを追加する作業の一連の手順については、[表 8-1](#)、「[作業マップ: クラスタノードの追加](#)」を参照してください。

ノードを既存のリソースグループに追加する方法については、『[Sun Cluster Data Services Planning and Administration Guide for Solaris OS](#)』を参照してください。

## グローバルクラスタ内の非投票ノード(ゾーン)の作成

ここでは、非投票ノード(単にゾーンと呼ばれる)をグローバルクラスタノード上に作成するための次の情報と手順を説明します。

### ▼ グローバルクラスタ内に非投票ノードを作成する

- 1 非投票ノードを作成するグローバルクラスタノードでスーパーユーザーになります。  
大域ゾーンで作業してください。
- 2 **Solaris 10 OS**の場合、各ノードでサービス管理機能(**Service Management Facility**、**SMF**)用のマルチユーザーサービスがオンラインであることを確認します。  
ノードのサービスがまだオンラインでない場合は、次のステップに進む前に状態がオンラインに変わるまで待ちます。

```
phys-schost# svcs multi-user-server node
STATE STIME FMRI
online 17:52:55 svc:/milestone/multi-user-server:default
```

- 3 新しいゾーンを構成、インストール、および起動します。

---

注-グローバルクラスタの非投票ノードでリソースグループ機能をサポートするには、`autoboot` プロパティを `true` に設定にします。

---

Solaris のマニュアルの手順に従ってください。

- a. 『**Solaris** のシステム管理 (**Solaris** コンテナ: 資源管理と **Solaris** ゾーン)』の第 18 章「非大域ゾーンの計画と構成(手順)」の手順を実行します。
  - b. 『**Solaris** のシステム管理 (**Solaris** コンテナ: 資源管理と **Solaris** ゾーン)』の「ゾーンのインストールと起動」の手順を実行します。
  - c. 『**Solaris** のシステム管理 (**Solaris** コンテナ: 資源管理と **Solaris** ゾーン)』の「ゾーンの起動方法」の手順を実行します。
- 4 ゾーンが ready 状態であることを確認します。

```
phys-schost# zoneadm list -v
ID NAME STATUS PATH
0 global running /
1 my-zone ready /zone-path
```

- 5 ip-type プロパティが `exclusive` に設定されている完全ルートゾーンの場合: ゾーンで論理ホスト名リソースをホストする可能性がある場合は、大域ゾーンからメソッドディレクトリをマウントするファイルシステムリソースを構成します。

```
phys-schost# zonecfg -z sczone
zonecfg:sczone> add fs
zonecfg:sczone:fs> set dir=/usr/cluster/lib/rgm
zonecfg:sczone:fs> set special=/usr/cluster/lib/rgm
zonecfg:sczone:fs> set type=lofs
zonecfg:sczone:fs> end
zonecfg:sczone> exit
```

- 6 (省略可能) 共有 IP ゾーンでは、プライベート IP アドレスとプライベートホスト名をゾーンに割り当てます。

次のコマンドは、クラスタのプライベート IP アドレスの範囲から、使用可能な IP アドレスを選択し、割り当てます。またこのコマンドは、指定されたプライベートホスト名、またはホスト別名をゾーンに割り当て、割り当てられたプライベート IP アドレスにそれをマッピングします。

```
phys-schost# clnode set -p zprivatehostname=hostalias node:zone
```

<code>-p</code>	プロパティを指定します。
<code>zprivatehostname=hostalias</code>	ゾーンプライベートホスト名、またはホスト別名を指定します。
<code>node</code>	ノードの名前。
<code>zone</code>	グローバルクラスタの非投票ノードの名前。

- 7 初期内部ゾーン構成を実行します。

『Solaris のシステム管理 (Solaris コンテナ: 資源管理と Solaris ゾーン)』の「初期内部ゾーン構成を実行する」の手順に従います。次のどちらかの方法を選択します。

- ゾーンにログインします。
- `/etc/sysidcfg` ファイルを使用します。

- 8 非投票ノードで、`nsswitch.conf` ファイルを変更します。

これらの変更により、クラスタ固有のホスト名と IP アドレスの検索をゾーンが解決できるようになります。

- a. ゾーンにログインします。

```
phys-schost# zlogin -c zonename
```

- b. 編集するため `/etc/nsswitch.conf` ファイルを開きます。

```
sczone# vi /etc/nsswitch.conf
```

- c. `hosts` エントリと `netmasks` エントリのルックアップの先頭に、`cluster` スイッチを追加し、その後 `files` スイッチを追加します。  
変更されたエントリは次のようになるはずです。

```
...
hosts: cluster files nis [NOTFOUND=return]
...
netmasks: cluster files nis [NOTFOUND=return]
...
```

- d. ほかのすべてのエントリでは、`files` スイッチが、エントリに一覧表示される最初のスイッチになるようにする必要があります。
- e. ゾーンを終了します。
- 9 排他的 IP ゾーンを作成した場合は、そのゾーン上の `/etc/hostname.interface` ファイルごとに IPMP グループを構成します。  
ゾーン内のデータサービストラフィックに使用されているパブリックネットワークアダプタごとに、IPMP グループを設定します。この情報は、大域ゾーンから継承されません。クラスタでの IPMP グループの設定については、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「パブリックネットワーク」を参照してください。
- 10 ゾーンにより使用されるすべての論理ホスト名リソースの名前とアドレスのマッピングを設定します。
- a. 名前とアドレスのマッピングを、ゾーン上の `/etc/inet/hosts` ファイルに追加します。  
この情報は、大域ゾーンから継承されません。
- b. ネームサーバーを使用している場合は、名前とアドレスのマッピングを追加します。

## クラスタからのノードの削除

このセクションでは、グローバルクラスタまたはゾーンクラスタ上のノードを削除する手順について説明します。グローバルクラスタから特定のゾーンクラスタを削除することもできます。次の表に、ノードを既存のクラスタから削除するときに行う作業を示します。作業は、示されている順に実行してください。



注意-RAC構成の場合、この手順のみを使用してノードを削除すると、再起動中のノードでパニックが発生する可能性があります。RAC構成からノードを削除する方法については、『[Sun Cluster Data Service for Oracle RAC Guide for Solaris OS](#)』の「[How to Remove Sun Cluster Support for Oracle RAC From Selected Nodes](#)」を参照してください。このプロセスが終了したら、次の適切な手順に従ってください。

表 8-2 作業マップ: ノードの削除

作業	参照先
削除するノードからすべてのリソースグループとデバイスグループを移動	<code>clnode evacuate node</code>
許可されたホストをチェックして、ノードを削除できることを確認	<code>claccess show node</code> <code>claccess allow -h node-to-remove</code>
ノードを削除できない場合に、クラスタ構成へのアクセス権をノードに付与	
すべてのデバイスグループからノードを削除	143 ページの「デバイスグループからノードを削除する (Solaris ボリュームマネージャー)」
削除するノードに接続されているすべての定足数デバイスを削除	2 ノードクラスタのノードを削除する場合、この手順は省略可能です。  213 ページの「定足数デバイスを削除する」  次の手順では、ストレージデバイスを削除する前に定足数デバイスを削除する必要がありますが、定足数デバイスはその後追加し直すことができます。  215 ページの「クラスタから最後の定足数デバイスを削除する」
削除するノードを非クラスタモードにする	285 ページの「ノードを保守状態にする」
ゾーンクラスタからノードを削除	263 ページの「ゾーンクラスタからノードを削除する」
クラスタソフトウェア構成からノードを削除	263 ページの「クラスタソフトウェア構成からノードを削除する」
(省略可能) Sun Cluster ソフトウェアをクラスタノードからアンインストール	295 ページの「Sun Cluster ソフトウェアをクラスタノードからアンインストールする」
ゾーンクラスタ全体の削除	290 ページの「ゾーンクラスタを削除する」

## ▼ ゾーンクラスタからノードを削除する

ノードを停止し、アンインストールして、構成からノードを削除することで、ゾーンクラスタからノードを削除することができます。あとでノードをゾーンクラスタに戻す場合は、表 8-1 の手順に従います。ここからの手順のほとんどは、グローバルクラスタノードから実行します。

- 1 グローバルクラスタのノードのスーパーユーザーになります。
- 2 ノードとそのゾーンクラスタを指定して、削除するゾーンクラスタノードを停止します。  

```
phys-schost# clzonecluster halt -n node zoneclustername
```

ゾーンクラスタ内で `clnode evacuate` コマンドと `shutdown` コマンドを使用することもできます。
- 3 ゾーンクラスタノードをアンインストールします。  

```
phys-schost# clzonecluster uninstall -n node zoneclustername
```
- 4 ゾーンクラスタノードを構成から削除します。  
 次のコマンドを使用します。  

```
phys-schost# clzonecluster configure zoneclustername
```

```
clzc:sczone> remove node physical-host=zoneclusternodename
```
- 5 ノードがゾーンクラスタから削除されたことを確認します。  

```
phys-schost# clzonecluster status
```

## ▼ クラスタソフトウェア構成からノードを削除する

この手順を実行して、ノードをグローバルクラスタから削除します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 この手順を実行する前に、ノードをすべてのリソースグループ、デバイスグループ、および定数デバイスの構成から削除していること、および、このノードを保守状態にしていることを確認します。

- 2 削除するノードで、スーパーユーザーになるか、**RBAC**の承認 `solaris.cluster.modify` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 削除するグローバルクラスタノードを非クラスタモードで起動します。ゾーンクラスタノードの場合は、この手順を実行する前に、[263 ページの「ゾーンクラスタからノードを削除する」](#)の手順を実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
shutdown -g -y -i0
```

```
Press any key to continue
```

- a. **GRUB** メニューで矢印キーを使用して該当する **Solaris** エントリを選択し、`e` と入力してコマンドを編集します。

GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

GRUB ベースの起動についての詳細は、『[Solaris のシステム管理 \(基本編\)](#)』の「[GRUB を使用して x86 システムをブートする \(作業マップ\)](#)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、`e` を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
```

selected line, or escape to go back to the main menu.

- c. コマンドに `-x` を追加して、システムを非クラスタモードで起動するように指定します。

```
[Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits.]
```

```
grub edit> kernel /platform/i86pc/multiboot -x
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot -x |
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b** と入力して、ノードを非クラスタモードで起動します。

カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-x` オプションを追加します。

---

注 - 削除するノードが使用できない場合や、起動できなくなっている場合は、アクティブな任意のクラスタノードで `clnode clear -F <node-to-be-removed>` コマンドを実行します。 `clnode status <nodename>` を実行して、ノードが削除されていることを確認します。

---

- 4 削除するノードで、ノードをクラスタから削除します。

```
phys-schost# clnode remove -F
```

`clnode remove` コマンドが失敗し、無効なノード参照が存在したままになっている場合は、アクティブなノードで `clnode clear -F nodename` を実行します。

注-クラスタ内の最後のノードを削除する場合は、そのノードがクラスタモードモードでないこと、およびクラスタ内にアクティブなノードがないことが必要です。

- 5 別のクラスタノードから、ノードの削除を確認します。

```
phys-schost# clnode status nodename
```

- 6 ノードの削除を完了します。

- 削除するノードから **Sun Cluster** ソフトウェアをアンインストールする場合は、[295 ページの「Sun Cluster ソフトウェアをクラスタノードからアンインストールする」](#)に進んでください。
- 削除するノードから **Sun Cluster** ソフトウェアをアンインストールしない場合は、『[Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS](#)』で説明されているように、ハードウェア接続を削除することにより、クラスタからノードを物理的に削除できます。

## 例 8-2 クラスタソフトウェア構成からのノードの削除

次に、ノード `phys-schost-2` をクラスタから削除する例を示します。`clnode remove` コマンドは、クラスタから削除するノード (`phys-schost-2`) から非クラスタモードで実行されます。

```
[Remove the node from the cluster:]
phys-schost-2# clnode remove
phys-schost-1# clnode clear -F phys-schost-2
[Verify node removal:]
phys-schost-1# clnode status
-- Cluster Nodes --
 Node name Status
 ----- -
Cluster node: phys-schost-1 OnLine
```

参照 削除するノードから **Sun Cluster** ソフトウェアをアンインストールする方法については、[295 ページの「Sun Cluster ソフトウェアをクラスタノードからアンインストールする」](#)を参照してください。

ハードウェア手順については、『[Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS](#)』を参照してください。

クラスタノードを削除する作業の概要については、[表 8-2](#)を参照してください。

ノードを既存のクラスタに追加する方法については、[256 ページの「ノードを認証ノードリストに追加する」](#)を参照してください。

## ▼ グローバルクラスタから非投票ノード(ゾーン)を削除する

- 1 非投票ノードを作成したグローバルクラスタノードでスーパーユーザーになります。
- 2 システムから非投票ノードを削除します。  
『Solaris のシステム管理 (Solaris コンテナ: 資源管理と Solaris ゾーン)』の「システムから非大域ゾーンを削除する」の手順に従います。

## ▼ 2 ノード接続より大きなクラスタでアレイと単一ノード間の接続を削除する

3 ノードまたは4 ノード接続のクラスタでストレージアレイを単一クラスタノードから取り外すには、次の手順を使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 取り外す予定のストレージアレイに関連付けられているすべてのデータベーステーブル、データサービス、ボリュームのバックアップを作成します。
- 2 切断する予定のノードで動作しているリソースグループとデバイスグループを判別します。  
`phys-schost# clresourcegroup status`  
`phys-schost# cldevicegroup status`
- 3 必要であれば、切断する予定のノードからすべてのリソースグループとデバイスグループを移動します。



---

**Caution (SPARC only)** – Oracle RAC ソフトウェアをクラスタで実行している場合、グループをノードから移動する前に、動作している Oracle RAC データベースのインスタンスを停止します。手順については、『Oracle Database Administration Guide』を参照してください。

---

```
phys-schost# clnode evacuate node
```

`clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定ノード上の投票ノードまたは非投票ノードから、次に優先される投票ノードまたは非投票ノードへ、すべてのリソースグループを切り替えます。

4 デバイスグループを保守状態にします。

Veritas 共有ディスクグループへの入出力活動を休止させる手順については、VxVMのマニュアルを参照してください。

デバイスグループを保守状態にする手順については、285 ページの「ノードを保守状態にする」を参照してください。

5 デバイスグループからノードを削除します。

- VxVM または raw ディスクを使用している場合は、`cldevicegroup(1CL)` コマンドを使用して、デバイスグループを削除します。
- Solstice DiskSuite を使用している場合は、`metaset` コマンドを使用して、デバイスグループを削除します。

6 HAStoragePlus リソースが含まれる各リソースグループで、リソースグループのノードリストからノードを削除します。

```
phys-schost# clresourcegroup remove-node -z zone -n node + | resourcegroup
```

*node* ノードの名前。

*zone* リソースグループをマスターできる非投票ノードの名前。リソースグループを作成した際に非投票ノードを指定した場合にのみ、*zone* を指定します。

リソースグループのノードリストを変更する方法の詳細は、『[Sun Cluster Data Services Planning and Administration Guide for Solaris OS](#)』を参照してください。

注-`clresourcegroup` を実行するときには、リソースタイプ、リソースグループ、およびリソースのプロパティ名には大文字と小文字の区別があります。

7 削除する予定のストレージレイがノードに接続されている最後のストレージレイである場合、当該ストレージレイに接続されているハブまたはスイッチとノードの間にある光ケーブルを取り外します(そうでない場合、この手順は省略します)。

8 切断するノードからホストアダプタを削除する場合、ノードの電源を切ります。切断するノードからホストアダプタを削除する場合は、[手順 11](#)に進みます。

9 ノードからホストアダプタを削除します。

ホストアダプタの削除手順については、ノード用ドキュメントを参照してください。

- 10 起動が行われないうようにして、ノードに電源を入れます。
- 11 Oracle RAC ソフトウェアがインストールされている場合、切断する予定のノードからそのパッケージを削除します。

```
phys-schost# pkgrm SUNWscucm
```



**Caution (SPARC only)** – 切断したノードから Oracle RAC ソフトウェアを削除しない場合、そのノードをクラスタに導入し直すときにノードでパニックが発生し、データの可用性が失われる可能性があります。

- 12 クラスタモードでノードを起動します。
- SPARC ベースのシステム上で、次のコマンドを実行します。
 

```
ok boot
```
  - x86 ベースのシステム上で、次のコマンドを実行します。
 

GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```
- 13 ノードの /devices と /dev エントリを更新して、デバイスの名前空間を更新します。

```
phys-schost# devfsadm -C
cldevice refresh
```

- 14 デバイスグループをオンラインに戻します。
- Veritas 共有ディスクグループをオンラインにする手順については、Veritas Volume Manager のマニュアルを参照してください。
- デバイスグループをオンラインにする方法については、287 ページの「ノードを保守状態から戻す」を参照してください。

## ▼ エラーメッセージを修正する

クラスタノードの削除手順のいずれかを実行中に発生したエラーメッセージを修正するには、次の手順を実行します。

- 1 グローバルクラスタへのノードの再結合を試みます。この手順は、グローバルクラスタ上のみで実行します。

```
phys-schost# boot
```

- 2 ノードがクラスタに正常に再結合されているかどうかを確認します。
  - 再結合されていない場合は、[手順 b](#)に進みます。
  - 再結合されている場合は、次の各手順を行なってノードをデバイスグループから削除します。
    - a. ノードが正常にクラスタに再結合された場合は、残っているデバイスグループからノードを削除します。  
[142 ページ](#)の「すべてのデバイスグループからノードを削除する」の作業を行います。
    - b. すべてのデバイスグループからノードを削除したあと、[295 ページ](#)の「[Sun Cluster ソフトウェアをクラスタノードからアンインストールする](#)」に戻り、その手順を繰り返します。
- 3 ノードがクラスタに再結合されなかった場合は、ノードの `/etc/cluster/ccr` ファイルを他の名前に変更します (たとえば、`ccr.old`)。

```
mv /etc/cluster/ccr /etc/cluster/ccr.old
```

- 4 [295 ページ](#)の「[Sun Cluster ソフトウェアをクラスタノードからアンインストールする](#)」に戻り、その手順を繰り返します。

# クラスタの管理

---

この章では、グローバルクラスタやゾーンクラスタ全体に影響する管理手順について説明します。

- 271 ページの「クラスタの管理の概要」
- 289 ページの「ゾーンクラスタ管理タスクの実行」
- 305 ページの「障害追跡」

クラスタへのノードの追加または削除に関する詳細は、第 8 章「ノードの追加と削除」を参照してください。

## クラスタの管理の概要

このセクションでは、グローバルクラスタやゾーンクラスタ全体の管理作業を実行する方法を説明します。次の表に、これらの管理作業と、関連する手順を示します。Solaris 10 OS の場合、クラスタの管理作業は通常は大域ゾーンで行います。ゾーンクラスタを管理するには、そのゾーンクラスタをホストするマシンが 1 台以上クラスタモードで起動していることが必要です。すべてのゾーンクラスタノードが起動し動作している必要はありません。現在クラスタ外にあるノードがクラスタに再結合すると、構成の変更点が Sun Cluster によって再現されます。

この章での `phys-schost#` は、グローバルクラスタのプロンプトを表します。`clzonecluster` の対話型シェルプロンプトは `clzc:schost>` です。

表 9-1 作業リスト:クラスタの管理

作業	参照先
クラスタへのノードの追加または削除	第 8 章「ノードの追加と削除」
クラスタ名を変更	272 ページの「クラスタ名を変更する」

作業	参照先
ノード ID およびそれらの対応するノード名の一覧の表示	273 ページの「ノード ID をノード名にマップする」
クラスタへの新しいノードの追加を許可または拒否	274 ページの「新しいクラスタノード認証で作業する」
Network Time Protocol (NTP) を使用してクラスタの時刻を変更	276 ページの「クラスタの時刻をリセットする」
ノードを停止し、SPARC ベースのシステムでは OpenBoot PROM ok プロンプト、x86 ベースのシステムでは GRUB メニューで「Press any key to continue」というメッセージを表示	278 ページの「SPARC: ノードで OpenBoot PROM (OBP) を表示する」
プライベートホスト名の追加または変更	282 ページの「グローバルクラスタ上の非投票ノードのプライベートホスト名を追加する」 279 ページの「ノードのプライベートホスト名を変更する」
クラスタノードを保守状態に変更	285 ページの「ノードを保守状態にする」
クラスタノードを保守状態から復帰	287 ページの「ノードを保守状態から戻す」
ゾーンクラスタの移動、アプリケーション用ゾーンクラスタの準備、ゾーンクラスタの削除	289 ページの「ゾーンクラスタ管理タスクの実行」
ノードから Sun Cluster ソフトウェアをアンインストールします。	295 ページの「Sun Cluster ソフトウェアをクラスタノードからアンインストールする」
SNMP Event MIB の追加および管理	299 ページの「SNMP イベント MIB を有効にする」 303 ページの「SNMP ユーザーをノードに追加する」

## ▼ クラスタ名を変更する

必要に応じて、初期インストール後にクラスタ名を変更できます。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 グローバルクラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 `clsetup`ユーティリティーを起動します。  

```
phys-schost# clsetup
```

 メインメニューが表示されます。
- 3 クラスタ名を変更するには、クラスタその他のプロパティのオプションに対応する番号を入力します。  
 「クラスタその他のプロパティ」メニューが表示されます。
- 4 メニューから選択を行って、画面の指示に従います。
- 5 **Sun Cluster** のサービスタグに新しいクラスタ名を反映させる場合は、既存の **Sun Cluster** タグを削除してクラスタを再起動します。 **Sun Cluster** サービスタグインスタンスを削除するには、クラスタ内のすべてのノードで次のサブステップを完了します。
  - a. すべてのサービスタグの一覧を表示します。  

```
phys-schost# stclient -x
```
  - b. **Sun Cluster** サービスタグインスタンス番号を見つけて、次のコマンドを実行します。  

```
phys-schost# stclient -d -i service_tag_instance_number
```
  - c. クラスタ内のすべてのノードを再起動します。  

```
phys-schost# reboot
```

#### 例 9-1 クラスタ名の変更

次の例に、新しいクラスタ名 `dromedary` へ変更するために、`clsetup(1CL)` ユーティリティーから生成される `cluster(1CL)` コマンドを示します。

```
phys-schost# cluster -c dromedary
```

## ▼ ノード ID をノード名にマップする

**Sun Cluster** のインストール中、ノードにはそれぞれ一意のノード ID 番号が自動で割り当てられます。このノード ID 番号は、最初にクラスタに加わったときの順番でノードに割り当てられます。ノード ID 番号が割り当てられたあとでは、番号は変更できません。ノード ID 番号は、通常、エラーメッセージが発生したクラスタノードを識別するために、エラーメッセージで使用されます。この手順を使用し、ノード ID とノード名間のマッピングを判別します。

グローバルクラスタまたはゾーンクラスタ用の構成情報を表示するために、スーパーユーザーになる必要はありません。グローバルクラスタのノードから、このプロシーチャーの1ステップが実行されます。他のステップはゾーンクラスタノードから実行されます。

- 1 `clnode(1CL)` コマンドを使用して、グローバルクラスタに対するクラスタ構成情報を一覧表示します。

```
phys-schost# clnode show | grep Node
```

- 2 1つのゾーンクラスタに対して、複数のノードIDを一覧表示することも可能です。ゾーンクラスタノードは、実行中のグローバルクラスタノードと同じノードIDを持っています。

```
phys-schost# zlogin sczone clnode -v | grep Node
```

### 例 9-2 ノード名のノードIDへのマップ

次の例は、グローバルクラスタに対するノードIDの割り当てを示しています。

```
phys-schost# clnode show | grep Node
=== Cluster Nodes ===
Node Name: phys-schost1
Node ID: 1
Node Name: phys-schost2
Node ID: 2
Node Name: phys-schost3
Node ID: 3
```

## ▼ 新しいクラスタノード認証で作業する

Sun Cluster では、新しいノードをグローバルクラスタに追加できるようにするかどうかと、使用する認証の種類を指定できます。パブリックネットワーク上のクラスタに加わる新しいノードを許可したり、新しいノードがクラスタに加わることを拒否したり、クラスタに加わるノードを特定できます。新しいノードは、標準 UNIX または Diffie-Hellman (DES) 認証を使用し、認証することができます。DES 認証を使用して認証する場合、ノードに加わるには、すべての必要な暗号化鍵を構成する必要があります。詳細は、[keyserv\(1M\)](#) と [publickey\(4\)](#) のマニュアルページを参照してください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 グローバルクラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 `clsetup(1CL)` ユーティリティーを起動します。  
`phys-schost# clsetup`  
メインメニューが表示されます。
- 3 クラスタ認証で作業するため、新規ノードのオプションに対応する番号を入力します。  
「新規ノード」メニューが表示されます。
- 4 メニューから選択を行って、画面の指示に従います。

#### 例 9-3 新しいマシンがグローバルクラスタに追加されないようにする

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、新しいマシンがクラスタに追加されないようにする `claccess` コマンドを示しています。

```
phys-schost# claccess deny -h hostname
```

#### 例 9-4 すべての新しいマシンがグローバルクラスに追加されることを許可する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、すべての新しいマシンをクラスタに追加できるようにする `claccess` コマンドを示しています。

```
phys-schost# claccess allow-all
```

#### 例 9-5 グローバルクラスタに追加される新しいマシンを指定する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、1台の新しいマシンをクラスタに追加できるようにする `claccess` コマンドを示しています。

```
phys-schost# claccess allow -h hostname
```

#### 例 9-6 認証を標準 UNIX に設定する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、クラスタに参加している新規ノードの標準 UNIX 認証に対し、リセットを行う `claccess` コマンドを示しています。

```
phys-schost# claccess set -p protocol=sys
```

### 例 9-7 認証を DES に設定する

clsetup ユーティリティにより、claccess コマンドを生成します。次の例は、クラスタに参加している新規ノードの DES 認証を使用する claccess コマンドを示しています。

```
phys-schost# claccess set -p protocol=des
```

DES 認証を使用する場合、クラスタにノードが加わるには、すべての必要な暗号化鍵を構成します。詳細は、[keyerv\(1M\)](#) と [publickey\(4\)](#) のマニュアルページを参照してください。

## ▼ クラスタの時刻をリセットする

Sun Cluster ソフトウェアは、Network Time Protocol (NTP) を使用し、クラスタノード間で時刻を同期させています。グローバルクラスタの時刻の調整は、ノードが時刻を同期するときに、必要に応じて自動的に行われます。詳細は、『[Sun Cluster Concepts Guide for Solaris OS](#)』および『[Network Time Protocol User's Guide](#)』を参照してください。



注意 - NTP を使用する場合、クラスタの稼動中はクラスタの時刻を調整しないでください。[date\(1\)](#)、[rdate\(1M\)](#)、[xntpd\(1M\)](#)、[svcadm\(1M\)](#) などのコマンドを、対話的に使用したり、[cron\(1M\)](#) スクリプト内で使用して時刻を調整しないでください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 グローバルクラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 グローバルクラスタを停止します。

```
phys-schost# cluster shutdown -g0 -y -i 0
```

- 3 SPARC ベースのシステムではノードが ok プロンプトを表示し、x86 ベースのシステムでは GRUB メニューで「Press any key to continue」というメッセージが表示されていることを確認します。
- 4 非クラスタモードでノードを起動します。
  - SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
shutdown -g -y -i0
```

```
Press any key to continue
```

- GRUB メニューで矢印キーを使用して該当する **Solaris** エントリを選択し、e と入力してコマンドを編集します。

GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

GRUB ベースの起動についての詳細は、『[Solaris のシステム管理 \(基本編\)](#)』の「[GRUB を使用して x86 システムをブートする \(作業マップ\)](#)」を参照してください。

- ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、e を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.
```

- コマンドに -x を追加して、システムを非クラスタモードで起動するように指定します。

```
[Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits.]
```

```
grub edit> kernel /platform/i86pc/multiboot -x
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot -x |
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-x` オプションを追加します。

---

- 5 単一のノードで、`date` コマンドを実行して時刻を設定します。  
`phys-schost# date HHMM.SS`
- 6 ほかのマシンで、`rdate(1M)` コマンドを実行し、時刻をこのノードに同期化します。  
`phys-schost# rdate hostname`
- 7 各ノードを起動し、クラスタを再起動します。  
`phys-schost# reboot`
- 8 すべてのクラスタノードで変更が行われたことを確認します。  
 各ノードで、`date` コマンドを実行します。  
`phys-schost# date`

## ▼ SPARC: ノードで OpenBoot PROM (OBP) を表示する

OpenBoot™ PROM 設定を構成または変更する必要がある場合は、この手順を使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 停止するノード上でコンソールに接続します。

```
telnet tc_name tc_port_number
```

*tc\_name*                    端末集配信装置 (コンセントレータ) の名前を指定します。

*tc\_port\_number*          端末集配信装置のポート番号を指定します。ポート番号は構成に依存します。通常、ポート 2 (5002) とポート 3 (5003) は、サイトで最初に設置されたクラスタで使用されています。

- 2 `clnode evacuate` コマンドを使用してから、`shutdown` コマンドを使用することで、クラスタノードを正常に停止します。`clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、グローバルクラスタ内の指定された投票ノードまたは非投票ノードから、次に優先される投票ノードまたは非投票ノードに、すべてのリソースグループを切り替えます。

```
phys-schost# clnode evacuate node
```

```
shutdown -g0 -y
```



注意 - クラスタノードを停止する場合は、クラスタコンソール上で `send brk` を使用してはいけません。

- 3 OBP コマンドを実行します。

## ▼ ノードのプライベートホスト名を変更する

このプロシージャを使用して、インストール終了後にクラスタノードのプライベートホスト名を変更します。

デフォルトのプライベートホスト名は、クラスタの初期インストール時に割り当てられます。デフォルトのプライベートホスト名の形式は、`clusternode<nodeid>-priv` です (`clusternode3-priv` など)。名前がすでにドメイン内で使用されている場合にかぎり、プライベートホスト名を変更します。



注意 - 新しいプライベートホスト名には IP アドレスを割り当てないでください。クラスタソフトウェアが IP アドレスを割り当てます。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 クラスタ内のすべてのノード上で、プライベートホスト名をキャッシュする可能性があるデータサービスリソースやアプリケーションをすべて無効にします。

```
phys-schost# clresource disable resource[,...]
```

無効にするアプリケーションには次のようなものがあります。

- HA-DNS と HA-NFS サービス (構成している場合)
- プライベートホスト名を使用するためにカスタム構成されたアプリケーション
- クライアントがプライベートインターコネクト経由で使用しているアプリケーション

clresource コマンドの使用に関する詳細は、clresource(1CL) のマニュアルページおよび『Sun Cluster Data Services Planning and Administration Guide for Solaris OS』を参照してください。

- 2 NTP 構成ファイルが、変更しようとするプライベートホスト名を参照している場合、クラスタの各ノード上で Network Time Protocol (NTP) デーモンを停止します。

- SPARC: Solaris 9 OS を使用している場合は、xntpd コマンドを使用して NTP デーモンを停止してください。NTP デーモンについての詳細は、xntpd(1M) のマニュアルページを参照してください。

```
phys-schost# /etc/init.d/xntpd.cluster stop
```

- Solaris 10 OS を使用している場合は、svcadm コマンドを使用して NTP デーモンを停止してください。NTP デーモンについての詳細は、svcadm(1M) のマニュアルページを参照してください。

```
phys-schost# svcadm disable ntp
```

- 3 clsetup(1CL) ユーティリティーを実行して、適切なノードのプライベートホスト名を変更します。

クラスタ内の 1 つのノードでのみユーティリティーを実行します。

---

注-新しいプライベートホスト名を選択するときには、その名前がクラスタノード内で一意であることを確認してください。

---

- 4 プライベートホスト名用のオプションに対応する番号を入力します。

- 5 プライベートホスト名を変更するためのオプションに対応する番号を入力します。表示される質問に答えます。変更しようとしているプライベートホスト名のノード名 (`clusternode<nodeid> -priv`) および新しいプライベートホスト名を入力してください。
- 6 ネームサービスキャッシュをフラッシュします。

クラスタの各ノードで次の手順を実行します。フラッシュすることによって、クラスタアプリケーションとデータサービスが古いプライベートホスト名にアクセスしないようにします。

```
phys-schost# nscd -i hosts
```
- 7 NTP 構成ファイルでプライベートホスト名を変更した場合、各ノード上で NTP 構成ファイル (`ntp.conf` または `ntp.conf.cluster`) を更新します。
  - a. 任意のエディタを使用してください。

この手順をインストール時に行う場合は、構成するノードの名前を削除する必要があります。デフォルトのテンプレートには 16 のノードが事前構成されています。通常 `ntp.conf.cluster` ファイルは各クラスタノード上で同じです。
  - b. すべてのクラスタノードから新しいプライベートホスト名に **ping** を実行できることを確認します。
  - c. NTP デーモンを再起動します。

クラスタの各ノードで次の手順を実行します。

    - SPARC: Solaris 9 OS を使用している場合は、`xntpd` コマンドを使用して NTP デーモンを再起動してください。

`ntp.conf.cluster` ファイルを使用している場合は、次のように入力します。

```
/etc/init.d/xntpd.cluster start
```

`ntp.conf` ファイルを使用している場合は、次のように入力します。

```
/etc/init.d/xntpd start
```
    - Solaris 10 OS を使用している場合は、`svcadm` コマンドを使用して NTP デーモンを再起動してください。

```
svcadm enable ntp
```
- 8 手順 1 で無効にしたデータサービスリソースとそのほかのアプリケーションをすべて有効にします。

```
phys-schost# clresource disable resource[,...]
```

scswitch コマンドの使用に関する詳細は、`clresource(1CL)` のマニュアルページおよび『Sun Cluster Data Services Planning and Administration Guide for Solaris OS』を参照してください。

### 例 9-8 プライベートホスト名を変更する

次に、ノード `phys-schost-2` 上のプライベートホスト名 `clusternode2-priv` を `clusternode4-priv` に変更する例を示します。

```
[Disable all applications and data services as necessary.]
phys-schost-1# /etc/init.d/xntpd stop
phys-schost-1# clnode show | grep node
...
private hostname: clusternode1-priv
private hostname: clusternode2-priv
private hostname: clusternode3-priv
...
phys-schost-1# clsetup
phys-schost-1# nscd -i hosts
phys-schost-1# vi /etc/inet/ntp.conf
...
peer clusternode1-priv
peer clusternode4-priv
peer clusternode3-priv
phys-schost-1# ping clusternode4-priv
phys-schost-1# /etc/init.d/xntpd start
[Enable all applications and data services disabled at the beginning of the procedure.]
```

## ▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を追加する

インストール完了後、グローバルクラスタ上の非投票ノードのプライベートホスト名を追加するには、次の手順を使用します。この章の手順の `phys-schost#` は、グローバルクラスタプロンプトを表します。この手順は、グローバルクラスタ上のみで実行します。

- 1 `clsetup(1CL)` ユーティリティを実行して、適切なゾーンにプライベートホスト名を追加します。  
`phys-schost# clsetup`
- 2 プライベートホスト名のオプションに対応する番号を入力し、**Return** キーを押します。

- 3 ゾーンプライベートホスト名を追加するためのオプションに対応する番号を入力し、**Return** キーを押します。
- 表示される質問に答えます。グローバルクラスタの非投票ノードには、デフォルトのプライベートホスト名はありません。ホスト名を入力する必要があります。

## ▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を変更する

インストール完了後、非投票ノードのプライベートホスト名を変更するには、次の手順を使用します。

プライベートホスト名は、クラスタの初期インストール時に割り当てられます。プライベートホスト名の形式は、`clusternode<nodeid>-priv` です (`clusternode3-priv` など)。名前がすでにドメイン内で使用されている場合にかぎり、プライベートホスト名を変更します。



注意 - 新しいプライベートホスト名には IP アドレスを割り当てないでください。クラスタソフトウェアが IP アドレスを割り当てます。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 グローバルクラスタ内のすべてのノード上で、プライベートホスト名をキャッシュする可能性があるデータサービスリソースやアプリケーションをすべて無効にします。

```
phys-schost# clresource disable resource1, resource2
```

無効にするアプリケーションには次のようなものがあります。

- HA-DNS と HA-NFS サービス (構成している場合)
- プライベートホスト名を使用するためにカスタム構成されたアプリケーション
- クライアントがプライベートインターコネクト経由で使用しているアプリケーション

`clresource` コマンドの使用に関する詳細は、`clresource(1CL)` のマニュアルページおよび『[Sun Cluster Data Services Planning and Administration Guide for Solaris OS](#)』を参照してください。

- 2 **clsetup(1CL)** ユーティリティーを実行して、グローバルクラスタ上の適切な非投票ノードのプライベートホスト名を変更します。

```
phys-schost# clsetup
```

