



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: 820-0296-11
2007 年 5 月、Revision A

Sun Microsystems, Inc. (以下 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。また、それらに限定されるものではありません。

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。

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 ポリユーモマネージャー、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, Inc. (以下、米国 Sun Microsystems 社とします) 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。

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。

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コービジュアルシステムズ株式会社からライセンス供与されたタイプフェイ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イ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。

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。

Wnn6 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 OMRON Co., Ltd. 1995-2000. All Rights Reserved. Copyright OMRON SOFTWARE Co., Ltd. 1995-2002 All Rights Reserved.

「ATOK」は、株式会社ジャストシステムの登録商標です。

「ATOK Server/ATOK12」は、株式会社ジャストシステムの著作物であり、「ATOK Server/ATOK12」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 Server/ATOK12」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)。

「ATOK Server/ATOK12」に含まれ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。

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。

原典: Sun Cluster System Administration Guide for Solaris OS

Part No: 819-2971-11

Revision A

目次

はじめに	13
1 Sun Cluster の管理の概要	17
Sun Cluster の管理の概要	17
Solaris オペレーティングシステム (Solaris OS) の機能制限	18
管理ツール	18
グラフィカルユーザーインターフェース	19
コマンド行インターフェース	19
クラスタ管理の準備	20
Sun Cluster ハードウェア構成の記録	21
管理コンソールの使用	21
クラスタのバックアップ	22
クラスタ管理の開始	22
▼ リモートからクラスタにログインする	24
▼ クラスタコンソールに安全に接続する	25
▼ clsetup ユーティリティにアクセスする	25
▼ Sun Cluster のパッチ情報を表示する	26
▼ 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	28
▼ クラスタコンポーネントの状態を確認する	30
▼ パブリックネットワークの状態を確認する	32
▼ クラスタ構成を表示する	33
▼ 基本的なクラスタ構成を検証する	42
▼ グローバルマウントポイントを確認する	44
▼ Sun Cluster のコマンドログの内容を表示する	45

2 Sun Cluster と RBAC	49
RBAC の設定と Sun Cluster での使用	49
Sun Cluster RBAC の権限プロファイル	50
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	51
▼管理役割ツールを使用して役割を作成する方法	51
▼コマンド行から役割を作成する方法	53
ユーザーの RBAC プロパティの変更	55
▼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	55
▼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	56
3 クラスタの停止と起動	57
クラスタの停止と起動の概要	57
▼クラスタを停止する	58
▼クラスタを起動する	60
▼クラスタを再起動する	62
単一クラスタノードの停止と起動	64
▼クラスタノードを停止する	65
▼クラスタノードを起動する	68
▼クラスタノードを再起動する	70
▼非クラスタモードでクラスタノードを起動する	73
満杯の /var ファイルシステムを修復する	76
▼満杯の /var ファイルシステムを修復する	76
4 データ複製のアプローチ	77
データ複製についての理解	77
ホストベースのデータ複製の使用法	79
ストレージベースのデータ複製の使用法	80
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	81
ストレージベースの複製を装備した自動フェイルオーバーに関する要件と制 限	82
ストレージベースのデータ複製を使用する際の手動復旧の考慮事項	82
ストレージベースのデータ複製に TrueCopy を使用する際のベストプラク ティス	83
例: 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	83

クラスタにおける Sun StorageTek Availability Suite ソフトウェアの理解	84
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	87
作業マップ: データ複製の構成例	92
クラスタの接続とインストール	93
デバイスグループとリソースグループの構成例	95
▼主クラスタでデバイスグループを構成する	97
▼二次クラスタでデバイスグループを構成する	98
▼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	99
▼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	100
▼主クラスタで複製リソースグループを作成する	102
▼二次クラスタで複製リソースグループを作成する	103
▼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	105
▼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	107
データ複製の有効化例	109
▼主クラスタで複製を有効にする	110
▼二次クラスタで複製を有効にする	113
データ複製の実行例	114
▼リモートミラー複製を実行する	114
▼ポイントインタイムスナップショットを実行する	116
▼複製が正しく構成されていることを確認する	118
フェイルオーバーとスイッチオーバーの管理例	121
▼スイッチオーバーを呼び出す	122
▼DNS エントリを更新する	123
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	125
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	125
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	126
グローバルデバイスでの動的再構成	126
SPARC: VERITAS Volume Manager による管理に関する注意事項	127
ストレージベースの複製されたデバイスの管理	129
▼Hitachi TrueCopy 複製グループを構成する	130
▼複製用に DID デバイスを構成する	131
▼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	133
例: Sun Cluster 向けの TrueCopy 複製グループの構成	134

クラスタファイルシステムの管理の概要	140
クラスタファイルシステムの制限事項	141
SPARC: VxFS サポートについてのガイドライン	141
デバイスグループの管理	142
▼ グローバルデバイス名前空間を更新する	145
▼ 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)	146
デバイスグループを削除して登録を解除する (Solaris ポリリュームマネージャー)	148
▼ すべてのデバイスグループからノードを削除する	148
▼ デバイスグループからノードを削除する (Solaris ポリリュームマネージャー)	149
▼ 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	151
▼ SPARC: 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)	153
▼ SPARC: 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)	154
▼ SPARC: 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)	155
▼ SPARC: 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)	157
▼ SPARC: 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)	157
▼ SPARC: 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)	158
▼ SPARC: ディスクグループの構成変更を登録する (VERITAS Volume Manager) ...	161
▼ 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)	162
▼ 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)	163
▼ SPARC: デバイスグループからボリュームを削除する (VERITAS Volume Manager)	164
▼ SPARC: デバイスグループを削除して登録を解除する (VERITAS Volume Manager)	165
▼ SPARC: デバイスグループにノードを追加する (VERITAS Volume Manager)	166
▼ SPARC: デバイスグループからノードを削除する (VERITAS Volume Manager) ...	167
▼ raw ディスクデバイスグループからノードを削除する	169
▼ デバイスグループのプロパティを変更する	171
▼ デバイスグループの二次ノードの希望数を設定する	173
▼ デバイスグループ構成の一覧を表示する	176
▼ デバイスグループの主ノードを切り替える	177
▼ デバイスグループを保守状態にする	178

ストレージデバイス用の SCSI プロトコル設定の管理	180
▼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	181
▼単一ストレージデバイスの SCSI プロトコルを表示する	182
▼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 を変更する	182
▼単一ストレージデバイスの SCSI プロトコルを変更する	183
クラスタファイルシステムの管理	184
▼クラスタファイルシステムを追加する	185
▼クラスタファイルシステムを削除する	189
▼クラスタ内のグローバルマウントを確認する	191
ディスクパス監視の管理	191
▼ディスクパスを監視する	192
▼ディスクパスの監視を解除する方法	194
▼障害のあるディスクパスを表示する	195
▼ファイルからディスクパスを監視する	195
▼すべての監視対象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有 効にする	198
▼すべての監視対象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 効にする	198
6 定足数の管理	199
定足数の管理の概要	199
定足数デバイスへの動的再構成	201
定足数デバイスの追加	202
▼SCSI 定足数デバイスを追加する	202
▼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	204
▼定足数サーバーを定足数デバイスとして追加する	207
定足数デバイスの削除または交換	211
▼定足数デバイスを削除する	211
▼クラスタから最後の定足数デバイスを削除する	212
▼定足数デバイスを交換する	214
定足数デバイスの保守	214
▼定足数デバイスのノードリストを変更する	215
▼定足数デバイスを保守状態にする	217

▼ 定足数デバイスを保守状態から戻す	219
▼ クラスタ構成を一覧表示する	220
▼ 定足数デバイスを修復する	221
7 クラスタインターコネクとパブリックネットワークの管理	223
クラスタインターコネクの管理	223
クラスタインターコネクでの動的再構成	224
▼ クラスタインターコネクの状態を確認する	225
▼ 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	226
▼ 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	229
▼ クラスタトランスポートケーブルを有効にする	232
▼ クラスタトランスポートケーブルを無効にする	233
▼ 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	235
▼ 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	236
パブリックネットワークの管理	239
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	239
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	241
8 クラスタの管理	243
クラスタの管理の概要	243
▼ クラスタ名を変更する	244
▼ ノード ID をノード名にマップする	245
▼ 新しいクラスタノード認証で作業する	245
▼ クラスタの時刻をリセットする	247
▼ SPARC: ノードで OpenBoot PROM (OBP) を表示する	249
▼ ノードのプライベートホスト名を変更する	250
▼ 非大域ゾーンのプライベートホスト名を追加する	253
▼ 非大域ゾーンのプライベートホスト名を変更する	254
▼ 非大域ゾーンのプライベートホスト名を削除する	255
▼ ノードを保守状態にする	255
▼ ノードを保守状態から戻す	257
クラスタノードの追加	260

▼ ノードを認証ノードリストに追加する	260
ノード上での非大域ゾーンの管理	262
▼ ノード上で非大域ゾーンを作成する	262
▼ ノード上で非大域ゾーンを削除する	264
クラスタノードの削除	264
▼ クラスタソフトウェア構成からノードを削除する	266
▼ 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 ..	269
▼ 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	271
▼ エラーメッセージを修正する	273
ノードのアンインストールに伴う問題の解決	274
Sun Cluster SNMP イベント MIB の作成、設定、および管理	275
▼ SNMP イベント MIB を有効にする	276
▼ SNMP イベント MIB を無効にする	276
▼ SNMP イベント MIB を変更する	277
▼ SNMP ホストを有効にしてノード上の SNMP トラップを受信する	278
▼ SNMP ホストを無効にしてノード上の SNMP トラップを受信しない	278
▼ ノード上に SNMP ユーザーを追加する	279
▼ ノードから SNMP ユーザーを削除する	280
9 CPU 使用率の制御の構成	281
CPU 制御の概要	281
シナリオの選択	282
公平配分スケジューラ	283
CPU 制御の構成	283
▼ SPARC: Solaris 9 OS で CPU 使用率を制御する	283
▼ 大域ゾーンで CPU 使用率を制御する	285
▼ デフォルトのプロセッサセットを使用して非大域ゾーンの CPU 使用率を制御する	287
▼ 専用のプロセッサセットを使用して非大域ゾーンの CPU 使用率を制御する	290
10 Sun Cluster ソフトウェアとファームウェアのパッチ	295
Sun Cluster へのパッチの適用の概要	295
Sun Cluster パッチの適用に関する注意事項	296
クラスタへのパッチの適用	297
▼ 再起動パッチを適用する (ノード)	297

▼再起動パッチを適用する(クラスタ)	301
▼非再起動 Sun Cluster パッチを適用する	305
▼シングルユーザーモードでフェイルオーバーゾーンにパッチを適用する	306
Sun Cluster パッチの変更	309
▼非再起動 Sun Cluster パッチを削除する	309
▼再起動 Sun Cluster パッチを削除する	309
11 クラスタのバックアップと復元	311
クラスタのバックアップ	311
▼バックアップするファイルシステム名を確認する	312
▼完全バックアップに必要なテープ数を判別する	312
▼ルート(/)ファイルシステムをバックアップする	313
▼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)	
)	316
▼SPARC: ボリュームのオンラインバックアップの実行 (VERITAS Volume	
Manager)	319
▼クラスタ構成をバックアップする	323
クラスタファイルの復元の作業マップ	324
▼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)	325
▼ルート(/)ファイルシステムを復元する (/)	325
▼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	327
▼SPARC: 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)	333
▼SPARC: 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)	335
12 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	341
Sun Cluster Manager の概要	341
SPARC: Sun Management Center の概要	342
Sun Cluster Manager の構成	343
RBAC の役割の設定	343
▼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	343
▼Sun Cluster Manager のサーバーアドレスを変更する	344
▼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	345

Sun Cluster Manager ソフトウェアの起動	346
▼ Sun Cluster Manager を起動する	346
A Sun Cluster オブジェクト指向コマンド	349
オブジェクト指向コマンド名および別名	349
オブジェクト指向コマンドセットの概要	350
索引	361

はじめに

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- このマニュアルでは、「x86」という用語は、Intel 32ビット系列のマイクロプロセッサチップ、および AMD が提供する互換マイクロプロセッサチップを意味します。

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。

注- Sun Cluster ソフトウェアは SPARC と x86 の 2 つのプラットフォームで動作します。このマニュアルで説明する情報は、章、節、注、箇条書き、図、表、または例において特に明記しないかぎり、両方のプラットフォームに該当します。

UNIX コマンドの使用

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。 system%
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。	system% su password:
AaBbCc123	変数を示します。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。
『』	参照する書名を示します。	『コードマネージャ・ユーザーズガイド』を参照してください。
「」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第 5 章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	sun% grep '^#define \ XV_VERSION_STRING'

コード例は次のように表示されます。

- C シェル

```
machine_name% command y|n [filename]
```

- C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

- Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

- Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します (例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ (-) は 2 つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

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。すべての Sun Cluster マニュアルは <http://docs.sun.com> で参照できます。

トピック	マニュアル
概要	『Sun Cluster の概要 (Solaris OS 版)』
概念	『Sun Cluster の概念 (Solaris OS 版)』
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 データサービスの計画と管理 (Solaris OS 版)』 各データサービスガイド
データサービスの開発	『Sun Cluster データサービス開発ガイド (Solaris OS 版)』
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 マニュアルの詳細なリストについては、ご使用の Sun Cluster ソフトウェアリリースに対応したリリースノートを <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。

- マニュアル (<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/>)

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)

ご購入先に知らせるシステムの情報を収集するには、次のコマンドを使用してください。

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、以下のクラスタ管理に関する情報と、Sun Cluster 管理ツールの使用手順について説明します。

- 17 ページの「Sun Cluster の管理の概要」
- 18 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」

Sun Cluster の管理の概要

Sun Cluster の高可用性環境によって、エンドユーザーに対して重要なアプリケーションの可用性が保証されます。システム管理者の業務は、Sun Cluster 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 の概念 (Solaris OS 版)』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。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。

- 定期的に(多くの場合は毎日)クラスタを管理および保守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 データサービスの計画と管理 (Solaris OS 版)』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。

ほとんどの場合、Sun Cluster の管理作業はクラスタの稼動中に実行できます。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。

す。Solaris 10 では、明記していないかぎり、Sun Cluster の管理作業は大域ゾーンで行うべきです。クラスタ全体の停止を必要とする手順の場合は、システムへの影響がもっとも少ない稼働時間外に停止時間を予定してください。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。

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以下の Sun Cluster サービスを有効または無効にしないようにしてください。

Sun Cluster サービス	FMRI
pnm	svc:/system/cluster/pnm:default
cl_event	svc:/system/cluster/cl_event:default
cl_eventlog	svc:/system/cluster/cl_eventlog:default
rpc_pmf	svc:/system/cluster/rpc_pmf:default
rpc_fed	svc:/system/cluster/rpc_fed:default
rgm	svc:/system/cluster/rgm:default
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 で管理作業を行うときは、グラフィカルユーザーインタフェース (GUI) またはコマンド行を使用できます。以降の節では、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 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 ツールには、Sun Cluster Manager と Sun Management Center があります (Sun Management Center は SPARC ベースのシステムで Sun Cluster を使用する場合に利用可)。Sun Cluster Manager と Sun Management Center の構成についての詳細と手順は、[第 12 章](#)を参照してください。Sun Cluster Manager の使い方については、各 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 の管理に使用するその他のコマンドを、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 アクセスポリシーの管理します。
<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 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 データサービスのためにリソースを管理します。
<code>clresourcegroup(1CL)</code>	Sun Cluster データサービスのためにリソースを管理します。
<code>clresourcetype(1CL)</code>	Sun Cluster データサービスのために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 を対話的に構成します。
<code>clsnmp(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 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>clvxvm</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。

さらに、コマンドを使用して Sun Cluster 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ボリュームマネージャー (Solstice DiskSuite™、VERITAS Volume Manager、または Solaris ボリュームマネージャー™) によって変わります。

クラスタ管理の準備

この節では、クラスタ管理の準備を整える上で必要な作業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労力を節約できます。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イス(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「リモートからクラスタにログインする」](#)を参照してください。Sun Management Center 用のクラスタコントロールパネルモジュールと 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します。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 ~* を送信し、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 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 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、あるいは、致命的な障害には対処できません。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 または Solaris Volume Manager を使用している場合)

クラスタ管理の開始

表 1-1 に、クラスタ管理の開始について示します。

表 1-1 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。次に以下のアイコンのうちの 1 つを選択します。 cconsolecrlogin、cssh、または ctelnet	24 ページの「リモートからクラスタにログインする」 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	clsetup(1CL) ユーティリティーを起動します。	25 ページの「clsetup 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) コマンドを、 show-rev --v -node サブコマンドおよびオプションを指定して使用してください。	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」

表 1-1 Sun Cluster 3.2 管理ツール (続き)

作業	ツール	参照先
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"> ■ <code>clresource(1CL)</code> ■ <code>clresourcegroup(1CL)</code> ■ <code>clresourcetype(1CL)</code> 	28 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してください。
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 Cluster Manager または Sun Management Center 用の 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 については、オンラインヘルプを参照してください。 Sun Management Center については、Sun Management Center のマニュアルを参照してください。
クラスタコンポーネントの状態を確認します。	<code>status</code> サブコマンドとともに <code>cluster(1CL)</code> コマンドを使用します。	30 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	<code>-m</code> オプションを付けて <code>clnode(1CL) status</code> コマンドを使用します。	32 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。	<code>show</code> サブコマンドとともに <code>cluster(1CL)</code> コマンドを使用します。	33 ページの「クラスタ構成を表示する」
大域マウントポイントの確認	<code>sccheck(1M)</code> コマンドを使用します。	42 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。	45 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
Solaris ボリュームマネージャーの状態を監視する (Solaris 9 または Solaris 10 が動作している場合)	<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管理』