この手順は、クラスタ内の1つのノードからのみ実行する必要があります。

---

注-新しいプライベートホスト名を選択するときには、その名前がクラスタ内で一意であることを確認してください。

---

- 3 プライベートホスト名のオプションに対応する番号を入力し、**Return** キーを押します。
- 4 ゾーンプライベートホスト名を追加するためのオプションに対応する番号を入力し、**Return** キーを押します。  
グローバルクラスタのプライベートホスト名の非投票ノードには、デフォルトは存在しません。ホスト名を入力する必要があります。
- 5 ゾーンプライベートホスト名を変更するためのオプションに対応する番号を入力します。  
表示される質問に答えます。ここでは、プライベートホスト名を変更する非投票ノードの名前 (`clusternode<nodeid> -priv`) と新しいプライベートホスト名を入力してください。
- 6 ネームサービスキャッシュをフラッシュします。  
クラスタの各ノードで次の手順を実行します。フラッシュすることによって、クラスタアプリケーションとデータサービスが古いプライベートホスト名にアクセスしないようにします。  

```
phys-schost# nscd -i hosts
```
- 7 **手順1**で無効にしたデータサービスリソースとそのほかのアプリケーションをすべて有効にします。

## ▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を削除する

グローバルクラスタ上の非投票ノードのプライベートホスト名を削除するには、次の手順を使用します。この手順は、グローバルクラスタ上のみで実行します。

- 1 **clsetup(1CL)** ユーティリティーを実行して、適切なゾーン上のプライベートホスト名を削除します。

- 2 ゾーンのプライベートホスト名用のオプションに対応する番号を入力します。
- 3 ゾーンプライベートホスト名を削除するためのオプションに対応する番号を入力します。
- 4 削除する非投票ノードのプライベートホスト名の名前を入力します。

## ▼ ノードを保守状態にする

サービスからグローバルクラスタノードを長時間はずす場合は、そのノードを保守状態にします。保守状態のノードは、サービス対象中に定足数確立の投票に参加しません。クラスタノードを保守状態にするには、`clnode(1CL) evacuate` および `cluster(1CL) shutdown` コマンドを使用して、ノードを停止しておく必要があります。

---

注 - ノードを1つだけ停止する場合は、Solaris の `shutdown` コマンドを使用します。クラスタ全体を停止する場合にだけ、`cluster shutdown` コマンドを使用します。

---

クラスタノードが停止されて保守状態になると、そのノードのポートで構成されるすべての定足数デバイスの、定足数投票数 (quorum vote count) が1つ減ります。このノードが保守状態から移動してオンラインに戻されると、ノードおよび定足数デバイスの投票数は1つ増えます。

クラスタノードを保守状態にするには、`clquorum(1CL) disable` コマンドを使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 保守状態にするグローバルクラスタノード上で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 すべてのリソースグループとデバイスグループをノードから退避します。`clnode evacuate` コマンドは、すべての非投票ノードを含むすべてのリソースグループとデバイスグループを、指定ノードから次に優先されるノードに切り替えます。

```
phys-schost# clnode evacuate node
```

- 3 退避させたノードを停止します。

```
phys-schost# shutdown -g0 -y-i 0
```

- 4 クラスタ内の別のノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になり、手順3で停止したノードを保守状態にします。

```
phys-schost# clquorum disable node
```

node                    保守モードにするノードの名前を指定します。

- 5 グローバルクラスタノードが保守状態にあることを確認します。

```
phys-schost# clquorum status node
```

保守状態にしたノードのStatusはofflineで、そのPresentとPossibleの定足数投票数は0(ゼロ)である必要があります。

### 例9-9 グローバルクラスタノードを保守状態にする

次に、クラスタノードを保守状態にして、その結果を確認する例を示します。 `clnode status` の出力では、 `phys-schost-1` のノードの `Node votes` は0(ゼロ)で、その状態はOfflineです。 `Quorum Summary` では、投票数も減っているはずですが、構成によって異なりますが、 `Quorum Votes by Device` の出力では、いくつかの定足数ディスクデバイスもオフラインである可能性があります。

[On the node to be put into maintenance state:]

```
phys-schost-1# clnode evacuate phys-schost-1
phys-schost-1# shutdown -g0 -y -i0
```

[On another node in the cluster:]

```
phys-schost-2# clquorum disable phys-schost-1
phys-schost-2# clquorum status phys-schost-1
```

-- Quorum Votes by Node --

Node Name	Present	Possible	Status
-----	-----	-----	-----
phys-schost-1	0	0	Offline
phys-schost-2	1	1	Online
phys-schost-3	1	1	Online

参照 ノードをオンライン状態に戻す方法については、287ページの「ノードを保守状態から戻す」を参照してください。

## ▼ ノードを保守状態から戻す

次の手順を使用して、グローバルクラスタノードをオンラインに戻し、定足数投票数をリセットしてデフォルト設定に戻します。クラスタノードのデフォルトの投票数は1です。定足数デバイスのデフォルトの投票数は $N-1$ です。 $N$ は、投票数が0以外で、定足数デバイスが構成されているポートを持つノードの数を示します。

ノードが保守状態になると、そのノードの投票数は1つ減ります。また、このノードのポートに構成されているすべての定足数デバイスの投票数も(1つ)減ります。投票数がリセットされ、ノードが保守状態から戻されると、ノードの投票数と定足数デバイスの投票数の両方が1つ増えます。

保守状態にしたグローバルクラスタノードを保守状態から戻した場合は、必ずこの手順を実行してください。



注意 - `globaldev` または `node` オプションのどちらも指定しない場合、定足数投票数はクラスタ全体でリセットされます。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 グローバルクラスタの、保守状態のノード以外の任意のノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。
- 2 グローバルクラスタ構成内にあるノードの数に応じて、次の手順のいずれかを実行します。
  - クラスタ構成内に2つのノードがある場合は、手順4に進みます。
  - クラスタ構成内に3つ以上のノードがある場合は、手順3に進みます。

- 3 保守状態から解除するノードに定足数デバイスがある場合は、保守状態にあるノード以外のノードからクラスタ定足数のカウントをリセットします。

保守状態ではないノードの定足数投票数をリセットするのは、そのノードを再起動する前である必要があります。そうしないと、定足数の確立を待機中にハングアップすることがあります。

```
phys-schost# clquorum reset
```

reset                    定足数をリセットする変更フラグです。

- 4 保守状態を解除するノードを起動します。
- 5 定足数投票数を確認します。

```
phys-schost# clquorum status
```

保守状態を解除したノードの状態はonlineであり、PresentとPossibleの定足数投票数は適切な値である必要があります。

### 例 9-10 クラスタノードの保守状態を解除して、定足数投票数をリセットする

次に、クラスタノードの定足数投票数をリセットして、その定足数デバイスをデフォルトに戻し、その結果を確認する例を示します。scstat -q の出力では、phys-schost-1 のNode votes は1であり、その状態はonlineです。Quorum Summary では、投票数も増えているはずです。

```
phys-schost-2# clquorum reset
```

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Solaris エントリを選択し Enter キーを押します。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

```
phys-schost-1# clquorum status
```

```
--- Quorum Votes Summary ---
```

Needed	Present	Possible
-----	-----	-----
4	6	6

```
--- Quorum Votes by Node ---
```

```

Node Name Present Possible Status

phys-schost-2 1 1 Online
phys-schost-3 1 1 Online

```

--- Quorum Votes by Device ---

```

Device Name Present Possible Status

/dev/did/rdisk/d3s2 1 1 Online
/dev/did/rdisk/d17s2 0 1 Online
/dev/did/rdisk/d31s2 1 1 Online

```

## ゾーンクラスタ管理タスクの実行

ゾーンクラスタに関するほかの管理作業(ゾーンパスの移動、アプリケーションを実行するためのゾーンクラスタの準備、ゾーンクラスタの複製など)を実行できます。これらのコマンドは、必ずグローバルクラスタの投票ノードから実行してください。

注-グローバルクラスタ内の投票ノードからのみ実行する Sun Cluster コマンドは、ゾーンクラスタには使用できません。各種ゾーンでのコマンドの有効な使用方法については、Sun Cluster の該当するマニュアルページを参照してください。

表 9-2 そのほかのゾーンクラスタ作業

作業	参照先
新規ゾーンパスへのゾーンパスの移動	<code>clzonecluster move -f zonepath zoneclustername</code>
アプリケーション実行用のゾーンクラスタの準備	<code>clzonecluster ready -n nodename zoneclustername</code>
ゾーンクラスタの複製	<code>clzonecluster clone -Z source- zoneclustername [-m copymethod] zoneclustername</code>  clone サブコマンドを使用する前に、複製元のゾーンクラスタを停止してください。複製先のゾーンクラスタは、構成済みであることが必要です。
ゾーンクラスタの削除	290 ページの「ゾーンクラスタを削除する」

表 9-2 そのほかのゾーンクラスタ作業 (続き)

作業	参照先
ゾーンクラスタからファイルシステムを削除	291 ページの「ゾーンクラスタからファイルシステムを削除する」
ゾーンクラスタからストレージデバイスを削除	293 ページの「ゾーンクラスタからストレージデバイスを削除する」
ノードのアンインストールに関するトラブルシューティング	297 ページの「ノードのアンインストールに伴う問題の解決」
Sun Cluster SNMP Event MIB の作成、設定、および管理	298 ページの「Sun Cluster SNMP イベント MIB の作成、設定、および管理」 298 ページの「Sun Cluster SNMP イベント MIB の作成、設定、および管理」

## ▼ ゾーンクラスタを削除する

グローバルクラスタ上に構成されているゾーンクラスタは、特定の 1 つのゾーンクラスタを削除することも、ワイルドカードを使用してすべてのゾーンクラスタを削除することもできます。構成されていないゾーンクラスタは、削除できません。

- 1 グローバルクラスタのノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になりますグローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 2 ゾーンクラスタからすべてのリソースグループとそのリソースを削除します。

```
phys-schost# clresourcegroup delete -F -Z zoneclustername +
```

注-この手順は、グローバルクラスタノードから実行されます。この手順をゾーンクラスタのノードから実行するには、ゾーンクラスタノードにログインし、コマンドの「-Z zonecluster」を省略します。

- 3 ゾーンクラスタを停止します。

```
phys-schost# clzonecluster halt zoneclustername
```

- 4 ゾーンクラスタをアンインストールします。

```
phys-schost# clzonecluster uninstall zoneclustername
```

- 5 ゾーンクラスタを構成解除します。

```
phys-schost# clzonecluster delete zoneclustername
```

## 例 9-11 グローバルクラスタからのゾーンクラスタの削除

```
phys-schost# clresourcegroup delete -F -Z sczone +
```

```
phys-schost# clzonecluster halt sczone
```

```
phys-schost# clzonecluster uninstall sczone
```

```
phys-schost# clzonecluster delete sczone
```

## ▼ ゾーンクラスタからファイルシステムを削除する

この手順は、ゾーンクラスタからファイルシステムを削除する場合に実行します。ゾーンクラスタでサポートされるファイルシステムには、UFS、Vxfs、スタンドアロンQFS、ZFS(データセットとしてエクスポートされたもの)、およびループバックファイルシステムがあります。ファイルシステムをゾーンクラスタに追加する手順は、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の「ゾーンクラスタにファイルシステムを追加する」を参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 ゾーンクラスタをホストするグローバルクラスタのノードで、スーパーユーザーになります。グローバルクラスタのノードから、このプロシージャの一部のステップが実行されます。他のステップは、ゾーンクラスタのノードから実行されます。
- 2 削除するファイルシステムに関連するリソースを削除します。

- a. 削除するゾーンクラスタのファイルシステム用に構成されている Sun Cluster のリソースタイプ (HAStoragePlus、SUNW.ScalMountPoint など) を特定し、削除します。

```
phys-schost# clresource delete -F -Z zoneclustername fs_zone_resources
```

- b. 削除するファイルシステム用のグローバルクラスタ内に構成されている SUNW.qfs タイプの Sun Cluster リソースがあれば、そのリソースを特定し、削除します。

```
phys-schost# clresource delete -F fs_global_resources
```

-F オプションを指定すると、前もって無効にしているリソースも含め、指定したリソースがすべて強制的に削除されるため、このオプションは注意して使用してください。すべての指定リソースが、ほかのリソースのリソース関係設定から削除されるため、クラスタ内のサービスが失われることがあります。削除されて

いない依存リソースは、無効な状態やエラー状態になる可能性があります。詳細は、`clresource(1CL)`のマニュアルページを参照してください。

ヒント-削除したリソースのリソースグループがあとで空になると、そのリソースグループを安全に削除できます。

- 3 ファイルシステムのマウントポイントディレクトリのパスを調べます。たとえば、次のように使用します。

```
phys-schost# clzonecluster configure zoneclustername
```

- 4 ファイルシステムをゾーンクラスタの構成から削除します。

```
phys-schost# clzonecluster configure zoneclustername
```

```
clzc:zoneclustername> remove fs dir=filesystemdirectory
```

```
clzc:zoneclustername> commit
```

ファイルシステムのマウントポイントは、`dir=` で指定します。

- 5 ファイルシステムが削除されたことを確認します。

```
phys-schost# clzonecluster show -v zoneclustername
```

## 例 9-12 ゾーンクラスタ内の高可用性ファイルシステムの削除

この例は、`sczone` というゾーンクラスタ内に構成された、マウントポイントディレクトリ (`/local/ufs-1`) のあるファイルシステムを削除する方法を示しています。リソースは `hasp-rs` で、そのタイプは `HASStoragePlus` です。

```
phys-schost# clzonecluster show -v sczone
```

```
...
```

Resource Name:	fs
dir:	/local/ufs-1
special:	/dev/md/ds1/dsk/d0
raw:	/dev/md/ds1/rdisk/d0
type:	ufs
options:	[logging]

```
...
```

```
phys-schost# clresource delete -F -Z sczone hasp-rs
```

```
phys-schost# clzonecluster configure sczone
```

```
clzc:sczone> remove fs dir=/local/ufs-1
```

```
clzc:sczone> commit
```

```
phys-schost# clzonecluster show -v sczone
```

### 例 9-13 ゾーンクラスタ内の高可用性 ZFS ファイルシステムの削除

この例は、HAzpool という ZFS プール内の ZFS ファイルシステムを削除する方法を示しています。このファイルシステムは、sczone ゾーンクラスタ内に構成されており、リソースは hasp-rs で、そのタイプは SUNW.HAStoragePlus です。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name: dataset
 name: HAzpool
...
phys-schost# clresource delete -F -Z sczone hasp-rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove dataset name=HAzpool
clzc:sczone> commit
phys-schost# clzonecluster show -v sczone
```

## ▼ ゾーンクラスタからストレージデバイスを削除する

ストレージデバイス (SVM ディスクセット、DID デバイスなど) をゾーンクラスタから削除することが可能です。この手順は、ゾーンクラスタからストレージデバイスを削除する場合に実行します。

- 1 ゾーンクラスタをホストするグローバルクラスタのノードで、スーパーユーザーになります。グローバルクラスタのノードから、このプロシーチャーの一部のステップが実行されます。ほかのステップは、ゾーンクラスタのノードから実行することが可能です。
- 2 削除するデバイスに関連するリソースを削除します。削除するゾーンクラスタのデバイス用に構成されている **Sun Cluster** のリソースタイプ (**SUNW.HAStoragePlus**、**SUNW.ScalDeviceGroup** など) を特定し、削除します。

```
phys-schost# clresource delete -F -Z zoneclustername dev_zone_resources
```

- 3 削除するデバイスに対して一致するエントリを調べます。

```
phys-schost# clzonecluster show -v zoneclustername
...
Resource Name: device
 match: <device_match>
...

```

- 4 デバイスをゾーンクラスタの構成から削除します。

```
phys-schost# clzonecluster configure zoneclustername
clzc:zoneclustername> remove device match=<devices_match>
```

```
clzc:zoneclustername> commit
clzc:zoneclustername> end
```

- 5 ゾーンクラスタを再起動します。

```
phys-schost# clzonecluster reboot zoneclustername
```

- 6 デバイスの削除を確認します。

```
phys-schost# clzonecluster show -v zoneclustername
```

#### 例 9-14 SVM ディスクセットをゾーンクラスタから削除する

この例は、`sczone` というゾーンクラスタに構成された `apachedg` という SVM ディスクセットを削除する方法を示しています。`apachedg` ディスクセットのセット番号は 3 です。このデバイスは、クラスタに構成された `zc_rs` のリソースにより使用されます。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name: device
match: /dev/md/apachedg/*dsk/*
Resource Name: device
match: /dev/md/shared/3/*dsk/*
...
phys-schost# clresource delete -F -Z sczone zc_rs

phys-schost# ls -l /dev/md/apachedg
lrwxrwxrwx 1 root root 8 Jul 22 23:11 /dev/md/apachedg -> shared/3
phys-schost# clzonecluster configure sczone
clzc:sczone> remove device match=/dev/md/apachedg/*dsk/*
clzc:sczone> remove device match=/dev/md/shared/3/*dsk/*
clzc:sczone> commit
clzc:sczone> end
phys-schost# clzonecluster reboot sczone
phys-schost# clzonecluster show -v sczone
```

#### 例 9-15 DID デバイスをゾーンクラスタから削除する

この例は、DID デバイス `d10` および `d11` を削除する方法を示しています。このデバイスは、`sczone` というゾーンクラスタに構成されています。このデバイスは、クラスタに構成された `zc_rs` のリソースにより使用されます。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name: device
match: /dev/did/*dsk/d10*
Resource Name: device
```

```
match: /dev/did/*dsk/d11*
...
phys-schost# clresource delete -F -Z sczone zc_rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove device match=/dev/did/*dsk/d10*
clzc:sczone> remove device match=/dev/did/*dsk/d11*
clzc:sczone> commit
clzc:sczone> end
phys-schost# clzonecluster reboot sczone
phys-schost# clzonecluster show -v sczone
```

## ▼ Sun Cluster ソフトウェアをクラスタノードからアンインストールする

完全に確立されたクラスタ構成からグローバルクラスタノードを切り離す前に、そのノードから Sun Cluster ソフトウェアをアンインストールするには、この手順を使用します。この手順では、クラスタに存在する最後のノードからソフトウェアをアンインストールできます。

---

注-クラスタにまだ結合されていない、あるいはまだインストールモードであるノードから Sun Cluster ソフトウェアをアンインストールする場合、この手順を使用してはいけません。その代わりに、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の「Sun Cluster ソフトウェアをアンインストールしてインストール問題を解決する」に進みます。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#)を参照してください。

- 1 クラスタノードを削除するための前提作業(作業マップを参照)がすべて完了していることを確認します。  
[表 8-2](#)を参照してください。

---

注-この手順を続ける前に、`clnode remove`を使用してクラスタ構成からノードを削除します。

---

- 2 アンインストールするグローバルクラスタノード以外の、アクティブなグローバルクラスタメンバー上で、スーパーユーザーになります。この手順は、グローバルクラスタノードから実行します。

- 3 アクティブなクラスタメンバーから、アンインストールを行うノードをクラスタのノード認証リストに追加します。

```
phys-schost# claccess allow -h hostname
```

-h                      ノードの認証リストに追加するノードの名前を指定します。

または、`clsetup(1CL)` ユーティリティを使用できます。手順については、[256 ページの「ノードを認証ノードリストに追加する」](#)を参照してください。

- 4 アンインストールするノードでスーパーユーザーになります。

- 5 ゾーンクラスタがある場合は、アンインストールします。

```
phys-schost# clzonecluster uninstall -F zoneclustername
```

具体的な手順は、[290 ページの「ゾーンクラスタを削除する」](#)を参照してください。

- 6 ノードにグローバルデバイス名前空間用の専用パーティションがある場合、グローバルクラスタノードを非クラスタモードで再起動します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
shutdown -g0 -y -i0ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
shutdown -g0 -y -i0
```

```
...
```

```
<<< Current Boot Parameters >>>
```

```
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@7,1/
sd@0,0:a
```

```
Boot args:
```

```
Type b [file-name] [boot-flags] <ENTER> to boot with options
or i <ENTER> to enter boot interpreter
or <ENTER> to boot with defaults
```

```
<<< timeout in 5 seconds >>>
```

```
Select (b)oot or (i)nterpreter: b -x
```

- 7 /etc/vfstab ファイルから、グローバルにマウントされるすべてのファイルシステムエントリを削除します。ただし、/global/.devices グローバルマウントを除きません。

- 8 このノード上で **Sun Cluster** ソフトウェアを再インストールする場合は、**Sun Java Enterprise System (Java ES)** 製品のレジストリから **Sun Cluster** のエントリを削除します。Java ES 製品のレジストリに Sun Cluster ソフトウェアがインストールされたという記録が含まれていると、Java ES のインストーラは Sun Cluster のコンポーネントを灰色で表示し、再インストールを許可しません。
  - a. **Java ES** のアンインストーラを起動します。

次のコマンドを実行します。ver は Sun Cluster ソフトウェアのインストール元である Java ES ディストリビューションのバージョンです。

```
/var/sadm/prod/SUNWentsysver/uninstall
```
  - b. プロンプトに従い、アンインストールする Sun Cluster を選択します。

uninstall コマンドの詳細については、『Sun Java Enterprise System 5 Update 1 Installation Guide for UNIX』の第 8 章、「Uninstalling」を参照してください。
- 9 このクラスタ上で **Sun Cluster** ソフトウェアを再インストールしない場合は、ほかのクラスタデバイスからトランスポートケーブルとトランスポートスイッチを切断します (存在する場合)。
  - a. アンインストールしたノードが、並列 **SCSI** インタフェースを使用する記憶装置デバイスに接続されている場合は、トランスポートケーブルを切り離れた後で、この記憶装置デバイスのオープン **SCSI** コネクタに **SCSI** ターミネータを取り付ける必要があります。

アンインストールしたノードが、Fibre Channel インタフェースを使用する記憶装置デバイスに接続されている場合は、終端処理は必要ありません。
  - b. 切り離し手順については、ホストアダプタやサーバーに添付されているマニュアルを参照してください。

---

ヒント-ループバックファイルインタフェース (lofi) デバイスを使用する場合、Java ES アンインストーラが自動的に `/globaldevices` という lofi ファイルを削除します。グローバルデバイス名前空間の lofi への移行に関する詳細は、[133 ページの「グローバルデバイス名前空間を移行する」](#)を参照してください。

---

## ノードのアンインストールに伴う問題の解決

ここでは、`scinstall -r` コマンドを実行したときに出力される可能性があるエラーメッセージとその対処方法について説明します。

## 削除されないクラスタファイルシステムのエントリ

次のエラーメッセージは、削除したグローバルクラスタノードに、`vfstab` ファイルから参照されているクラスタファイルシステムがまだあることを示しています。

```
Verifying that no unexpected global mounts remain in /etc/vfstab ... failed
scinstall: global-mount1 is still configured as a global mount.
scinstall: global-mount1 is still configured as a global mount.
scinstall: /global/dg1 is still configured as a global mount.

scinstall: It is not safe to uninstall with these outstanding errors.
scinstall: Refer to the documentation for complete uninstall instructions.
scinstall: Uninstall failed.
```

このエラーを修正するためには、295 ページの「[Sun Cluster ソフトウェアをクラスタノードからアンインストールする](#)」に戻って、その手順を繰り返す必要があります。手順 7 コマンドを再度実行する前に、この Step 7 が正しく行われているか確認してください。

## デバイスグループに削除されていないリストがある場合

次のエラーメッセージは、削除したノードが依然としてデバイスグループにリストされていることを示しています。

```
Verifying that no device services still reference this node ... failed
scinstall: This node is still configured to host device service "
service".
scinstall: This node is still configured to host device service "
service2".
scinstall: This node is still configured to host device service "
service3".
scinstall: This node is still configured to host device service "
dg1".

scinstall: It is not safe to uninstall with these outstanding errors.
scinstall: Refer to the documentation for complete uninstall instructions.
scinstall: Uninstall failed.
```

## Sun Cluster SNMP イベント MIB の作成、設定、および管理

この節では、Simple Network Management Protocol (簡易ネットワーク管理プロトコル、SNMP) イベント Management Information Base (管理情報ベース、MIB) を作成、設

定、および管理する方法を説明します。またこのセクションでは、Sun Cluster SNMP イベント MIB を有効化、無効化、および変更する方法も説明します。

Sun Cluster ソフトウェアでは現在、イベント MIB という MIB を 1 つサポートしています。SNMP マネージャーソフトウェアがクラスタイベントをリアルタイムでトラップします。有効な場合、SNMP マネージャーはトラップ通知を `clsnmphost` コマンドによって定義されているすべてのホストに自動的に送信します。MIB には、最新の 50 イベントの読み取り専用のテーブルが保持されます。クラスタは多数の通知を生成するので、重要度が `warning` 以上のイベントだけがトラップ通知として送信されます。この情報は、レポートが実行されると消失します。

SNMP イベント MIB は、`sun-cluster-event-mib.mib` ファイルで定義されており、`/usr/cluster/lib/mib` ディレクトリにあります。この定義を使用して、SNMP トラップ情報を解釈できます。

イベント SNMP モジュールのデフォルトのポート番号は 11161 で、SNMP トラップのデフォルトのポートは 11162 です。これらのポート番号は、共通エージェントコンテナのプロパティファイル (`/etc/cacao/instances/default/private/cacao.properties`) を変更することによって変更できます。

Sun Cluster SNMP イベント MIB の作成、設定、および管理には次の作業が含まれます。

表 9-3 作業マップ: Sun Cluster SNMP イベント MIB の作成、設定、および管理

作業	参照先
SNMP イベント MIB の有効化	299 ページの「SNMP イベント MIB を有効にする」
SNMP イベント MIB の無効化	300 ページの「SNMP イベント MIB を無効にする」
SNMP イベント MIB の変更	300 ページの「SNMP イベント MIB を変更する」
MIB のトラップ通知を受信するホストリストへの SNMP ホストの追加	301 ページの「SNMP ホストがノード上の SNMP トラップを受信できるようにする」
SNMP ホストの削除	302 ページの「SNMP ホストがノード上の SNMP トラップを受信できないようにする」
SNMP ユーザーの追加	303 ページの「SNMP ユーザーをノードに追加する」
SNMP ユーザーの削除	304 ページの「SNMP ユーザーをノードから削除する」

## ▼ SNMP イベント MIB を有効にする

この手順では、SNMP イベント MIB を有効化する方法を説明します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 **SNMP イベント MIB** を有効にします。

```
phys-schost-1# clsnmpmib enable [-n node] MIB
```

`[-n node]` 有効にするイベント MIB がある `node` を指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`MIB` 有効にする MIB の名前を指定します。この場合、MIB 名は `event` にしてください。

### ▼ **SNMP イベント MIB** を無効にする

この手順では、SNMP イベント MIB を無効化にする方法を説明します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 **SNMP イベント MIB** を無効にします。

```
phys-schost-1# clsnmpmib disable -n node MIB
```

`-n node` 無効にするイベント MIB がある `node` を指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`MIB` 無効にする MIB の種類を指定します。この場合、`event` を指定してください。

### ▼ **SNMP イベント MIB** を変更する

この手順では、SNMP イベント MIB のプロトコルを変更する方法を説明します。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 スーパーユーザーになるか、RBAC 承認 solaris.cluster.modify を提供する役割になります。
- 2 **SNMP イベント MIB** のプロトコルを変更します。

```
phys-schost-1# clsnmpmib set -n node -p version=value MIB
```

-n *node*

変更するイベント MIB がある *node* を指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

-p *version=value*

MIB で使用する SNMP プロトコルのバージョンを指定します。*value* は次のように指定します。

- version=SNMPv2
- version=snmpv2
- version=2
- version=SNMPv3
- version=snmpv3
- version=3

*MIB*

サブコマンドが適用される単数または複数の MIB の名前を指定します。この場合、*event* を指定してください。このオペランドを指定しない場合は、デフォルトのプラス記号 (+) がサブコマンドで使用されます (すべての MIB を意味します)。MIB オペランドを使用する場合は、ほかのコマンドラインオプションの後ろに、スペース区切りリストで MIB を指定します。

## ▼ SNMP ホストがノード上の SNMP トラップを受信できるようにする

この手順では、ノード上の SNMP ホストを、MIB のトラップ通知を受信するホストのリストに追加する方法を説明します。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 ホストを、別のノード上のコミュニティの SNMP ホストリストに追加します。

```
phys-schost-1# cslsnmphost add -c SNMPcommunity [-n node] host
```

`-c SNMPcommunity`

ホスト名とともに使用される SNMP コミュニティ名を指定します。

ホストを `public` 以外のコミュニティに追加する場合は、コミュニティ名 `SNMPcommunity` を指定してください。add サブコマンドを `-c` オプションなしで使用すると、このサブコマンドは `public` をデフォルトのコミュニティ名として使用します。

指定されたコミュニティ名が存在しない場合、このコマンドはそのコミュニティを作成します。

`-n node`

クラスタ内の SNMP MIB に対するアクセス権を付与されている SNMP ホストの `node` の名前を指定します。ノード名またはノード ID を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`host`

クラスタ内の SNMP MIB に対するアクセス権が付与されたホストの名前、IP アドレス、または IPv6 アドレスを指定します。

## ▼ SNMP ホストがノード上の SNMP トラップを受信できないようにする

この手順では、ノード上の SNMP ホストを、MIB のトラップ通知を受信するホストのリストから削除する方法を説明します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。

- 2 指定のノード上のコミュニティの **SNMP** ホストリストからホストを削除します。

```
phys-schost-1# clnmphost remove -c SNMPcommunity -n node host
```

**remove**

指定のノードから指定の **SNMP** ホストを削除します。

**-c SNMPcommunity**

**SNMP** ホストを削除する **SNMP** コミュニティの名前を指定します。

**-n node**

構成から削除される **SNMP** ホストの *node* の名前を指定します。ノード名またはノード ID を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

**host**

構成から削除されるホストの名前、IP アドレス、または IPv6 アドレスを指定します。

指定の **SNMP** コミュニティ内のすべてのホストを削除するには、**-c** オプション付きの *host* に正符号 (+) を使用します。すべてのホストを削除するには、*host* に正符号 + を使用します。

## ▼ **SNMP** ユーザーをノードに追加する

この手順では、ノード上の **SNMP** ユーザー構成に **SNMP** ユーザーを追加する方法を説明します。

**phys-schost#** プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 スーパーユーザーになるか、**RBAC** 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 **SNMP** ユーザーを追加します。

```
phys-schost-1# clnmpuser create -n node -a authentication \
-f password user
```

**-n node**

**SNMP** ユーザーが追加されるノードを指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

**-a authentication**

ユーザーの承認に使用する認証プロトコルを指定します。認証プロトコルの値は **SHA** または **MD5** です。

*-f password* SNMP ユーザーパスワードを含むファイルを指定します。新しいユーザーを作成する際にこのオプションを指定しないと、コマンドはパスワードを求めるプロンプトを表示します。このオプションは、`add` サブコマンドとだけ有効です。

ユーザーパスワードは、次の形式で、独立した行の上に指定します。

*user:password*

パスワードには次に示す文字または空白文字を含めることはできません。

- ; (セミコロン)
- : (コロン)
- \ (バックスラッシュ)
- \n (復帰改行)

*user* 追加する SNMP ユーザーの名前を指定します。

## ▼ SNMP ユーザーをノードから削除する

この手順では、ノード上の SNMP ユーザー構成から SNMP ユーザーを削除する方法を説明します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 スーパーユーザーになるか、**RBAC 承認** `solaris.cluster.modify` を提供する役割になります。
- 2 **SNMP ユーザーを削除**します。

`phys-schost-1# clsnmpuser delete -n node user`

*-n node* SNMP ユーザーが削除されるノードを指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

*user* 削除する SNMP ユーザーの名前を指定します。

## 障害追跡

このセクションでは、テスト用に使用できる障害追跡手順について説明します。

### ▼ 非クラスタモードで起動したノードから **Solaris Volume Manager** メタセットを取得する

この手順を使用して、テスト用にグローバルクラスタ外でアプリケーションを実行します。

- 1 **Solaris Volume Manager** メタセットで定足数デバイスが使用されているかどうかを確認し、定足数デバイスが **SCSI2** または **SCSI3** 予約を使用するかどうかを確認します。

```
phys-schost# clquorum show
```

- a. 定足数デバイスが **Solaris Volume Manager** メタセットにある場合は、あとで非クラスタモードにするメタセットには含まれない、新しい定足数デバイスを追加します。

```
phys-schost# clquorum add did
```

- b. 古い定足数デバイスを削除します。

```
phys-schost# clquorum remove did
```

- c. 定足数デバイスが **SCSI2** 予約を使用する場合は、古い定足数からの **SCSI2** 予約をスクラブして、**SCSI2** 予約が残らないことを確認します。

```
phys-schost# /usr/cluster/lib/sc/pgre -c pgre_scrub -d /dev/did/rdisk/dids2
```

```
phys-schost# /usr/cluster/lib/sc/pgre -c pgre_inkeys -d /dev/did/rdisk/dids2
```

- 2 非クラスタモードで起動するグローバルクラスタノードを退避します。

```
phys-schost# clresourcegroup evacuate -n targetnode
```

- 3 **HASStorage** または **HASStoragePlus** リソースを含み、あとで非クラスタモードにするメタセットの影響を受けるデバイスまたはファイルシステムを含む、1つまたは複数のリソースグループをオフラインにします。

```
phys-schost# clresourcegroup offline resourcegroupname
```

- 4 オフラインにしたリソースグループ内のすべてのリソースを無効にします。

```
phys-schost# clresource disable resourcename
```

- 5 リソースグループを非管理状態に切り替えます。

```
phys-schost# clresourcegroup unmanage resourcegroupname
```

- 6 対応する1つまたは複数のデバイスグループをオフラインにします。  
`phys-schost# cldevicegroup offline devicegroupname`
- 7 1つまたは複数のデバイスグループを無効にします。  
`phys-schost# cldevicegroup disable devicegroupname`
- 8 パッシブノードを非クラスタモードで起動します。  
`phys-schost# reboot -x`
- 9 続ける前にパッシブノードで起動プロセスが完了していることを確認します。
  - Solaris 9  
ログインプロンプトは起動プロセスが完了したあとにのみ表示されるため、操作は不要です。
  - Solaris 10  
`phys-schost# svcs -x`
- 10 メタセット内のディスクにSCSI3予約があるかどうかを調べます。メタセット内のすべてのディスクで次のコマンドを実行します。  
`phys-schost# /usr/cluster/lib/sc/scsi -c inkeys -d /dev/did/rdisk/dids2`
- 11 ディスク上にSCSI3予約がある場合は、それらをスクラブします。  
`phys-schost# /usr/cluster/lib/sc/scsi -c scrub -d /dev/did/rdisk/dids2`
- 12 メタセットを退避したノードに移します。  
`phys-schost# metaset -s name -C take -f`
- 13 メタセット上で定義されたデバイスを含む1つまたは複数のファイルシステムをマウントします。  
`phys-schost# mount device mountpoint`
- 14 アプリケーションを起動して、必要なテストを実行します。テストが終了したら、アプリケーションを停止します。
- 15 ノードを再起動し、起動プロセスが終了するまで待ちます。  
`phys-schost# reboot`
- 16 1つまたは複数のデバイスグループをオンラインにします。  
`phys-schost# cldevicegroup online -e devicegroupname`

- 17 1つまたは複数のリソースグループを起動します。

```
phys-schost# clresourcegroup online -eM resourcegroupname
```



## CPU 使用率の制御の構成

---

CPU の使用率を制御したい場合は、CPU 制御機能を構成します。CPU 制御機能の構成についての詳細は、[rg\\_properties\(5\)](#) のマニュアルページを参照してください。この章では、次のトピックについて説明します。

- 309 ページの「CPU 制御の概要」
- 311 ページの「CPU 制御の構成」

### CPU 制御の概要

Sun Cluster ソフトウェアでは CPU の使用量を制御できます。Solaris 9 OS で可能な構成の選択肢は、Solaris 10 OS で可能な選択肢とは同じではありません。

CPU 制御機能は、Solaris OS で利用可能な機能に基づいて構築されています。ゾーン、プロジェクト、リソースプール、プロセッサセット、およびスケジューリングクラスについては、『[Solaris のシステム管理 \(Solaris コンテナ: 資源管理と Solaris ゾーン\)](#)』を参照してください。

SPARC: Solaris 9 OS では、CPU シェアをリソースグループに割り当てることができます。

Solaris 10 OS では、次の作業を実行できます。

- CPU シェアをリソースグループに割り当てる。
- プロセッサをリソースグループに割り当てる。

---

注 - この章のすべての手順は、Solaris 9 OS に固有であると明記していないかぎり、Solaris 10 OS で使用するためのものです。

---

## シナリオの選択

構成の選択肢と、選択するオペレーティングシステムのバージョンに応じて、さまざまなレベルのCPU制御を行うことができます。この章で説明するCPU制御のすべての局面は、リソースグループプロパティ `RG_SLM_TYPE` が `automated` に設定されていることに依存します。