▼ リモートからクラスタにログインする

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ノードコンソールに対して `telnet` 接続を使用します。代わりに、コンソールへのセキュアシェル接続を確立するには、`cconsole` ウィンドウで「オプション」メニューの「SSHの使用」チェックボックスを有効にします。あるいは、`ccp` または `cconsole` コマンドを発行するときに、`-s` オプションを指定します。

詳細は、`ccp(1M)` と `cconsole(1M)` のマニュアルページを参照してください。

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形式もあります。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形式については、[付録 A](#) 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。

- `SUNWccn` 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)` および `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、**System Service Processor (SSP)** にログインします。
 - a. `netcon` コマンドを使用して接続します。
 - 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。

```
# 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**の該当するアイコン (**cconsole**、**crlogin**、**cssh**、**ctelnet**)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

クラスタノードへのセキュアシェル接続を確立するには、この手順を実行します。

始める前に 端末集配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 ファイルではポート番号を使用し、セキュリティー保護された接続を各コンソールアクセスデバイスに使用するようにします。セキュアシェル接続のデフォルトのポート番号は 22 です。

詳細は、clusters(4) および 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 cconsole ユーティリティーをセキュリティー保護されたモードで起動します。

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

-s セキュアシェル接続を有効にします。

-l *username* リモート接続用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* 使用するセキュアシェル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、デフォルトポート番号 22 がセキュリティー保護された接続に使用されます。

▼ clsetup ユーティリティーにアクセスする

clsetup(1CL) ユーティリティーを使用すると、定足数 (quorum)、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、クラスタの新しいノードのオプションを対話形式で構成できます。

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形式もあります。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形式については、[付録 A](#) 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 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。

参照 詳細については、`clsetup` 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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

● Sun Cluster のパッチ情報の表示

```
% showrev -p
```

Sun Cluster 更新リリースは、製品のメインパッチ番号に更新バージョンを加えたものです。

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。

```
% 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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

● Sun Cluster のリリースとバージョン情報を表示する

```
% 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。

```
% clnode show-rev
3.2

% 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
SUNWscspm: 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
SUNWsc rsm: 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 12 章](#) または Sun Cluster Manager のオンラインヘルプを参照してください。

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形式もあります。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形式については、[付録 A](#) 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。

```
% 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。

- resource
- 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。

```
% cluster show 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:                               default
  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。

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形式もあります。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形式については、[付録 A](#) 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。

```
% cluster status
```

例 1-4 クラスタコンポーネントの状態確認

次に、クラスタ `cluster(1CL) status` 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。

```

% 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

▼ 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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

IP ネットワークマルチパスグループの状態を確認するには、`clnode(1CL)` コマンドを `status` サブコマンドを指定して使用します。

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。

```
% 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  qfe1     Online
phys-schost-2     test-rg        Online  qfe1     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。

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形式もあります。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形式については、[付録 A](#) 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。

- クラスタ構成を表示します。

```
% cluster show
```

`cluster` コマンドを使用して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。

例 1-6 クラスタ構成を表示する

次に、クラスタ構成の一覧の例を示します。

```
% 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
globalzoneshares:                 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:                   SUNWsctelemetry
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

```

▼ 基本的なクラスタ構成を検証する

sccheck(1M) コマンドはシステムの構成を検証して、クラスタが機能するために必要な正しい基本構成であるかどうかを判断します。エラーがない場合、sccheck は単にシェルプロンプトに戻ります。エラーがあると、sccheck は、指定された出力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出力します。sccheck を複数のノードに対して実行すると、sccheck は、ノードごとのレポートと複数ノード全体の報告を生成します。

sccheck コマンドは、データ収集のステップと分析のステップからなります。システム構成によっては、データ収集に長い時間がかかることがあります。sccheck に `-v1` フラグを指定し、冗長モードで実行することによって、進捗メッセージを表示できます。あるいは、sccheck に `-v2` フラグを指定し、高冗長モードで実行することによって、より詳細な進捗メッセージを表示できます (特にデータ収集時)。

注 - sccheck 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。

- 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。

```
% su
```

- 2 クラスタ構成を検証します。

```
# sccheck
```

例 1-7 クラスタ構成の検証 (エラーがない場合)

次の例は、sccheck を冗長モードで `phys-schost-1` と `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。

```
# sccheck -v1 -h phys-schost-1,phys-schost-2
```

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

```
sccheck: Multi-node checks finished
#
```

例1-8 クラスタ構成の検証(エラーがある場合)

次の例は、クラスタ suncluster 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/sccheck/myReports/ に作成されます。

```
# sccheck -v1 -h phys-schost-1,phys-schost-2 -o /var/cluster/sccheck/myReports
```

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

sccheck(1M) コマンドには、`/etc/vfstab` ファイルでクラスタファイルシステムとその広域マウントポイントの構成エラーを調べるチェックが含まれます。

注 - `sccheck` 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後で実行してください。

- 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。

```
% su
```

- 2 クラスタ構成を検証します。

```
# sccheck
```

例 1-9 グローバルマウントポイントの確認

次の例は、クラスタ `suncluster` 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。レポートが出力ディレクトリ `/var/cluster/sccheck/myReports/` に送信されます。

```
# sccheck -v1 -h phys-schost-1,phys-schost-2 -o /var/cluster/sccheck/myReports
```

```
sccheck: Requesting explorer data and node report from phys-schost-1.
sccheck: Requesting explorer data and node report from phys-schost-2.
sccheck: phys-schost-1: Explorer finished.
sccheck: phys-schost-1: Starting single-node checks.
sccheck: phys-schost-1: Single-node checks finished.
sccheck: phys-schost-2: Explorer finished.
sccheck: phys-schost-2: Starting single-node checks.
sccheck: phys-schost-2: Single-node checks finished.
sccheck: Starting multi-node checks.
sccheck: Multi-node checks finished.
sccheck: One or more checks failed.
sccheck: The greatest severity of all check failures was 3 (HIGH).
sccheck: Reports are in /var/cluster/sccheck/myReports.
#
# cat /var/cluster/sccheck/myReports/sccheck-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/sccheck/myReports/sccheck-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、このファイルにロギングされません。

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
- clsnmpghost
- clsnmpmib
- clsnmpuser
- cltelemetryattribute
- cluster
- scconf
- 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 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す。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 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。

```
# more /var/cluster/logs/commandlog
```

例 1-10 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。

- 49 ページの「RBAC の設定と Sun Cluster での使用」
- 50 ページの「Sun Cluster RBAC の権限プロファイル」
- 51 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」
- 55 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、それを Sun Cluster で使用するための手順については、この章の後の方で説明します。

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、RBAC 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、大域ゾーンと非大域ゾーンの両方に適用されます。

`solaris.cluster.read` 一覧表示、表示、およびその他の読み取り操作の承認。

`solaris.cluster.admin` クラスタオブジェクトの状態を変更する承認。

`solaris.cluster.modify` 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、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 Commands	なし。ただし、 <code>eid=0</code> 指定で実行される Sun Cluster コマンドのリストが含まれます。	以下のサブコマンドがあるすべての Sun Cluster コマンドを含む、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"> ■ <code>list</code> ■ <code>show</code> ■ <code>status</code> <code>scha_control(1HA)</code> <code>scha_resource_get(1HA)</code> <code>scha_resource_setstatus(1H)</code> <code>scha_resourcegroup_get(1HA)</code> <code>scha_resourcetype_get(1HA)</code>
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
	<code>solaris.cluster.read</code>	Sun Cluster Manager へのアクセス、ならびに Sun Cluster コマンドの一覧表示、表示、およびそのほかの読み取り操作の実行。
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、およびそのほか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

この作業を使用して、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。

管理役割ツールを実行し、Solaris Management Console を起動するには、『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。次に、「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)」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
役割の正式名称	正式名
説明	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割り当てます。
useradd(1M) コマンドを使って、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 ユーザーを追加したら、「完了 (Finish)」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、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。
 - また同じくローカルスコープの役割を作成する場合、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、入力ミスが発生しやすいため、緊急時以外はできるだけ使用しないでください。
 - 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 :: <primaryadmin パスワードを入力する>

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 :: <primaryadmin パスワードを入力する>

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。

- 55 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」
- 56 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、Primary Administrator 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。次に、「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の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。

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の承認 `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、入力ミスが発生しやすいため、緊急時以外はできるだけ使用しないでください。

- 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ときは、`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、クラスタと個々のクラスタ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の詳細は、『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。

- 57 ページの「クラスタの停止と起動の概要」
- 64 ページの「単一クラスタノードの停止と起動」
- 76 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概要は、[表 3-1](#) 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、クラスタサービスを正しい順序で停止し、クラスタ全体をクリーンに停止します。`cluster shutdown` コマンドは、クラスタの場所を移動するときにも使用できます。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、クラスタを停止するときにも、このコマンドを使用できます。

注-クラスタ全体を正しく停止するには、`shutdown` や `halt` コマンドではなく、`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、[58 ページの「クラスタを停止する」](#)、または [64 ページの「単一クラスタノードの停止と起動」](#) を参照してください。

`cluster shutdown` コマンドは、次のアクションを実行することによりクラスタ内のすべてのノードを停止します。

1. すべての実行中のリソースグループをオフラインにする。
2. 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. アクティブなデバイスサービスを停止する。
4. `init 0` を実行してすべてのノードを OpenBoot™ PROM ok プロンプトの状態にする (SPARC ベースシステムの場合) か、あるいは GRUB メニューの状態にする (x86 ベースシステムの場合)。GRUB メニューの詳細は、『Solaris のシステム管理 (基本編)』の第 11 章「GRUB ベースのブート (手順)」に記載されています。

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、73 ページの「非クラスタモードでクラスタノードを起動する」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 - <code>cluster(1CL shutdown</code> を使用	58 ページの「クラスタを停止する」を参照
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。	60 ページの「クラスタを起動する」を参照
クラスタの再起動 - <code>cluster shutdown</code> を使用 「Press any key to continue」というメッセージが表示された時点で、キーを押すことにより各ノードを個別に起動します。 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。	62 ページの「クラスタを再起動する」を参照

▼ クラスタを停止する



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。この機能はクラスタ内ではサポートされません。

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形式もあります。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形式については、付録 A を参照してください。

- 1 **SPARC:Oracle Parallel Server** または **Oracle Real Application Clusters (RAC)** が動作しているクラスタの場合、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。
- 3 ただちにクラスタを停止します。
クラスタ内の1つのノードから、次のコマンドを入力します。
cluster shutdown -g0 -y
- 4 **SPARC** ベースシステムでは、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** ベースシステムでは、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 ベースのシステムではすべてのクラスタノードが `ok` プロンプトになるまで、**x86** ベースシステムではすべてのクラスタノードが `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。
cluster status -t node
- 5 必要であればノードの電源を切ります。

例 3-1 SPARC: クラスタの停止

次に、正常な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、クラスタ内の他のノードのコンソールにも表示されます。

```
# 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 3-2 x86: クラスタの停止

次に、正常な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、クラスタ内の他のノードのコンソールにも表示されます。

```
# 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/K05initrgrm: 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 60 ページの「[クラスタを起動する](#)」を参照し、停止したクラスタを再起動します。

▼ クラスタを起動する

この手順では、ノードが停止され、`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) が表示されている状態の、クラスタを起動する方法を説明します。

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形式もあります。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形式については、[付録 A](#) を参照してください。

1 各ノードをクラスタモードで起動します。

- SPARC ベースのシステムでは、次の操作を実行します。

```
ok boot
```

- x86 ベースのシステムでは、次の操作を実行します。

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、クラスタ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 ベースの起動の詳細は、『Solaris のシステム管理 (基本編)』の第 11 章「GRUB ベースのブート (手順)」を参照してください。

- 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(1CL) ステータスコマンドはノードのステータスを報告します。
cluster status -t node

注-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。この問題が発生した場合は、[76 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 3-3 SPARC: クラスタの起動

次に、ノード phys-schost-1 を起動して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。クラスタ内の他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。

```

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

`cluster(1CL) shutdown` コマンドを実行し、クラスタを停止してから、各ノード上で `boot(1M)` コマンドを使用してクラスタを起動します。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- 1 **SPARC:Oracle Parallel Server** または **Oracle RAC** が動作しているクラスタの場合、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。
- 3 クラスタを停止します。

クラスタ内の1つのノードから、次のコマンドを入力します。

```
# cluster shutdown -g0 -y
```

各ノードが停止します。

注-クラスタメンバーシップを取得できるように、クラスタ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。

- 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 ベースのシステムでは、次の操作を実行します。

```
ok boot
```

- x86 ベースのシステムでは、次の操作を実行します。

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、クラスタ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 ベースの起動の詳細は、『Solaris のシステム管理 (基本編)』の第 11 章「GRUB ベースのブート (手順)」を参照してください。

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。

scstat コマンドを使用してノードの状態を表示します。

```
# cluster status -t node
```

注-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。この問題が発生した場合は、76 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: クラスタの再起動

次に、正常な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、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 オプションで猶予期間をゼロに設定し、-y 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、クラスタ内の他のノードのコンソールにも表示されます。

```

# 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:
```

単一クラスタノードの停止と起動

注 - `clnode(1CL) evacuate` コマンドを、Solaris の `shutdown(1M)` コマンドとともに使用して、個別のノードを停止します。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。非大域ゾーンの停止と起動についての詳細は、『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、およびアンインストール(手順)」を参照してください。

表 3-2 作業マップ:クラスタノードの停止と起動

作業	ツール	参照先
クラスタノードの停止	<code>clnode(1CL) evacuate</code> コマンドおよび <code>shutdown</code> コマンドを使用	65 ページの「クラスタノードを停止する」
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タとの動作中の接続が必要です。	<code>boot</code> または <code>b</code> コマンドを使用	68 ページの「クラスタノードを起動する」
クラスタ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タとの動作中の接続が必要です。	<code>clnode evacuate</code> および <code>shutdown</code> コマンドを使用	70 ページの「クラスタ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	<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> または <code>shutdown -g -y -i0</code> コマンドを使用	73 ページの「非クラスタモードでクラスタノードを起動する」

▼ クラスタノードを停止する



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。この機能はクラスタ内ではサポートされません。

非大域ゾーンの停止についての詳細は、『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、およびアンインストール (手順)」を参照してください。

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形式もあります。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形式については、付録 A を参照してください。

- 1 **SPARC:Oracle Parallel Server** または **Oracle RAC** が動作しているクラスタの場合、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。

- 3 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クラスタノードに切り替えます。

停止する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、指定のノードから、次に優先される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。

- 4 クラスタノードを停止します。

停止するノードで次のようにコマンドを入力します。

```
# shutdown -g0 -y -i0
```

SPARC ベースのシステムではクラスタノードが `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。

- 5 必要であればノードの電源を切ります。

例 3-5 SPARC: 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、クラスタ内の他のノードのコンソールにも表示されます。

```
# clnode evacuate -S -h phys-schost-1
# shutdown -T0 -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 3-6 x86: 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、クラスタ内の他のノードのコンソールにも表示されます。

```

# clnode evacuate phys-schost-1
# shutdown -T0 -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

```

参照 [68 ページの「クラスタノードを起動する」](#) を参照し、停止したクラスタノードを再起動します。

▼ クラスタノードを起動する

クラスタ内の他のアクティブノードを停止または再起動したい場合は、少なくとも起動中のノードで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、multi-user-server 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ぎません。非大域ゾーンの起動についての詳細は、『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、およびアンインストール (手順)」を参照してください。

注-クラスタノードの起動方法は、定足数 (quorum) 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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

1 停止した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。

- SPARC ベースのシステムでは、次の操作を実行します。

```
ok boot
```

- x86 ベースのシステムでは、次の操作を実行します。

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。

注-クラスタメンバーシップを取得できるように、クラスタ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。

- 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 コマンドは、ノードのステータスを報告します。

```
# cluster status -t node
```

注-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。この問題が発生した場合は、[76 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 3-7 SPARC: クラスタノードの起動

次に、ノード phys-schost-1 を起動して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:
```

▼ クラスタノードを再起動する

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したい場合は、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、multi-user-server 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ぎません。非大域ゾーンの再起動についての詳細は、『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、およびアンインストール (手順)」を参照してください。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- 1 **SPARC: Oracle Parallel Server/Real Application Clusters** が動作しているクラスタノードの場合、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。

- 3 `clnode evacuate` および `shutdown` コマンドを使用してクラスタノードを停止します。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 ベースのシステムでは、次の操作を実行します。

```
# clnode evacuate node  
# shutdown -g0 -y -i6
```

- x86 ベースのシステムでは、次の操作を実行します。

```
# clnode evacuate node
```

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、クラスタ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。

- 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。

```
# cluster status -t node
```

例 3-8 SPARC: 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、クラスタ内の他のノードのコンソールに表示されます。

```
# clnode evacuate phys-schost-1
# 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 3-9 x86: 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、クラスタ内の他のノードのコンソールに表示されます。

```
# clnode evacuate phys-schost-1
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:
```

▼ 非クラスタモードでクラスタノードを起動する

ノードは、クラスタメンバーシップに参加しないよう、つまり非クラスタモードで起動できます。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、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。

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形式もあります。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形式については、[付録 A](#) 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 `clnode evacuate` および `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。

```
# clnode evacuate node
# shutdown -g0 -y
```

- 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、以下のコマンドを実行します。

```
phys-schost# boot -xs
```

- x86 ベースのシステム上で、以下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適切な Solaris エントリを選択し、`e` を入力してその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 ベースの起動の詳細は、『Solaris のシステム管理 (基本編)』の第 11 章「GRUB ベースのブート (手順)」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 -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`を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 3-10 SPARC: 非クラスタモードで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` で起動します。このノードの停止メッセージは、クラスタ内の他のノードのコンソールにも表示されます。