表 10-1 で、使用可能なさまざまな構成シナリオを説明します。

表 10-1 CPU制御のシナリオ

説明	参照先
<p>SPARC:リソースグループが Solaris 9 OS 上で動作する</p> <p>CPU シェアをリソースグループに割り当て、<code>project.cpu-shares</code> の値を提供する</p>	<p>311 ページの「SPARC: Solaris 9 OS で CPU 使用率を制御する」</p>
<p>リソースグループが Solaris 10 OS 上のグローバルクラスタの投票ノードで動作する</p> <p>CPU シェアをリソースグループとゾーンに割り当て、<code>project.cpu-shares</code> および <code>zone.cpu-shares</code> の値を提供する</p> <p>この手順は、グローバルクラスタの非投票ノードが構成されているかどうかに関係なく実行できます。</p>	<p>313 ページの「グローバルクラスタの投票ノードで CPU 使用率を制御する」</p>
<p>リソースグループはデフォルトのプロセッサセットを使用することによりグローバルクラスタの非投票ノードで動作する</p> <p>CPU シェアをリソースグループとゾーンに割り当て、<code>project.cpu-shares</code> および <code>zone.cpu-shares</code> の値を提供する</p> <p>この手順は、プロセッサセットのサイズを制御する必要がない場合に実行します。</p>	<p>315 ページの「デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する」</p>

表 10-1 CPU 制御のシナリオ (続き)

説明	参照先
<p>リソースグループは専用のプロセッサセットを使用してグローバルクラスタの非投票ノードで動作する</p> <p>CPU シェアをリソースグループに割り当て、<code>project.cpu-shares</code>、<code>zone.cpu-shares</code> の値、および専用のプロセッサセット内のプロセッサの最大数を提供する</p> <p>専用のプロセッサセット内のプロセッサセットの最小数を設定します。</p> <p>CPU シェアと、プロセッサセットのサイズを制御したい場合に、この手順を実行します。この制御は、専用のプロセッサセットを使用することにより、グローバルクラスタの非投票ノードでのみ実行できます。</p>	<p>318 ページの「専用のプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する」</p>

## 公平配分スケジューラ

CPU シェアをリソースグループに割り当てる手順の最初のステップは、システムのスケジューラを公平配分スケジューラ (FSS) に設定することです。デフォルトでは、Solaris OS のスケジューリングクラスはタイムシェアスケジューラ (TS) です。スケジューラを FSS に設定し、シェア構成を有効にします。

選択するスケジューラクラスに関係なく、専用のプロセッサセットを作成できません。

## CPU 制御の構成

このセクションでは次の作業について説明します。

- 311 ページの「SPARC: Solaris 9 OS で CPU 使用率を制御する」
- 313 ページの「グローバルクラスタの投票ノードで CPU 使用率を制御する」
- 315 ページの「デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する」
- 318 ページの「専用のプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する」

### ▼ SPARC: Solaris 9 OS で CPU 使用率を制御する

Solaris 9 OS を実行するクラスタ上のリソースグループに CPU シェアを割り当てるには、この手順を実行します。

リソースグループにCPUシェアが割り当てられている場合、Sun Cluster ソフトウェアは、リソースグループの1つのリソースを起動する際に、次の作業を実行します。

- `SCSLM_resource_group_name` という名前のプロジェクトを作成する (そのプロジェクトがまだ存在しない場合)。このプロジェクトはリソースグループに固有で、指定された数のCPUシェア (`project.cpu_shares`) が割り当てられます。
- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

CPU 制御機能の構成についての詳細は、[rg\\_properties\(5\)](#) のマニュアルページを参照してください。

## 1 システムのスケジューラを、公平配分スケジューラ (FSS) に設定します。

```
dispadmin -d FSS
```

次の再起動時に、FSS がデフォルトのスケジューラになります。この構成をすぐに有効にするには、`priocntl` コマンドを使用します。

```
priocntl -s -c FSS
```

`priocntl` コマンドと `dispadmin` コマンドを組み合わせることで、FSS がすぐにデフォルトのスケジューラになり、再起動後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [priocntl\(1\)](#) のマニュアルページを参照してください。

---

注 - FSS がデフォルトのスケジューラでない場合、CPU シェアの割り当ては有効になりません。

---

## 2 CPU 制御機能を構成します。

```
clresourcegroup create -p RG_SLM_TYPE=automated \
[-p RG_SLM_CPU_SHARES=value] resource_group_name
```

`-p RG_SLM_TYPE=automated` CPU の使用状況を管理できるようにし、システム資源管理用に Solaris OS を設定する手順の一部を自動化します。

`-p RG_SLM_CPU_SHARES=value` リソースグループ固有のプロジェクト `project.cpu-shares` に割り当てられた CPU シェアの数を指定します。

`resource_group_name` リソースグループの名前を指定します。

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

### 3 構成の変更を有効にします。

```
clresourcegroup online -M resource_group_name
```

`resource_group_name` リソースグループの名前を指定します。

---

注-SCSLM `resource_group_name` プロジェクトは削除または変更しないでください。手動で、たとえば `project.max-lwps` プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

## ▼ グローバルクラスタの投票ノードで CPU 使用率を制御する

グローバルクラスタの投票ノードで実行されるリソースグループに CPU シェアを割り当てるには、この手順を実行します。

リソースグループに CPU シェアが割り当てられている場合、Sun Cluster ソフトウェアは、グローバルクラスタの投票ノードでリソースグループの 1 つのリソースを起動する際に、次の作業を実行します。

- 投票ノードに割り当てられている CPU シェア (`zone.cpu-shares`) の数を、指定された CPU シェアの数だけ増やします (まだ行われていない場合)。
- 投票ノードに `SCSLM_resourcegroup_name` という名前のプロジェクトを作成します (まだ行われていない場合)。このプロジェクトはリソースグループに固有で、指定された数の CPU シェア (`project.cpu-shares`) が割り当てられています。
- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

CPU 制御機能の構成についての詳細は、[rg\\_properties\(5\)](#) のマニュアルページを参照してください。

### 1 システムのデフォルトのスケジューラを、公平配分スケジューラ (FSS) に設定します。

```
dispadmin -d FSS
```

次の再起動時に、FSS がデフォルトのスケジューラになります。この構成をすぐに有効にするには、`priocntl` コマンドを使用します。

```
priocntl -s -C FSS
```

`priocntl` コマンドと `dispadmin` コマンドを組み合わせることで、FSS がすぐにデフォルトのスケジューラになり、再起動後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [priocntl\(1\)](#) のマニュアルページを参照してください。

注 - FSS がデフォルトのスケジューラでない場合、CPU シェアの割り当ては有効になりません。

- 2 各ノードで CPU 制御を使用するため、グローバルクラスタの投票ノードに対するシェア数と、デフォルトのプロセッサセットで使用可能な CPU の最小数を構成します。

これらのパラメータを設定することで、投票ノードで動作中のプロセスを、非投票ノードで動作中のプロセスと CPU を獲得する際に発生する競合から保護します。globalzoneshares および defaultpsetmin プロパティに値を割り当てないと、これらのプロパティはデフォルト値をとります。

```
clnode set [-p globalzoneshares=integer] \
[-p defaultpsetmin=integer] \
node
```

-p defaultpsetmin=*defaultpsetmininteger* デフォルトのプロセッサセットで利用可能な CPU シェアの最小数を設定します。デフォルト値は 1 です。

-p globalzoneshares=*integer* 投票ノードに割り当てられるシェアの数を設定します。デフォルト値は 1 です。

node プロパティを設定するノードを指定します。

これらのプロパティを設定する際には、投票ノードのプロパティを設定しています。これらのプロパティを設定しないと、非投票ノードで RG\_SLM\_PSET\_TYPE プロパティの恩恵を受けることができません。

- 3 これらのプロパティを正しく設定したことを確認します。

```
clnode show node
```

指定するノードに対して、clnode コマンドは、設定されているプロパティ、およびこれらのプロパティに設定されている値を出力します。clnode を使用して CPU 制御プロパティを設定しないと、これらはデフォルト値をとります。

- 4 CPU 制御機能を構成します。

```
clresourcegroup create -p RG_SLM_TYPE=automated \
[-p RG_SLM_CPU_SHARES=value] resource_group_name
```

-p RG\_SLM\_TYPE=automated CPU の使用状況を管理できるようにし、システム資源管理用に Solaris OS を設定する手順の一部を自動化します。

-p RG\_SLM\_CPU\_SHARES=*value* リソースグループ固有のプロジェクトに割り当てられる CPU シェアの数 (project.cpu-shares) を指定し、投

票ノードに割り当てられる CPU シェアの数  
(`zone.cpu-shares`) を決定します。

`resource_group_name` リソースグループの名前を指定します。

この手順では、`RG_SLM_PSET_TYPE` プロパティは設定しません。投票ノードでは、このプロパティは値 `default` をとります。

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

## 5 構成の変更を有効にします。

```
clresourcegroup online -M resource_group_name
```

`resource_group_name` リソースグループの名前を指定します。

---

注 - `SCSLM_resource_group_name` プロジェクトは削除または変更しないでください。手動で、たとえば `project.max-lwps` プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

## ▼ デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する

グローバルクラスタの非投票ノードのリソースグループに対して CPU シェアを割り当てるが、専用のプロセッサセットを作成する必要がない場合は、この手順を実行します。

リソースグループに CPU シェアが割り当てられている場合、Sun Cluster ソフトウェアは、非投票ノードのリソースグループの 1 つのリソースを起動する際に、次の作業を実行します。

- `SCSLM_resource_group_name` という名前のプールを作成します (まだ行われていない場合)。
- `SCSLM_pool_zone_name` プールを、デフォルトのプロセッサセットに関連付けます。
- 非投票ノードを `SCSLM_pool_zone_name` プールに動的にバインドします。
- 非投票ノードに割り当てられている CPU シェア (`zone.cpu-shares`) の数を、指定された CPU シェアの数だけ増やします (まだ行われていない場合)。

- 非投票ノードに `SCSLM_resourcegroup_name` という名前のプロジェクトを作成します(まだ行われていない場合)。このプロジェクトはリソースグループに固有で、指定された数のCPUシェア (`project.cpu_shares`) が割り当てられます。
- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

CPU制御機能の構成についての詳細は、[rg\\_properties\(5\)](#)のマニュアルページを参照してください。

- 1 システムのデフォルトのスケジューラを、公平配分スケジューラ (FSS) に設定します。

```
dispadmin -d FSS
```

次の再起動時に、FSSがデフォルトのスケジューラになります。この構成をすぐに有効にするには、`priocntl` コマンドを使用します。

```
priocntl -s -C FSS
```

`priocntl` コマンドと `dispadmin` コマンドを組み合わせることで、FSSがすぐにデフォルトのスケジューラになり、再起動後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [priocntl\(1\)](#) のマニュアルページを参照してください。

---

注-FSSがデフォルトのスケジューラでない場合、CPUシェアの割り当ては有効になりません。

---

- 2 各ノードでCPU制御を使用するため、グローバルクラスタの投票ノードに対するシェア数と、デフォルトのプロセッサセットで使用可能なCPUの最小数を構成します。

これらのパラメータを設定することで、投票ノードで動作中のプロセスを、グローバルクラスタの非投票ノードで動作中のプロセスとCPUを獲得する際に発生する競合から保護します。`globalzonestshares` および `defaultpsetmin` プロパティに値を割り当てないと、これらのプロパティはデフォルト値をとります。

```
clnode set [-p globalzonestshares=integer] \
[-p defaultpsetmin=integer] \
node
```

```
-p globalzonestshares=integer
```

投票ノードに割り当てられるシェアの数を設定します。デフォルト値は1です。

```
-p defaultpsetmin=defaultpsetmininteger
```

デフォルトのプロセッサセットで利用可能なCPUの最小数を設定します。デフォルト値は1です。

```
node
```

プロパティを設定するノードを指定します。

これらのプロパティを設定する際には、投票ノードのプロパティを設定しています。

- 3 これらのプロパティを正しく設定したことを確認します。

```
clnode show node
```

指定するノードに対して、`clnode` コマンドは、設定されているプロパティ、およびこれらのプロパティに設定されている値を出力します。`clnode` を使用して CPU 制御プロパティを設定しないと、これらはデフォルト値をとります。

- 4 CPU 制御機能を構成します。

```
clresourcegroup create -p RG_SLM_TYPE=automated \
[-p RG_SLM_CPU_SHARES=value] resource_group_name
```

`-p RG_SLM_TYPE=automated` CPU の使用状況を管理できるようにし、システム資源管理用に Solaris OS を設定する手順の一部を自動化します。

`-p RG_SLM_CPU_SHARES=value` リソースグループ固有のプロジェクトに割り当てられる CPU シェアの数 (`project.cpu-shares`) を指定し、グローバルクラスタの非投票ノードに割り当てられる CPU シェアの数 (`zone.cpu-shares`) を決定します。

*resource\_group\_name* リソースグループの名前を指定します。

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

デフォルト以外のプールがゾーン構成内にある場合、またはゾーンがデフォルト以外のプールに動的にバインドされている場合は、非投票ノードで `RG_SLM_TYPE` を `automated` に設定できません。ゾーン構成とプールのバインディングについては、それぞれ [zonecfg\(1M\)](#) と [poolbind\(1M\)](#) のマニュアルページを参照してください。ゾーン構成を次のように表示します。

```
zonecfg -z zone_name info pool
```

---

注 -HASStoragePlus、LogicalHostname などのリソースは、非投票ノードで起動するように構成されていても、`GLOBAL_ZONE` プロパティが `TRUE` に設定されている場合は、投票ノードで起動されます。`RG_SLM_TYPE` プロパティを `automated` に設定していても、このリソースは CPU シェア構成の恩恵を受けることはなく、`RG_SLM_TYPE` が手動に設定されているリソースグループと同様に扱われます。

---

この手順では、`RG_SLM_PSET_TYPE` プロパティは設定しません。Sun Cluster はデフォルトのプロセッサセットを使用します。

## 5 構成の変更を有効にします。

```
clresourcegroup online -M resource_group_name
```

*resource\_group\_name* リソースグループの名前を指定します。

RG\_SLM\_PSET\_TYPE に default を設定すると、Sun Cluster はプール SCSLM\_pool\_zone\_name を作成しますが、プロセッサセットは作成しません。この場合、SCSLM\_pool\_zone\_name はデフォルトのプロセッサセットに関連付けられます。

オンラインリソースグループが、非投票ノード内のCPU制御に対して設定されなくなった場合、非投票ノードのCPUシェア値はゾーン構成内の zone.cpu-shares の値をとります。このパラメータの値はデフォルトで1です。ゾーン構成の詳細は、[zonecfg\(1M\)](#) のマニュアルページを参照してください。

---

注 - SCSLM\_resource\_group\_name プロジェクトは削除または変更しないでください。手動で、たとえば project.max-lwps プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

## ▼ 専用のプロセッサセットを使用してグローバルクラスタの非投票ノードのCPU使用率を制御する

リソースグループを専用のプロセッサセットで実行する場合は、この手順を実行します。

リソースグループが専用のプロセッサセットで実行するよう構成されている場合、Sun Cluster ソフトウェアは、グローバルクラスタ非投票ノードでリソースグループのリソースを起動する際に、次の作業を実行します。

- SCSLM\_pool\_zone\_name という名前のプールを作成します(まだ行われていない場合)。
- 専用のプロセッサセットを作成します。プロセッサセットのサイズは、RG\_SLM\_CPU\_SHARES および RG\_SLM\_PSET\_MIN プロパティを使用することで決定されます。
- SCSLM\_pool\_zone\_name プールを、作成されたプロセッサセットに関連付けます。
- 非投票ノードを SCSLM\_pool\_zone\_name プールに動的にバインドします。
- 非投票ノードに割り当てられているCPUシェアの数を、指定されたCPUシェアの数だけ増やします(まだ行われていない場合)。
- 非投票ノードに SCSLM\_resourcegroup\_name という名前のプロジェクトを作成します(まだ行われていない場合)。このプロジェクトはリソースグループに固有で、指定された数のCPUシェア(project.cpu\_shares)が割り当てられます。

- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

## 1 システムのスケジューラを、公平配分スケジューラ (FSS) に設定します。

```
dispadmin -d FSS
```

次の再起動時に、FSSがデフォルトのスケジューラになります。この構成をすぐに有効にするには、`priocntl` コマンドを使用します。

```
priocntl -s -C FSS
```

`priocntl` コマンドと `dispadmin` コマンドを組み合わせて使用することで、FSSがすぐにデフォルトのスケジューラになり、再起動後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [priocntl\(1\)](#) のマニュアルページを参照してください。

---

注-FSSがデフォルトのスケジューラでない場合、CPUシェアの割り当ては有効になりません。

---

## 2 各ノードでCPU制御を使用するため、グローバルクラスタの投票ノードに対するシェア数と、デフォルトのプロセッサセットで使用可能なCPUの最小数を構成します。

これらのパラメータを設定することで、投票ノードで動作中のプロセスを、非投票ノードで動作中のプロセスとCPUを獲得する際に発生する競合から保護します。`globalzoneshares` および `defaultpsetmin` プロパティに値を割り当てないと、これらのプロパティはデフォルト値をとります。

```
clnode set [-p globalzoneshares=integer] \
[-p defaultpsetmin=integer] \
node
```

`-p defaultpsetmin=defaultpsetmininteger` デフォルトのプロセッサセットで利用可能なCPUの最小数を設定します。デフォルトは1です。

`-p globalzoneshares=integer` 投票ノードに割り当てられるシェアの数を設定します。デフォルトは1です。

*node* プロパティを設定するノードを指定します。

これらのプロパティを設定する際には、投票ノードのプロパティを設定しています。

## 3 これらのプロパティを正しく設定したことを確認します。

```
clnode show node
```

指定するノードに対して、`clnode` コマンドは、設定されているプロパティ、およびこれらのプロパティに設定されている値を出力します。`clnode` を使用して CPU 制御プロパティを設定しないと、これらはデフォルト値をとります。

#### 4 CPU 制御機能を構成します。

```
clresourcegroup create -p RG_SLM_TYPE=automated \
 [-p RG_SLM_CPU_SHARES=value] \
-p -y RG_SLM_PSET_TYPE=value \
[-p RG_SLM_PSET_MIN=value] resource_group_name
```

- p `RG_SLM_TYPE=automated` CPU 制御の使用状況を管理できるようにし、システム資源管理用に Solaris OS を設定する手順の一部を自動化します。
- p `RG_SLM_CPU_SHARES=value` リソースグループ固有のプロジェクトに割り当てられる CPU シェアの数 (`project.cpu-shares`) を指定し、非投票ノードに割り当てられる CPU シェアの数 (`zone.cpu-shares`) とプロセッサセット内のプロセッサの最大数を決定します。
- p `RG_SLM_PSET_TYPE=value` 専用のプロセッサセットの作成を可能にします。専用のプロセッサセットを使用するには、このプロパティを `strong` または `weak` に設定します。値 `strong` と `weak` は相互に排他的です。つまり、同じゾーン内のリソースグループを、`strong` と `weak` が混在するように構成することはできません。
- p `RG_SLM_PSET_MIN=value` プロセッサセット内のプロセッサの最小数を判別します。
- resource\_group\_name* リソースグループの名前を指定します。

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

デフォルト以外のプールがゾーン構成内にある場合、またはゾーンがデフォルト以外のプールに動的にバインドされている場合は、非投票ノードで `RG_SLM_TYPE` を `automated` に設定できません。ゾーン構成とプールのバインディングについては、それぞれ `zonectfg(1M)` と `poolbind(1M)` のマニュアルページを参照してください。ゾーン構成を次のように表示します。

```
zonectfg -z zone_name info pool
```

---

注 - `HASStoragePlus`、`LogicalHostname` などのリソースは、非投票ノードで起動するように構成されていても、`GLOBAL_ZONE` プロパティが `TRUE` に設定されている場合は、投票ノードで起動されます。`RG_SLM_TYPE` プロパティを `automated` に設定していても、このリソースは CPU シェアと専用のプロセッサセット構成の恩恵を受けることはなく、`RG_SLM_TYPE` が手動に設定されているリソースグループと同様に扱われます。

---

## 5 構成の変更を有効にします。

`resource_group_name` リソースグループの名前を指定します。

---

注 - `SCSLM_resource_group_name` プロジェクトは削除または変更しないでください。手動で、たとえば `project.max-lwps` プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

リソースグループがオンラインの間に `RG_SLM_CPU_SHARES` と `RG_SLM_PSET_MIN` に行われた変更は、動的に考慮されます。しかし、`RG_SLM_PSET_TYPE` に `strong` が設定されている場合、および、変更を受け入れるための CPU が十分に存在しない場合、`RG_SLM_PSET_MIN` に要求された変更は適用されません。この場合は、警告メッセージが表示されます。次のスイッチオーバーでは、`RG_SLM_PSET_MIN` に対して構成した値を受け入れる十分な CPU が使用できない場合、不十分な CPU 数によるエラーが発生する可能性があります。

オンラインリソースグループが、非投票ノード内の CPU 制御に対して設定されなくなった場合、非投票ノードの CPU シェア値は `zone.cpu-shares` の値をとります。このパラメータの値はデフォルトで 1 です。



# Sun Cluster ソフトウェアとファームウェアのパッチ

---

この章では、Sun Cluster 構成のパッチの追加および削除手順について説明します。この章で説明する手順は次のとおりです。

- 323 ページの「Sun Cluster へのパッチの適用の概要」
- 325 ページの「Sun Cluster ソフトウェアへのパッチの適用」

## Sun Cluster へのパッチの適用の概要

クラスタの性質上、クラスタを正しく動作させるには、すべてのクラスタメンバーノードが同じパッチレベルにある必要があります。&fmv1 パッチをノードに適用するときは、パッチをインストールする前に、クラスタメンバーシップからノードを一時的に削除するか、全体のクラスタを停止しておく必要があります。この節では、これらの手順について説明します。

Sun Cluster パッチを適用する前に、パッチの README ファイルを確認します。また、ストレージデバイスのアップグレード要件を確認して、必要なパッチ方法を判別します。

---

注 - Sun Cluster パッチを適用する場合は、この章の説明よりも新しい注意事項がないかどうか、README ファイルと SunSolve を参照してください。

---

すべてのクラスタノードにパッチをインストールする作業は、次のいずれかの状況に該当します。

再起動パッチ(ノード)      パッチを提供するには、`boot -sx` または `shutdown -g -y -i0` コマンドを使用して、ノードをシングルユーザーモードで起動してから、クラスタに結合するために再起動します。まず、任意のリソースグループまたはデバイスグループを、パッチを適用するノードから別のクラスタメンバーに切り替え、ノードをオフライン状

態にする必要があります。また、クラスタ全体が停止しないように、パッチまたはファームウェアは1つのクラスタノードに適用します。

たとえ個々のノードが一時的に使用不可能になっても、クラスタ自体はこの種類のパッチアプリケーションの間は使用可能のままです。パッチを適用したノードは、他のノードが同じパッチレベルになくても、メンバーノードとしてクラスタに結合できます。

**Rebooting patch (cluster)** ソフトウェアまたはファームウェアのパッチを適用するには、`boot -sx` または `shutdown -g -y -i0` コマンドを使用して、クラスタを停止して各ノードをシングルユーザーモードで起動する必要があります。次に、ノードを再起動してクラスタに結合します。このようなパッチでは、パッチ適用中にクラスタを使用できません。

**非再起動パッチ** ノードをオフライン状態にする必要はありません(引き続きリソースグループやデバイスグループのマスターとして動作可能)。また、パッチの適用時にノードをオフライン状態に、または再起動する必要もありません。ただし、パッチは一度に1つのノードに適用し、次のノードに適用する前に、パッチが動作することを確認する必要があります。

---

注-パッチの適用によって配下のクラスタプロトコルが変更されることはありません。

---

パッチをクラスタに適用するには `patchadd` コマンドを、パッチを削除するには(可能な場合) `patchrm` コマンドをそれぞれ使用します。

## Sun Cluster パッチの適用に関する注意事項

Sun Cluster パッチをより効率的に適用するために、次の点に注意してください。

- パッチを適用する前に、必ずパッチの README ファイルを参照してください。
- ストレージデバイスのアップグレード要件を確認して、必要なパッチ方法を判別します。
- クラスタを実際の環境で実行する前に、すべてのパッチ(必須および推奨)を適用します。

- ハードウェアのファームウェアレベルを確認し、必要と思われる必須ファームウェアアップデートをインストールします。
- クラスタメンバーとして機能するノードには、すべて同じパッチを適用する必要があります。
- クラスタサブシステムパッチの状態を最新の状態に保ちます。これらのパッチには、たとえば、ボリューム管理、ストレージデバイスのファームウェア、クラスタトランスポートなどが含まれます。
- 定期的に (四半期に一度など) パッチレポートを確認し、推奨パッチを Sun Cluster 構成に適用します。
- ご購入先が推奨するパッチを適用します。
- 主要なパッチを更新したならフェイルオーバーをテストします。クラスタの動作が低下または悪化した場合に備えて、パッチを取り消す準備をしておきます。

## Sun Cluster ソフトウェアへのパッチの適用

表 11-1 作業リスト:クラスタへのパッチの適用

作業	参照先
ノードを停止せずに、非再起動 Sun Cluster パッチを一度に 1 つのノードだけに適用	334 ページの「非再起動 Sun Cluster パッチを適用する」
クラスタメンバーを非クラスタモードにした後で、再起動 Sun Cluster パッチを適用	325 ページの「再起動パッチを適用する (ノード)」 330 ページの「再起動パッチを適用する (クラスタ)」
クラスタにフェイルオーバーノードが含まれる場合、シングルユーザーモードでパッチを適用	335 ページの「フェイルオーバーノード構成をしている場合に、シングルユーザーモードでパッチを適用する」
Sun Cluster パッチを削除	339 ページの「Sun Cluster パッチの変更」

### ▼ 再起動パッチを適用する (ノード)

パッチを一度にクラスタの 1 つのノードだけに適用し、パッチ処理中でもクラスタ自体は動作したままにします。この手順で、パッチを適用する前に、`boot -sx` または `shutdown -g -y -i0` コマンドを使用して、まずクラスタのノードを停止してシングルユーザーモードで起動する必要があります。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 パッチを適用する前に、Sun Cluster 製品の Web サイトで、インストール前後の特別な注意事項があるかどうかを確認してください。

- 2 パッチの適用先であるノード上で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。

- 3 パッチを適用するノードのリソースグループおよびデバイスグループの一覧を表示します。

```
clresourcegroup status -n node
cldevicegroup status -n node
```

- 4 すべてのリソースグループ、リソース、デバイスグループを、パッチを適用するノードから他のクラスタメンバーに切り替えます。

```
clnode evacuate -n node
```

`evacuate` グローバルクラスタのすべての非投票ノードも含め、すべてのデバイスグループとリソースグループを退避させます。

`-n node` リソースグループとデバイスグループを切り替えるノードを指定します。

- 5 ノードを停止します。

```
shutdown -g0 [-y]
[-i0]
```

- 6 ノードを非クラスタのシングルユーザーモードで起動します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -sx
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# shutdown -g -y -i0
```

```
Press any key to continue
```

- a. GRUB メニューで矢印キーを使用して該当する Solaris エントリを選択し、e と入力してコマンドを編集します。

GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
```

```
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+

```

Use the ^ and v keys to select which entry is highlighted.  
Press enter to boot the selected OS, 'e' to edit the  
commands before booting, or 'c' for a command-line.

GRUB ベースの起動についての詳細は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、e を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+

```

Use the ^ and v keys to select which entry is highlighted.  
Press 'b' to boot, 'e' to edit the selected command in the  
boot sequence, 'c' for a command-line, 'o' to open a new line  
after ('O' for before) the selected line, 'd' to remove the  
selected line, or escape to go back to the main menu.

- c. コマンドに -sx を追加して、システムを非クラスタモードで起動することを指定します。

[ Minimal BASH-like line editing is supported. For the first word, TAB lists possible command completions. Anywhere else TAB lists the possible completions of a device/filename. ESC at any time exits. ]

```
grub edit> kernel /platform/i86pc/multiboot -sx
```

- d. Enter キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot -sx |
| module /platform/i86pc/boot_archive |
+-----+

```

Use the ^ and v keys to select which entry is highlighted.  
Press 'b' to boot, 'e' to edit the selected command in the

boot sequence, 'c' for a command-line, 'o' to open a new line after ('O' for before) the selected line, 'd' to remove the selected line, or escape to go back to the main menu.-

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-sx` オプションを追加します。

---

- 7 ソフトウェアパッチまたはファームウェアパッチを適用します。

```
patchadd -M patch-dir patch-id
```

*patch-dir*            パッチのディレクトリの場所を指定します。

*patch-id*            特定のパッチのパッチ番号を指定します。

---

注-パッチディレクトリに、この章の手順よりも新しい注意事項がないかどうかを必ず確認してください。

---

- 8 パッチが正常にインストールされていることを確認します。

```
showrev -p | grep patch-id
```

- 9 ノードを再起動してクラスタに結合します。

```
reboot
```

- 10 パッチが機能しており、ノードとクラスタが正常に動作していることを確認します。

- 11 残りのすべてのクラスタノードで、**手順 2** から **手順 10** を繰り返します。

- 12 必要に応じて、リソースグループとデバイスグループを切り替えます。

すべてのノードを再起動した後、最後に再起動したノードのリソースグループとデバイスグループはオンラインになりません。

```
cldevicegroup switch -n node +|devicegroup ...
```

```
clresourcegroup switch -n node[:zone][,...] +|resource-group ...
```

*node*    リソースグループとデバイスグループの切り替え先のノードの名前。

*zone*    リソースグループをマスターできるグローバルクラスタ非投票ノード (*node*) の名前。リソースグループを作成した際に非投票ノードを指定した場合にの

み、zone を指定します。

- 13 `scversions` コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。

```
/usr/cluster/bin/scversions
```

結果として、次に示すメッセージのどちらか一方が表示されます。

```
Upgrade commit is needed.
```

```
Upgrade commit is NOT needed. All versions match.
```

- 14 コミットが必要な場合は、パッチソフトウェアをコミットします。

```
scversions -c
```

---

注 - `scversions` を実行すると、状況に応じて1つ以上のCMM再構成が発生します。

---

#### 例 11-1 再起動パッチの適用(ノード)

次に、ノードに Sun Cluster 再起動パッチを適用する例を示します。

```
clresourcegroup status -n rg1
...Resource Group Resource

rg1 rs-2
rg1 rs-3
...
cldevicegroup status -n nodedg-schost-1
...
Device Group Name: dg-schost-1
...
clnode evacuate phys-schost-2
shutdown -g0 -y -i0
...
```

ノードを非クラスタのシングルユーザーモードで起動します。

- SPARC:次のように入力します。

```
ok boot -sx
```

- x86:ノードを非クラスタのシングルユーザーモードで起動します。続きの手順で起動ステップを確認します。

```
patchadd -M /var/tmp/patches 234567-05
...
```

```
showrev -p | grep 234567-05

...
reboot
...
cldevicegroup switch -n phys-schost-1 dg-schost-1
clresourcegroup switch -n phys-schost-1 schost-sa-1
scversions
Upgrade commit is needed.
scversions -c
```

参照 パッチを取り消す必要がある場合は、339 ページの「Sun Cluster パッチの変更」を参照してください。

## ▼ 再起動パッチを適用する(クラスタ)

この手順で、パッチを適用する前に、`boot -sx` または `shtudown -g -y -i0` コマンドを使用して、まずクラスタを停止して各ノードをシングルユーザーモードで起動する必要があります。

- 1 パッチを適用する前に、Sun Cluster 製品の Web サイトで、インストール前後の特別な注意事項があるかどうかを確認してください。

- 2 クラスタ内にある任意のノード上でスーパーユーザーになります。

- 3 クラスタを停止します。

```
cluster shutdown -y -g grace-period "message"

-y 確認プロンプトで yes と答えます。
-g grace-period 停止までの待機時間を秒単位で指定します。デフォルトの猶予期
 間は 60 秒です。
message 送信する警告メッセージを指定します。message が複数の単語の場
 合は、引用符で囲みます。
```

- 4 各ノードを非クラスタのシングルユーザーモードで起動します。  
各ノードのコンソールで、次のコマンドを実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -sx
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# shutdown -g -y -i0
```

```
Press any key to continue
```

- a. GRUB メニューで矢印キーを使用して該当する Solaris エントリを選択し、e と入力してコマンドを編集します。

GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.

Press enter to boot the selected OS, 'e' to edit the commands before booting, or 'c' for a command-line.

GRUB ベースの起動についての詳細は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、e を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.

Press 'b' to boot, 'e' to edit the selected command in the boot sequence, 'c' for a command-line, 'o' to open a new line after ('O' for before) the selected line, 'd' to remove the selected line, or escape to go back to the main menu.

- c. コマンドに -sx を追加して、システムを非クラスタモードで起動することを指定します。

```
[Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits.]
```

```
grub edit> kernel /platform/i86pc/multiboot -sx
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot -sx |
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-sx` オプションを追加します。

---

- 5 ソフトウェアパッチまたはファームウェアパッチを適用します。一度に1つのノードずつ、次のコマンドを実行します。

```
patchadd -M patch-dir patch-id
patch-dir パッチのディレクトリの場所を指定します。
patch-id 特定のパッチのパッチ番号を指定します。
```

---

注-パッチディレクトリに、この章の手順よりも新しい注意事項がないかどうかを必ず確認してください。

---

- 6 パッチが各ノードに正常にインストールされていることを確認します。

```
showrev -p | grep patch-id
```

- 7 パッチをすべてのノードに適用したなら、ノードを再起動してクラスタに結合します。各ノードで次のコマンドを実行します。

```
reboot
```

- 8 `scversions` コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。

```
/usr/cluster/bin/scversions
```

結果として、次に示すメッセージのどちらか一方が表示されます。

```
Upgrade commit is needed.
```

```
Upgrade commit is NOT needed. All versions match.
```

- 9 コミットが必要な場合は、パッチソフトウェアをコミットします。

```
scversions -c
```

---

注 - `scversions` を実行すると、状況に応じて1つ以上のCMM再構成が発生します。

---

- 10 パッチが機能しており、ノードとクラスタが正常に動作していることを確認します。

#### 例 11-2 再起動パッチの適用 (クラスタ)

次に、クラスタに Sun Cluster 再起動パッチを適用する例を示します。

```
cluster shutdown -g0 -y
```

```
...
```

クラスタを非クラスタのシングルユーザーモードで起動します。

- SPARC:次のように入力します。

```
ok boot -sx
```

- x86:各ノードを非クラスタのシングルユーザーモードで起動します。ステップの続きの手順を確認します。

```
...
```

```
patchadd -M /var/tmp/patches 234567-05
(Apply patch to other cluster nodes)
```

```
...
```

```
showrev -p | grep 234567-05
```

```
reboot
```

```
scversions
```

```
Upgrade commit is needed.
```

```
scversions -c
```

参照 パッチを取り消す必要がある場合は、[339 ページの「Sun Cluster パッチの変更」](#)を参照してください。

## ▼ 非再起動 Sun Cluster パッチを適用する

パッチを一度にクラスタの1つのノードだけに適用します。非再起動パッチを適用するときは、パッチを適用するノードを停止する必要はありません。

- 1 パッチを適用する前に、**Sun Cluster** 製品の **Web** ページで、インストール前後の特別な注意事項があるかどうかを確認してください。

- 2 ひとつのノードにパッチを適用します。

```
patchadd -M patch-dir patch-id
```

*patch-dir*           パッチのディレクトリの場所を指定します。

*patch-id*           特定のパッチのパッチ番号を指定します。

- 3 パッチが正常にインストールされていることを確認します。

```
showrev -p | grep patch-id
```

- 4 パッチが機能しており、ノードとクラスタが正常に動作していることを確認します。

- 5 残りのクラスタノードで、[手順2](#)から[手順4](#)を繰り返します。

- 6 `scversions` コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。

```
/usr/cluster/bin/scversions
```

結果として、次に示すメッセージのどちらか一方が表示されます。

```
Upgrade commit is needed.
```

```
Upgrade commit is NOT needed. All versions match.
```

- 7 コミットが必要な場合は、パッチソフトウェアをコミットします。

```
scversions -c
```

---

注 - `scversions` を実行すると、状況に応じて1つ以上のCMM再構成が発生します。

---

## 例 11-3 非再起動 Sun Cluster パッチの適用

```
patchadd -M /tmp/patches 234567-05
...
showrev -p | grep 234567-05
scversions
Upgrade commit is needed.
scversions -c
```