```
# clnode evacuate phys-schost-1
# cluster shutdown -T0 -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「システムメッセージの表示」を参照してください。

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形式もあります。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形式については、[付録 A](#) 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 で利用可能なデータ複製のアプローチについて説明します。クラスタに最適なサービスを提供する複製アプローチの組み合わせを選択するためには、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。

Sun Cluster のこのリリースは、Sun の Availability Suite ソフトウェアの次のリリースをサポートしています。

- Sun StorageTek Availability Suite 4
- Sun StorEdge Availability Suite 3.2.1

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。

この章の内容は次のとおりです。

- 77 ページの「データ複製についての理解」
- 79 ページの「ホストベースのデータ複製の使用法」
- 80 ページの「ストレージベースのデータ複製の使用法」
- 83 ページの「例: 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」

データ複製についての理解

「データ複製」とは、主ストレージデバイスからバックアップまたは二次ストレージデバイスへのデータのコピーです。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。このようにして、データ複製を使用すると、クラスタの高可用性と耐災害性を確保できます。

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、特別なソフトウェアを使用して、地理的に離れたノード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、主ノードのマスターボリュームのデータを、地理的に離れた二次ノード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。

ホストベースのデータ複製は、ストレージレイを必要としないため、比較的安価なデータ複製ソリューションです。ホストベースのデータ複製は、ローカルに接続されたディスクを使用します。ただし、ホストベースのデータ複製はデータの複製にホストのリソースを消費します。また、Oracle RACなどのスケラブルアプリケーションをサポートしません。構内クラスタ環境でのホストベースのデータ複製の使用法の詳細については、[79 ページの「ホストベースのデータ複製の使用法」](#)を参照してください。2つ以上のクラスタ間でのホストベースのデータ複製の使用法についての詳細は、『Sun Cluster Geographic Edition Data Replication Guide for Hitachi TrueCopy』を参照してください。

- 「ストレージベースのデータ複製」は、特別な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。このソフトウェアリロケーション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。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場合があります。これは、この種類のデータ複製はスケラブルアプリケーションをサポートし、ホストの負担を軽減するためです。また、ストレージベースの複製は、Oracle RACなどのスケラブルアプリケーションもサポートします。構内クラスタ環境でのストレージベースのデータ複製の使用法の詳細については、[80 ページの「ストレージベースのデータ複製の使用法」](#)を参照してください。2つ以上のクラスタ間でのストレージベースの複製の使用法、およびこの手順を自動化する Sun Cluster GeoEdition 製品についての詳細は、『Sun Cluster Geographic Edition Data Replication Guide for Sun StorageTek Availability Suite』を参照してください。

Sun StorageTek Availability Suite ソフトウェアは、地理的に離れたクラスタにわたるホストベースのデータ複製のためのメカニズムを提供します。この章の最後にある[83 ページの「例: 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」](#)では、このようなクラスタ構成の完全な例を示します。

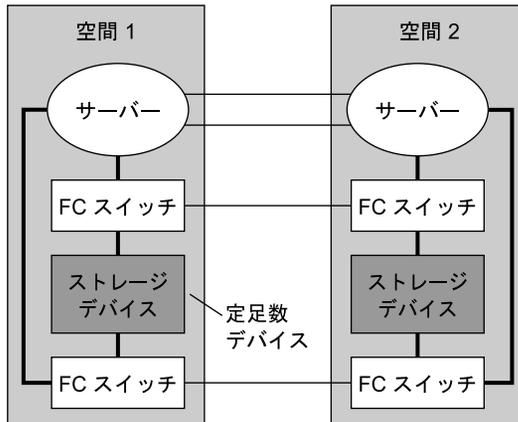
ホストベースのデータ複製の使用法

この節では、2つの場所に設置された構内クラスタにおけるホストベースのデータ複製を説明します。ホストベースのデータ複製を装備した、2つの場所に設置されたクラスタ構成は次のように定義されます。

- 2つの独立した空間。
- 各空間にはノード1個と複数のディスクサブシステムを配置。
- 空間内のディスクサブシステム間でデータを複製。
- 少なくとも1つのディスクサブシステム。これは両方のホストに接続され、定足数デバイスとして使用され、またいずれかの空間にあります。

注- この節の例は一般的な構内クラスタ構成を示したもので、必須構成や推奨構成を示すものではありません。説明を簡単にするため、図や説明は、構内クラスタリングの理解に固有な機能のみを集中的に扱います。たとえば、パブリックネットワークのEthernet接続は示してありません。

この構成では、定足数ディスクが失われると、システムは自動的に復旧できなくなります。復旧にはSunのサービスプロバイダによる介入が必要になります。



—— シングルモードファイバ

—— マルチモードファイバ

図 4-1 ホストベースのデータ複製を装備した2ヶ所に設置された構内クラスタ (マルチパスなし)

図 4-1 は、標準的な非構内構成に似ています。構内クラスタでは、マルチモードからシングルモードファイバに切り替えるため、ファイバチャネルスイッチが追加されています。

ストレージベースのデータ複製の使用法

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複製を管理します。このようなソフトウェアは、使用するそれぞれのストレージデバイスに固有なものです。ストレージベースのデータ複製を構成する際には、常に、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 ソフトウェアを使用して、手動と自動両方による複製物の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 4-2 に、データが2つのストレージアレイ間で複製される2ヶ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アレイがあり、これが両方の場所のノードにデータを提供します。また主ストレージアレイは、複製されたデータを二次ストレージアレイに提供します。

通常のクラスタ稼働中、二次ストレージアレイはクラスタからは認識できません。ただし、主ストレージアレイが使用不能になると、Sun のサービスプロバイダにより、二次ストレージアレイを手動でクラスタ内に構成することができます。

注 - 図 4-2 に示すように、複製された側ではないボリューム上に定足数デバイスがあります。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。

使用されるアプリケーションの種類に応じて、Sun Cluster 環境では、ストレージ

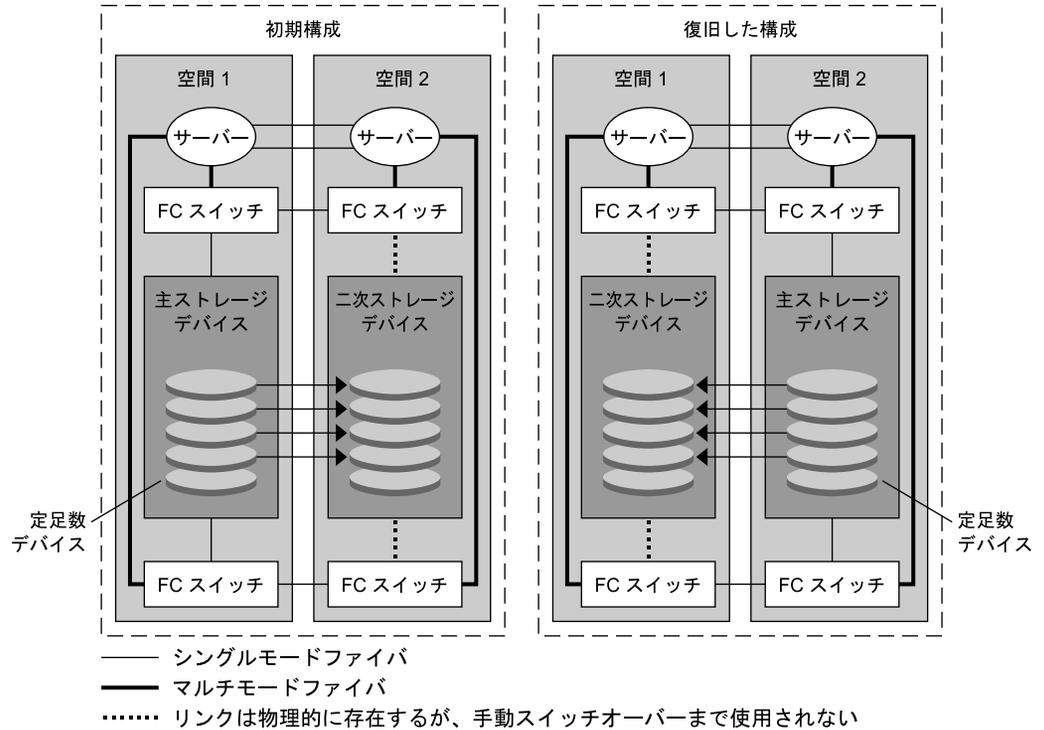


図4-2 ストレージベースのデータ複製を装備した2ヶ所に設置されたクラスタ構成ベースのデータ複製を同期または非同期に実行できます。

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

データの完全性を確保するため、マルチパスと適切な 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構内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 とインターコネクトインフ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。すべての定足数デバイスは、複製された側ではないボリューム上に配置します。

- 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アクセスしようとする場合があり、二次コピーは読み取り専用であるため、データ破損が発生する可能性があります。
- 複製されたデバイスを使用するディスクグループまたはディスクセットを作成する場合は、ディスクグループ、またはディスクセットと Hitachi TrueCopy レプリカのペアには同じ名前を使用します。
- 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レイに付属するマニュアルを参照してください。
- 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、[129 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。

ストレージベースの複製を装備した自動フェイルオーバーに関する要件と制限

ストレージベースのデータ複製と自動フェイルオーバーの併用には、次の制限が適用されます。

- Oracle Real Application Clusters (RAC) はサポートされません。
- 同期モードのみがサポートされます。
- 複製されたデバイスは定足数デバイスになることができません。
- CVM および Sun Cluster 用 Solaris ボリュームマネージャーはサポートされません。

ストレージベースのデータ複製を使用する際の手動復旧の考慮事項

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、[\(図 4-2 に示すように\)](#) 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い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に TrueCopy 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 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、次の手順を守ってください。

- データの古いコピーへのフェイルオーバーを回避するため、常に最高のフェンスレベル data を使用します。
- リソースグループ 1 つにつき 1 つの Hitachi TrueCopy デバイスグループを作成します。クラスタリソースグループ、クラスタデバイスグループ、VxVM ディスクグループ、および Hitachi TrueCopy デバイスグループの間には一対一の関係が存在すべきです。
- 同一の Hitachi TrueCopy 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。
- 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。

例: 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

この節では、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、ホストベースのデータをクラスタ間で複製する完全な構成例を示します。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。すべてのタスクは大域ゾーンで行うべきです。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 の役割の使用法の詳細については、[第 2 章](#)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

この節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わ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。耐障害性のベースは、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(1RPC)` と `iiadm(1II)` 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 4-3](#) は、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。

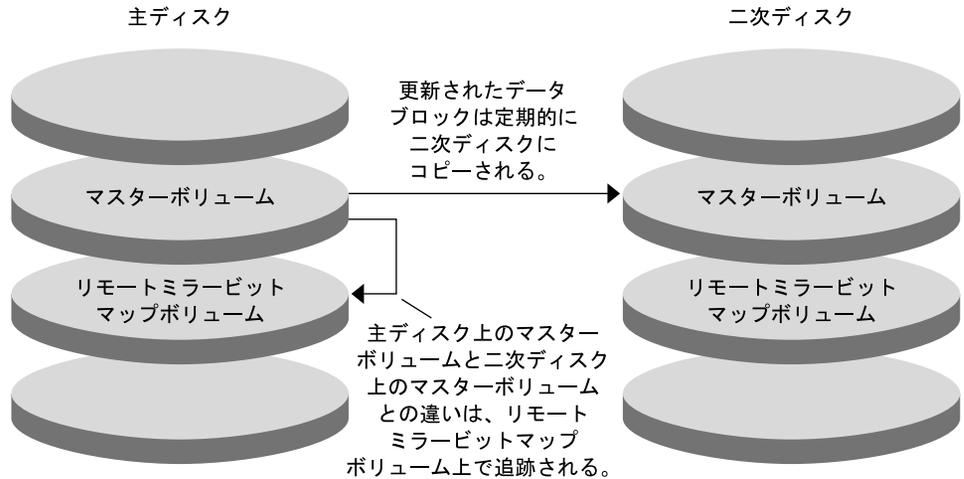


図4-3 リモートミラー複製

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4-4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。

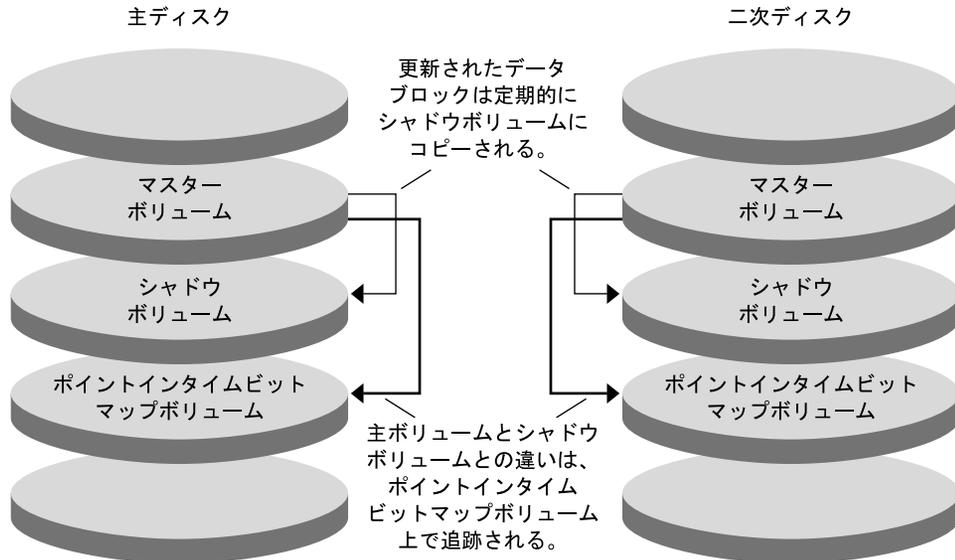


図4-4 ポイントインタイムスナップショット

構成例での複製

図4-5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。

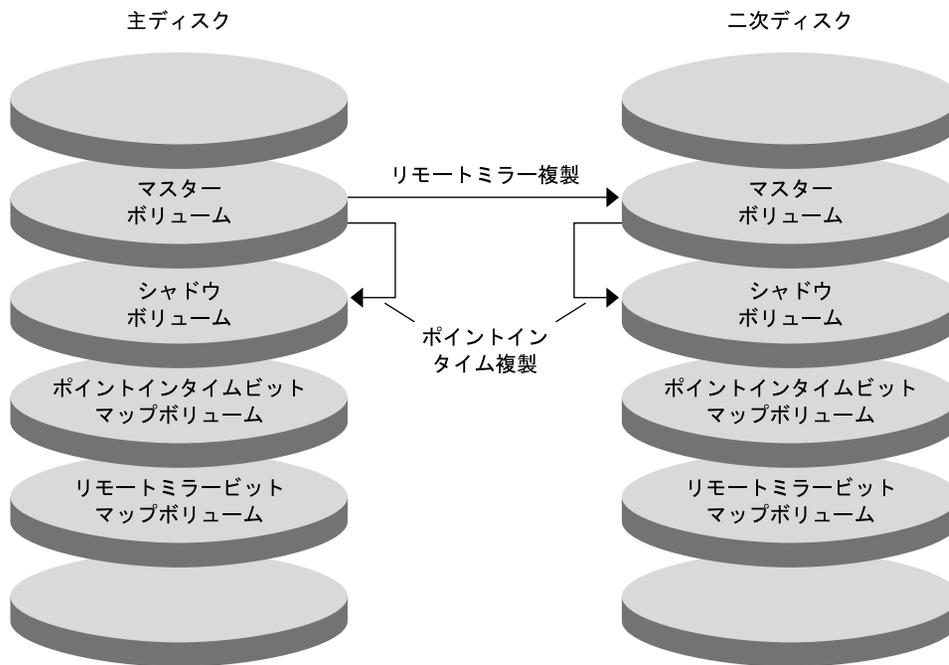


図4-5 構成例での複製

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。

- 88 ページの「複製リソースグループの構成」
- 88 ページの「アプリケーションリソースグループの構成」
 - 89 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」
 - 90 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」
- 92 ページの「フェイルオーバーまたはスイッチオーバーの管理ガイドライン」

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。フェイルオーバーまたはスイッチオーバーの後は、二次クラスタが論理ホスト名のホストになる必要があります。ドメインネームシステム (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スイッチオーバーします。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スイッチオーバーします。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す。
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スイッチオーバーと呼ばれます。

HAStoragePlus については、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。

- 89 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」
- 90 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」

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スイッチオーバーさせる

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スイッチオーバーすると、デバイスグループと複製リソースグループもスイッチオーバーします。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スイッチオーバーまたは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スイッチオーバーや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らないません。

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スイッチオーバーまたは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スイッチオーバーやフェイルオーバーは行われません。スイッチオーバーや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からは制御されません。

HAStoragePlus については、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 4-6 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す図

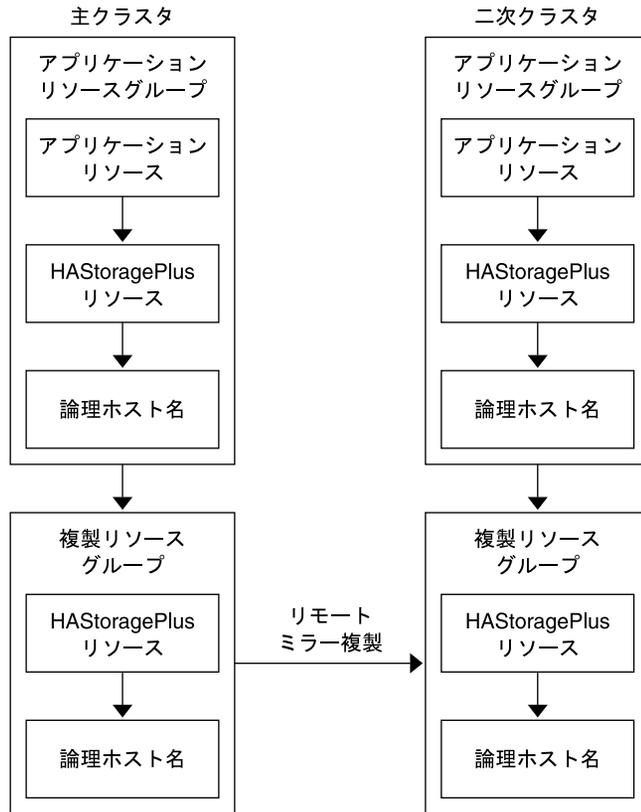


図 4-6 フェイルオーバーアプリケーションでのリソースグループの構成

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 4-7 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す図