参照 パッチを取り消す必要がある場合は、339 ページの「Sun Cluster パッチの変更」を参照してください。

## ▼ フェイルオーバーノード構成をしている場合に、シングルユーザーモードでパッチを適用する

フェイルオーバーノードを構成している場合に、シングルユーザーモードでパッチを適用するには、次の作業を実行します。このパッチ方法は、Sun Cluster ソフトウェアによるフェイルオーバー構成で Solaris コンテナ用 Sun Cluster データサービスを使用する場合に必要です。

- 1 この手順で、手動で取得されるゾーンパスを含むディスクセットの一部となる共有ストレージとして使用される LUN のいずれかに、定足数デバイスが構成されていないことを確認します。
  - a. 定足数デバイスがゾーンパスを含むディスクセットで使用されているかどうかを確認し、さらに定足数デバイスが SCSI2 または SCSI3 予約を使用するかどうかを確認します。
 

```
clquorum show
```
  - b. 定足数デバイスがディスクセットの LUN 内にある場合は、ゾーンパスを含むディスクセットの一部ではない定足数デバイスとして、新しい LUN を追加します。
 

```
clquorum add new-didname
```
  - c. 古い定足数デバイスを削除します。
 

```
clquorum remove old-didname
```
  - d. 古い定足数デバイスで SCSI2 予約が使用されている場合は、古い定足数デバイスから SCSI2 予約をスクラブして、SCSI2 予約が残っていないことを確認します。
 

```
/usr/cluster/lib/sc/pgre -c pgre_scrub -d /dev/did/rdsk/old-didnames2
/usr/cluster/lib/sc/pgre -c pgre_inkeys -d /dev/did/rdsk/old-didnames2
```

---

注-有効な定足数デバイスで誤って予約キーをスクラブした場合は、定足数デバイスを削除してからもう一度追加して、その定足数デバイスに新しい予約キーを付与します。

---

- 2 パッチを適用するノードを退避させます。  
# `clresourcegroup evacuate -n node1`
- 3 HA Solaris コンテナリソースを含む1つまたは複数のリソースグループをオフラインにします。  
# `clresourcegroup offline resourcegroupname`
- 4 オフラインにしたリソースグループ内のすべてのリソースを無効にします。  
# `clresource disable resourcename`
- 5 オフラインにしたリソースグループをアンマネージします。  
# `clresourcegroup unmanage resourcegroupname`
- 6 対応する1つまたは複数のデバイスグループをオフラインにします。  
# `cldevicegroup offline cldevicegroupname`
- 7 オフラインにしたデバイスグループを無効にします。  
# `cldevicegroup disable devicegroupname`
- 8 クラスタ外のパッシブノードを起動します。  
# `reboot -- -x`
- 9 続ける前に、パッシブノードでSMF起動方法が完了していることを確認します。  
# `svcs -x`
- 10 アクティブノード上で再構成プロセスが完了していることを確認します。  
# `cluster status`
- 11 ディスクセット内のディスクにSCSI-2予約があるかどうかを調べて、キーを解放します。次の手順に従って、SCSI-2予約が存在するかどうかを確認し、SCSI-2予約を解放します。
  - ディスクセット内のすべてのディスクについて、コマンド `/usr/cluster/lib/sc/scsi -c disfailfast -d /dev/did/rdisk/d#s2` を実行します。
  - キーがリストされている場合は、コマンド `/usr/cluster/lib/sc/scsi -c release -d /dev/did/rdisk/d#s2` を実行してキーを解放します。

予約キーの解放が終了したら、手順 12 をスキップして、手順 13 に進みます。

- 12 ディスクセット内のディスクに **SCSI-3** 予約があるかどうかを調べます。
  - a. ディスクセット内のすべてのディスクで次のコマンドを実行します。

```
/usr/cluster/lib/sc/scsi -c inkeys -d /dev/did/rdisk/didnames2
```
  - b. キーがリストされている場合は、それらをスクラブします。

```
/usr/cluster/lib/sc/scsi -c scrub -d /dev/did/rdisk/didnames2
```
- 13 パッシブノード上のメタセットの所有権を得ます。

```
metaset -s disksetname -C take -f
```
- 14 ゾーンパスを含む 1 つまたは複数のファイルシステムをパッシブノードにマウントします。

```
mount device mountpoint
```
- 15 パッシブノードでシングルユーザーモードに切り替えます。

```
init s
```
- 16 **Solaris** コンテナ用の **Sun Cluster** データサービスの制御下にない、起動されているゾーンをすべて停止します。

```
zoneadm -z zonename halt
```
- 17 (省略可能) 複数のパッチをインストールする場合は、パフォーマンス上の理由から、すべての構成済みゾーンをシングルユーザーモードで起動することを選択できます。

```
zoneadm -z zonename boot -s
```
- 18 パッチを適用します。
- 19 ノードを再起動して、**SMF** 起動方法が終了するまで待ちます。ノードが再起動したあとではじめて `svcs -a` コマンドを実行します。

```
reboot
```

```
svcs -a
```

最初のノードの準備ができました。
- 20 パッチを適用する 2 番目のノードを退避させます。

```
clresourcegroup evacuate -n node2
```
- 21 2 番目のノードで手順 8 ~ 13 を繰り返します。

- 22 パッチプロセスを高速化するために、すでにパッチを適用したゾーンを切り離します。
- ```
# zoneadm -z zonename detach
```
- 23 パッシブノードでシングルユーザーモードに切り替えます。
- ```
init s
```
- 24 Solaris コンテナ用の Sun Cluster データサービスの制御下にない、起動されているゾーンをすべて停止します。
- ```
# zoneadm -z zonename halt
```
- 25 (省略可能) 複数のパッチをインストールする場合は、パフォーマンス上の理由から、すべての構成済みゾーンをシングルユーザーモードで起動することを選択できます。
- ```
zoneadm -z zonename boot -s
```
- 26 パッチを適用します。
- 27 切り離したゾーンを接続します。
- ```
# zoneadm -z zonename attach -F
```
- 28 ノードを再起動して、クラスタモードにします。
- ```
reboot
```
- 29 1つまたは複数のデバイスグループをオンラインにします。
- 30 リソースグループを起動します。
- 31 `scversions` コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。
- ```
# /usr/cluster/bin/scversions
```
- 結果として、次に示すメッセージのどちらか一方が表示されます。
- ```
Upgrade commit is needed.
```
- ```
Upgrade commit is NOT needed. All versions match.
```
- 32 コミットが必要な場合は、パッチソフトウェアをコミットします。
- ```
scversions -c
```

---

注 - `scversions` を実行すると、状況に応じて1つ以上の CMM 再構成が発生します。

---

## Sun Cluster パッチの変更

クラスタに適用した Sun Cluster パッチを削除するには、まず新しい Sun Cluster パッチを削除して、次に以前のパッチまたは更新リリースを再適用します。適用した Sun Cluster パッチを削除するには、次の手順を参照してください。以前の Sun Cluster パッチをもう一度適用するには、次の手順のいずれかを参照してください。

- 325 ページの「再起動パッチを適用する(ノード)」
- 330 ページの「再起動パッチを適用する(クラスタ)」
- 334 ページの「非再起動 Sun Cluster パッチを適用する」

---

注 - Sun Cluster パッチを適用する前に、パッチの README ファイルを確認します。

---

### ▼ 非再起動 Sun Cluster パッチを削除する

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 非再起動パッチを削除します。

```
patchrm patchid
```

### ▼ 再起動 Sun Cluster パッチを削除する

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 クラスタノードを非クラスタモードで起動します。ノードを非クラスタモードで起動する方法については、[83 ページの「非クラスタモードでノードを起動する」](#)を参照してください。

- 3 再起動パッチを削除します。

```
patchrm patchid
```

- 4 クラスタノードを再起動して、クラスタモードに戻します。

```
reboot
```

- 5 各クラスタノードで手順 2～4 を繰り返します。



# ◆◆◆ 第 12 章

## クラスタのバックアップと復元

---

この章は次の節から構成されています。

- 341 ページの「クラスタのバックアップ」
- 354 ページの「クラスタファイルの復元の作業マップ」

### クラスタのバックアップ

表 12-1 作業リスト:クラスタファイルのバックアップ

作業	参照先
バックアップするファイルシステムの名前の検索	342 ページの「バックアップするファイルシステム名を確認する」
フルバックアップを作成するのに必要なテープ数の計算	342 ページの「完全なバックアップに必要なテープ数を決定する」
ルートファイルシステムのバックアップの作成	343 ページの「ルート(/) ファイルシステムをバックアップする」
ミラーまたはプレックスファイルシステムのオンラインバックアップの実行	346 ページの「ミラーのオンラインバックアップを実行する (Solaris ボリュームマネージャー)」  349 ページの「ボリュームのオンラインバックアップの実行 (VERITAS Volume Manager)」
クラスタ構成のバックアップ	353 ページの「クラスタ構成をバックアップする」
ストレージディスクのディスクパーティション分割構成のバックアップ	ストレージディスクのマニュアルを参照

---

## ▼ バックアップするファイルシステム名を確認する

この手順を使用し、バックアップするファイルシステムの名前を判定します。

- 1 /etc/vfstab ファイルの内容を表示します。  
このコマンドを実行するためにスーパーユーザーまたは同等の役割である必要はありません。

```
more /etc/vfstab
```

- 2 バックアップするファイルシステムの名前のマウントポイントの列を調べます。  
この名前は、ファイルシステムをバックアップするときに使用します。

```
more /etc/vfstab
```

### 例 12-1 バックアップするファイルシステム名の確認

次に、/etc/vfstab ファイルに記述されている使用可能なファイルシステム名の例を示します。

```
more /etc/vfstab
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
#/dev/dsk/c1d0s2 /dev/rdisk/c1d0s2 /usr ufs 1 yes -
f - /dev/fd fd - no -
/proc - /proc proc - no -
/dev/dsk/c1t6d0s1 - - swap - no -
/dev/dsk/c1t6d0s0 /dev/rdisk/c1t6d0s0 / ufs 1 no -
/dev/dsk/c1t6d0s3 /dev/rdisk/c1t6d0s3 /cache ufs 2 yes -
swap - /tmp tmpfs - yes -
```

## ▼ 完全なバックアップに必要なテープ数を決定する

この手順を使用し、ファイルシステムのバックアップに必要なテープ数を計算します。

- 1 バックアップするクラスタノード上で、スーパーユーザーまたは同等の役割になります。
- 2 バックアップのサイズをバイト単位で予測します。

```
ufsdump S filesystem
```

```
S バックアップの実行に必要な予測バイト数を表示します。
```

```
filesystem バックアップするファイルシステムの名前を指定します。
```

- 3 予測サイズをテープの容量で割り、必要なテープの数を確認します。

### 例 12-2 必要なテープ数の判別

次の例では、ファイルシステムのサイズは 905,881,620 バイトなので、4G バイトのテープに収めることができます (905,881,620 ÷ 4,000,000,000)。

```
ufsdump S /global/phys-schost-1
905881620
```

## ▼ ルート(/)ファイルシステムをバックアップする

この手順を使用し、クラスタノードのルート (/) ファイルシステムをバックアップします。バックアップ手順を実行する前に、クラスタが正常に動作していることを確認してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 バックアップするクラスタノードで、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。
- 2 実行中の各データサービスを、バックアップを作成するノードからクラスタ内の別のノードに切り替えます。

```
clnode evacuate node
```

node リソースグループとデバイスグループを切り替えるノードを指定します。

- 3 ノードを停止します。

```
shutdown -g0 -y -i0
```

- 4 ノードを非クラスタモードで再起動します。
  - SPARC ベースのシステム上で、次のコマンドを実行します。
 

```
ok boot -xs
```
  - x86 ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# shutdown -g -y -i0
```

Press any key to continue

- a. GRUB メニューで矢印キーを使用して該当する Solaris エントリを選択し、e と入力してコマンドを編集します。

GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86 |
| Solaris failsafe |
| |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.  
Press enter to boot the selected OS, 'e' to edit the  
commands before booting, or 'c' for a command-line.

GRUB ベースの起動についての詳細は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、e を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.  
Press 'b' to boot, 'e' to edit the selected command in the  
boot sequence, 'c' for a command-line, 'o' to open a new line  
after ('O' for before) the selected line, 'd' to remove the  
selected line, or escape to go back to the main menu.

- c. コマンドに -x を追加して、システムが非クラスタモードで起動するように指定します。

```
[Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits.]
```

```
grub edit> kernel /platform/i86pc/multiboot -x
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a) |
| kernel /platform/i86pc/multiboot -x |
| module /platform/i86pc/boot_archive |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.  
Press 'b' to boot, 'e' to edit the selected command in the boot sequence, 'c' for a command-line, 'o' to open a new line after ('O' for before) the selected line, 'd' to remove the selected line, or escape to go back to the main menu.-

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-x` オプションを追加します。

---

- 5 **UFS** スナップショットを作成して、ルート (/) ファイルシステムのバックアップを作成します。
- ファイルシステムに、バックイングストアファイル用の十分なディスク容量が存在することを確認してください。
 

```
df -k
```
  - 同じ場所に同じ名前の既存のバックイングストアファイルが存在していないことを確認します。
 

```
ls /backing-store-file
```
  - UFS** スナップショットを作成します。
 

```
fssnap -F ufs -o bs=/backing-store-file /file-system
```
  - スナップショットが作成されたことを確認します。
 

```
/usr/lib/fs/ufs/fssnap -i /file-system
```
- 6 ノードをクラスタモードで再起動します。
- ```
# init 6
```

例 12-3 ルート (/) ファイルシステムのバックアップ

次の例では、ルート (/) ファイルシステムのスナップショットは /usr ディレクトリ内の /scratch/usr.back.file に保存されています。‘

```
# fssnap -F ufs -o bs=/scratch/usr.back.file /usr
/dev/fssnap/1
```

▼ ミラーのオンラインバックアップを実行する (Solaris ボリュームマネージャー)

ミラー化した Solstice DiskSuite メタデバイスまたは Solaris ボリュームマネージャーのボリュームは、マウント解除したりミラー全体をオフラインにすることなくバックアップできます。サブミラーの1つを一時的にオフラインにする必要があるのも、ミラー化の状態ではなくなりますが、バックアップ完了後ただちにオンラインに戻し、再度同期をとることができます。システムを停止したり、データへのユーザーアクセスを拒否する必要はありません。ミラーを使用してオンラインバックアップを実行すると、アクティブなファイルシステムの「スナップショット」であるバックアップが作成されます。

lockfs コマンドを実行する直前にプログラムがボリュームにデータを書き込むと、問題が生じることがあります。この問題を防ぐには、このノードで実行中のすべてのサービスを一時的に停止します。また、バックアップ手順を実行する前に、クラスタが正常に動作していることを確認してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 バックアップするクラスタノード上で、スーパーユーザーまたは同等の役割になります。
- 2 **metaset(1M)** コマンドを使用し、バックアップするボリュームの所有権を持つノードを判別します。

```
# metaset -s setname
```

-s *setname* ディスクセット名を指定します。

- 3 `-w` オプションを指定して `lockfs(1M)` コマンドを使用し、ファイルシステムへの書き込みをロックします。

```
# lockfs -w mountpoint
```

注-ファイルシステムをロックする必要があるのは、UFS ファイルシステムがミラー上にある場合だけです。たとえば、Solstice DiskSuite メタデバイスや Solaris ボリュームマネージャーボリュームがデータベース管理ソフトやその他の特定のアプリケーションに使用する raw デバイスとして設定されている場合、`lockfs` コマンドを使用する必要はありません。ただし、ソフトウェアアプリケーション固有の適切なユーティリティーを実行し、任意のバッファをフラッシュしてアクセスをロックしてもかまいません。

- 4 `metastat(1M)` コマンドを使用し、サブミラーの名前を判別します。

```
# metastat -s setname -p
```

`-p` `md.tab` ファイルと同様の形式で状態を表示します。

- 5 `metadetach(1M)` コマンドを使用し、ミラーから1つのサブミラーをオフラインにします。

```
# metadetach -s setname mirror submirror
```

注-読み取り操作は引き続きそのほかのサブミラーから行われます。読み取り操作は引き続きそのほかのサブミラーから実行できますが、オフラインのサブミラーは、ミラーに最初に書き込んだ直後から同期がとれなくなります。この不一致は、オフラインのサブミラーをオンラインに戻したときに修正されます。`fsck` を実行する必要はありません。

- 6 `-u` オプションを指定して `lockfs` コマンドを使用し、ファイルシステムのロックを解除して書き込みを続行できるようにします。

```
# lockfs -u mountpoint
```

- 7 ファイルシステムを検査します。

```
# fsck /dev/md/diskset/rdisk/submirror
```

- 8 オフラインのサブミラーをテープなどのメディアにバックアップします。

`ufsdump(1M)` コマンドか、通常使用しているバックアップユーティリティーを使用します。

```
# ufsdump 0ucf dump-device submirror
```

注- ブロックデバイス (/dsk) 名ではなく、サブミラーの raw デバイス (/rdsk) 名を使用してください。

- 9 **metattach(1M)** コマンドを使用し、メタデバイスまたはボリュームをオンラインに戻します。

```
# metattach -s setname mirror submirror
```

メタデバイスまたはボリュームをオンラインに戻すと、自動的にミラーとの再同期が行われます。

- 10 **metastat** コマンドを使用し、サブミラーが再同期されていることを確認します。

```
# metastat -s setname mirror
```

例 12-4 ミラーのオンラインバックアップの実行 (Solaris ボリュームマネージャー)

次の例では、クラスタノード `phys-schost-1` がメタセット `schost-1` の所有者なので、バックアップ作成手順は `phys-schost-1` から実行します。ミラー `/dev/md/schost-1/dsk/d0` は、サブミラー `d10`、`d20`、`d30` で構成されています。

```
[Determine the owner of the metaset:]
# metaset -s schost-1
Set name = schost-1, Set number = 1
Host          Owner
  phys-schost-1  Yes
...
[Lock the file system from writes:]
# lockfs -w /global/schost-1
[List the submirrors:]
# metastat -s schost-1 -p
schost-1/d0 -m schost-1/d10 schost-1/d20 schost-1/d30 1
schost-1/d10 1 1 d4s0
schost-1/d20 1 1 d6s0
schost-1/d30 1 1 d8s0
[Take a submirror offline:]
# metadetach -s schost-1 d0 d30
[Unlock the file system:]
# lockfs -u /
[Check the file system:]
# fsck /dev/md/schost-1/rdsk/d30
[Copy the submirror to the backup device:]
# ufsdump 0ucf /dev/rmt/0 /dev/md/schost-1/rdsk/d30
DUMP: Writing 63 Kilobyte records
DUMP: Date of this level 0 dump: Tue Apr 25 16:15:51 2000
```

```

DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/md/schost-1/rdisk/d30 to /dev/rdsk/clt9d0s0.
...
DUMP: DUMP IS DONE
[Bring the submirror back online:]
# metattach -s schost-1 d0 d30
schost-1/d0: submirror schost-1/d30 is attached
[Resynchronize the submirror:]
# metastat -s schost-1 d0
schost-1/d0: Mirror
  Submirror 0: schost-0/d10
    State: Okay
  Submirror 1: schost-0/d20
    State: Okay
  Submirror 2: schost-0/d30
    State: Resyncing
Resync in progress: 42% done
Pass: 1
Read option: roundrobin (default)
...

```

▼ ボリュームのオンラインバックアップの実行 (VERITAS Volume Manager)

Veritas Volume Manager では、ミラー化ボリュームはプレックスと認識されます。プレックスは、マウント解除したり、ボリューム全体をオフラインにしなくてもバックアップできます。プレックスは、ボリュームのスナップショットコピーを作成し、この一時ボリュームをバックアップします。システムを停止したり、データへのユーザーアクセスを拒否する必要はありません。

バックアップ手順を実行する前に、クラスタが正常に動作していることを確認してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、[付録 B 「Sun Cluster オブジェクト指向コマンド」](#) を参照してください。

- 1 クラスタ内の任意のノードにログオンし、クラスタ上のディスクグループの現在の主ノード上で、スーパーユーザーになるか、RBAC の承認 solaris.cluster.admin を提供する役割になります。

- 2 ディスクグループ情報を表示します。

```
# vxprint -g diskgroup
```

- 3 どのノードに現在インポートされているディスクグループがあるかを判別します (これはそのノードがディスクグループの主ノードであることを示します)。

```
# cldevicegroup status
```

- 4 ボリュームのスナップショットを作成します。

```
# vxassist -g diskgroup snapstart volume
```

注-ボリュームのサイズによっては、スナップショットの作成に時間がかかることがあります。

- 5 新しいボリュームが作成されたことを確認します。

```
# vxprint -g diskgroup
```

スナップショットの作成が完了すると、選択したディスクグループの State フィールドに Snapdone と表示されます。

- 6 ファイルシステムにアクセスしているデータサービスを停止します。

```
# clresourcegroup offline resource-group
```

注-データファイルシステムが正しくバックアップされるように、すべてのデータサービスを停止します。データサービスが実行中でない場合は、[手順6](#)と[手順8](#)を実行する必要はありません。

- 7 bkup-vol という名前のバックアップボリュームを作成し、それにスナップショットボリュームを添付します。

```
# vxassist -g diskgroup snapshot volume bkup-vol
```

- 8 clresourcegroup コマンドを使用して、[手順6](#)で停止されたデータサービスを再起動します。

```
# clresourcegroup online -zone -n node resourcegroup
```

node ノードの名前。

zone リソースグループをマスターできるグローバルクラスタ非投票ノード (*node*) の名前。リソースグループを作成した際に非投票ノードを指定した場合のみ、*zone* を指定します。

- 9 そのボリュームが新しいボリューム bkup-vol に添付されていることを確認します。

```
# vxprint -g diskgroup
```

- 10 デバイスグループ構成変更を登録します。
`cldevicegroup sync diskgroup`
- 11 バックアップボリュームを確認します。
`fsck -y /dev/vx/rdisk/diskgroup/bkup-vol`
- 12 テープなどのメディアにボリューム `bkup-vol` をバックアップします。
`ufsdump(1M)` コマンドか、通常使用しているバックアップユーティリティを使用します。
`ufsdump 0ucf dump-device /dev/vx/dsk/diskgroup/bkup-vol`
- 13 一時ボリュームを削除します。
`vxedit -rf rm bkup-vol`
- 14 ディスクグループ構成変更を登録します。
`cldevicegroup sync diskgroup`

例 12-5 ボリュームのオンラインバックアップの実行 (Veritas Volume Manager)

次の例では、クラスタノード `phys-schost-2` はデバイスグループ `schost-1` の主所有者です。そのため、バックアップ手順は `phys-schost-2` から実行します。ボリューム `/vol01` がコピーされ、新しいボリューム `bkup-vol` と関連付けられます。

[Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization on the primary node.]

[Identify the current primary node for the device group:]

```
# cldevicegroup status
```

```
-- Device Group Servers --
```

| | Device Group | Primary | Secondary |
|-----------------------|--------------|---------------|---------------|
| | ----- | ----- | ----- |
| Device group servers: | rmt/1 | - | - |
| Device group servers: | schost-1 | phys-schost-2 | phys-schost-1 |

```
-- Device Group Status --
```

| | Device Group | Status |
|----------------------|--------------|---------|
| | ----- | ----- |
| Device group status: | rmt/1 | Offline |
| Device group status: | schost-1 | Online |

[List the device group information:]

```
# vxprint -g schost-1
```

| TY NAME | ASSOC | KSTATE | LENGTH | PLOFFS | STATE | TUTIL0 | PUTIL0 |
|---------------|----------|--------|----------|--------|-------|--------|--------|
| dg schost-1 | schost-1 | - | - | - | - | - | - |
| dm schost-101 | clt1d0s2 | - | 17678493 | - | - | - | - |

```

dm schost-102    c1t2d0s2 -      17678493 -    -    -
dm schost-103    c2t1d0s2 -      8378640  -    -    -
dm schost-104    c2t2d0s2 -      17678493 -    -    -
dm schost-105    c1t3d0s2 -      17678493 -    -    -
dm schost-106    c2t3d0s2 -      17678493 -    -    -

v  vol01         gen      ENABLED 204800 -    ACTIVE -
pl  vol01-01     vol01    ENABLED 208331 -    ACTIVE -
sd  schost-101-01 vol01-01 ENABLED 104139 0    -    -
sd  schost-102-01 vol01-01 ENABLED 104139 0    -    -
pl  vol01-02     vol01    ENABLED 208331 -    ACTIVE -
sd  schost-103-01 vol01-02 ENABLED 103680 0    -    -
sd  schost-104-01 vol01-02 ENABLED 104139 0    -    -
pl  vol01-03     vol01    ENABLED LOGONLY -    ACTIVE -
sd  schost-103-02 vol01-03 ENABLED 5      LOG  -    -

[Start the snapshot operation:]
# vxassist -g schost-1 snapstart vol01
[Verify the new volume was created:]
# vxprint -g schost-1
TY NAME          ASSOC      KSTATE   LENGTH  PLOFFS  STATE   TUTIL0  PUTIL0
dg schost-1      schost-1   -        -        -        -        -        -

dm schost-101    c1t1d0s2 -      17678493 -    -    -
dm schost-102    c1t2d0s2 -      17678493 -    -    -
dm schost-103    c2t1d0s2 -      8378640  -    -    -
dm schost-104    c2t2d0s2 -      17678493 -    -    -
dm schost-105    c1t3d0s2 -      17678493 -    -    -
dm schost-106    c2t3d0s2 -      17678493 -    -    -

v  vol01         gen      ENABLED 204800 -    ACTIVE -
pl  vol01-01     vol01    ENABLED 208331 -    ACTIVE -
sd  schost-101-01 vol01-01 ENABLED 104139 0    -    -
sd  schost-102-01 vol01-01 ENABLED 104139 0    -    -
pl  vol01-02     vol01    ENABLED 208331 -    ACTIVE -
sd  schost-103-01 vol01-02 ENABLED 103680 0    -    -
sd  schost-104-01 vol01-02 ENABLED 104139 0    -    -
pl  vol01-03     vol01    ENABLED LOGONLY -    ACTIVE -
sd  schost-103-02 vol01-03 ENABLED 5      LOG  -    -
pl  vol01-04     vol01    ENABLED 208331 -    SNAPDONE -
sd  schost-105-01 vol01-04 ENABLED 104139 0    -    -
sd  schost-106-01 vol01-04 ENABLED 104139 0    -    -

[Stop data services, if necessary:]
# clresourcegroup offline nfs-rg
[Create a copy of the volume:]
# vxassist -g schost-1 snapshot vol01 bkup-vol
[Restart data services, if necessary:]
# clresourcegroup online -n phys-schost-1 nfs-rg

```

```

[Verify bkup-vol was created:]
# vxprint -g schost-1
TY NAME          ASSOC      KSTATE  LENGTH  PLOFFS STATE  TUTIL0  PUTIL0
dg schost-1      schost-1  -       -       -       -       -       -

dm schost-101    clt1d0s2  -       17678493 -       -       -       -
...

v  bkup-vol      gen        ENABLED  204800  -       ACTIVE  -       -
pl bkup-vol-01  bkup-vol  ENABLED  208331  -       ACTIVE  -       -
sd schost-105-01 bkup-vol-01 ENABLED  104139  0       -       -       -
sd schost-106-01 bkup-vol-01 ENABLED  104139  0       -       -       -

v  vol01         gen        ENABLED  204800  -       ACTIVE  -       -
pl vol01-01     vol01     ENABLED  208331  -       ACTIVE  -       -
sd schost-101-01 vol01-01  ENABLED  104139  0       -       -       -
sd schost-102-01 vol01-01  ENABLED  104139  0       -       -       -
pl vol01-02     vol01     ENABLED  208331  -       ACTIVE  -       -
sd schost-103-01 vol01-02  ENABLED  103680  0       -       -       -
sd schost-104-01 vol01-02  ENABLED  104139  0       -       -       -
pl vol01-03     vol01     ENABLED  LOGONLY -       ACTIVE  -       -
sd schost-103-02 vol01-03  ENABLED  5        LOG     -       -       -

[Synchronize the disk group with cluster framework:]
# cldevicegroup sync schost-1
[Check the file systems:]
# fsck -y /dev/vx/rdisk/schost-1/bkup-vol
[Copy bkup-vol to the backup device:]
# ufsdump 0ucf /dev/rmt/0 /dev/vx/rdisk/schost-1/bkup-vol
DUMP: Writing 63 Kilobyte records
DUMP: Date of this level 0 dump: Tue Apr 25 16:15:51 2000
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/vx/dsk/schost-2/bkup-vol to /dev/rmt/0.
...
DUMP: DUMP IS DONE

[Remove the bkup-volume:]
# vxedit -rf rm bkup-vol
[Synchronize the disk group:]
# cldevicegroup sync schost-1

```

▼ クラスタ構成をバックアップする

クラスタ構成をアーカイブし、クラスタ構成の簡単な復元を実現するため、定期的にクラスタ構成をバックアップします。&fmv1 3.2には、クラスタ構成をXML (eXtensible Markup Language) ファイルにエクスポートする機能があります。

- 1 クラスタ内の任意のノードにログオンし、スーパーユーザーになるか、RBACの承認 `solaris.cluster.read` を提供する役割になります。

- 2 クラスタ構成情報をファイルにエクスポートします。

```
# /usr/cluster/bin/cluster export -o configfile
```

configfile クラスタコマンドのクラスタ構成情報のエクスポート先であるXML構成ファイルの名前。XML構成ファイルについては、[clconfiguration\(5CL\)](#)を参照してください。

- 3 クラスタ構成情報が正常にXMLファイルにエクスポートされたことを確認します。

```
# vi configfile
```

クラスタファイルの復元の作業マップ

`ufsrestore(1M)` コマンドを実行すると、`ufsdump(1M)` コマンドで作成されたバックアップのファイルが、現在の作業ディレクトリを基準として指定されるディスク上の位置にコピーされます。`ufsrestore` を使用すると、レベル0のダンプとそれ以降の増分ダンプからファイルシステム階層全体を読み込み直したり、任意のダンプテープから個々のファイルを復元できます。スーパーユーザーまたは同等の役割として `ufsrestore` を実行すると、元の所有者、最終修正時刻、モード(アクセス権)を保持したままファイルを復元できます。

ファイルまたはファイルシステムの復元を開始する前に、次の点を確認してください。

- 必要なテープ
- ファイルシステムを復元する raw デバイス名
- 使用するテープドライブの種類
- テープドライブのデバイス名(ローカルまたはリモート)
- 障害が発生したディスクのパーティション分割方式。これは、パーティションとファイルシステムを交換用ディスクに正確に複製しなければならないためです。

表 12-2 作業リスト:クラスタファイルの復元

| 作業 | 参照先 |
|--|---|
| Solaris ボリュームマネージャーの場合、対話形式でファイルを復元 | 355 ページの「個々のファイルを対話形式で復元する (Solaris ボリュームマネージャー)」 |
| Solaris ボリュームマネージャーの場合、ルート (/) ファイルシステムを復元 | 355 ページの「ルート (/) ファイルシステムを復元する (/)」 |

表 12-2 作業リスト:クラスタファイルの復元 (続き)

| 作業 | 参照先 |
|--|---|
| | 357 ページの「Solstice DiskSuite メタデバイスまたは Solaris ボリュームマネージャー上に存在していたルート (/) ファイルシステムを復元する」 |
| Veritas Volume Manager の場合、ルート (/) ファイルシステムを復元 | 363 ページの「カプセル化されていないルート (/) ファイルシステムを復元する (Veritas Volume Manager)」 |
| Veritas Volume Manager の場合、カプセル化されたルート (/) ファイルシステムを復元 | 365 ページの「カプセル化されたルート (/) ファイルシステムを復元する (VERITAS Volume Manager)」 |

▼ 個々のファイルを対話形式で復元する (Solaris ボリュームマネージャー)

この手順を使用し、1つまたは複数の個々のファイルを復元します。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

- 1 復元するクラスタノード上で、スーパーユーザーになるか、**RBAC**の承認 `solaris.cluster.admin` を提供する役割になります。
- 2 復元するファイルを使用しているデータサービスをすべて停止します。
`clresourcegroup offline resource-group`
- 3 ファイルを復元します。
`ufsrestore`

▼ ルート (/) ファイルシステムを復元する (/)

障害の発生したルートディスクを交換した後などに、この手順を使用してルート (/) ファイルシステムを新しいディスクに復元します。復元中のノードは起動しなさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

注-新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 復元するノードの添付先であるディスクセットへのアクセス権があるクラスタノード上で、スーパーユーザーになるか、`solaris.cluster.modifyRBAC` 権限を提供する役割になります。
復元する以外のノードを使用します。
- 2 すべてのメタセットから、復元するノードのホスト名を削除します。
このコマンドは、削除するノード以外のメタセットのノードから実行します。復元を行っているノードはオフラインであるため、システムは「RPC: Rpcbnd failure - RPC: Timed out」というエラーを表示します。このエラーを無視し、次のステップを続けます。

```
# metaset -s setname -f -d -h nodelist
```

| | |
|-------------|----------------------------|
| -s setname | ディスクセット名を指定します。 |
| -f | ディスクセットから最後のホストを削除します。 |
| -d | ディスクセットから削除します。 |
| -h nodelist | ディスクセットから削除するノードの名前を指定します。 |

- 3 `root (/)` ファイルシステムと `/usr` ファイルシステムを復元します。
`root` ファイルシステムと `/usr` ファイルシステムを復元するには、『Solaris のシステム管理 (デバイスとファイルシステム)』の第 26 章「UFS ファイルとファイルシステムの復元 (手順)」の手順に従ってください。Solaris OS の手順にあるシステムを再起動する手順は省略してください。

注 - `/global/.devices/node@nodeid` ファイルシステムが作成されていることを確認します。

- 4 ノードをマルチユーザーモードで再起動します。

```
# reboot
```
- 5 デバイス ID を交換します。

```
# cldevice repair rootdisk
```
- 6 `metadb(1M)` コマンドを使用し、状態データベースの複製を再作成します。

```
# metadb -c copies -af raw-disk-device
```

| | |
|-----------|-----------------|
| -c copies | 作成する複製の数を指定します。 |
|-----------|-----------------|

-f *raw-disk-device* 複製の作成先の raw ディスクデバイス名を指定します。
 -a 複製を追加します。

- 7 復元するノード以外のクラスタノードから、復元するノードをすべてのディスクセットに追加します。

```
phys-schost-2# metaset -s setname -a -h nodelist
```

-a ホストを作成してディスクセットに追加します。

ノードがクラスタモードで再起動します。これでクラスタを使用できるようになります。

例 12-6 ルート (/) ファイルシステムの復元 (Solaris ポリリュームマネージャー)

次に、テープデバイス /dev/rmt/0 からノード phys-schost-1 に復元したルート (/) ファイルシステムの例を示します。metaset コマンドは、クラスタの別のノード phys-schost-2 から実行し、ノード phys-schost-1 を削除し、後でディスクセット schost-1 に追加します。そのコマンドはすべて phys-schost-1 から実行します。新しいブートブロックが /dev/rdisk/c0t0d0s0 に作成され、3つの状態データベースの複製が /dev/rdisk/c0t0d0s4 に再作成されます。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on a cluster node other than the local node.]
```

```
[Remove the node from the metaset:]
```

```
phys-schost-2# metaset -s schost-1 -f -d -h phys-schost-1
```

```
[Replace the failed disk and boot the node:]
```

```
Restore the root (/) and /usr file system using the procedure in the Solaris system administration documentation
```

```
[Reboot:]
```

```
# reboot
```

```
[Replace the disk ID:]
```

```
# cldevice repair /dev/dsk/c0t0d0
```

```
[Re-create state database replicas:]
```

```
# metadb -c 3 -af /dev/rdisk/c0t0d0s4
```

```
[Add the node back to the metaset:]
```

```
phys-schost-2# metaset -s schost-1 -a -h phys-schost-1
```

▼ Solstice DiskSuite メタデバイスまたは Solaris ポリリュームマネージャー上に存在していたルート (/) ファイルシステムを復元する

この手順を使用して、バックアップ実行時に Solstice DiskSuite メタデバイスまたは Solaris Volume Manager のポリリューム上に存在していたルート (/) ファイルシステムを復元します。この手順は、破損したルートディスクを新しいディスクに交換する場

合などに実行します。復元中のノードは起動しなおさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

注-新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 ディスクセットへのアクセス権があるクラスタノード、ただし復元するノード以外のノード上で、スーパーユーザーになるか、**RBAC**の承認 `solaris.cluster.modify` を提供する役割になります。
復元する以外のノードを使用します。

- 2 すべてのディスクセットから、復元するノードのホスト名を削除します。