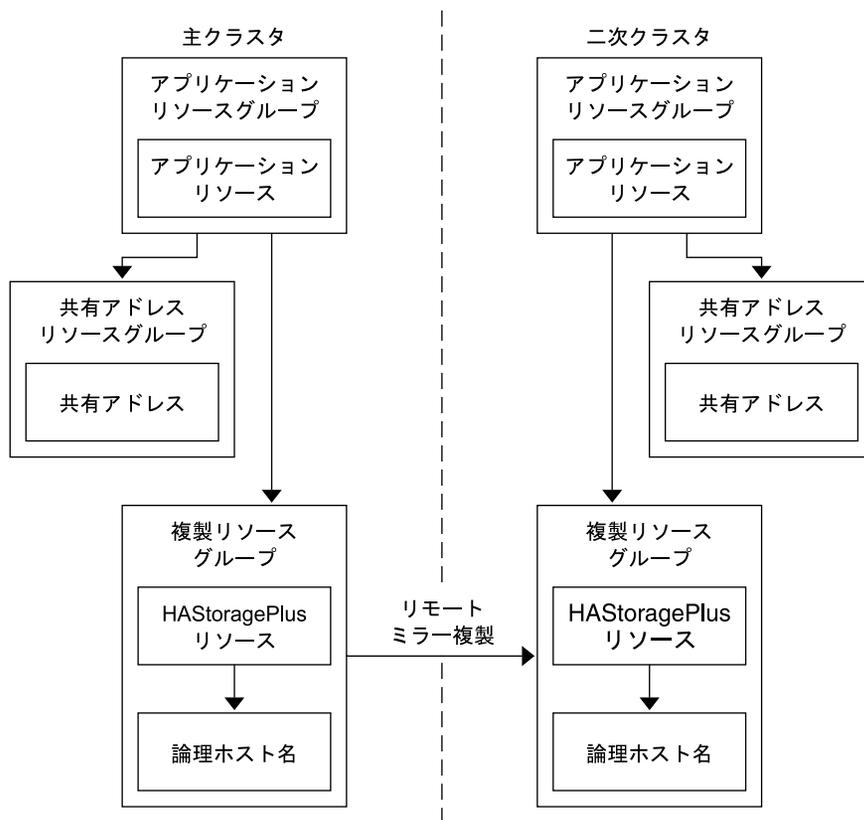


図 4-7 スケーラブルアプリケーションでのリソースグループの構成

フェイルオーバーまたはスイッチオーバーの管理ガイドライン

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。フェイルオーバーまたはスイッチオーバーの後、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 4-8 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す図

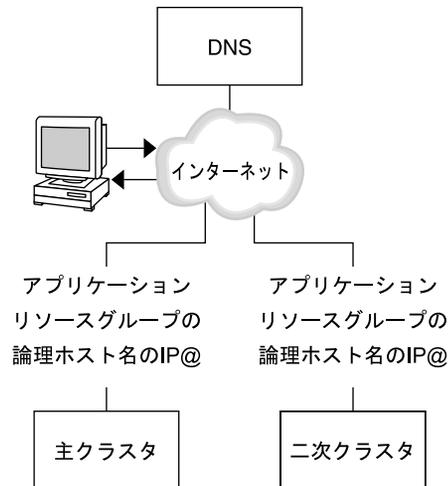


図 4-8 クライアントからクラスタへの DNS マッピング

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。フェイルオーバーやスイッチオーバーの管理方法の例については、121 ページの「フェイルオーバーとスイッチオーバーの管理例」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 4-1 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データ複製を構成する方法の、この例の作業を示します。

表 4-1 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。	93 ページの「クラスタの接続とインストール」
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。	95 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。	110 ページの「主クラスタで複製を有効にする」 113 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。	114 ページの「リモートミラー複製を実行する」 116 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。	118 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 4-9 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。

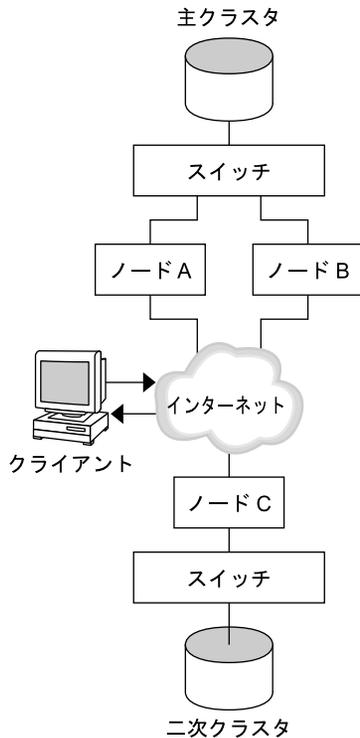


図4-9 クラスタ構成例

表4-2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4-2 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、『Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS』を参照してください。
ディスクの空き容量	約 15M バイト

表 4-2 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。 インストールの詳細は、『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。
Sun Cluster ソフトウェア	Sun Cluster 3.2 ソフトウェア インストールの詳細は、『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。
ボリューム管理ソフトウェア	Solstice DiskSuite または 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。 インストールの詳細は、『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"> ■ Sun StorEdge Availability Suite 3.1 - Sun StorEdge Availability のマニュアル ■ Sun StorEdge Availability Suite 3.2 - Sun StorEdge Availability のマニュアル ■ Sun StorageTek Availability Suite 4.0 - Sun StorageTek Availability のマニュアル
Sun StorageTek Availability Suite ソフトウェアパッチ	最新のパッチについては、 http://www.sunsolve.com を参照

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、88 ページの「複製リソースグループの構成」および 88 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。

ここで説明する手順は次のとおりです。

- 97 ページの「主クラスタでデバイスグループを構成する」
- 98 ページの「二次クラスタでデバイスグループを構成する」
- 99 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」
- 100 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」
- 102 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」
- 103 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」
- 105 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」
- 107 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」
- 118 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 4-3 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。Solstice DiskSuite または Solaris ボリュームマネージャ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。

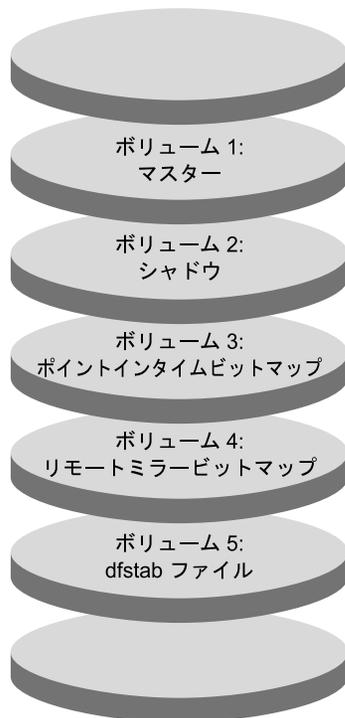


図 4-10 デバイスグループのボリューム

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なりません。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。
 - 84 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」
 - 87 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」

- 93 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または RBAC の承認 `solaris.cluster.modify` を提供する役割を使用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 4-9](#)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 [98 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97 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または RBAC の承認 `solaris.cluster.modify` を提供する役割を使用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 99 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98 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、スーパーユーザーまたは 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、次のように指定します。

```
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 4-10](#) 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 100 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99 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、スーパーユーザーまたは RBAC の承認 solaris.cluster.admin を提供する役割になります。

- 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、次のように指定します。

```
nodeC# mkdir /global/mountpoint
```

- 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
```

- 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。

- 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
```

- 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
```

- 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
```

- 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
```

- nodeC に vol05 をマウントします。

```
nodeC# mount /global/etc
```

- 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。

- 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 102 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100 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または RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割を使用してアクセスします。
- 2 SUNW.HASStoragePlus というリソース型を登録します。
nodeA# `clresourcetype register SUNW.HASStoragePlus`
- 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.HASStoragePlus リソースを追加します。
nodeA# `clresource create -g devgrp-stor-rg -t SUNW.HASStoragePlus \`
`-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.HASStoragePlus 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は、SUNW.HASStoragePlus(5)のマニュアルページを参照してください。

- 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。

- 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。

- 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 [103 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102 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。