```
# metaset -s setname -f -d -h nodelist
```

| | |
|------------------------------------|---------------------------------------|
| <code>-s setname</code> | メタセット名を指定します。 |
| <code>-f</code> | ディスクセットから最後のホストを削除します。 |
| <code>-d</code> | メタセットから削除します。 |
| <code>-h nodelist</code> | メタセットから削除するノードの名前を指定します。 |
| <code>-m mediator_host_list</code> | ディスクセットから追加または削除するメディアータホストの名前を指定します。 |

- 3 ルート (/) ファイルシステムを復元するノードで、障害の発生したディスクを交換します。
ディスク交換手順については、サーバーに付属のマニュアルを参照してください。

- 4 復元するノードを起動します。

- Solaris OS CD を使用している場合は、次の点に注意してください。
 - SPARC:次のように入力します。

```
ok boot cdrom -s
```

- x86:CD をシステムの CD ドライブに挿入し、システムを停止して、電源を切って入れなおすことにより、システムを起動します。「Current Boot Parameters」画面で **b** または **i** を入力します。

```

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@
7,1/sd@0,0:a
Boot args:

Type b [file-name] [boot-flags] <ENTER> to boot with options
or i <ENTER> to enter boot interpreter
or <ENTER> to boot with defaults

```

```

<<< timeout in 5 seconds >>>
Select (b)oot or (i)nterpreter: b -s

```

- Solaris JumpStart™ サーバーを使用している場合は、次の点に注意してください。
 - SPARC:次のように入力します。

```
ok boot net -s
```

- x86:CD をシステムの CD ドライブに挿入し、システムを停止して、電源を切って入れなおすことにより、システムを起動します。「Current Boot Parameters」画面で **b** または **i** を入力します。

```

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@
7,1/sd@0,0:a
Boot args:

Type b [file-name] [boot-flags] <ENTER> to boot with options
or i <ENTER> to enter boot interpreter
or <ENTER> to boot with defaults

```

```

<<< timeout in 5 seconds >>>
Select (b)oot or (i)nterpreter: b -s

```

- 5 `format` コマンドを使用し、ルートディスクのすべてのパーティションとスワップ空間を作成します。
障害の発生したディスクに存在していた元のパーティションの分割方式を再作成します。
- 6 `newfs` コマンドを使用し、必要に応じてルート (/) ファイルシステムやその他のファイルシステムを作成します。
障害の発生したディスクに存在していた元のファイルシステムを再作成します。

注 - /global/.devices/node@nodeid ファイルシステムが作成されていることを確認します。

- 7 ルート (/) ファイルシステムを一時マウントポイントにマウントします。

```
# mount device temp-mountpoint
```
- 8 次のコマンドを使用し、ルート (/) ファイルシステムを復元します。

```
# cd temp-mountpoint
# ufsrestore rvf dump-device
# rm restoresymtable
```
- 9 新しい起動ブロックを新しいディスクにインストールします。

```
# /usr/sbin/installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk
raw-disk-device
```
- 10 /temp-mountpoint/etc/system ファイルの MDD ルート情報の行を削除します。

```
* Begin MDD root info (do not edit)
forceload: misc/md_trans
forceload: misc/md_raid
forceload: misc/md_mirror
forceload: misc/md_hotspares
forceload: misc/md_stripe
forceload: drv/pcipsy
forceload: drv/glm
forceload: drv/sd
rootdev:/pseudo/md@0:0,10,blk
* End MDD root info (do not edit)
```
- 11 /temp-mountpoint/etc/vfstab ファイルを編集して、ルートエントリを **Solstice DiskSuite** メタデバイスまたは **Solaris** ポリリュームマネージャー ポリリュームからメタデバイスまたはポリリュームの一部であるルートディスク上の各ファイルシステムの対応する正常なスライスに変更します。

Example:

Change from-

```
/dev/md/dsk/d10 /dev/md/rdisk/d10 / ufs 1 no -
```

Change to-

```
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
```
- 12 一時ファイルシステムをマウント解除し、**raw** ディスクデバイスを確認します。

```
# cd /
# umount temp-mountpoint
# fsck raw-disk-device
```

- 13 ディスクセットのメディアータホストのリストから復元したノードを削除します。
`# metaset -s setname -f -d -m hostname`
- 14 ノードをマルチユーザーモードで再起動します。
`# reboot`
- 15 デバイス ID を交換します。
`# cldevice repair rootdisk`
- 16 `metadb` コマンドを使用し、状態データベースの複製を再作成します。
`# metadb -c copies -af raw-disk-device`
`-c copies` 作成する複製の数を指定します。
`-af raw-disk-device` 指定した raw ディスクデバイスに初期状態のデータベースの複製を作成します。
- 17 復元したノード以外のクラスタノードから、復元したノードをすべてのディスクセットに追加します。
`phys-schost-2# metaset -s setname -a -h nodelist`
`-a` メタセットを追加 (作成) します。
 Solstice DiskSuite のマニュアルに従って、メタデバイスまたはボリューム/ミラーのルート (/) を設定します。
 ノードがクラスタモードで再起動します。これでクラスタを使用できるようになります。

例 12-7 Solstice DiskSuite メタデバイスまたは Solaris ボリュームマネージャー上に存在していたルート (/) ファイルシステムの復元

次に、テープデバイス `/dev/rmt/0` からノード `phys-schost-1` に復元したルート (/) ファイルシステムの例を示します。`metaset` コマンドは、クラスタの別のノード `phys-schost-2` から実行し、ノード `phys-schost-1` を削除し、後でメタセット `schost-1` に追加します。そのコマンドはすべて `phys-schost-1` から実行します。新しいブートブロックが `/dev/rdsk/c0t0d0s0` に作成され、3つの状態データベースの複製が `/dev/rdsk/c0t0d0s4` に再作成されます。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on a cluster node with
[Remove the node from the metaset:]
phys-schost-2# metaset -s schost-1 -f -d -h phys-schost-1
[Replace the failed disk and boot the node:]
```

次の操作で、Solaris OS CD からノードを起動します。

- SPARC:次のように入力します。

```
ok boot cdrom -s
```

- x86:CD をシステムの CD ドライブに挿入し、システムを停止して、電源を切って入れなおすことにより、システムを起動します。「Current Boot Parameters」画面で**b**または**i**を入力します。

```
<<< Current Boot Parameters >>>
```

```
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@7,1/
sd@0,0:a
```

```
Boot args:
```

```
Type  b [file-name] [boot-flags] <ENTER> to boot with options
or    i <ENTER>                          to enter boot interpreter
or    <ENTER>                             to boot with defaults
```

```
<<< timeout in 5 seconds >>>
```

```
Select (b)oot or (i)nterpreter: b -s
```

```
[Use format and newfs to recreate partitions and file systems
```

```
.]
```

```
[Mount the root file system on a temporary mount point:]
```

```
# mount /dev/dsk/c0t0d0s0 /a
```

```
[Restore the root file system:]
```

```
# cd /a
```

```
# ufsrestore rvf /dev/rmt/0
```

```
# rm restoresymtable
```

```
[Install a new boot block:]
```

```
# /usr/sbin/installboot /usr/platform/'uname \
-i'/lib/fs/ufs/bootblk /dev/rdisk/c0t0d0s0
```

```
[Remove the lines in / temp-mountpoint/etc/system file for MDD root information:
```

```
]
```

```
* Begin MDD root info (do not edit)
```

```
forceload: misc/md_trans
```

```
forceload: misc/md_raid
```

```
forceload: misc/md_mirror
```

```
forceload: misc/md_hotspares
```

```
forceload: misc/md_stripe
```

```
forceload: drv/pcipsy
```

```
forceload: drv/glm
```

```
forceload: drv/sd
```

```
rootdev:/pseudo/md@0:0,10,blk
```

```
* End MDD root info (do not edit)
```

```
[Edit the /temp-mountpoint/etc/vfstab file]
```

```
Example:
```

```
Change from-
```

```
/dev/md/dsk/d10 /dev/md/rdsk/d10 / ufs 1 no -
```

```
Change to-
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 /usr ufs 1 no -
[Unmount the temporary file system and check the raw disk device:]
# cd /
# umount /a
# fsck /dev/rdsk/c0t0d0s0
[Reboot:]
# reboot
[Replace the disk ID:]
# cldevice repair /dev/rdsk/c0t0d0
[Re-create state database replicas:]
# metadb -c 3 -af /dev/rdsk/c0t0d0s4
[Add the node back to the metaset:]
phys-schost-2# metaset -s schost-1 -a -h phys-schost-1
```

▼ カプセル化されていないルート (/) ファイルシステムを復元する (Veritas Volume Manager)

この手順を使用して、カプセル化されていないルート (/) ファイルシステムをノードに復元します。復元中のノードは起動しなおさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

注-新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 ルートファイルシステムを復元するノードで、障害が発生したディスクを交換します。
ディスク交換手順については、サーバーに付属のマニュアルを参照してください。
- 2 復元するノードを起動します。

- Solaris OS CD を使用している場合は、OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。

```
ok boot cdrom -s
```

- Solaris JumpStart サーバーを使用している場合は、OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。

```
ok boot net -s
```

- 3 format コマンドを使用し、ルートディスクのすべてのパーティションとスワップを作成します。

障害の発生したディスクに存在していた元のパーティションの分割方式を再作成します。

- 4 newfs コマンドを使用し、必要に応じてルート (/) ファイルシステムやその他のファイルシステムを作成します。

障害の発生したディスクに存在していた元のファイルシステムを再作成します。

注 - /global/.devices/ node@nodeid ファイルシステムが作成されていることを確認します。

- 5 ルート (/) ファイルシステムを一時マウントポイントにマウントします。

```
# mount device temp-mountpoint
```

- 6 バックアップからルート (/) ファイルシステムを復元し、ファイルシステムをマウント解除して確認します。

```
# cd temp-mountpoint
# ufsrestore rvf dump-device
# rm restoresymtable
# cd /
# umount temp-mountpoint
# fsck raw-disk-device
```

これでファイルシステムが復元されます。

- 7 新しい起動ブロックを新しいディスクにインストールします。

```
# /usr/sbin/installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk raw-disk-device
```

- 8 ノードをマルチユーザーモードで再起動します。

```
# reboot
```

- 9 デバイス ID を更新します。

```
# cldevice repair /dev/rdisk/disk-device
```

- 10 Control-D キーを押して、マルチユーザーモードで再起動します。

ノードがクラスタモードで再起動します。これでクラスタを使用できるようになります。

例 12-8 カプセル化されていないルート (/) ファイルシステムの復元 (Veritas Volume Manager)

次に、カプセル化されていないルート (/) ファイルシステムがテープデバイス /dev/rmt/0 からノード phys-schost-1 に復元される例を示します。

[Replace the failed disk and boot the node:]

Solaris OS CD からノードを起動します。OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。

```
ok boot cdrom -s
...
[Use format and newfs to create partitions and file systems]
[Mount the root file system on a temporary mount point:]
# mount /dev/dsk/c0t0d0s0 /a
[Restore the root file system:]
# cd /a
# ufsrestore rvf /dev/rmt/0
# rm restoresymtable
# cd /
# umount /a
# fsck /dev/rdisk/c0t0d0s0
[Install a new boot block:]
# /usr/sbin/installboot /usr/platform/'uname \
-i'/lib/fs/ufs/bootblk /dev/rdisk/c0t0d0s0

[Reboot:]
# reboot
[Update the disk ID:]
# cldevice repair /dev/rdisk/c0t0d0
```

▼ カプセル化されたルート (/) ファイルシステムを復元する (VERITAS Volume Manager)

この手順を使用して、カプセル化されているルート (/) ファイルシステムをノードに復元します。復元中のノードは起動しなさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

注-新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Sun Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。コマンドのリストとその短縮形については、付録 B 「Sun Cluster オブジェクト指向コマンド」を参照してください。

- 1 ルートファイルシステムを復元するノードで、障害が発生したディスクを交換します。
ディスク交換手順については、サーバーに付属のマニュアルを参照してください。
- 2 復元するノードを起動します。
 - Solaris OS CD を使用している場合は、OpenBoot PROM の `ok` プロンプトで、次のコマンドを入力します。

`ok boot cdrom -s`
 - Solaris JumpStart サーバーを使用している場合は、OpenBoot PROM の `ok` プロンプトで、次のコマンドを入力します。

`ok boot net -s`
- 3 `format` コマンドを使用し、ルートディスクのすべてのパーティションとスワップ空間を作成します。
障害の発生したディスクに存在していた元のパーティションの分割方式を再作成します。
- 4 `newfs` コマンドを使用し、必要に応じてルート (`/`) ファイルシステムやその他のファイルシステムを作成します。
障害の発生したディスクに存在していた元のファイルシステムを再作成します。

注 - `/global/.devices/node@nodeid` ファイルシステムが作成されていることを確認します。

- 5 ルート (`/`) ファイルシステムを一時マウントポイントにマウントします。
`# mount device temp-mountpoint`

- 6 バックアップからルート(/)ファイルシステムを復元します。
- ```
cd temp-mountpoint
ufsrestore rvf dump-device
rm restoresymtable
```
- 7 空の install-db ファイルを作成します。  
このファイルによって、次回起動時にノードが VxVM インストールモードになります。
- ```
# touch \
/temp-mountpoint/etc/vx/reconfig.d/state.d/install-db
```
- 8 /temp-mountpoint/etc/system ファイル内の次のエントリを削除します。
- ```
* rootdev:/pseudo/vxio@0:0
* set vxio:vol_rootdev_is_volume=1
```
- 9 /temp-mountpoint/etc/vfstab ファイルを編集し、すべての VxVM マウントポイントをルートディスクの標準ディスクデバイス (/dev/dsk/c0t0d0s0 など) に置換します。
- Example:  
Change from-
- ```
/dev/vx/dsk/rootdg/rootvol /dev/vx/rdisk/rootdg/rootvol / ufs 1 no -
```
- Change to-
- ```
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
```
- 10 一時ファイルシステムをマウント解除してファイルシステムを確認します。
- ```
# cd /
# umount temp-mountpoint
# fsck raw-disk-device
```
- 11 起動ブロックを新しいディスクにインストールします。
- ```
/usr/sbin/installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk raw-disk-device
```
- 12 ノードをマルチユーザーモードで再起動します。
- ```
# reboot
```
- 13 **scdidadm(1M)** を使用し、デバイス ID を更新します。
- ```
cldevice repair /dev/rdisk/c0t0d0
```
- 14 ディスクをカプセル化して再起動するために、clvxdm コマンドを実行します。
- 15 マイナー番号が他のシステムと衝突している場合は、グローバルデバイスをマウント解除し、ディスクグループに別のマイナー番号を割り当てます。

- クラスタノードのグローバルデバイスファイルシステムをマウント解除します。  
# **umount /global/.devices/node@nodeid**
- クラスタノードの rootdg ディスクグループに別のマイナー番号を割り当てます。  
# **vx dg remminor rootdg 100**

16 ノードを停止し、クラスタモードで再起動します。

```
shutdown -g0 -i6 -y
```

### 例 12-9 カプセル化されたルート (/) ファイルシステムの復元 (Veritas Volume Manager)

次に、カプセル化されたルート (/) ファイルシステムがテープデバイス /dev/rmt/0 からノード phys-schost-1 に復元される例を示します。

[Replace the failed disk and boot the node:]

Solaris OS CD からノードを起動します。OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。

```
ok boot cdrom -s
...
[Use format and newfs to create partitions and file systems]
[Mount the root file system on a temporary mount point:]
mount /dev/dsk/c0t0d0s0 /a
[Restore the root file system:]
cd /a
ufsrestore rvf /dev/rmt/0
rm restoresymtable
[Create an empty install-db file:]
touch /a/etc/vx/reconfig.d/state.d/install-db
[Edit /etc/system on the temporary file system and
remove or comment out the following entries:]
rootdev:/pseudo/vxio@0:0
set vxio:vol_rootdev_is_volume=1
[Edit /etc/vfstab on the temporary file system:]
Example:
Change from-
/dev/vx/dsk/rootdg/rootvol /dev/vx/rdisk/rootdg/rootvol / ufs 1 no-

Change to-
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
[Unmount the temporary file system, then check the file system:]
cd /
umount /a
```

```
fsck /dev/rdisk/c0t0d0s0
[Install a new boot block:]
/usr/sbin/installboot /usr/platform/'uname \
-i'/lib/fs/ufs/bootblk /dev/rdisk/c0t0d0s0
[Reboot:]
reboot
[Update the disk ID:]
cldevice repair /dev/rdisk/c0t0d0
[Encapsulate the disk:]
vxinstall
Choose to encapsulate the root disk.
[If a conflict in minor number occurs, remind the rootdg disk group:]
umount /global/.devices/node@nodeid
vxdg remind rootdg 100
shutdown -g0 -i6 -y
```

参照 カプセル化されたルートディスクをミラー化する手順については、[『Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)』](#)を参照してください。



# グラフィカルユーザーインターフェースによる Sun Cluster の管理

---

この章では、グラフィカルユーザーインターフェース (Graphical User Interface、GUI) ツールの、Sun Cluster Manager と Sun Management Center について説明します。これらのツールを使用すると、クラスタをさまざまな面から管理できます。また、Sun Cluster Manager を構成および起動する手順も説明します。Sun Cluster Manager GUI に含まれるオンラインヘルプでは、さまざまな Sun Cluster 管理作業の手順を説明しています。

この章の内容は、次のとおりです。

- 371 ページの「Sun Cluster Manager の概要」
- 372 ページの「SPARC: Sun Management Center の概要」
- 373 ページの「Sun Cluster Manager の構成」
- 376 ページの「Sun Cluster Manager ソフトウェアの起動」

## Sun Cluster Manager の概要

Sun Cluster Manager は、クラスタ情報のグラフィカルな表示、構成の変更の監視、およびクラスタコンポーネントのチェックを可能にする GUI です。Sun Cluster Manager では、以下の Sun Cluster コンポーネントを対象としたさまざまな管理作業も行えます。

- アダプタ
- ケーブル
- データサービス
- グローバルデバイス
- 相互接続
- 接続点
- ノード
- 定足数デバイス
- リソースグループ
- リソース

Sun Cluster Manager をインストールおよび使用方法については、次の文書を参照してください。

- **Sun Cluster Manager** のインストール: 『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』を参照してください。
- **Sun Cluster Manager** の起動: 376 ページの「[Sun Cluster Manager ソフトウェアの起動](#)」を参照してください。
- ポート番号、サーバーアドレス、セキュリティー証明書、ユーザーの構成: 373 ページの「[Sun Cluster Manager の構成](#)」を参照してください。
- **Sun Cluster Manager** によるクラスタのインストールと管理: Sun Cluster Manager に付属のオンラインヘルプを参照してください。
- **Sun Cluster Manager** セキュリティー鍵の再生成: 375 ページの「[共通エージェントコンテナのセキュリティー鍵を再生成する](#)」を参照してください。

---

注-ただし、Sun Cluster Manager は現在、Sun Cluster のすべての管理作業を実行できるわけではありません。一部の作業には、コマンド行インタフェースを使用する必要があります。

---

## SPARC: Sun Management Center の概要

Sun Management Center™ (旧 Sun Enterprise SyMON™) 用の Sun Cluster モジュールの GUI コンソールを使用すると、クラスタリソース、リソースタイプ、リソースグループをグラフィカルに表示できます。また、構成の変更を監視したり、クラスタコンポーネントの状態を検査できます。ただし、Sun Management Center 用の Sun Cluster モジュールは、Sun Cluster の構成作業を行えません。構成処理には、コマンド行インタフェースを使用する必要があります。詳細については、第 1 章「コマンド行インタフェース」を参照してください。

Sun Management Center 用 Sun Cluster モジュールのインストールと起動の詳細は、『[Sun Cluster ソフトウェアのインストール \(Solaris OS 版\)](#)』の第 8 章「[Sun Cluster モジュールの Sun Management Center へのインストール](#)」を参照してください。

Sun Management Center 用の Sun Cluster モジュールは Simple Network Management Protocol (SNMP) に準拠しています。したがって、SNMP に基づくサン以外の管理ステーションは、Sun Cluster が作成する管理情報ベース (MIB) をデータ定義として使用できます。

Sun Cluster MIB ファイルは、任意のクラスタノード上の `/opt/SUNwsymon/modules/cfg/sun-cluster-mib.mib` にあります。

Sun Cluster の MIB ファイルは、モデル化された Sun Cluster データの ASN.1 仕様です。この仕様は、Sun Management Center のすべての MIB で使用される仕様と同じで

す。Sun Cluster MIB を使用する方法については、『Sun Management Center 3.6 User's Guide』の「SNMP MIBs for Sun Management Centre Modules」にある、ほかの Sun Management Center MIB を使用するための手順を参照してください。

## Sun Cluster Manager の構成

&fmv426 は、定足数デバイス、IPMP グループ、インターコネクトコンポーネント、グローバルデバイスなどのあらゆる局面の状態を管理、表示できる GUI です。この GUI は、多くの Sun Cluster CLI コマンドの代わりに使用できます。

Sun Cluster Manager をクラスタにインストールする手順については、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』を参照してください。GUI を使用してさまざまな作業を行う方法については、Sun Cluster Manager のオンラインヘルプを参照してください。

この節では、初期インストール後、Sun Cluster Manager を再構成するための次のような手順について説明します。

- 373 ページの「RBAC の役割の設定」
- 374 ページの「Sun Cluster Manager のサーバーアドレスを変更する」
- 375 ページの「共通エージェントコンテナのセキュリティキーを再生成する」

## RBAC の役割の設定

Sun Cluster Manager は、RBAC を使用して、誰がクラスタを管理する権限を持っているかを判定します。Sun Cluster ソフトウェアには、いくつかの RBAC 権限プロファイルが含まれています。これらの権限プロファイルをユーザーまたは役割に割り当てることで、Sun Cluster に対するさまざまなレベルのアクセス権をユーザーに与えることができます。Sun Cluster ソフトウェアの RBAC を設定および管理する方法についての詳細は、第 2 章「Sun Cluster と RBAC」を参照してください。

### ▼ 共通エージェントコンテナを使用して、サービスまたは管理エージェントのポート番号を変更する

共通エージェントコンテナサービスのデフォルトのポート番号 (6789) が実行中の別のプロセスと衝突する場合、`cacaoadm` コマンドを使用し、各クラスタノード上で、衝突しているサービスまたは管理エージェントのポート番号を変更できます。

- 1 すべてのクラスタ上で共通エージェントコンテナ管理デーモンを停止します。

```
/opt/bin/cacaoadm stop
```

- 2 Sun Java Web Console を停止します。

```
/usr/sbin/smcwebserver stop
```

- 3 get-param サブコマンドを使用して、共通エージェントコンテナサービスにより現在使用されているポート番号を取得します。

```
/opt/bin/cacaoadm get-param parameterName
```

cacaoadm コマンドを使用して、以下の共通エージェントコンテナサービスのポート番号を変更できます。次のリストは、共通エージェントコンテナで管理できるサービスとエージェント、および対応するパラメータ名の例を示しています。

JMX コネクタポート	jmxmp-connector-port
SNMP ポート	snmp-adaptor-port
SNMP トラップポート	snmp-adaptor-trap-port
コマンドストリームポート	commandstream-adaptor-port

- 4 ポート番号を変更します。

```
/opt/bin/cacaoadm set-param parameterName=parameterValue
=parameterValue
```

- 5 クラスタの各ノード上で手順4を繰り返します。

- 6 Sun Java Web Console を再起動します。

```
/usr/sbin/smcwebserver start
```

- 7 すべてのクラスタノード上で共通エージェントコンテナ管理デーモンを再起動します。

```
/opt/bin/cacaoadm start
```

## ▼ Sun Cluster Manager のサーバーアドレスを変更する

クラスタノードのホスト名を変更する場合、Sun Cluster Manager を実行するアドレスを変更する必要があります。デフォルトのセキュリティー証明書は、Sun Cluster Manager がインストールされる時点でノードのホスト名に基づいて生成されます。ノードのホスト名をリセットするには、証明書ファイルである keystore を削除し、Sun Cluster Manager を再起動します。Sun Cluster Manager は、新しいホスト名を使用して新しい証明書ファイルを自動的に作成します。この手順は、ホスト名を変更したすべてのノード上で行う必要があります。

- 1 /etc/opt/webconsole にある証明書ファイル keystore を削除します。

```
cd /etc/opt/webconsole
pkgrm keystore
```

- 2 Sun Cluster Manager を再起動します。

```
/usr/sbin/smcwebserver restart
```

## ▼ 共通エージェントコンテナのセキュリティー鍵を再生成する

Sun Cluster Manager は、強力な暗号化技術を使用して、Sun Cluster Manager Web サーバーと各クラスタノード間の安全な通信を確保しています。

Sun Cluster Manager が使用する鍵は、各ノードの /etc/opt/SUNWcacao/security ディレクトリに格納されています。これらの鍵は、すべてのクラスタノードで同一でなければなりません。

通常の動作では、これらのキーはデフォルトの構成のままとなります。クラスタノードのホスト名を変更する場合は、共通エージェントコンテナのセキュリティー鍵を再生成する必要があります。また、鍵が攻撃の対象となる恐れがある場合 (マシンのルート侵入など) にも鍵の再生成が必要となります。セキュリティー鍵を再生成するには、次の手順を実行します。

- 1 すべてのクラスタ上で共通エージェントコンテナ管理デーモンを停止します。

```
/opt/bin/cacaoadm stop
```

- 2 クラスタの1つのノード上で、セキュリティー鍵を再生成します。

```
phys-schost-1# /opt/bin/cacaoadm create-keys --force
```

- 3 セキュリティー鍵を再生成したノード上で共通エージェントコンテナ管理デーモンを再起動します。

```
phys-schost-1# /opt/bin/cacaoadm start
```

- 4 /etc/cacao/instances/default ディレクトリの tar ファイルを作成します。

```
phys-schost-1# cd /etc/cacao/instances/default
phys-schost-1# tar cf /tmp/SECURITY.tar security
```

- 5 /tmp/Security.tar ファイルを各クラスタノードにコピーします。

- 6 /tmp/SECURITY.tar ファイルをコピーした各ノード上で、セキュリティーファイルを解凍します。  
/etc/opt/SUNWcacao/ ディレクトリに既にセキュリティーファイルがある場合は、すべて上書きされます。  

```
phys-schost-2# cd /etc/cacao/instances/default
phys-schost-2# tar xf /tmp/SECURITY.tar
```
- 7 クラスタの各ノードから /tmp/SECURITY.tar ファイルを削除します。  
セキュリティーのリスクを避けるために tar ファイルの各コピーを削除する必要があります。  

```
phys-schost-1# rm /tmp/SECURITY.tar

phys-schost-2# rm /tmp/SECURITY.tar
```
- 8 すべてのノード上で共通エージェントコンテナ 管理デーモンを再起動します。  

```
phys-schost-1# /opt/bin/cacaoadm start
```
- 9 **Sun Cluster Manager** を再起動します。  

```
/usr/sbin/smcwebserver restart
```

## Sun Cluster Manager ソフトウェアの起動

Sun Cluster Manager グラフィカルユーザーインタフェース (Graphical User Interface, GUI) は、Sun Cluster ソフトウェアをさまざまな面から簡単に管理する方法を提供します。詳細については、Sun Cluster Manager のオンラインヘルプを参照してください。

クラスタを起動すると、Sun Java Web コンソールと共通エージェントコンテナの両方が起動します。Sun Java Web コンソールと共通エージェントコンテナが実行されていることを確認するには、この手順のすぐあとの障害追跡のセクションを参照してください。

### ▼ Sun Cluster Manager を起動する

この手順では、クラスタ上で Sun Cluster Manager を起動する方法を示します。

- 1 **Sun Cluster Manager** にアクセスするときに、クラスタノードの root のユーザー名とパスワードを使用するか、異なるユーザー名とパスワードを設定するかを決定します。

- クラスタノードのルートのユーザー名を使用して Sun Cluster Manager にアクセスする場合は、[手順 5](#)に進みます。
  - 別のユーザー名とパスワードを設定する場合は、[手順 3](#)に進んで Sun Cluster Manager ユーザーアカウントを設定します。
- 2 クラスタノード上にインストールするクラスタノード上でスーパーユーザーになります。
  - 3 **Sun Cluster Manager** 経由でクラスタにアクセスするためのユーザーアカウントを作成します。

`useradd(1M)` コマンドを使用して、ユーザーアカウントをシステムに追加します。root システムアカウントを使用しない場合、Sun Cluster Manager にアクセスするには、少なくとも1つのユーザーアカウントを設定する必要があります。Sun Cluster Manager のユーザーアカウントは、Sun Cluster Manager だけで使用されます。これらのアカウントは、Solaris OS システムのユーザーアカウントとの関連はありません。RBAC の役割を作成し、それをユーザーアカウントに割り当てる方法については、[55 ページの「Sun Cluster 管理権限プロファイルによる RBAC 役割の作成と割り当て」](#)を参照してください。

---

注-ノードにユーザーアカウントが設定されていない場合、そのユーザーはそのノードからは SunPlex Manager 経由でクラスタにアクセスできません。また、アクセス権を持っている別のクラスタノードからも、そのノードを管理することはできません。

---

- 4 (省略可能)追加するユーザーアカウントごとに [手順 3](#)を繰り返します。
- 5 管理コンソール、またはクラスタの外部に存在する他のマシンから、ブラウザを起動します。
- 6 ブラウザのディスクとメモリーキャッシュのサイズが、0 より大きな値に設定されていることを確認します。
- 7 ブラウザで **Java** および **Javascript** が有効になっていることを確認します。
- 8 ブラウザから、クラスタ内の任意のノード上にある **Sun Cluster Manager** のポートに接続します。  
デフォルトのポート番号は 6789 です。  
`https://node:6789/`
- 9 **Web** ブラウザにより提示されたすべての証明書を受け入れます。  
Java Web Console ログインページが表示されます。

- 10 Sun Cluster Manager にアクセスするユーザーのユーザー名とパスワードを入力します。
- 11 「Log In」 ボタンをクリックします。  
Java Web Console のアプリケーション起動ページが表示されます。
- 12 Systems カテゴリの下の Sun Cluster Manager リンクをクリックします。
- 13 Web ブラウザにより提示されたすべての追加の証明書を受け入れます。
- 14 Sun Cluster Manager に接続できない場合は、次のサブステップを実行して、Solaris のインストール中に制限されたネットワークプロファイルが選択されたかどうかを判別し、Java Web コンソールサービスへの外部アクセスを復元します。  
Solaris のインストール中に制限されたネットワークプロファイルを選択すると、Sun Java Web コンソールサービスの外部アクセスは制限されます。このネットワークは、Sun Cluster Manager GUI を使用するために必要です。
  - a. Java Web コンソールサービスが制限されているかどうかを調べます。

```
svcprop /system/webconsole:console | grep tcp_listen
```

tcp\_listen プロパティの値が true でない場合、Web コンソールサービスは制限されます。
  - b. Java Web Console サービスへの外部アクセスを復元します。

```
svccfg
svc:> select system/webconsole
svc:/system/webconsole> setprop options/tcp_listen=true
svc:/system/webconsole> quit
/usr/sbin/smcwebserver restart
```
  - c. サービスが使用可能になっていることを確認します。

```
netstat -a | grep 6789
```

サービスが使用可能な場合、コマンド出力により 6789 のエントリが返されません。これは Java Web コンソールへの接続に使用されるポート番号です。

**注意事項** この手順の実行後に Sun Cluster Manager に接続できない場合、`/usr/sbin/smcwebserver status` を入力して、Sun Java Web コンソールが実行されているかどうかを調べます。Sun Java Web コンソールが実行されていない場合、`/usr/sbin/smcwebserver start` を入力して手動で起動します。それでも Sun Cluster Manager に接続できない場合は、`usr/bin/cacoadm status` を入力して、共通

---

エージェントコンテナが実行されているかどうかを調べます。共通エージェントコンテナが実行されていない場合は、`/usr/sbin/cacoadm start` を入力して手動で起動します。



## Sun StorEdge Availability Suite または Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成

この付録では、Sun Cluster Geographic Edition を使用しないホストベースの複製の代替方法を説明します。Sun では、ホストベースの複製に Sun Cluster Geographic Edition を使用して、クラスタ内のホストベースの複製の構成と操作を簡素化することをお勧めします。90 ページの「[データ複製についての理解](#)」を参照してください。

この付録の例は、Sun StorageTek Availability Suite 3.1 または 3.2 ソフトウェアあるいは Sun StorageTek Availability Suite 4.0 ソフトウェアを使用してクラスタ間のホストベースのデータ複製を構成する方法を示しています。この例では、NFS アプリケーション用の完全なクラスタ構成を示し、個別のタスクの実行方法に関する詳細情報を提供します。すべてのタスクはグローバルクラスタの投票ノードで行われます。例には、ほかのアプリケーションやクラスタ構成に必要な手順がすべて含まれているわけではありません。

スーパーユーザーの代わりに役割に基づくアクセス制御 (RBAC) を使用してクラスタノードにアクセスする場合は、すべての Sun Cluster コマンドの承認を提供する RBAC の役割になることができるようにします。ユーザーがスーパーユーザーでない場合、一連のデータ複製手順には、次の Sun Cluster RBAC の承認が必要です。

- `solaris.cluster.modify`
- `solaris.cluster.admin`
- `solaris.cluster.read`

RBAC の役割についての詳細は、『[Solaris のシステム管理 \(セキュリティサービス\)](#)』を参照してください。各 Sun Cluster サブコマンドで必要となる RBAC の承認については、Sun Cluster のマニュアルページを参照してください。

## クラスタにおける Sun StorageTek Availability Suite ソフトウェアの理解

ここでは、耐障害性について紹介し、Sun StorageTek Availability Suite ソフトウェアが使用するデータ複製方式について説明します。

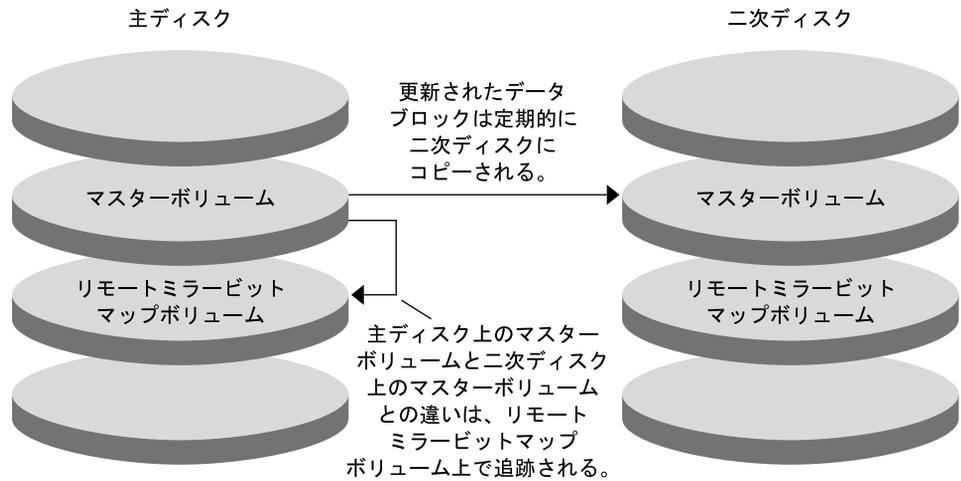
耐障害性は、主クラスタで障害が発生した場合に代替りのクラスタ上でアプリケーションを復元するシステムの機能です。災害耐性のベースは、データ複製とフェイルオーバーです。フェイルオーバーとは、主クラスタから二次クラスタへの、リソースグループまたはデバイスグループの自動再配置です。主クラスタに障害が発生した場合でも、アプリケーションとデータは二次クラスタで即座に使用できます。

### Sun StorageTek Availability Suite ソフトウェアが使用するデータ複製方式

この節では、Sun StorageTek Availability Suite が使用するリモートミラー複製方式とポイントインタイムスナップショット方式について説明します。このソフトウェアは、`sndradm (IRPC)` と `iiadm (III)` コマンドを使用してデータを複製します。

#### リモートミラー複製

[図 A-1](#) はリモートミラー複製を示しています。主ディスクのマスターボリュームのデータは、TCP/IP 接続を経由して二次ディスクのマスターボリュームに複製されます。リモートミラービットマップは、主ディスク上のマスターボリュームと、二次ディスク上のマスターボリュームの差分を追跡します。



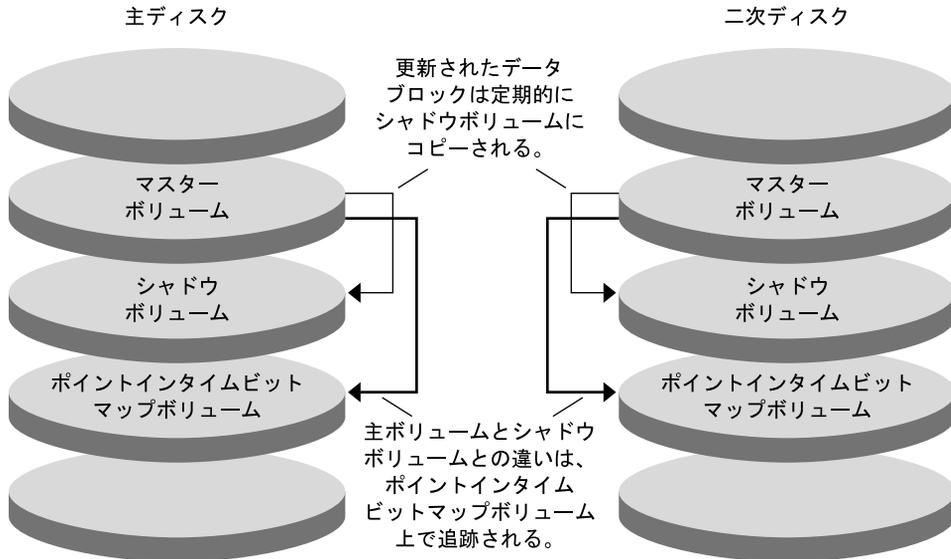
図A-1 リモートミラー複製

リモートミラー複製は、リアルタイムに同期で実行することも非同期で実行することもできます。各クラスタの各ボリュームセットはそれぞれ、同期複製または非同期複製に構成できます。

- 同期データ複製では、リモートボリュームが更新されるまで、書き込み操作は完了したとは確認されません。
- 非同期データ複製では、リモートボリュームが更新される前に書き込み操作が完了したと確認されます。非同期データ複製は、長い距離や低い帯域幅で大きな柔軟性を発揮します。

## ポイントインタイムスナップショット

図A-2は、ポイントインタイムスナップショットを示しています。各ディスクのマスターボリュームのデータは、同じディスクのシャドウボリュームにコピーされます。ポイントインタイムビットマップは、マスターボリュームとシャドウボリューム間の違いを追跡調査します。データがシャドウボリュームにコピーされると、ポイントインタイムビットマップはリセットされます。



図A-2 ポイントインタイムスナップショット

### 構成例での複製

図A-3に、この構成例でミラー複製とポイントインタイムスナップショットがどのように使用されているかを示します。

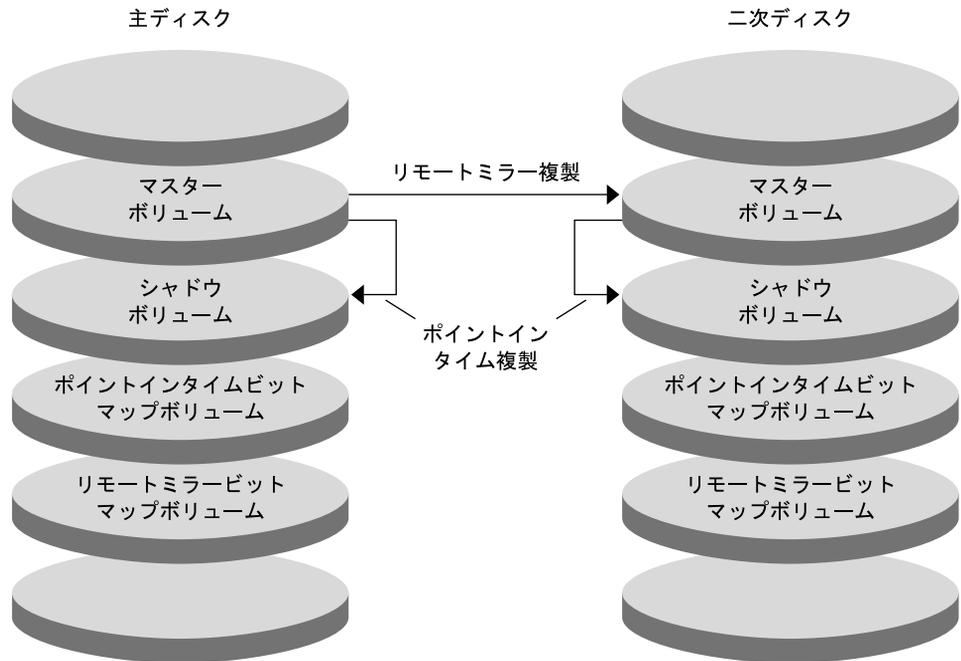


図 A-3 構成例での複製

## クラスタ間でホストベースのデータ複製を構成するためのガイドライン

この節では、クラスタ間のデータ複製の構成ガイドラインを提供します。また、複製リソースグループとアプリケーションリソースグループの構成のコツも紹介します。これらのガイドラインは、クラスタのデータ複製を構成する際に使用してください。

この節では、次の項目について説明します。

- [386 ページの「複製リソースグループの構成」](#)
- [387 ページの「アプリケーションリソースグループの構成」](#)
  - [387 ページの「フェイルオーバーアプリケーション向けのリソースグループの構成」](#)
  - [389 ページの「スケーラブルアプリケーション向けのリソースグループの構成」](#)
- [390 ページの「フェイルオーバーの管理のガイドライン」](#)

## 複製リソースグループの構成

複製リソースグループは、Sun StorageTek Availability Suite ソフトウェアが制御するデバイスグループと論理ホスト名リソースを相互に関連付けます。複製リソースグループには、次の特徴があります。

- フェイルオーバーリソースグループである  
フェイルオーバーリソースは、常に単一のノード上で実行されます。フェイルオーバーが発生すると、フェイルオーバーリソースがフェイルオーバーに加わります。
- 論理ホスト名リソースを持つ  
論理ホスト名は、主クラスタがホストでなければなりません。フェイルオーバーの後は、論理ホスト名は二次クラスタがホストでなければなりません。ドメインネームシステム (Domain Name System、DNS) は、論理ホスト名とクラスタを関連付けるために使用されます。
- HAStoragePlus リソースを持つ  
HAStoragePlus リソースは、複製リソースグループがスイッチオーバーまたはフェイルオーバーしたときに、デバイスグループをフェイルオーバーします。Sun Cluster ソフトウェアはまた、デバイスグループがスイッチオーバーしたときに、複製リソースグループをフェイルオーバーします。このように複製リソースグループとデバイスグループは常に結び付き、同じノードから制御されません。

HAStoragePlus リソース内に次の拡張プロパティを定義する必要があります。

- *GlobalDevicePaths*。この拡張プロパティは、ボリュームが属するデバイスグループを定義します。
- *AffinityOn property = True*。この拡張プロパティは、複製リソースグループがスイッチオーバーまたはフェイルオーバーしたときに、デバイスグループをスイッチオーバーまたはフェイルオーバーします。この機能はアフィニティースイッチオーバーと呼ばれます。
- *ZPoolsSearchDir*。この拡張プロパティは、ZFS ファイルシステムを使用するために必要です。

HAStoragePlus についての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 結び付いているデバイスグループに `-stor-rg` を付けた名前になる  
たとえば、`devgrp-stor-rg` などです。
- 主クラスタと二次クラスタでオンラインになる

## アプリケーションリソースグループの構成

高可用性を実現するためには、アプリケーションはアプリケーションリソースグループのリソースとして管理される必要があります。アプリケーションリソースグループは、フェイルオーバーアプリケーションまたはスケラブルアプリケーション向けに構成できます。

主クラスタ上に構成したアプリケーションリソースとアプリケーションリソースグループは、二次クラスタ上でも構成される必要があります。また、アプリケーションリソースがアクセスするデータは、二次クラスタに複製する必要があります。

この節では、次のアプリケーションリソースグループを構成するためのガイドラインを紹介します。

- [387 ページの「フェイルオーバーアプリケーション向けのリソースグループの構成」](#)
- [389 ページの「スケラブルアプリケーション向けのリソースグループの構成」](#)

## フェイルオーバーアプリケーション向けのリソースグループの構成

フェイルオーバーアプリケーションでは、1つのアプリケーションが1度に1ノード上で動作します。ノードで障害が発生すると、アプリケーションは同じクラスタ内の別のノードにフェイルオーバーします。フェイルオーバーアプリケーション向けリソースグループは、以下の特徴を持っていないければなりません。

- アプリケーションリソースグループがスイッチオーバーまたはフェイルオーバーされた場合、HAStoragePlus リソースにデバイスグループをフェイルオーバーさせる

デバイスグループは、複製リソースグループとアプリケーションリソースグループに結び付けられています。したがって、アプリケーションリソースグループがフェイルオーバーすると、デバイスグループと複製リソースグループもフェイルオーバーします。アプリケーションリソースグループ、複製リソースグループおよびデバイスグループは、同じノードによって制御されます。

ただし、デバイスグループや複製リソースグループがフェイルオーバーしても、アプリケーションリソースグループはフェイルオーバーを行いません。

- アプリケーションデータがグローバルマウントされている場合は、アプリケーションリソースグループにHAStoragePlus リソースを必ず入れなければならないわけではありませんが、入れることをお勧めします。
- アプリケーションデータがローカルマウントされている場合は、アプリケーションリソースグループにHAStoragePlus リソースを必ず入れなければなりません。

HAStoragePlus リソースがないと、アプリケーションリソースグループがフェイルオーバーしても、複製リソースグループとデバイスグループのフェイルオーバーは行われません。フェイルオーバーの後は、アプリケーションリソースグループ、複製リソースグループおよびデバイスグループは同じノードに制御されません。

HAStoragePlus についての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 主クラスタでオンライン、二次クラスタでオフラインとなる  
二次クラスタが主クラスタをテイクオーバーした場合は、二次クラスタ上のアプリケーションリソースグループをオンラインにします。

図 A-4 に、フェイルオーバーアプリケーションでのアプリケーションリソースグループと複製リソースグループの構成を示します。

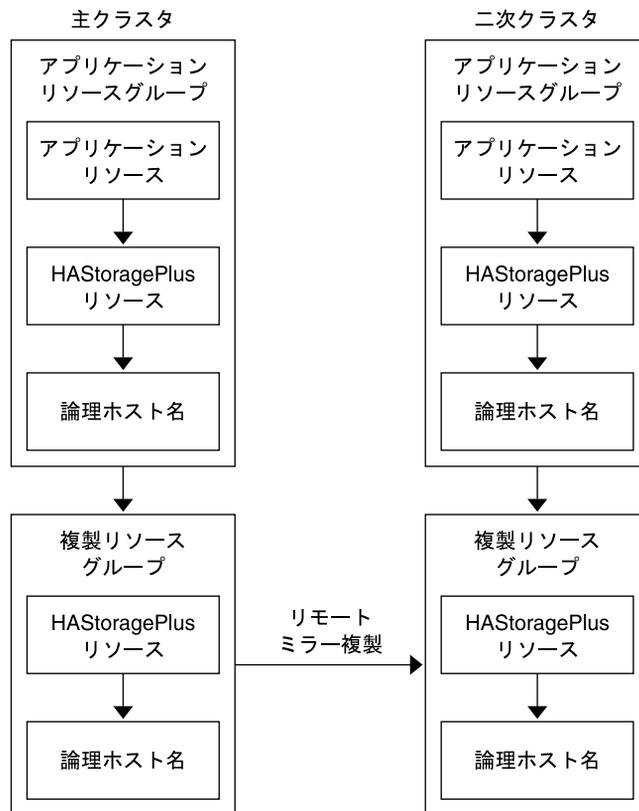


図 A-4 フェイルオーバーアプリケーションでのリソースグループの構成

## スケーラブルアプリケーション向けのリソースグループの構成

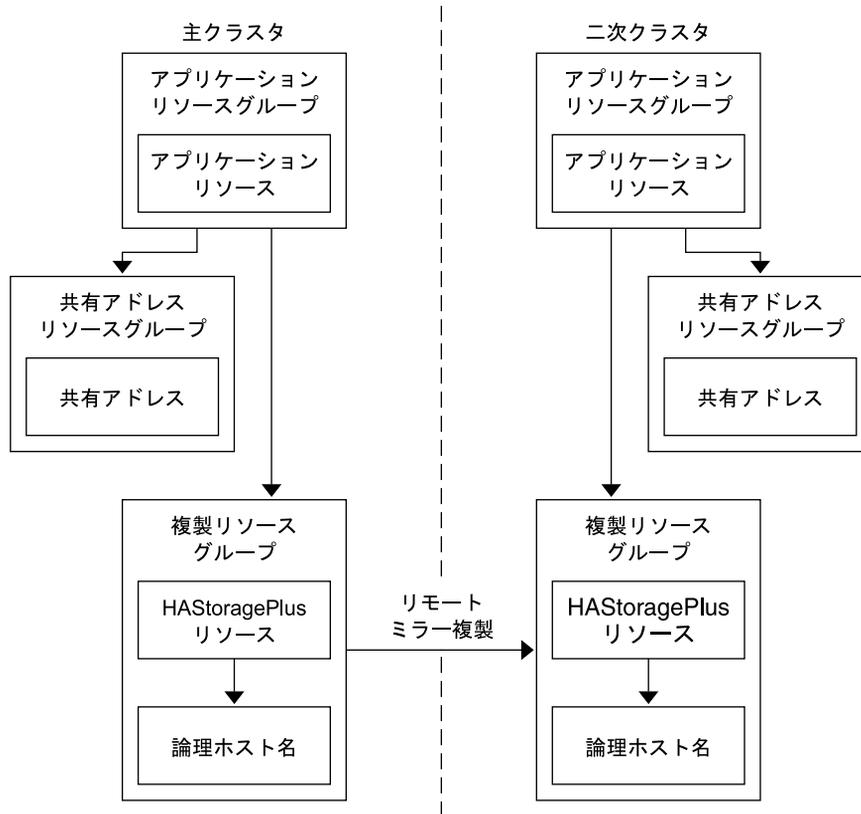
スケーラブルアプリケーションでは、アプリケーションは複数のノードで実行されて、1つの論理サービスを作成します。スケーラブルアプリケーションを実行しているノードで障害が発生しても、フェイルオーバーは起こりません。アプリケーションは別のノードで引き続き実行されます。

スケーラブルアプリケーションをアプリケーションリソースグループのリソースとして管理している場合は、アプリケーションリソースグループをデバイスグループと結び付ける必要はありません。したがって、アプリケーションリソースグループ向けに HAStoragePlus リソースを作成する必要はありません。

スケーラブルアプリケーション向けリソースグループは、以下の特徴を持っていないければなりません。

- 共有アドレスのリソースグループに依存する  
共有アドレスは、受信データを配信するためにスケーラブルアプリケーションを実行するノードで使用されます。
- 主クラスタでオンライン、二次クラスタでオフラインとなる

図 A-5 に、スケーラブルアプリケーションでのリソースグループの構成を示します。



図A-5 スケーラブルアプリケーションでのリソースグループの構成

## フェイルオーバーの管理のガイドライン

主クラスタで障害が発生した場合、できるだけ速やかにアプリケーションを二次クラスタにスイッチオーバーする必要があります。二次クラスタがテイクオーバーできるようにするには、DNSを更新する必要があります。

DNSは、クライアントをアプリケーションの論理ホスト名に関連付けます。フェイルオーバーの後、主クラスタへのDNSマッピングを削除し、二次クラスタへのDNSマッピングを作成します。図A-6に、DNSがどのようにクライアントをクラスタにマッピングするかを示します。

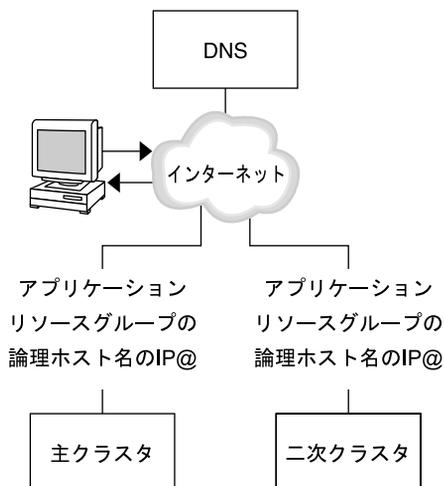


図 A-6 クライアントからクラスタへの DNS マッピング

DNS を更新するには、`nsupdate` コマンドを使用します。詳細は、[nsupdate\(1M\)](#) のマニュアルページを参照してください。フェイルオーバーの管理方法の例については、[420 ページの「フェイルオーバーの管理方法の例」](#)を参照してください。

修復後は、主クラスタをオンラインに戻せます。元の主クラスタにスイッチバックするには、次の手順を実行します。

1. 主クラスタと二次クラスタを同期させ、主ボリュームが最新のものであることを確認します。
2. クライアントが主クラスタのアプリケーションにアクセスできるように、DNS を更新します。

## 作業マップ: データ複製の構成例

[表 A-1](#) に、Sun StorageTek Availability Suite ソフトウェアを使用して NFS アプリケーション向けにどのようにデータ複製を構成するかを示すこの例での作業を示します。

表 A-1 作業マップ: データ複製の構成例

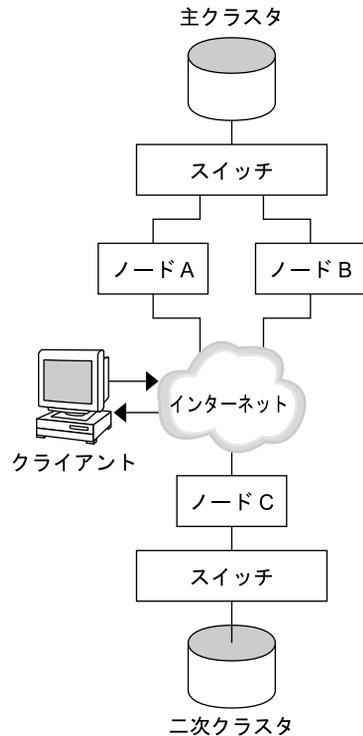
作業	参照先
1. クラスタを接続およびインストールする	<a href="#">392 ページの「クラスタの接続とインストール」</a>

表 A-1 作業マップ: データ複製の構成例 (続き)

作業	参照先
2. 主クラスタと二次クラスタで、デバイスグループ、NFS アプリケーション用のファイルシステム、およびリソースグループを構成する	394 ページの「デバイスグループとリソースグループの構成例」
3. 主クラスタと二次クラスタでデータ複製を有効にする	409 ページの「主クラスタで複製を有効にする」 412 ページの「二次クラスタで複製を有効にする」
4. データ複製を実行する	413 ページの「リモートミラー複製を実行する」 415 ページの「ポイントインタイムスナップショットを実行する」
5. データ複製の構成を確認する	417 ページの「複製が正しく構成されていることを確認する」

## クラスタの接続とインストール

図 A-7 に、構成例で使用するクラスタ構成を示します。構成例の二次クラスタにはノードが 1 つ含まれていますが、これ以外のクラスタ構成も使用できます。



図A-7 クラスタ構成例

表A-2に、構成例で必要となるハードウェアとソフトウェアをまとめました。Solaris OS、Sun Cluster ソフトウェア、ボリューム管理ソフトウェアは、Sun StorageTek Availability Suite ソフトウェアとパッチをインストールする前にクラスタノードにインストールしてください。

表A-2 必要なハードウェアとソフトウェア

ハードウェアまたはソフトウェア	要件
ノードハードウェア	Sun StorageTek Availability Suite ソフトウェアは、Solaris OS を使用するすべてのサーバー上でサポートされます。  使用するハードウェアについては、『 <a href="#">Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS</a> 』を参照してください。
ディスク容量	約 15M バイト

表 A-2 必要なハードウェアとソフトウェア (続き)

ハードウェアまたはソフトウェア	要件
Solaris OS	Sun Cluster ソフトウェアがサポートする Solaris OS のリリース。 すべてのノードが同じバージョンの Solaris OS を使用する必要があります。  インストールについては、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』を参照してください。
Sun Cluster ソフトウェア	Sun Cluster 3.2 2/08 ソフトウェア  インストールについては、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』を参照してください。
ボリューム管理ソフトウェア	Solaris ボリュームマネージャーソフトウェアまたは Veritas Volume Manager (VxVM) ソフトウェア  すべてのノードで、同じバージョンのボリューム管理ソフトウェアを使用する。  インストールについては、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の第 4 章「Solaris ボリュームマネージャーソフトウェアの構成」および『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の第 5 章「Veritas Volume Manager をインストールして構成する」を参照してください。
Sun StorageTek Availability Suite ソフトウェア	ソフトウェアのインストール方法については、使用しているリリースの Sun StorageTek Availability Suite または Sun StorageTek Availability Suite ソフトウェアのインストールマニュアルを参照してください。  <ul style="list-style-type: none"> <li>■ Sun StorEdge Availability Suite 3.1 - Sun StorEdge Availability のマニュアル</li> <li>■ Sun StorEdge Availability Suite 3.2 - Sun StorEdge Availability のマニュアル</li> <li>■ Sun StorageTek Availability Suite 4.0 - Sun StorageTek Availability のマニュアル</li> </ul>
Sun StorageTek Availability Suite ソフトウェアパッチ	最新のパッチについては、 <a href="http://www.sunsolve.com">http://www.sunsolve.com</a> を参照してください。

## デバイスグループとリソースグループの構成例

この節では、NFS アプリケーション向けにディスクデバイスグループとリソースグループをどのように構成するかを説明します。追加情報については、386 ページの「複製リソースグループの構成」および 387 ページの「アプリケーションリソースグループの構成」を参照してください。

ここでは、次の手順について説明します。

- 396 ページの「主クラスタでデバイスグループを構成する」
- 397 ページの「二次クラスタでデバイスグループを構成する」
- 398 ページの「主クラスタのファイルシステムを NFS アプリケーション向けに構成する」
- 399 ページの「二次クラスタのファイルシステムを NFS アプリケーション向けに構成する」
- 401 ページの「主クラスタで複製リソースグループを作成する」
- 402 ページの「二次クラスタで複製リソースグループを作成する」
- 404 ページの「主クラスタで NFS アプリケーションリソースグループを作成する」
- 406 ページの「二次クラスタで NFS アプリケーションリソースグループを作成する」
- 417 ページの「複製が正しく構成されていることを確認する」

構成例のために作成されたグループとリソースの名前を次の表に示します。

表 A-3 構成例内のグループとリソースのまとめ

グループまたはリソース	名前	説明
デバイスグループ	devgrp	デバイスグループ
複製リソースグループ とリソース	devgrp-stor-rg	複製リソースグループ
	lhost-reprg-prim、lhost-reprg-sec	主クラスタと二次クラスタの複製リソースグループの論理ホスト名
	devgrp-stor	複製リソースグループの HAStoragePlus リソース
アプリケーションリ ソースグループとリ ソース	nfs-rg	アプリケーションリソースグループ
	lhost-nfsrg-prim、lhost-nfsrg-sec	主クラスタと二次クラスタのアプリケーションリソースグループの論理ホスト名
	nfs-dg-rs	アプリケーションの HAStoragePlus リソース
	nfs-rs	NFS リソース

devgrp-stor-rg 以外のグループとリソースの名前は一例で、必要に応じて変更可能です。複製リソースグループは、*devicegroupname-stor-rg* というフォーマットでなければなりません。

この構成例では VxVM ソフトウェアを使用しています。Solaris ボリュームマネージャーソフトウェアについては、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の第 4 章「Solaris ボリュームマネージャーソフトウェアの構成」を参照してください。

デバイスグループで作成済みのボリュームを下図に示します。

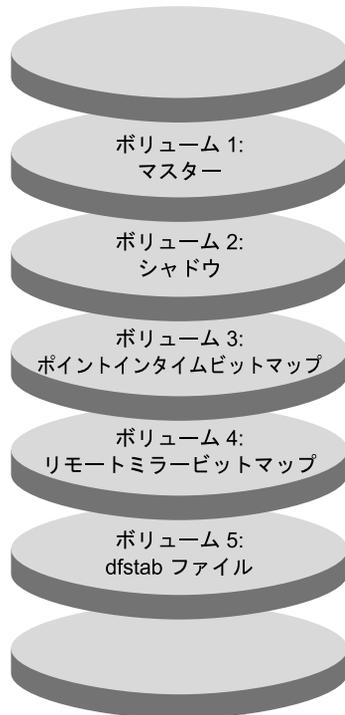


図 A-8 デバイスグループのボリューム

---

注-この手順で定義されたボリュームに、シリンダ 0 などのディスクラベルのプライベート領域を含めてはいけません。VxVM ソフトウェアは、この制限を自動管理します。

---

## ▼ 主クラスタでデバイスグループを構成する

始める前に 次の作業を完成していることを確認してください。

- 次の節のガイドラインと要件を確認します。

- 382 ページの「クラスタにおける Sun StorageTek Availability Suite ソフトウェアの理解」
  - 385 ページの「クラスタ間でホストベースのデータ複製を構成するためのガイドライン」
  - 392 ページの「クラスタの接続とインストール」で説明されているように、主クラスタおよび二次クラスタを設定します。
- 1 nodeA にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になりますしてアクセスします。  
nodeA は、主クラスタの最初のノードです。どのノードが nodeA であるかを確認するには、[図 A-7](#) を参照してください。
  - 2 nodeA でボリューム 1 vol01 からボリューム 4 vol04 を含むディスクグループを作成します。  
VxVM ソフトウェアを使用したディスクグループの構成については、『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』の第 5 章「Veritas Volume Manager をインストールして構成する」を参照してください。
  - 3 ディスクグループを構成して、デバイスグループを作成します。  

```
nodeA# cldevicegroup create -t vxvm -n nodeA nodeB devgrp
```

  
デバイスグループは devgrp と呼ばれます。
  - 4 デバイスグループのファイルシステムを作成します。  

```
nodeA# newfs /dev/vx/rdisk/devgrp/vol01 < /dev/null
nodeA# newfs /dev/vx/rdisk/devgrp/vol02 < /dev/null
```

  
vol03 と vol04 は raw ボリュームとして使用されるため、ファイルシステムは必要ありません。

次の手順 [397 ページの「二次クラスタでデバイスグループを構成する」](#)に進みます。

## ▼ 二次クラスタでデバイスグループを構成する

始める前に [手順396 ページの「主クラスタでデバイスグループを構成する」](#)を完了します。

- 1 nodeC にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になりますしてアクセスします。
- 2 nodeC でボリューム 1 vol01 からボリューム 4 vol04 までの 4 つのボリュームを含むディスクグループを作成します。
- 3 ディスクグループを構成して、デバイスグループを作成します。  

```
nodeC# cldevicegroup create -t vxvm -n nodeC devgrp
```

デバイスグループは `devgrp` という名前です。

- 4 デバイスグループのファイルシステムを作成します。

```
nodeC# newfs /dev/vx/rdisk/devgrp/vol01 < /dev/null
nodeC# newfs /dev/vx/rdisk/devgrp/vol02 < /dev/null
```

`vol03` と `vol04` は raw ボリュームとして使用されるため、ファイルシステムは必要ありません。

次の手順 398 ページの「主クラスタのファイルシステムを NFS アプリケーション向けに構成する」に進みます。

## ▼ 主クラスタのファイルシステムを NFS アプリケーション向けに構成する

始める前に 手順397 ページの「二次クラスタでデバイスグループを構成する」を完了します。

- 1 nodeA および nodeB で、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になります。
- 2 nodeA と nodeB で、**NFS** ファイルシステム向けのマウントポイントディレクトリを作成します。  
たとえば、次のように使用します。  

```
nodeA# mkdir /global/mountpoint
```
- 3 nodeA と nodeB で、マウントポイントに自動でマウントされるようにマスターボリュームを構成します。  
nodeA と nodeB の `/etc/vfstab` ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは 1 行で記述してください。  

```
/dev/vx/dsk/devgrp/vol01 /dev/vx/rdisk/devgrp/vol01 \
/global/mountpoint ufs 3 no global,logging
```

  
デバイスグループで使用されているボリューム名とボリューム番号を確認するには、[図 A-8](#) を参照してください。
- 4 nodeA で、**Sun Cluster HA for NFS** データサービスが使用するファイルのシステム情報向けのボリュームを作成します。  

```
nodeA# vxassist -g devgrp make vol05 120m disk1
```

  
ボリューム 5 `vol05` には Sun Cluster HA for NFS データサービスが使用するファイルシステム情報が含まれています。
- 5 nodeA で、デバイスグループと **Sun Cluster** ソフトウェアを再同期化します。  

```
nodeA# cldevicegroup sync devgrp
```

- 6 nodeA で、vol05 用のファイルシステムを作成します。  
nodeA# `newfs /dev/vx/rdisk/devgrp/vol05`
- 7 nodeA と nodeB で、vol05 のマウントポイントを作成します。  
次の例では、マウントポイント /global/etc を作成しています。  
nodeA# `mkdir /global/etc`
- 8 nodeA と nodeB で、マウントポイントに自動でマウントされるように vol05 を構成します。  
nodeA と nodeB の /etc/vfstab ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは 1 行で記述してください。  
`/dev/vx/dsk/devgrp/vol05 /dev/vx/rdisk/devgrp/vol05 \  
/global/etc ufs 3 yes global,logging`
- 9 nodeA に vol05 をマウントします。  
nodeA# `mount /global/etc`
- 10 vol05 がリモートシステムからアクセスできるようにします。
  - a. nodeA に /global/etc/SUNW.nfs というディレクトリを作成します。  
nodeA# `mkdir -p /global/etc/SUNW.nfs`
  - b. nodeA に /global/etc/SUNW.nfs/dfstab.nfs-rs ファイルを作成します。  
nodeA# `touch /global/etc/SUNW.nfs/dfstab.nfs-rs`
  - c. nodeA の /global/etc/SUNW.nfs/dfstab.nfs-rs ファイルに次の行を追加します。  
`share -F nfs -o rw -d "HA NFS" /global/mountpoint`

次の手順 [399 ページの「二次クラスタのファイルシステムを NFS アプリケーション向けに構成する」](#)に進みます。

## ▼ 二次クラスタのファイルシステムを **NFS** アプリケーション向けに構成する

始める前に 手順[398 ページの「主クラスタのファイルシステムを NFS アプリケーション向けに構成する」](#)を完了します。

- 1 nodeC で、スーパーユーザーになるか、RBAC の承認 solaris.cluster.admin を提供する役割になります。

- 2 nodeC で、**NFS** ファイルシステム向けのマウントポイントディレクトリを作成します。  
たとえば、次のように使用します。  

```
nodeC# mkdir /global/mountpoint
```
- 3 nodeC で、マウントポイントに自動でマウントされるようにマスターボリュームを構成します。  
nodeC の `/etc/vfstab` ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは 1 行で記述してください。  

```
/dev/vx/dsk/devgrp/vol01 /dev/vx/rdisk/devgrp/vol01 \
/global/mountpoint ufs 3 no global,logging
```
- 4 nodeC で、**Sun Cluster HA for NFS** データサービスが使用するファイルのシステム情報向けのボリュームを作成します。  

```
nodeC# vxassist -g devgrp make vol05 120m disk1
```

ボリューム 5 vol05 には Sun Cluster HA for NFS データサービスが使用するファイルシステム情報が含まれています。
- 5 nodeC で、デバイスグループと **Sun Cluster** ソフトウェアを再同期化します。  

```
nodeC# cldevicegroup sync devgrp
```
- 6 nodeC で、vol05 用のファイルシステムを作成します。  

```
nodeC# newfs /dev/vx/rdisk/devgrp/vol05
```
- 7 nodeC で、vol05 用のマウントポイントを作成します。  
次の例では、マウントポイント `/global/etc` を作成しています。  

```
nodeC# mkdir /global/etc
```
- 8 nodeC で、vol05 がマウントポイントで自動的にマウントされるよう構成します。  
nodeC の `/etc/vfstab` ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは 1 行で記述してください。  

```
/dev/vx/dsk/devgrp/vol05 /dev/vx/rdisk/devgrp/vol05 \
/global/etc ufs 3 yes global,logging
```
- 9 nodeC に vol05 をマウントします。  

```
nodeC# mount /global/etc
```
- 10 vol05 がリモートシステムからアクセスできるようにします。
  - a. nodeC に `/global/etc/SUNW.nfs` というディレクトリを作成します。  

```
nodeC# mkdir -p /global/etc/SUNW.nfs
```

- b. nodeC に /global/etc/SUNW.nfs/dfstab.nfs-rs ファイルを作成します。  
nodeC# **touch /global/etc/SUNW.nfs/dfstab.nfs-rs**
- c. nodeC の /global/etc/SUNW.nfs/dfstab.nfs-rs ファイルに次の行を追加します。  
**share -F nfs -o rw -d "HA NFS" /global/mountpoint**

次の手順 401 ページの「主クラスタで複製リソースグループを作成する」に進みます。

## ▼ 主クラスタで複製リソースグループを作成する

始める前に 手順399 ページの「二次クラスタのファイルシステムを NFS アプリケーション向けに構成する」を完了します。

- 1 nodeA にスーパーユーザーになるか、RBAC の承認 solaris.cluster.modify、solaris.cluster.admin、および solaris.cluster.read を提供する役割になりますしてアクセスします。
- 2 SUNW.HAStoragePlus というリソースタイプを登録します。  
nodeA# **clresourcetype register SUNW.HAStoragePlus**
- 3 デバイスグループの複製リソースグループを作成します。  
nodeA# **clresourcegroup create -n nodeA,nodeB devgrp-stor-rg**  
  
-n nodeA,nodeB クラスタノード nodeA および nodeB が複製リソースグループをマスターできることを指定します。  
  
devgrp-stor-rg 複製リソースグループの名前。この名前で、devgrp はデバイスグループの名前を指定します。
- 4 複製リソースグループに SUNW.HAStoragePlus リソースを追加します。  
nodeA# **clresource create -g devgrp-stor-rg -t SUNW.HAStoragePlus \**  
**-p GlobalDevicePaths=devgrp \**  
**-p AffinityOn=True \**  
**devgrp-stor**  
  
-g リソースを追加するリソースグループを指定します。  
  
-p GlobalDevicePaths= Sun StorageTek Availability Suite ソフトウェアが依存する拡張プロパティを指定します。  
  
-p AffinityOn=True SUNW.HAStoragePlus リソースが、-x GlobalDevicePaths= で定義されたグローバルデバイスおよびクラスタファイルシステムに対して、アフィニティスイッチオーバーを実行することを指定します。したがって、複製リソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#)のマニュアルページを参照してください。

- 5 論理ホスト名リソースを複製リソースグループに追加します。

```
nodeA# clreslogicalhostname create -g devgrp-stor-rg lhost-reprg-prim
```

主クラスタ上の複製リソースグループの論理ホスト名は `lhost-reprg-prim` です。

- 6 リソースを有効にし、リソースグループを管理し、リソースグループをオンラインにします。

```
nodeA# clresourcegroup online -e -M -n nodeA devgrp-stor-rg
```

-e 関連付けられたリソースを有効にします。

-M リソースグループを管理状態にします。

-n リソースグループをオンラインにするノードを指定します。

- 7 リソースグループがオンラインであることを確認します。

```
nodeA# clresourcegroup status devgrp-stor-rg
```

リソースグループの状態フィールドを調べ、複製リソースグループが `nodeA` でオンラインとなっていることを確認します。

次の手順 [402](#) ページの「二次クラスタで複製リソースグループを作成する」に進みます。

## ▼ 二次クラスタで複製リソースグループを作成する

始める前に 手順 [401](#) ページの「主クラスタで複製リソースグループを作成する」を完了します。

- 1 `nodeC` にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割になりますしてアクセスします。
- 2 `SUNW.HAStoragePlus` というリソースタイプを登録します。  

```
nodeC# clresourcetype register SUNW.HAStoragePlus
```
- 3 デバイスグループの複製リソースグループを作成します。  

```
nodeC# clresourcegroup create -n nodeC devgrp-stor-rg
```

`create` リソースグループを作成します。

`-n` リソースグループのノードリストを指定します。

`devgrp` デバイスグループの名前。

`devgrp-stor-rg` 複製リソースグループの名前。

#### 4 複製リソースグループに SUNW.HAStoragePlus リソースを追加します。

```
nodeC# clresource create \
-t SUNW.HAStoragePlus \
-p GlobalDevicePaths=devgrp \
-p AffinityOn=True \
devgrp-stor
```

create リソースを作成します。

-t リソースタイプを指定します。

**-p GlobalDevicePaths=** Sun StorageTek Availability Suite ソフトウェアが依存する拡張プロパティを指定します。

**-p AffinityOn=True** SUNW.HAStoragePlus リソースが、`-x GlobalDevicePaths=` で定義されたグローバルデバイスおよびクラスタファイルシステムに対して、アフィニティスイッチオーバーを実行することを指定します。したがって、複製リソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

devgrp-stor 複製リソースグループの HAStoragePlus リソース

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

#### 5 論理ホスト名リソースを複製リソースグループに追加します。

```
nodeC# cllogicalhostname create -g devgrp-stor-rg lhost-reprg-sec
```

主クラスタ上の複製リソースグループの論理ホスト名は `lhost-reprg-sec` です。

#### 6 リソースを有効にし、リソースグループを管理し、リソースグループをオンラインにします。

```
nodeC# clresourcegroup online -e -M -n nodeC devgrp-stor-rg
```

online オンラインにします。

-e 関連付けられたリソースを有効にします。

-M リソースグループを管理状態にします。

-n リソースグループをオンラインにするノードを指定します。

#### 7 リソースグループがオンラインであることを確認します。

```
nodeC# clresourcegroup status devgrp-stor-rg
```

リソースグループの状態フィールドを調べ、複製リソースグループが `nodeC` でオンラインとなっていることを確認します。

次の手順 [404 ページの「主クラスタで NFS アプリケーションリソースグループを作成する」](#)に進みます。

## ▼ 主クラスタで NFS アプリケーションリソースグループを作成する

この手順では、アプリケーションリソースグループを NFS に対して作成する方法を説明します。この手順はこのアプリケーションに固有で、別の種類のアプリケーションには使用できません。

始める前に 手順 [402 ページの「二次クラスタで複製リソースグループを作成する」](#) を完了します。

1 nodeA にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割になりますしてアクセスします。

2 SUNW.nfs をリソースタイプとして登録します。

```
nodeA# clresourcetype register SUNW.nfs
```

3 SUNW.HASStoragePlus をリソースタイプとして登録していない場合は、登録します。

```
nodeA# clresourcetype register SUNW.HASStoragePlus
```

4 デバイスグループ `devgrp` のアプリケーションリソースグループを作成します。

```
nodeA# clresourcegroup create \
-p Pathprefix=/global/etc \
-p Auto_start_on_new_cluster=False \
-p RG_dependencies=devgrp-stor-rg \
nfs-rg
```

Pathprefix=/global/etc

グループのリソースが管理ファイルを書き込むディレクトリを指定します。

Auto\_start\_on\_new\_cluster=False

アプリケーションリソースグループが自動的に起動しないように指定します。

RG\_dependencies=devgrp-stor-rg

アプリケーションリソースグループが依存するリソースグループを指定します。この例では、アプリケーションリソースグループは複製リソースグループ `devgrp-stor-rg` に依存しています。

アプリケーションリソースグループが新しい主ノードにスイッチオーバーすると、複製リソースグループが自動的にスイッチオーバーします。ただし、複製リソースグループが新しい主ノードにスイッチオーバーした場合は、アプリケーションリソースグループを手動でスイッチオーバーする必要があります。

nfs-rg  
アプリケーションリソースグループの名前。

- 5 アプリケーションリソースグループに SUNW.HAStoragePlus リソースを追加します。

```
nodeA# clresource create -g nfs-rg \
-t SUNW.HAStoragePlus \
-p FileSystemMountPoints=/global/mountpoint \
-p AffinityOn=True \
nfs-dg-rs
```

create  
リソースを作成します。

-g  
リソースを追加するリソースグループを指定します。

-t SUNW.HAStoragePlus  
リソースのタイプに SUNW.HAStoragePlus を指定します。

-p FileSystemMountPoints=/global/  
ファイルシステムのマウントポイントがグローバルであることを指定します。

-p AffinityOn=True  
アプリケーションリソースが -p GlobalDevicePaths= で定義されたグローバルデバイスとクラスタファイルシステム向けにアフィニティスイッチオーバーを実行するように指定します。したがって、アプリケーションリソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

nfs-dg-rs  
NFS アプリケーション向けの HAStoragePlus リソースの名前。

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 6 論理ホスト名リソースをアプリケーションリソースグループに追加します。

```
nodeA# clreslogicalhostname create -g nfs-rg \
lhost-nfsrg-prim
```

主クラスタ上のアプリケーションリソースグループの論理ホスト名は lhost-nfsrg-prim です。

- 7 リソースを有効にし、アプリケーションリソースグループを管理し、アプリケーションリソースグループをオンラインにします。

- a. NFS アプリケーション向けの HAStoragePlus リソースを有効にします。

```
nodeA# clresource enable nfs-rs
```

- b. nodeA でアプリケーションリソースグループをオンラインにします。

```
nodeA# clresourcegroup online -e -M -n nodeA nfs-rg
```

online リソースグループをオンラインにします。

-e 関連付けられたリソースを有効にします。

-M リソースグループを管理状態にします。

-n リソースグループをオンラインにするノードを指定します。

nfs-rg リソースグループの名前。

- 8 アプリケーションリソースグループがオンラインであることを確認します。

```
nodeA# clresourcegroup status
```

アプリケーションリソースグループの状態フィールドを調べ、複製リソースグループが nodeA と nodeB でオンラインとなっているかどうかを調べます。

次の手順 [406 ページの「二次クラスタで NFS アプリケーションリソースグループを作成する」](#)に進みます。

## ▼ 二次クラスタで NFS アプリケーションリソースグループを作成する

始める前に 手順[404 ページの「主クラスタで NFS アプリケーションリソースグループを作成する」](#)を完了します。

- 1 nodeC にスーパーユーザーになるか、RBAC の承認 solaris.cluster.modify、solaris.cluster.admin、および solaris.cluster.read を提供する役割になりますしてアクセスします。

- 2 SUNW.nfs をリソースタイプとして登録します。

```
nodeC# clresourcetype register SUNW.nfs
```

- 3 SUNW.HAStoragePlus をリソースタイプとして登録していない場合は、登録します。

```
nodeC# clresourcetype register SUNW.HAStoragePlus
```

- 4 デバイスグループのアプリケーションリソースグループを作成します。

```
nodeC# clresourcegroup create \
-p Pathprefix=/global/etc \
-p Auto_start_on_new_cluster=False \
-p RG_dependencies=devgrp-stor-rg \
nfs-rg
```

**create**

リソースグループを作成します。

**-p**

リソースグループのプロパティを指定します。

**Pathprefix=/global/etc**

グループのリソースが管理ファイルを書き込むディレクトリを指定します。

**Auto\_start\_on\_new\_cluster=False**

アプリケーションリソースグループが自動的に起動しないように指定します。

**RG\_dependencies=devgrp-stor-rg**

アプリケーションリソースグループが依存するリソースグループを指定します。この例では、アプリケーションリソースグループは複製リソースグループに依存しています。

アプリケーションリソースグループが新しい主ノードにスイッチオーバーすると、複製リソースグループが自動的にスイッチオーバーします。ただし、複製リソースグループが新しい主ノードにスイッチオーバーした場合は、アプリケーションリソースグループを手動でスイッチオーバーする必要があります。

**nfs-rg**

アプリケーションリソースグループの名前。

## 5 アプリケーションリソースグループに SUNW.HASStoragePlus リソースを追加します。

```
nodeC# clresource create -g nfs-rg \
-t SUNW.HASStoragePlus \
-p FileSystemMountPoints=/global/mountpoint \
-p AffinityOn=True \
nfs-dg-rs
```

**create**

リソースを作成します。

**-g**

リソースを追加するリソースグループを指定します。

**-t SUNW.HASStoragePlus**

リソースのタイプに SUNW.HASStoragePlus を指定します。

**-p**

リソースのプロパティを指定します。

**FileSystemMountPoints=/global/**

ファイルシステムのマウントポイントがグローバルであることを指定します。

**AffinityOn=True**

アプリケーションリソースが **-x GlobalDevicePaths=** で定義されたグローバルデバイスとクラスタファイルシステム向けにアフィニティスイッチオーバーを実行す

るように指定します。したがって、アプリケーションリソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

`nfs-dg-rs`

NFS アプリケーション向けの HAStoragePlus リソースの名前。

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 6 論理ホスト名リソースをアプリケーションリソースグループに追加します。

```
nodeC# clreslogicalhostname create -g nfs-rg \
lhost-nfsrg-sec
```

二次クラスタ上のアプリケーションリソースグループの論理ホスト名は `lhost-nfsrg-sec` です。

- 7 NFS リソースをアプリケーションリソースグループに追加します。

```
nodeC# clresource create -g nfs-rg \
-t SUNW.nfs -p Resource_dependencies=nfs-dg-rs nfs-rg
```

- 8 アプリケーションリソースグループが `nodeC` でオンラインになっていないことを確認します。

```
nodeC# clresource disable -n nodeC nfs-rs
nodeC# clresource disable -n nodeC nfs-dg-rs
nodeC# clresource disable -n nodeC lhost-nfsrg-sec
nodeC# clresourcegroup online -n "" nfs-rg
```

`Auto_start_on_new_cluster=False` によって、リソースグループは再起動後もオフラインのままになります。

- 9 グローバルボリュームが主クラスタにマウントされている場合は、二次クラスタのグローバルボリュームのマウントを解除します。

```
nodeC# umount /global/mountpoint
```

ボリュームが二次クラスタにマウントされていると、同期が失敗します。

次の手順 [408 ページ](#) の「[データ複製の有効化例](#)」に進みます。

## データ複製の有効化例

この節では、構成例のデータ複製をどのように有効にするかを説明します。この節では、Sun StorageTek Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、Sun StorageTek Availability のマニュアルを参照してください。

ここでは、次の手順について説明します。

- 409 ページの「主クラスタで複製を有効にする」
- 412 ページの「二次クラスタで複製を有効にする」

## ▼ 主クラスタで複製を有効にする

- 1 nodeA にスーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.read` を提供する役割になりますしてアクセスします。

- 2 すべてのトランザクションをフラッシュします。

```
nodeA# lockfs -a -f
```

- 3 論理ホスト名 `lhost-reprg-prim` と `lhost-reprg-sec` がオンラインであることを確認します。

```
nodeA# clresourcegroup status
```

```
nodeC# clresourcegroup status
```

リソースグループの状態フィールドを調べます。

- 4 主クラスタから二次クラスタへのリモートミラー複製を有効にします。

この手順によって、主クラスタのマスターボリュームから二次クラスタのマスターボリュームへの複製が有効になります。さらに、`vol04` のリモートミラービットマップへの複製も有効になります。

- 主クラスタと二次クラスタが同期されていない場合は、次のコマンドを実行します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -e lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -e lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- 主クラスタと二次クラスタが同期されている場合は、次のコマンドを実行します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -E lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -E lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

## 5 自動同期機能を有効にします。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -a on lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -a on lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

この手順で自動同期が有効になります。自動同期のアクティブ状態が on に設定されている場合、システムが再起動されたり障害が発生すると、ボリュームセットは再度同期化されます。

## 6 クラスタがロギングモードであることを確認します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -P
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01
autosync: off, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: logging
```

ロギングモードでは、状態は logging で、自動同期のアクティブ状態は off です。ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。

## 7 ポイントインタイムスナップショットを有効にします。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/iiadm -e ind \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
nodeA# /usr/opt/SUNWesm/sbin/iiadm -w \
/dev/vx/rdisk/devgrp/vol02
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/iiadm -e ind \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
nodeA# /usr/sbin/iiadm -w \
/dev/vx/rdisk/devgrp/vol02
```

この手順によって、主クラスタのマスターボリュームが同じクラスタのシャドウボリュームにコピーされるようになります。マスターボリューム、シャドウボリューム、およびポイントインタイムビットマップボリュームは同じデバイスグループに存在する必要があります。この例では、マスターボリュームは vol01、シャドウボリュームは vol02、ポイントインタイムビットマップボリュームは vol03 になります。

## 8 ポイントインタイムスナップショットをリモートミラーセットに設定します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -I a \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -I a \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
```

この手順によって、ポイントインタイムスナップショットがリモートミラーボリュームセットに関連付けられます。Sun StorageTek Availability Suite ソフトウェアは、リモートミラー複製の前にポイントインタイムスナップショットを必ず取りません。

次の手順 412 ページの「二次クラスタで複製を有効にする」に進みます。

## ▼ 二次クラスタで複製を有効にする

始める前に 手順409 ページの「主クラスタで複製を有効にする」を完了します。

- 1 スーパーユーザーとして nodeC にアクセスします。
- 2 すべてのトランザクションをフラッシュします。  
nodeC# **lockfs -a -f**
- 3 主クラスタから二次クラスタへのリモートミラー複製を有効にします。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeC# /usr/opt/SUNWesm/sbin/sndradm -n -e lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -n -e lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

主クラスタが二次クラスタの存在を認識し、同期を開始します。クラスタのステータスについては、Sun StorEdge Availability Suite のシステムログファイル /var/opt/SUNWesm/ds.log、または Sun StorageTek Availability Suite の /var/adm を参照してください。

- 4 それぞれのポイントインタイムスナップショットを有効にします。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeC# /usr/opt/SUNWesm/sbin/iiadm -e ind \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
nodeC# /usr/opt/SUNWesm/sbin/iiadm -w \
/dev/vx/rdisk/devgrp/vol02
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/iiadm -e ind \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
```

```
nodeC# /usr/sbin/iiadm -w \
/dev/vx/rdisk/devgrp/vol02
```

## 5 ポイントインタイムスナップショットをリモートミラーセットに設定します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeC# /usr/opt/SUNWesm/sbin/sndradm -I a \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -I a \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
```

次の手順 [413 ページの「データ複製の実行例」](#)に進みます。

## データ複製の実行例

この節では、構成例のデータ複製をどのように実行するかを説明します。この節では、Sun StorageTek Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、Sun StorageTek Availability Suite のマニュアル。

ここでは、次の手順について説明します。

- [413 ページの「リモートミラー複製を実行する」](#)
- [415 ページの「ポイントインタイムスナップショットを実行する」](#)
- [417 ページの「複製が正しく構成されていることを確認する」](#)

### ▼ リモートミラー複製を実行する

この手順では、主ディスクのマスターボリュームが二次ディスクのマスターボリュームに複製されます。マスターボリュームは `vol01` で、リモートミラービットマップボリュームは `vol04` です。

#### 1 スーパーユーザーとして nodeA にアクセスします。

#### 2 クラスタがロギングモードであることを確認します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -P
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01
autosync: off, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: logging
```

ロギングモードでは、状態は logging で、自動同期のアクティブ状態は off です。ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。

- 3 すべてのトランザクションをフラッシュします。

```
nodeA# lockfs -a -f
```

- 4 nodeC で手順 1 から手順 3 を繰り返します。

- 5 nodeA のマスターボリュームを nodeC のマスターボリュームにコピーします。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -m lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -m lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- 6 複製が完了し、ボリュームが同期化されるのを待ちます。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -w lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -w lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

## 7 クラスタが複製モードであることを確認します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -P
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01
autosync: on, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: replicating
```

複製モードでは、状態は `replicating` で、自動同期のアクティブ状態は `on` です。主ボリュームに書き込みが行われると、Sun StorageTek Availability Suite ソフトウェアが二次ボリュームを更新します。

次の手順 [415 ページの「ポイントインタイムスナップショットを実行する」](#)に進みます。

## ▼ ポイントインタイムスナップショットを実行する

この手順では、ポイントインタイムスナップショットを使用して、主クラスタのシャドウボリュームを主クラスタのマスターボリュームに同期させます。マスターボリュームは `vol01`、ビットマップボリュームは `vol04`、シャドウボリュームは `vol02` です。

始める前に [手順413 ページの「リモートミラー複製を実行する」](#)を完了します。

- 1 nodeA にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` および `solaris.cluster.admin` を提供する役割になりますしてアクセスします。

- 2 nodeA で実行されているリソースを無効にします。

```
nodeA# clresource disable -n nodeA nfs-rs
```

- 3 主クラスタをロギングモードに変更します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。複製は行われません。

4 主クラスタのシャドウボリュームを主クラスタのマスターボリュームに同期化させます。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/iiadm -u s /dev/vx/rdisk/devgrp/vol02
nodeA# /usr/opt/SUNWesm/sbin/iiadm -w /dev/vx/rdisk/devgrp/vol02
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/iiadm -u s /dev/vx/rdisk/devgrp/vol02
nodeA# /usr/sbin/iiadm -w /dev/vx/rdisk/devgrp/vol02
```

5 二次クラスタのシャドウボリュームを二次クラスタのマスターボリュームに同期化させます。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeC# /usr/opt/SUNWesm/sbin/iiadm -u s /dev/vx/rdisk/devgrp/vol02
nodeC# /usr/opt/SUNWesm/sbin/iiadm -w /dev/vx/rdisk/devgrp/vol02
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/iiadm -u s /dev/vx/rdisk/devgrp/vol02
nodeC# /usr/sbin/iiadm -w /dev/vx/rdisk/devgrp/vol02
```

6 nodeA でアプリケーションを再起動します。

```
nodeA# clresource enable -n nodeA nfs-rs
```

7 二次ボリュームを主ボリュームと再同期化させます。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
```

```
/dev/vx/rdsk/devgrp/vol01 \
/dev/vx/rdsk/devgrp/vol04 ip sync
```

次の手順 417 ページの「複製が正しく構成されていることを確認する」に進みます。

### ▼ 複製が正しく構成されていることを確認する

始める前に 手順415 ページの「ポイントインタイムスナップショットを実行する」を完了します。

- 1 nodeA および nodeC にスーパーユーザーになるか、RBAC の承認 solaris.cluster.admin を提供する役割になりますを使用してアクセスします。
- 2 主クラスタが複製モードで、自動同期機能がオンになっていることを確認します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -P
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdsk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdsk/devgrp/vol01
autosync: on, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: replicating
```

複製モードでは、状態は replicating で、自動同期のアクティブ状態は on です。主ボリュームに書き込みが行われると、Sun StorageTek Availability Suite ソフトウェアが二次ボリュームを更新します。

- 3 主クラスタが複製モードでない場合は、複製モードにします。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdsk/devgrp/vol01 \
/dev/vx/rdsk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdsk/devgrp/vol01 \
/dev/vx/rdsk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdsk/devgrp/vol01 \
/dev/vx/rdsk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdsk/devgrp/vol01 \
/dev/vx/rdsk/devgrp/vol04 ip sync
```

- 4 クライアントマシンにディレクトリを作成します。
  - a. クライアントマシンにスーパーユーザーとしてログインします。  
次のようなプロンプトが表示されます。  
*client-machine#*
  - b. クライアントマシンにディレクトリを作成します。  
*client-machine# mkdir /dir*
- 5 ディレクトリを主クラスタのアプリケーションにマウントし、マウントしたディレクトリを表示します。
  - a. ディレクトリを主クラスタのアプリケーションにマウントします。  
*client-machine# mount -o rw lhost-nfsrg-prim:/global/mountpoint /dir*
  - b. マウントしたディレクトリを表示します。  
*client-machine# ls /dir*
- 6 ディレクトリを二次クラスタのアプリケーションにマウントし、マウントしたディレクトリを表示します。
  - a. 主クラスタのアプリケーションからディレクトリのマウントを解除します。  
*client-machine# umount /dir*
  - b. 主クラスタのアプリケーションリソースグループをオフラインにします。  
*nodeA# clresource disable -n nodeA nfs-rs*  
*nodeA# clresource disable -n nodeA nfs-dg-rs*  
*nodeA# clresource disable -n nodeA lhost-nfsrg-prim*  
*nodeA# clresourcegroup online -n "" nfs-rg*
  - c. 主クラスタをロギングモードに変更します。
    - Sun StorEdge Availability Suite ソフトウェアの場合:  
*nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync*
    - Sun StorageTek Availability Suite ソフトウェアの場合:  
*nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \*

```
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。複製は行われません。

- d. PathPrefix ディレクトリが使用可能であることを確認します。

```
nodeC# mount | grep /global/etc
```

- e. 二次クラスタのアプリケーションリソースグループをオンラインにします。

```
nodeC# clresourcegroup online -n nodeC nfs-rg
```

- f. クライアントマシンにスーパーユーザーとしてアクセスします。

次のようなプロンプトが表示されます。

```
client-machine#
```

- g. **手順4**で作成したディレクトリを二次クラスタのアプリケーションにマウントします。

```
client-machine# mount -o rw lhost-nfsrg-sec:/global/mountpoint /dir
```

- h. マウントしたディレクトリを表示します。

```
client-machine# ls /dir
```

- 7 **手順5**で表示されたディレクトリが**手順6**で表示されたディレクトリと同じであることを確認します。

- 8 主クラスタのアプリケーションをマウントされたディレクトリに戻します。

- a. 二次クラスタのアプリケーションリソースグループをオフラインにします。

```
nodeC# clresource disable -n nodeC nfs-rs
nodeC# clresource disable -n nodeC nfs-dg-rs
nodeC# clresource disable -n nodeC lhost-nfsrg-sec
nodeC# clresourcegroup online -n "" nfs-rg
```

- b. グローバルボリュームを二次クラスタからマウント解除します。

```
nodeC# umount /global/mountpoint
```

- c. 主クラスタのアプリケーションリソースグループをオンラインにします。

```
nodeA# clresourcegroup online -n nodeA nfs-rg
```

- d. 主クラスタを複製モードに変更します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

主ボリュームに書き込みが行われると、Sun StorageTek Availability Suite ソフトウェアが二次ボリュームを更新します。

参照 [420 ページの「フェイルオーバーの管理方法の例」](#)

## フェイルオーバーの管理方法の例

ここでは、フェイルオーバーの開始方法と、アプリケーションがどのように二次クラスタに転送されるかを説明します。フェイルオーバーの後、DNS エントリを更新します。詳細については、[390 ページの「フェイルオーバーの管理のガイドライン」](#)を参照してください。

ここでは、次の手順について説明します。

- [420 ページの「スイッチオーバーを呼び出す」](#)
- [422 ページの「DNS エントリを更新する」](#)

### ▼ スwitchオーバーを呼び出す

- 1 nodeA および nodeC にスーパーユーザーになるか、RBAC の承認 solaris.cluster.admin を提供する役割になりますを使用してアクセスします。
- 2 主クラスタをロギングモードに変更します。
  - Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -n -l lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

ディスクのデータボリュームに書き込みが行われると、同じデバイスグループのビットマップボリュームが更新されます。複製は行われません。

- 3 主クラスタと二次クラスタがロギングモードで、自動同期がオフであることを確認します。

- a. nodeA で、モードと設定を確認します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeA# /usr/opt/SUNWesm/sbin/sndradm -P
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01
autosync:off, max q writes:4194304,max q fbas:16384,mode:sync,ctag:
devgrp, state: logging
```

- b. nodeC で、モードと設定を確認します。

- Sun StorEdge Availability Suite ソフトウェアの場合:

```
nodeC# /usr/opt/SUNWesm/sbin/sndradm -P
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 <-
lhost-reprg-prim:/dev/vx/rdisk/devgrp/vol01
autosync:off, max q writes:4194304,max q fbas:16384,mode:sync,ctag:
devgrp, state: logging
```

nodeA と nodeC の状態は logging で、非同期のアクティブ状態は off でなければなりません。

- 4 二次クラスタで主クラスタからのテイクオーバーの準備ができていることを確認します。

```
nodeC# fsck -y /dev/vx/rdisk/devgrp/vol01
```

- 5 二次クラスタにスイッチオーバーします。

```
nodeC# clresourcegroup switch -n nodeC nfs-rg
```

次の手順 422 ページの「DNS エントリを更新する」に進みます。

## ▼ DNS エントリを更新する

DNS がクライアントをクラスタにどのようにマッピングするかについては、[図 A-6](#) を参照してください。

始める前に 手順420 ページの「スイッチオーバーを呼び出す」を完了します。

- 1 nsupdate コマンドを開始します。  
詳細は、[nsupdate\(1M\)](#) のマニュアルページを参照してください。
- 2 両方のクラスタについて、アプリケーションリソースグループの論理ホスト名とクラスタ IP アドレス間の現在の DNS マッピングを削除します。

```
> update delete lhost-nfsrg-prim A
> update delete lhost-nfsrg-sec A
> update delete ipaddress1rev.in-addr.arpa ttl PTR lhost-nfsrg-prim
> update delete ipaddress2rev.in-addr.arpa ttl PTR lhost-nfsrg-sec
```

*ipaddress1rev* 主クラスタの IP アドレス (逆順) です。

*ipaddress2rev* 二次クラスタの IP アドレス (逆順) です。

*ttl* 秒単位の有効時間です。一般的な値は 3600 になります。

- 3 両方のクラスタについて、アプリケーションリソースグループの論理ホスト名とクラスタ IP アドレス間の、新しい DNS マッピングを作成します。  
主論理ホスト名を二次クラスタの IP アドレスにマッピングし、二次論理ホスト名を主クラスタの IP アドレスにマッピングします。

```
> update add lhost-nfsrg-prim ttl A ipaddress2fwd
> update add lhost-nfsrg-sec ttl A ipaddress1fwd
> update add ipaddress2rev.in-addr.arpa ttl PTR lhost-nfsrg-prim
> update add ipaddress1rev.in-addr.arpa ttl PTR lhost-nfsrg-sec
```

*ipaddress2fwd* 二次クラスタの IP アドレス (正順) です。

*ipaddress1fwd* 主クラスタの IP アドレス (正順) です。

## Sun Cluster オブジェクト指向コマンド

---

この付録では、オブジェクト指向コマンド、その短縮形、およびそのサブコマンドの概要を説明します。

### オブジェクト指向コマンド名および別名

多くの Sun Cluster コマンドには、長い説明的な形式以外にも、ユーザーの入力量を大幅に減らす、短縮形つまり別名もあります。次の表に、コマンドとその短い別名を示します。

表 B-1 オブジェクト指向コマンドと別名 (短縮名)

完全なコマンド	別名	目的
claccess	なし	Sun Cluster のアクセスポリシーの管理
cldevice	cldev	Sun Cluster デバイスの管理
cldevicegroup	cldg	Sun Cluster デバイスグループの管理
clinterconnect	clintr	Sun Cluster インターコネクットの管理
clnasdevice	clnas	Sun Cluster の NAS デバイスへのアクセスの管理
clnode	なし	Sun Cluster ノードの管理
clquorum	clq	Sun Cluster 定足数の管理
clquorumserver	clqs	定足数サーバーホスト上での定足数サーバープロセスの構成と管理
clreslogicalhostname	clrslh	論理ホスト名のための Sun Cluster リソースの管理
clresource	clrs	Sun Cluster データサービスのリソースの管理

表 B-1 オブジェクト指向コマンドと別名 (短縮名) (続き)

完全なコマンド	別名	目的
clresourcegroup	clrg	Sun Cluster データサービスのリソースグループの管理
clresourcetype	clrt	Sun Cluster データサービスのリソースタイプの管理
clrssharedaddress	clrssa	共有アドレスのための Sun Cluster リソースの管理
clsetup	なし	Sun Cluster の対話型での構成。このコマンドにはサブコマンドはありません。
clsnmphost	なし	Sun Cluster SNMP ホストの管理
clsnmpmib	なし	Sun Cluster SNMP MIB の管理
clsnmpuser	なし	Sun Cluster SNMP ユーザーの管理
cltelemetryattribute	clta	システムリソース監視の構成
cluster	なし	Sun Cluster の広域構成と状態の管理
clvxvm	なし	Veritas Volume Manager for Sun Cluster の構成
clzonecluster	clzc	ゾーンクラスタの管理

## オブジェクト指向コマンドセットの概要

次の表に、オブジェクト指向コマンドセットのコマンドと各コマンドで使用可能なサブコマンドのリストを示します。

表 B-2 claccess: ノード用の Sun Cluster アクセスポリシーの管理

サブコマンド	目的
allow	指定されたマシン (1 つまたは複数) がクラスタ構成にアクセスすることを許可します。
allow-all	すべてのノードがクラスタ構成にアクセスすることを許可します。
deny	指定されたマシン (1 つまたは複数) がクラスタ構成にアクセスすることを禁止します。
deny-all	すべてのノードがクラスタ構成にアクセスすることを禁止します。
list	クラスタ構成へのアクセス権を持っているマシンの名前を表示します。
set	承認プロトコルを <code>-a</code> オプションで指定した値に設定します。
show	クラスタ構成へのアクセス権を持っているマシンの名前を表示します。

表 B-3 cldevice、cldev: Sun Cluster デバイスの管理

サブコマンド	目的
check	デバイスの物理デバイスに対する整合性検査を、カーネル表現と比較して実行します。
clear	現在のノードから排除されたデバイスに関して、すべての DID 参照を削除するよう指定します。
combine	指定された DID インスタンスを新しい宛先インスタンスに結合します。
export	クラスタデバイスの構成情報をエクスポートします。
list	すべてのデバイスパスを表示します。
monitor	指定したディスクパスの監視をオンにします。
populate	広域デバイス名前空間を生成します。
refresh	クラスタノード上にある現在のデバイスツリーに対してデバイス構成情報を更新します。
rename	指定された DID インスタンスを新しい DID インスタンスに移動します。
repair	指定されたデバイスインスタンスに対して修復手順を実行します。
replicate	コントローラベースの複製で使用する DID デバイスを構成します。
set	指定されたデバイスのプロパティを設定します。
show	指定されたすべてのデバイスパスの構成レポートを表示します。
status	コマンドに対するオペランドとして指定されたディスクパスの状態を表示します。
unmonitor	コマンドのオペランドとして指定されたディスクパスの監視をオフにします。

表 B-4 cldevicegroup、cldg: Sun Cluster デバイスグループの管理

サブコマンド	目的
add-device	新しいメンバーディスクデバイスを既存の raw ディスクデバイスグループに追加します。
add-node	新しいノードを既存のデバイスグループに追加します。
create	新しいデバイスグループを作成します。
delete	デバイスグループを削除します。
disable	オフラインのデバイスグループを無効にします。
enable	デバイスグループを有効にします。
export	デバイスグループ構成情報をエクスポートします。

表 B-4 cldevicegroup、cldg: Sun Cluster デバイスグループの管理 (続き)

サブコマンド	目的
list	デバイスグループのリストを表示します。
offline	デバイスグループをオフラインにします。
online	指定されたノードでデバイスグループをオンラインにします。
remove-device	メンバーディスクデバイスを raw ディスクデバイスグループから削除します。
remove-node	既存のデバイスグループからノードを削除します。
set	デバイスグループに関連付けられている属性を設定します。
show	デバイスグループの構成レポートを作成します。
status	デバイスグループのステータスレポートを作成します。
switch	Sun Cluster 構成内の、ある主ノードから別のノードにデバイスグループを転送します。
sync	クラスタリングソフトウェアとデバイスグループ情報の同期をとります。

表 B-5 clinterconnect、clintr: Sun Cluster インターコネクットの管理

サブコマンド	目的
add	コマンドへのオペランドとして指定された新しいクラスタインターコネクットコンポーネントを追加します。
disable	コマンドへのオペランドとして指定されたインターコネクットコンポーネントを無効にします。
enable	コマンドへのオペランドとして指定されたインターコネクットコンポーネントを有効にします。
export	クラスタインターコネクットの構成情報をエクスポートします。
remove	コマンドへのオペランドとして提供されたクラスタインターコネクットコンポーネントを削除します。
show	インターコネクットコンポーネントの構成を表示します。
status	インターコネクットパスのステータスを表示します。

表 B-6 clnasdevice、clnas: Sun Cluster の NAS デバイスへのアクセスの管理

サブコマンド	目的
add	NAS デバイスを Sun Cluster 構成に追加します。
add-dir	すでに構成されている NAS デバイスの指定されたディレクトリをクラスタ構成に追加します。

表 B-6 clnasdevice、clnas:Sun Cluster の NAS デバイスへのアクセスの管理 (続き)

サブコマンド	目的
export	クラスタ NAS デバイス構成情報をエクスポートします。
list	クラスタに構成されている NAS デバイス構成を表示します。
remove	指定された NAS デバイス (1 つまたは複数) を Sun Cluster 構成から削除します。
remove-dir	指定された NAS ディレクトリ (1 つまたは複数) を Sun Cluster 構成から削除します。
set	特定の NAS デバイスの指定されたプロパティを設定します。
show	クラスタ内の NAS デバイスの構成情報を表示します。

表 B-7 clnode:Sun Cluster ノードの管理

サブコマンド	目的
add	ノードをクラスタに構成および追加します。
add-farm	ファームノードをクラスタに追加します。
clear	Sun Cluster ソフトウェア構成からノードを削除します。
evacuate	指定されたノードから新しい主ノードに、すべてのリソースグループおよびデバイスグループを切り替えます。
export	ノードまたはファーム構成情報をファイルまたは標準出力 (stdout) にエクスポートします。
list	クラスタまたはファームで構成されているノードの名前を表示します。
remove	ノードをクラスタから削除します。
remove-farm	ファームノードをクラスタから削除します。
set	指定したノードに関連するプロパティを設定します。
show	指定されたノード (1 つまたは複数) の構成を表示します。
show-rev	ノードにインストールされている Sun Cluster パッケージの名前と、そのノードについてのリリース情報を表示します。
status	指定したノード (1 つまたは複数) のステータスを表示します。

表 B-8 clquorum、clq:Sun Cluster の定足数構成の管理

サブコマンド	目的
add	指定した共有デバイスを定足数デバイスとして追加します。
disable	定足数デバイスまたはノードを定足数保守状態に置きます。

表 B-8 clquorum、clq: Sun Cluster の定足数構成の管理 (続き)

サブコマンド	目的
enable	定足数デバイスまたはノードを定足数保守状態から解除します。
export	クラスタ定足数の構成情報をエクスポートします。
list	クラスタ内で設定されている定足数デバイスの名前を表示します。
remove	指定された定足数デバイス (1 つまたは複数) を、Sun Cluster 定足数構成から削除します。
reset	定足数構成全体をリセットし、デフォルトの投票数にします。
show	定足数デバイスのプロパティを表示します。
status	定足数デバイスの状態と投票数を表示します。

表 B-9 clquorumserver、clqs: 定足数サーバーの管理

サブコマンド	目的
clear	期限切れのクラスタ情報を定足数サーバーから削除します。
show	定足数サーバーについての構成情報を表示します。
start	ホストマシン上で定足数サーバープロセスを起動します。
stop	定足数サーバープロセスを停止します。

表 B-10 clreslogicalhostname、clrslh: Sun Cluster 論理ホスト名のリソースの管理

サブコマンド	目的
create	新しい論理ホスト名リソースを作成します。
delete	論理ホスト名リソースを削除します。
disable	論理ホスト名リソースを無効にします。
enable	論理ホスト名リソースを有効にします。
export	論理ホスト名リソース構成をエクスポートします。
list	論理ホスト名リソースのリストを表示します。
list-props	論理ホスト名リソースのプロパティのリストを表示します。
monitor	論理ホスト名リソースに対する監視をオンにします。
reset	論理ホスト名リソースと関連するエラーフラグをクリアします。
set	論理ホスト名リソースの指定されたプロパティを設定します。
show	論理ホスト名リソースの構成を表示します。

表 B-10 `clreslogicalhostname`、`clrs1h`: Sun Cluster 論理ホスト名のリソースの管理 (続き)

サブコマンド	目的
<code>status</code>	論理ホスト名リソースのステータスを表示します。
<code>unmonitor</code>	論理ホスト名リソースに対する監視をオフにします。

表 B-11 `clresource`、`clrs`: Sun Cluster データサービスのリソースの管理

サブコマンド	目的
<code>create</code>	コマンドに対するオペラントとして指定されたリソースを作成します。
<code>delete</code>	コマンドに対するオペラントとして指定されたリソースを削除します。
<code>disable</code>	リソースを無効にします。
<code>enable</code>	リソースを有効にします。
<code>export</code>	クラスタリソース構成をエクスポートします。
<code>list</code>	クラスタリソースのリストを表示します。
<code>list-props</code>	リソースプロパティのリストを表示します。
<code>monitor</code>	リソースの監視をオンにします。
<code>reset</code>	クラスタリソースと関連しているエラーフラグをクリアします。
<code>set</code>	リソースプロパティを設定します。
<code>show</code>	リソース構成を表示します。
<code>status</code>	リソースのステータスを表示します。
<code>unmonitor</code>	リソースの監視をオフにします。

表 B-12 `clresourcegroup`、`clrg`: Sun Cluster データサービスのリソースグループの管理

サブコマンド	目的
<code>add-node</code>	ノードをリソースグループの <code>NodeList</code> プロパティの最後に追加します。
<code>create</code>	新しいリソースグループを作成します。
<code>delete</code>	リソースグループを削除します。
<code>evacuate</code>	<code>-n</code> オプションで指定したノード上のすべてのリソースグループをオフラインにします。
<code>export</code>	リソースグループの構成情報をファイルまたは標準出力 <code>stdout</code> に書き込みます。
<code>list</code>	リソースグループのリストを表示します。

表 B-12 clresourcegroup、clrg:Sun Cluster データサービスのリソースグループの管理 (続き)

サブコマンド	目的
manage	指定したリソースグループを管理状態にします。
offline	指定したリソースグループをオフライン状態にします。
online	指定したリソースグループをオンライン状態にします。
quiesce	指定されたリソースグループを休止状態にします。
remaster	指定したリソースグループを、最も優先されるノードに切り替えます。
remove-node	ノードをリソースグループの <code>NodeList</code> プロパティから削除します。
restart	もともとリソースグループをホストしていた主ノードの同じセット上でリソースグループをオフラインにしてからオンラインに戻します。
resume	保存停止にある指定されたソースグループの保存停止状態をクリアします。
set	指定したリソースグループに関連付けられているプロパティを設定します。
show	指定したリソースグループの構成レポートを生成します。
status	指定したリソースグループのステータスレポートを生成します。
suspend	指定したリソースグループにより管理されているすべてのアプリケーションに対して、RGM の制御を保存停止します。
switch	指定したリソースグループをマスターするノードまたはノードのセットを変更します。
unmanage	指定したリソースグループを管理されない状態にします。

表 B-13 clresourcetype、clrt:Sun Cluster データサービスのリソースタイプの管理

サブコマンド	目的
add-node	指定されたノードを、リソースタイプのノードリストに追加します。
export	クラスタリソースタイプ構成をエクスポートします。
list	リソースタイプのリストを表示します。
list-props	リソースタイプのリソース拡張プロパティまたはリソースタイププロパティのリストを表示します。
register	リソースタイプを登録します。
remove-node	オペランドリスト内のリソースタイプが登録されるノードのリストからノードを削除します。
set	リソースタイプのプロパティを設定します。

表 B-13 clresourcetype、clrt:Sun Cluster データサービスのリソースタイプの管理 (続き)

サブコマンド	目的
show	クラスタ内に登録されているリソースタイプについての構成情報を表示します。
unregister	リソースタイプを登録解除します。

表 B-14 clressharedaddress、clrssa:共有アドレスの Sun Cluster リソースの管理

サブコマンド	目的
create	共有アドレスリソースを作成します。
delete	共有アドレスリソースを削除します。
disable	共有アドレスリソースを無効にします。
enable	共有アドレスリソースを有効にします。
export	共有アドレスリソース構成をエクスポートします。
list	共有アドレスリソースのリストを表示します。
list-props	共有アドレスリソースのプロパティのリストを表示します。
monitor	共有アドレスリソースの監視をオンにします。
reset	共有アドレスリソースと関連付けられたエラーフラグをクリアします。
set	共有アドレスリソースの指定されたプロパティを設定します。
show	共有アドレスリソースの構成を表示します。
status	共有アドレスリソースのステータスを表示します。
unmonitor	共有アドレスリソースの監視をオフにします。

表 B-15 clsnmphot:Sun Cluster SNMP ホストのリストの管理

サブコマンド	目的
add	SNMP ホストを、指定されたノード構成に追加します。
export	指定されたノードから SNMP ホスト情報をエクスポートします。
list	指定されたノード上で構成されている SNMP ホストを一覧表示します。
remove	SNMP ホストをノード構成から削除します。
show	指定されたノード上の SNMP ホスト構成情報を表示します。

表 B-16 clsnmpmib: Sun Cluster SNMP MIB の管理

サブコマンド	目的
disable	指定されたノード上の 1 つ以上のクラスタの MIB を無効にします。
enable	指定されたノード上にある 1 つ以上のクラスタの MIB を有効にします。
export	クラスタの MIB の構成情報をエクスポートします。
list	指定されたノード上のクラスタの MIB のリストを表示します。
set	1 つまたは複数の MIB で使用されている SNMP プロトコル設定を設定します。
show	指定されたノード上の MIB の構成情報を表示します。

表 B-17 clsnmpuser: Sun Cluster SNMP ユーザーの管理

サブコマンド	目的
create	指定されたノード上の SNMP ユーザー構成にユーザーを追加します。
delete	SNMPv3 ユーザーを指定されたノードから削除します。
export	SNMP ユーザー情報を指定されたノードからエクスポートします。
list	指定されたノードで構成されている SNMPv3 ユーザーのリストを出力します。
set	指定されたノード上のユーザーの構成を設定します。
set-default	SNMPv3 を使用してトラップを送信する際に使用する、デフォルトのユーザーおよびセキュリティレベルを設定します。
show	指定されたノード上のユーザーについての情報を表示します。

表 B-18 cltelemetryattribute、clta: システムリソースの監視の構成

サブコマンド	目的
disable	指定されたオブジェクトタイプの指定されたテレメトリ属性を無効にします。
enable	指定されたオブジェクトタイプの指定されたテレメトリ属性のデータ収集を有効にします。
export	オブジェクトタイプおよびオブジェクトインスタンスのテレメトリ属性の構成をファイルまたは標準出力 stdout にエクスポートします。
list	指定されたオブジェクトタイプに対して構成されているテレメトリ属性を表示します。
print	指定されたオブジェクトインスタンスまたはオブジェクトタイプに対して有効な、指定されているテレメトリ属性のシステムリソースの使用状況を表示します。

表 B-18 cltelemetryattribute、clta: システムリソースの監視の構成 (続き)

サブコマンド	目的
set-threshold	ノード上の指定されたオブジェクトの指定されたテレメトリ属性のしきい値の設定を変更します。
show	オブジェクトタイプまたはオブジェクトインスタンスのテレメトリ属性に対して設定されているプロパティを表示します。

表 B-19 cluster: クラスタの広域構成とステータスの管理

サブコマンド	目的
check	クラスタが正しく構成されているかどうかをチェックして報告します。
create	clconfigfile ファイルに格納されている構成情報を使用してクラスタを作成します。
export	クラスタ構成ファイルの構成情報をエクスポートします。
list	クラスタコマンドの発行対象であるクラスタの名前を表示します。
list-checks	使用可能な各チェックについて、そのチェック ID と説明のリストを出力します。
list-cmds	使用可能なすべての Sun Cluster コマンドのリストを出力します。
rename	クラスタコマンドの発行対象であるクラスタの名前を変更します。
restore-netprops	クラスタコマンドの発行対象であるクラスタの、クラスタプライベートネットワーク設定を修復します。
set	クラスタコマンドの発行対象であるクラスタのプロパティを設定します。
set-netprops	クラスタプライベートネットワークアドレスのプロパティを設定します。
show	指定されたクラスタのクラスタコンポーネントに関する詳細な構成情報を表示します。
show-netprops	プライベートネットワークアドレスの設定を表示します。
shutdown	クラスタコマンドの発行対象であるクラスタを適切な順序で停止します。
status	指定されたクラスタのクラスタコンポーネントのステータスを表示します。

表 B-20 clvxvm: Sun Cluster の VERITAS Volume Manager の構成

サブコマンド	目的
encapsulate	ルートディスクをカプセル化し、ほかの Sun Cluster 固有のタスクを実行します。
initialize	VxVM を初期化し、その他の Sun Cluster 固有のタスクを実行します。

表 B-21 clzonecluster: Sun Cluster 用のゾーンクラスタの作成と管理

サブコマンド	目的
boot	ゾーンクラスタを起動します。
clone	ゾーンクラスタを複製します。
configure	対話型ユーティリティを起動して、ゾーンクラスタの構成と作成を行います。
delete	特定のゾーンクラスタを削除します。
halt	ゾーンクラスタまたはゾーンクラスタ上の特定のノードを停止します。
install	ゾーンクラスタをインストールします。
list	構成されているゾーンクラスタの名を一覧表示します。
move	ゾーンパスを新しいゾーンパスに移動します。
ready	アプリケーション用のゾーンを準備します。
reboot	ゾーンクラスタを再起動します。
show	ゾーンクラスタのプロパティを表示します。
status	ゾーンクラスタノードがゾーンクラスタのメンバーかどうかを判定します。
uninstall	ゾーンクラスタをアンインストールします。
verify	指定された情報の構文が正しいかどうかをチェックします。

# 索引

---

## A

Add Administrative Role ウィザード, 説明, 55  
autoboot プロパティ, 259  
Availability Suite, データ複製に使用, 381

## B

boot コマンド, 65-67

## C

cconsole コマンド, 20, 24  
ccp コマンド, 19, 24  
claccess コマンド, 19, 424  
cldev コマンド, 425  
cldevice コマンド, 19, 425  
cldevicegroup コマンド, 19, 426  
cldg コマンド, 426  
clinterconnect コマンド, 19, 426  
clintr コマンド, 426  
clnas コマンド, 427  
clnasdevice コマンド, 19, 427  
clnode check コマンド, 19  
clnode コマンド, 427  
clq コマンド, 428  
clqs コマンド, 428  
clquorum コマンド, 19, 428  
clquorumserver コマンド, 428  
clreslogicalhostname コマンド, 19, 429  
clresource コマンド, 19, 429

clresource コマンド (続き)

リソースとリソースグループの削除, 290

clresourcegroup コマンド, 19, 430

clresourcetype コマンド, 19, 431

clressharedaddress コマンド, 19, 431

clrg コマンド, 430

clrs コマンド, 429

clrslh コマンド, 429

clrssa コマンド, 431

clrt コマンド, 431

clsetup コーティリティー, 18, 19, 26

clsnmphot コマンド, 19, 431

clsnmpmib コマンド, 19, 432

clsnmpuser コマンド, 19, 432

clta コマンド, 433

cltelemattribute コマンド, 19

cltelemetryattribute コマンド, 433

cluster check コマンド, 19, 433

vfstab ファイル確認, 183

cluster shutdown コマンド, 61-72

cluster コマンド, 433

clvxvm コマンド, 433

clzonecluster

boot, 65-67

halt, 61-72

説明, 26

clzonecluster コマンド, 19, 434

CPU シェア

グローバルクラスタの投票ノード, 313

グローバルクラスタの非投票ノード, 315

グローバルクラスタの非投票ノード、専用のプロセッサセット, 318

**CPU シェア (続き)**

構成, 309

制御, 309

crlogin コマンド, 24

cssh コマンド, 24

ctelnet コマンド, 24

**D**

DID 情報, 手動更新, 192

DID 情報の手動更新, 192

DR, 「動的再構成」を参照

**E****EMC SRDF**

DID デバイスを構成する, 115-117

を管理する, 113-126

キャンパスクラスタの主ルームの完全なフェイルオーバー後の復元, 124-126

ドミノモード, 93

ベストプラクティス, 95

構成を確認する, 117-118

構成例, 118-126

制限, 94

適応型コピー, 93

複製グループを構成する, 114-115

要件, 94

/etc/inet/hosts ファイル, 排他的 IP ゾーンでの設定, 261

/etc/nsswitch.conf ファイル, 非大域ゾーンの変更, 260

/etc/vfstab ファイル

マウントポイントの追加, 182

設定の確認, 183

/etc/vfstab ファイル, 48

**F**

failback プロパティ, 165

**G**

GUI 管理ツール, 18, 371-379

Sun Cluster Manager, 371

Sun Management Center, 372

**H****Hitachi TrueCopy**

DID デバイスの構成, 104-106

データまたはステータスモード, 93

ベストプラクティス, 95

管理, 101-113

構成の確認, 106-107

構成例, 107-113

制限, 94

複製グループの構成, 102-104

要件, 94

hosts ファイル, 排他的 IP ゾーンでの設定, 261

**I**

IP アドレス, 排他的 IP ゾーンのネームサービスへの追加, 261

**IPMP**

ステータス, 34

管理, 250

排他的 IP ゾーン上のグループ

構成, 261

**K**

/kernel/drv/,md.conf ファイル, 136

**L**

lofi ファイル, アインストール, 297

**M**

md.tab ファイル, 21

**MIB**

- SNMP イベントの有効化と無効化, 299, 300
- SNMP イベントプロトコルの変更, 300

**N**

- NAS, 「ネットワーク接続ストレージ定足数デバイス」を参照
- NetApp, 「ネットワーク接続ストレージ定足数デバイス」を参照
- netcon コマンド, 20
- Network Appliance, 「ネットワーク接続ストレージ定足数デバイス」を参照
- nsswitch.conf ファイル, 非大域ゾーンの変更, 260
- ntp.conf.cluster ファイル, 281
- numsecondaries プロパティ, 166

**O**

- OpenBoot PROM (OBP), 278

**R**

- raw ディスクデバイス, 命名規約, 181
- raw ディスクデバイスグループ, 追加, 138-139
- RBAC, 53-60
  - Sun Cluster Manager, 373
    - グローバルクラスタの投票ノードの場合, 54
    - 権限プロファイル(説明), 54-55
    - 作業
      - カスタム役割を追加, 58
      - ユーザーの変更, 59
      - 使用, 53
      - 設定, 53
      - 役割の追加, 55
    - 非投票ノードの場合, 54
  - Role-Based Access Control, 「RBAC」を参照

**S**

- SATA, 200, 202
- Secure Shell, 24
- secure shell, 25
- showrev -p コマンド, 27
- SME, オンラインサービスの確認, 259
- SNMP
  - イベント MIB の有効化と無効化, 299, 300
  - プロトコルの変更, 300
  - ホストの無効化, 302
  - ホストの有効化, 301
  - ユーザーの削除, 304
  - ユーザーの追加, 303
- SNMP イベント MIB の有効化と無効化, 299, 300
- SNMP イベント MIB プロトコルの変更, 300
- Solaris 10 OS
  - CPU 制御, 309
  - SME, 259
  - svcadm コマンド, 279
  - グローバルクラスタの管理作業, 16
  - グローバルクラスタ定義, 15
  - ゾーンクラスタ定義, 15
  - ノードの起動に関する特別な指示, 77-79
  - ノードの再起動に関する特別な指示, 79-83
  - ホストベースの複製, 91
- Solaris 9 OS
  - CPU シェアの制御, 311
  - ノードの起動に関する特別な指示, 77-79
  - ノードの再起動に関する特別な指示, 79-83
  - ホストベースの複製, 91
- Solaris ゾーン
  - autoboot プロパティ, 259
  - nsswitch.conf ファイルの変更, 260
  - 共有 IP ゾーン, 260
  - 排他的 IP ゾーン
    - hosts ファイルの設定, 261
    - IPMP グループの構成, 261
    - 完全ルートの要件, 260
- Solaris ボリュームマネージャー, raw ディスクデバイス名, 181
- SRDF
  - 「EMC SRDF」を参照
- ssh, 25
- Sun Cluster Manager, 18, 371

**Sun Cluster Manager (続き)**

RBAC の役割、設定, 373

サーバーアドレスの変更, 374

起動, 376

**Sun Cluster へのパッチの適用, 323-325****Sun Cluster ソフトウェアのアンインストール, 295****Sun Management Center**

インストール, 20

概要, 372

説明, 18

**Sun NAS 定足数デバイス, 追加, 203****Sun StorageTek Availability Suite, データ複製に使用, 381****SunMC, 「Sun Management Center」を参照****System Service Processor (SSP), 20****T****TrueCopy**

「Hitachi TrueCopy」を参照

**U****/usr/cluster/bin/clresource, リソースグループの削除, 290****/usr/cluster/bin/cluster check コマンド, vfstab ファイル確認, 183****V****/var/adm/messages ファイル, 87****Veritas**

オンラインバックアップ, 349

カプセル化されていないルートファイルシステムを復元, 363

カプセル化されているルートファイルシステムを復元, 365

管理, 99-100

**Veritas File System (VxFS)**

クラスタファイルシステムのマウント, 183

管理, 184

**Veritas Volume Manager (VxVM), raw ディスクデバイス名, 181****vfstab ファイル**

マウントポイントの追加, 182

設定の確認, 183

**VxFS によってサポートされる機能, 127****VxVM, 99-100****Z****ZFS**

デバイスグループ追加, 139

ファイルシステムの削除, 291-293

ルートファイルシステムの制限, 127

複製, 139

**ア**

インストール, lofi デバイスファイル, 297

アクセス権、グローバルデバイスの, 98

アダプタ、トランスポート, 239

アフィニティスイッチオーバー、データ複製用の構成, 401

アフィニティフェイルオーバー、データ複製の拡張プロパティ, 386

アプリケーションリソースグループ

ガイドライン, 387

データ複製の構成, 404-406

**イ****イベント MIB**

SNMP の有効化と無効化, 299, 300

SNMP プロトコルの変更, 300

**エ****エラーメッセージ****/var/adm/messages ファイル, 87**

ノードの削除, 270

## ク

## クラスタ

- ノード認証, 274
- バックアップ, 21, 341-354
- ファイルの復元, 354
- 再起動パッチの適用, 330
- 時刻の設定, 276
- 名前の変更, 272-273
- クラスタの時刻の設定, 276
- クラスタインターコネクト
  - 管理, 233-253
  - 状態の確認, 235
  - 動的再構成, 235
- クラスタコンソールへの安全な接続, 25
- クラスタコントロールパネル (CCP), 20
- クラスタファイルシステム, 97-195
  - グローバルクラスタ投票ノード, 127
  - グローバルクラスタ非投票ノード, 127
  - マウントオプション, 182
  - 管理, 127
  - 削除, 185-187
  - 設定の確認, 183
  - 追加, 180-184
- クラスタファイルシステムのマウントオプション, 要件, 182

## グ

## グローバル

- デバイス, 97-195
  - アクセス権の設定, 98
  - 動的再構成, 98-99
- マウントポイント、チェック, 48
- マウントポイント、確認, 187
- 名前空間, 97-100, 132
- グローバルクラスタ
  - コンポーネントの状態, 31
  - ノードの削除, 263
  - 管理, 271-307
  - 起動, 61-87
  - 構成の検証, 45
  - 構成の表示, 35
  - 再起動, 68
  - 停止, 61-87

## グローバルクラスタ (続き)

- 定義, 16
- グローバルクラスタの投票ノード, CPU シェア, 313
- グローバルクラスタの非投票ノード
  - CPU シェア, 315, 318
  - パッチの適用, 328
  - プライベートホスト名の追加, 282
- グローバルクラスタノード
  - シャットダウン, 72-87
  - 起動, 72-87
  - 再起動, 79-83
- グローバルクラスタ投票ノード, クラスタファイルシステムの管理, 127
- グローバルクラスタ非投票ノード
  - クラスタファイルシステムの管理, 127
  - プライベートホスト名、削除, 284
  - プライベートホスト名の変更, 283
  - 管理, 16
  - 停止と再起動, 72
- グローバルファイルシステム、「クラスタファイルシステム」を参照
- グローバル名前空間、移行, 133
- グローバル名前空間の更新, 132

## ケ

- ケーブル、トランスポート, 239

## コ

- コマンド, 423-434
  - boot, 65-67
  - cconsole, 20, 24
  - ccp, 19, 24
  - claccess, 19
  - cldevice, 19
  - cldevicegroup, 19
  - clinterconnect, 19
  - clnasdevice, 19
  - clnode check, 19
  - clquorum, 19
  - clreslogicalhostname, 19

## コマンド (続き)

- clresource, 19
- clresourcegroup, 19
- clresourcecetype, 19
- clressharedaddress, 19
- clsetup, 19
- clsnmpghost, 19
- clsnmpmib, 19
- clsnmpuser, 19
- cltelemetryattribute, 19
- cluster check, 19, 22, 45, 48
- clzonecluster, 19, 61-72
- clzonecluster boot, 65-67
- clzonecluster verify, 45
- crlogin, 24
- cssh, 24
- ctelnet, 24
- netcon, 20
- scshutdown, 61-72
- メタセット, 97-100

コマンド行管理ツール, 18

## コンソール

- への接続, 24
- 安全な接続, 25

## サ

- サブコマンド, 423-434
- サポートされる機能、VxFS によって, 127

## シ

## シャットダウン

- グローバルクラスタノード, 72-87
- ゾーンクラスタノード, 72-87
- ノード, 72-87

## ス

- スイッチ、トランスポート, 239
- スイッチバック、データ複数での実行ガイドライン, 391

- ストレージアレイ、削除, 267
- ストレージベースのデータ複製, 92-96
  - と定足数デバイス, 95
  - ベストプラクティス, 95
  - 制限, 94
  - 定義, 90
  - 復旧, 95
  - 要件, 94

ストレージベースの複製されたデバイスを、管理する, 101-126

## スナップショット

- 「ストレージベースの複製」を参照
- ポイントインタイム, 383

## セ

セキュリティー鍵、再生成, 375

## ゾ

## ゾーンクラスタ

- アプリケーション用に準備, 289
- コンポーネントの状態, 31
- ゾーンパスの移動, 289
- ファイルシステムの削除, 289
- 管理, 271-307
- 起動, 61-87
- 構成の検証, 45
- 構成の表示, 35
- 再起動, 68
- 停止, 61-87
- 定義, 16
- 複製, 289

## ゾーンクラスタノード

- シャットダウン, 72-87
- 起動, 72-87
- 再起動, 79-83
- ゾーンパス、移動, 289

## チ

チェック、グローバルマウントポイント, 48

## デ

- データ複製, 89-96
  - DNS エントリの更新, 422
  - ガイドライン
    - スイッチオーバーの管理, 390
    - フェイルオーバーの管理, 390
    - リソースグループの構成, 385
  - ストレージベースの, 90, 92-96
  - フェイルオーバーの管理, 420-422
  - ホストベースの, 90
  - ポイントインタイムスナップショット, 383, 415-417
  - リソースグループ
    - アプリケーション, 387
    - スケラブルアプリケーション, 389
    - フェイルオーバーアプリケーション, 387-388
    - 共有アドレス, 389
    - 構成, 386
    - 作成, 401-402
    - 命名規則, 386
  - リモートミラー, 382, 413-415
  - 概要, 382
  - 構成
    - NFS アプリケーションリソースグループ, 404-406
    - NFS アプリケーション用ファイルシステム, 398-399
    - アフィニティスイッチオーバー, 386, 401
    - デバイスグループ, 396
  - 構成の確認, 417-420
  - 構成例, 391
  - 定義, 90-91
  - 同期, 383
  - 非同期, 383
  - 必要なハードウェアとソフトウェア, 393
  - 有効化, 408-413
  - 例, 413-420
- データ複製のためにスイッチオーバー, 実行, 420-422
- データ複製のためのスケラブルアプリケーション, 389
- データ複製のためのフェイルオーバー管理, 420-422
  - データ複製のためのフェイルオーバー (続き)
    - 管理のガイドライン, 390
  - データ複製のためのフェイルオーバーアプリケーション
    - ガイドライン
      - フェイルオーバーの管理, 390
  - データ複製のフェイルオーバーアプリケーション
    - ガイドライン
      - リソースグループ, 387-388
  - データ複製の拡張プロパティ
    - アプリケーションリソース, 405, 407
    - 複製リソース, 401, 403
  - データ複製用のスイッチオーバー, アフィニティスイッチオーバー, 386
  - データ複製用のフェイルオーバーアプリケーション, アフィニティスイッチオーバー, 386
  - データ複製用の共有アドレスリソースグループ, 389
  - ディスクのカプセル化, 147
  - ディスクグループ
    - 構成変更の登録, 154
    - 作成, 141
    - 登録, 151
    - 変更, 150
  - ディスクパス
    - 監視, 97-195
      - 障害のあるディスクパスを表示, 191
      - 監視解除, 190-191
  - ディスクパスの, 状態エラーの解決, 192
  - デバイス, グローバル, 97-195
  - デバイスグループ
    - データ複製用の構成, 396
    - 確認登録, 156
    - 管理の概要, 129
    - 構成の表示, 170
    - 削除および登録解除, 142
    - 削除と登録解除, 158
    - 主所有者権, 165
    - 新しいマイナー番号の割当て, 150
    - 追加, 138
    - 変更プロパティ, 165
    - 保守状態, 172
  - デバイスグループ、raw ディスク, 追加, 138-139

デバイスグループ、SVM, 追加, 136  
デバイスグループの主ノードの切り替え, 171-172  
デバイスグループの主所有者権, 165

## ト

トランスポートアダプタ、追加, 236  
トランスポートアダプタの追加, 239  
トランスポートケーブル  
追加, 236, 239  
無効にする, 244  
有効にする, 243  
トランスポートスイッチ、追加, 236  
トランスポートスイッチの追加, 239

## ド

ドメインネームシステム (DNS), データ複製での更新, 422  
ドメインネームシステム (Domain Name System、DNS), 更新のガイドライン, 390

## ネ

ネームサービス, 排他的 IP ゾーンの IP アドレスマッピングの追加, 261  
ネットワークファイルシステム (Network File System、NFS), データ複製用アプリケーションファイルシステムの構成, 398-399  
ネットワーク接続ストレージ定足数デバイス, 追加およびインストール, 206

## ノ

ノード  
ID 検索, 273  
への接続, 24  
グローバルクラスタからのノードの削除, 263  
グローバルクラスタからの非投票ノードの削除, 267  
シャットダウン, 72-87

## ノード (続き)

ゾーンクラスタからの削除, 263  
デバイスグループから削除, 142, 161  
デバイスグループに追加, 159  
プライマリ, 98-99  
起動, 72-87  
再起動パッチの適用, 325  
削除  
エラーメッセージ, 270  
主, 165  
追加, 255-261  
二次, 165  
認証, 274  
保守状態にする, 285

## バ

バックアップ  
クラスタ, 21, 341-354  
ファイルシステム, 342  
ボリュームをオンライン, 349  
ミラーのオンライン, 346  
ルートファイルシステム, 343

## パ

パッチ  
クラスタおよびファームウェアへの適用, 330  
グローバルクラスタの非投票ノードでの, 328  
再起動パッチの適用, 325  
注意事項, 324  
非再起動を適用, 334  
パブリックネットワーク  
管理, 233-253  
動的再構成, 251

## ビ

ビットマップ  
ポイントインタイムスナップショット, 383  
リモートミラー複製, 382

## フ

## ファイル

- /etc/vfstab, 48
- md.conf, 136
- md.tab, 21
- ntp.conf.cluster, 281
- 対話形式で1つずつ復元, 355

## ファイルシステム

- NFS アプリケーション
  - データ複製の構成, 398-399
- カプセル化されていないルートを復元, 363
- カプセル化されているルートを復元, 365
- ゾーンクラスタ内で削除, 289
- バックアップ, 342
- ルートの復元
  - 説明, 355
- ルートを復元
  - ボリュームから, 357
  - メタデバイスから, 357
- 名前の確認, 342

## プ

## プライベートホスト名

- グローバルクラスタの非投票ノード, 282
- グローバルクラスタ非投票ノードの変更, 283
- グローバルクラスタ非投票ノード上で削除, 284
- ゾーンへの割り当て, 260
- 変更, 279

## プロパティ

- failback, 165
- numsecondaries, 166
- preferenced, 165

## プロファイル, RBAC 権限, 54-55

## ベ

## ベストプラクティス

- EMC SRDF, 95
- Hitachi TrueCopy, 95
- ストレージベースのデータ複製, 95

## ホ

## ホスト

- SNMP の追加と削除, 301, 302
- ホストベースのデータ複製
  - 定義, 90
  - 例, 381-422

## ボ

## ボリューム

- 「ストレージベースの複製」を参照
- オンラインでバックアップ, 349
- デバイスグループからの削除, 157-158
- デバイスグループに追加, 149
- ボリュームマネージャー, Veritas, 99-100

## ポ

- ポート番号, 共通エージェントコンテナの使用の変更, 373
- ポイントインタイムスナップショット
  - 実行, 415-417
  - 定義, 383

## マ

- マウントポイント, /etc/vfstab ファイルの修正, 182
- マウントポイント, グローバル, 48
- マルチユーザーサービス, 確認, 259

## ミ

- ミラー, オンラインバックアップ, 346

## メ

- メタセットコマンド, 97-100

## モ

モニタリング, 共有ディスクパス, 195

## ユ

### ユーザー

SNMP の削除, 304

SNMP の追加, 303

プロパティの変更, 59

ユーザーアカウントツール, 説明, 59

## リ

### リソース

構成済みタイプを表示, 29

削除, 290

### リソースグループ

データ複製

フェイルオーバーでの役割, 386

構成, 386

構成のガイドライン, 385

リモートミラー化, 「ストレージベースの複製」を参照

### リモートミラー複製

実行, 413-415

定義, 382

リモート複製, 「ストレージベースの複製」を参照

リリース情報, 27

## ロ

ローカルミラー化, 「ストレージベースの複製」を参照

ログイン, 遠隔, 24

## 移

移行, グローバルデバイス名前空間, 133

## 一

### 一覧表示

デバイスグループ構成, 170

定足数構成, 223

## 遠

遠隔ログイン, 24

## 開

### 開始

グローバルクラスタノード, 72-87

ゾーンクラスタノード, 72-87

ノード, 72-87

## 概

概要, 定足数, 197-225

## 確

### 確認

SME, 259

vfstab 設定, 183

クラスタインターコネクトの状態, 235

グローバルマウントポイント, 187

データ複製構成, 417-420

ファイルシステム名, 342

## 完

完全ルートゾーン, 排他的 IP ゾーンの要件, 260

## 監

監視, ディスクパス, 188-190

監視解除, ディスクパス, 190-191

## 管 管理

Hitachi TrueCopy で複製されたデバイス, 101-113  
 IPMP, 233-253  
 クラスタインターコネクとパブリックネットワーク, 233-253  
 クラスタファイルシステム, 127  
 グラフィカルユーザインタフェース (Graphical User Interface、GUI) ツールによる  
 クラスタ, 371-379  
 グローバルクラスタ, 16  
 グローバルクラスタ設定, 271-307  
 グローバルクラスタ非投票ノード, 16  
 ゾーンクラスタ, 16, 289  
 定足数, 197-225

## 管理する

EMC SRDF 複製デバイス, 113-126  
 ストレージベースの複製されたデバイス, 101-126

## 管理コンソール, 20

## 起 起動

Sun Cluster Manager, 376  
 グローバルクラスタ, 61-87  
 グローバルクラスタノード, 72-87  
 ゾーンクラスタ, 61-87  
 ゾーンクラスタノード, 72-87  
 ノード, 72-87  
 非クラスタモード, 83  
 非大域ゾーン, 72

## 共

共通エージェントコンテナ  
 セキュリティー鍵の再生成, 375  
 ポート番号の変更, 373  
 共有 IP ゾーン, 「Solaris ゾーン」を参照  
 共有ディスクパス  
 監視, 187-195  
 自動再起動の無効化, 195

共有ディスクパス (続き)  
 自動再起動の有効化, 195

## 検 検索

グローバルクラスタ用ノード ID, 273  
 ゾーンクラスタ用ノード ID, 273

## 検証

グローバルクラスタ構成, 45  
 ゾーンクラスタ構成, 45

## 権

権限プロファイル, RBAC, 54-55

## 公

公平配分スケジューラ  
 CPU シェアの構成, 311  
 構成, 312

## 構

### 構成

データ複製, 381-422  
 デバイスグループのマイナー番号, 150

構成済みリソースを表示, 29

### 構成例 (構内クラスタ化)

2か所、ストレージベースのデータ複製, 92-96  
 2か所、ストレージベースの複製, 92-96

### 構内クラスタ

ストレージベースのデータ複製, 92-96  
 ストレージベースのデータ複製を使用した復  
 旧, 95

## 再

### 再起動

グローバルクラスタ, 68

## 再起動 (続き)

- グローバルクラスタノード, 79-83
  - ゾーンクラスタ, 68
  - ゾーンクラスタノード, 79-83
- 再生成, セキュリティー鍵, 375

## 最

- 最後の定足数デバイス, 削除, 215

## 作

- 作成, 新しいディスクグループ, 141

## 削

## 削除

- SNMP ホスト, 302
- SNMP ユーザー, 304
- Solstice DiskSuite デバイスグループ, 142
- すべてのデバイスグループからノードを, 142
- クラスタファイルシステム, 185-187
- グローバルクラスタ上の非投票ノード, 267
- ストレージレイ, 267
- ゾーンクラスタから, 263
- デバイスグループ, 158
- デバイスグループから, ボリューム, 157-158
- デバイスグループからノードを, 161
- トランスポートケーブル, アダプタ, スイッチ, 239
- ノード, 261, 263
- リソースとリソースグループをゾーンクラスタから削除, 290
- 最後の定足数デバイス, 215
- 定足数デバイス, 199, 213

## 使

- 使用, 役割 (RBAC), 53

## 修

- 修復, 定足数デバイス, 224
- 修復, 満杯の /var/adm/messages ファイル, 87

## 状

## 状態

- グローバルクラスタコンポーネント, 31
- ゾーンクラスタコンポーネント, 31

## 切

- 切り換え, デバイスグループの主ノード, 171-172

## 設

- 設定, 役割 (RBAC), 53

## 専

- 専用のプロセッサセット, 構成, 318

## 属

- 属性, 「プロパティ」を参照

## 耐

- 耐障害性, 定義, 382

## 短

- 短いコマンド, 423-434

## 直

- 直接接続共有ディスク定足数デバイス, 追加, 202

**追****追加**

- SNMP ホスト, 301
- SNMP ユーザー, 303
- Solstice DiskSuite デバイスグループ, 138
- Sun NAS 定足数デバイス, 203
- ZFS デバイスグループ, 139
- カスタム役割 (RBAC), 58
- クラスタファイルシステム, 180-184
- デバイスグループ, 136, 138-139
- デバイスグループにノードを, 159
- トランスポートケーブル、アダプタ、およびスイッチ, 236
- ネットワーク接続ストレージ定足数デバイス, 206
- ノード, 255-261
- ノードをグローバルクラスタに, 256
- ノードをゾーンクラスタに, 256
- 新しいボリュームをデバイスグループに, 149
- 直接接続共有ディスク定足数デバイス, 202
- 定足数サーバー定足数デバイス, 209
- 定足数デバイス, 201
- 役割 (RBAC), 55

**停****停止**

- グローバルクラスタ, 61-87
- グローバルクラスタノード, 72-87
- ゾーンクラスタ, 61-87
- ゾーンクラスタノード, 72-87
- ノード, 72-87
- 非大域ゾーン, 72

**定****定足数**

- 概要, 197-225
- 管理, 197-225
- 定足数サーバー、「定足数サーバー定足数デバイス」を参照
- 定足数サーバー定足数デバイス  
インストールの要件, 209

**定足数サーバー定足数デバイス (続き)**

- 削除のトラブルシューティング, 214
- 追加, 209
- 定足数デバイス
  - とストレージベースの複製, 95
  - デバイスの動的再構成, 199
  - ノードリストの変更, 217
  - 交換, 216-217
  - 構成の一覧表示, 223
  - 最後の定足数デバイスの削除, 215
  - 削除, 199, 213
  - 修復, 224
  - 追加, 201
    - Sun NAS 定足数デバイス, 203
    - ネットワーク接続ストレージ定足数デバイス, 206
    - 直接接続共有ディスク定足数デバイス, 202
    - 定足数サーバー定足数デバイス, 209
    - 保守状態、デバイスを, 220
    - 保守状態、デバイスを保守状態から戻す, 221
- 定足数デバイスの交換, 216-217

**適****適用**

- パッチ, 325
- 非再起動パッチ, 334

**登****登録**

- ディスクグループをデバイスグループとして, 151
- ディスクグループ構成の変更, 154
- 登録解除
  - Solstice DiskSuite デバイスグループ, 142
  - デバイスグループ, 158

**動****動的再構成, 98-99**

- クラスタインターコネクト, 235

## 動的再構成 (続き)

- パブリックネットワークインタフェース, 251
- 定足数デバイス, 199

## 同

- 同期データ複製, 93, 383

## 二

## 二次ノード

- デフォルト数, 165
- 希望数の設定, 166

## 排

- 排他的 IP ゾーン, 「Solaris ゾーン」を参照

## 非

- 非クラスタモードでの起動, 83
- 非同期データ複製, 93, 383

## 表

## 表示

- グローバルクラスタ構成, 35
- ゾーンクラスタ構成, 35
- 障害のあるディスクパス, 191

## 復

- 復旧, ストレージベースのデータ複製を装備したクラスタ, 95

## 復元

- カプセル化されていないルートファイルシステム, 363
- カプセル化されているルートファイルシステム, 365

## 復元 (続き)

- クラスタファイル, 354
- ルートファイルシステム, 355
  - ボリュームから, 357
  - メタデバイスから, 357
- 個々のファイルを対話形式で, 355

## 複

## 複製

- 「データ複製」を参照
- Hitachi TrueCopy で複製されたデバイス, 102-104
- 複製、ストレージベースの, 92-96

## 別

- 別名, 423-434

## 変

## 変更

- numsecondaries プロパティ, 166
- Sun Cluster Manager サーバーアドレス, 374
- クラスタ名, 272-273
- ディスクグループ, 150
- プライベートホスト名, 279
- プロパティ, 165
- ポート番号、共通エージェントコンテナの使用, 373
- ユーザー (RBAC), 59
- 主ノード, 171-172
- 定足数デバイスのノードリスト, 217

## 保

## 保守状態

- ノード, 285
- 定足数デバイス, 220
- 定足数デバイスを保守状態から戻す, 221

## 保守状態にする

定足数デバイス, 220

定足数デバイスを, 220

## 無

無効にする、トランスポートケーブルを, 244

## 名

### 名前空間

グローバル, 97-100

移行, 133

## 命

命名規則, 複製リソースグループ, 386

命名規約, raw ディスクデバイス, 181

## 役

### 役割

カスタム役割を追加, 58

設定, 53

役割の追加, 55

## 有

有効にする、トランスポートケーブルを, 243

## 例

例, クラスタファイルシステムの作成, 184

## 論

論理ホスト名リソース, データ複製フェイル

オーバーでの役割, 386