- nodeC にスーパーユーザーまたは RBAC の承認 solaris.cluster.modify、solaris.cluster.admin、および solaris.cluster.read を提供する役割を使用してアクセスします。

- SUNW.HASStoragePlus というリソースタイプを登録します。

```
nodeC# clresourcetype register SUNW.HASStoragePlus
```

- 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.HASStoragePlus リソースを追加します。

```
nodeC# clresource create \  
-t SUNW.HASStoragePlus \  
-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.HASStoragePlus 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 HASStoragePlus リソース

これらの拡張プロパティについては、SUNW.HASStoragePlus(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# clreslogicalhostname 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 105 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103 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または RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割を使用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 107 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105 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または RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割を使用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。
- 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。
- 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.HAStoragePlus リソースを追加します。

```
nodeC# 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

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は、SUNW.HAStoragePlus(5) のマニュアルページを参照してください。

- 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。

- 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
```

- 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。

- 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 [109 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。これらのコマンドについての詳細は、Sun StorageTek Availability のマニュアルを参照してください。

ここで説明する手順は次のとおりです。

- 110 ページの「主クラスタで複製を有効にする」
- 113 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または RBAC の承認 `solaris.cluster.read` を提供する役割を使用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みます。

次の手順 113 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110 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 114 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。これらのコマンドについての詳細は、Sun StorageTek Availability Suite のマニュアルを参照してください。

ここで説明する手順は次のとおりです。

- 114 ページの「リモートミラー複製を実行する」
- 116 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」
- 118 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 [116 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114 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または RBAC の承認 `solaris.cluster.modify` および `solaris.cluster.admin` を提供する役割を使用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/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 -l 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
```

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/rdsk/devgrp/vol02
nodeA# /usr/opt/SUNWesm/sbin/iiadm -w /dev/vx/rdsk/devgrp/vol02
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/iiadm -u s /dev/vx/rdsk/devgrp/vol02
nodeA# /usr/sbin/iiadm -w /dev/vx/rdsk/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/rdsk/devgrp/vol02
nodeC# /usr/opt/SUNWesm/sbin/iiadm -w /dev/vx/rdsk/devgrp/vol02
```

- Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/iiadm -u s /dev/vx/rdsk/devgrp/vol02
nodeC# /usr/sbin/iiadm -w /dev/vx/rdsk/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/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync
```

次の手順 118 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116 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または **RBAC** の承認 `solaris.cluster.admin` を提供する役割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/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。

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/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
```

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 [121 ページの「フェイルオーバーとスイッチオーバーの管理例」](#)

フェイルオーバーとスイッチオーバーの管理例

この節では、スイッチ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。スイッチオーバーまたはフェイルオーバーのあと、DNS エントリを更新します。詳細については、[92 ページの「フェイルオーバーまたはスイッチオーバーの管理ガイドライン」](#)を参照してください。

ここで説明する手順は次のとおりです。

- 122 ページの「スイッチオーバーを呼び出す」
- 123 ページの「DNS エントリを更新する」

▼ スイッチオーバーを呼び出す

- 1 nodeA および nodeC にスーパーユーザーまたは **RBAC** の承認 `solaris.cluster.admin` を提供する役割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 [123 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 4-8](#)を参照してください。

始める前に 手順[122 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。

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。

- 125 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」
- 129 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」
- 140 ページの「クラスタファイルシステムの管理の概要」
- 142 ページの「デバイスグループの管理」
- 180 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」
- 184 ページの「クラスタファイルシステムの管理」
- 191 ページの「ディスクパス監視の管理」

この章に関連する手順の概要については、表 5-3 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 の概念 (Solaris OS 版)』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非大域ゾーン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

広域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 Detach 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、201 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	参照先
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替えを実行	177 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 削除操作を実行します。	「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』

SPARC: VERITAS Volume Manager による管理に関する注意事項

- Sun Cluster 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、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 Parallel Server または 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。Oracle Parallel Server または 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 Parallel Server または Oracle RAC 用の共有ディスクグループをインポートするには、`vxdg -s` を使用する必要があります。Oracle Parallel Server または 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、153 ページの「SPARC: 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 は、ストレージベースの複製用ソフトウェアとして、Hitachi TrueCopy ソフトウェアをサポートしています。

Hitachi TrueCopy ソフトウェアを使用してデータを複製するためには、Hitachi TrueCopy のマニュアルによく目を通し、Hitachi TrueCopy 製品と最新の Hitachi TrueCopy パッチを、使用しているシステムにインストールする必要があります。Hitachi TrueCopy ソフトウェアのインストール方法については、Hitachi TrueCopy 製品のマニュアルを参照してください。

Hitachi TrueCopy 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 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、`horcm takeover` を直接実行することによって主複製を移動すべきではありません。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。

次の表に、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: 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 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。
Hitachi 複製グループを構成する。	130 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。	131 ページの「複製用に DID デバイスを構成する」
複製されたグループを登録する。	146 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)」
構成を確認する。	133 ページの「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。

1 ストレージアレイに接続されたノードで、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。

```
# 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 DATA ,12345 29 -
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR DATA ,----- 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 DATA ,12345 29 -
```

```
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR DATA ,----- 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 DATA ,12345 29 -
```

```
group-name pair1(R) (CL1-A , 0, 29)12345 29..P-VOL PAIR DATA ,----- 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 DATA ,12345 29 -
```

```
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR DATA ,----- 58 -
```

次の手順 131 ページの「複製用に DID デバイスを構成する」の手順に従って、複製されたデバイスの構成を続けます。

▼ 複製用に DID デバイスを構成する

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。

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形式もあります。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形式については、[付録 A](#) 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。

```
# /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 DATA ,12345 29 -
group-name pair1(R) (CL1-A , 0, 29)12345 29..S-VOL PAIR DATA ,----- 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により両方のノード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 device
```

- 6 TrueCopy 複製が設定されていることを確認します。

```
# cldevice show device |
```

- 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、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ノード上で `cldevice combine` コマンドを実行します。このノードはローカルでなければなりません。

```
# 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。

- 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。

- 146 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)」または 158 ページの「SPARC: 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際には、必ず TrueCopy 複製グループと同じ名前を指定します。

- 133 ページの「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」

▼ 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、まずそれらを作成します。Solstice DiskSuite または Solaris ボリュームマネージャ デバイスグループの作成についての詳細は、146 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)」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の詳細は、154 ページの「SPARC: 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。

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形式もあります。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形式については、[付録 A](#) 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 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。

```
# 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、[130 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 DATA ,----- 58 -
VG01 pair1(R) (CL1-C , 0, 9)20064 58..P-VOL PAIR DATA ,61114 29 -
VG01 pair2(L) (CL1-A , 0, 30)61114 30..S-VOL PAIR DATA ,----- 59 -
```

例 5-4 ノード 1 上の pairdisplay コマンドの出力 (続き)

```
VG01 pair2(R) (CL1-C , 0, 10)20064 59..P-VOL PAIR DATA ,61114 30 -
VG01 pair3(L) (CL1-A , 0, 31)61114 31..S-VOL PAIR DATA ,----- 60 -
VG01 pair3(R) (CL1-C , 0, 11)20064 60..P-VOL PAIR DATA ,61114 31 -
```

例 5-5 ノード 2 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 DATA ,----- 58 -
VG01 pair1(R) (CL1-C , 0, 9)20064 58..P-VOL PAIR DATA ,61114 29 -
VG01 pair2(L) (CL1-A , 0, 30)61114 30..S-VOL PAIR DATA ,----- 59 -
VG01 pair2(R) (CL1-C , 0, 10)20064 59..P-VOL PAIR DATA ,61114 30 -
VG01 pair3(L) (CL1-A , 0, 31)61114 31..S-VOL PAIR DATA ,----- 60 -
VG01 pair3(R) (CL1-C , 0, 11)20064 60..P-VOL PAIR DATA ,61114 31 -
```

例 5-6 ノード 3 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-C , 0, 9)20064 58..P-VOL PAIR DATA ,61114 29 -
VG01 pair1(R) (CL1-A , 0, 29)61114 29..S-VOL PAIR DATA ,----- 58 -
VG01 pair2(L) (CL1-C , 0, 10)20064 59..P-VOL PAIR DATA ,61114 30 -
VG01 pair2(R) (CL1-A , 0, 30)61114 30..S-VOL PAIR DATA ,----- 59 -
VG01 pair3(L) (CL1-C , 0, 11)20064 60..P-VOL PAIR DATA ,61114 31 -
VG01 pair3(R) (CL1-A , 0, 31)61114 31..S-VOL PAIR DATA ,----- 60 -
```

どのディスクが使用されているかを確認するには、次の例に示すように、pairdisplay コマンドの -fd オプションを使用します。

例 5-7 使用されているディスクを示す、ノード 1 上の pairdisplay コマンドの出力

```
# pairdisplay -fd -g VG01
Group PairVol(L/R) Device_File ,Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
VG01 pair1(L) c6t500060E80000000000000000E8BA0000001Dd0s2 61114 29..S-VOL PAIR DATA ,----- 58 -
VG01 pair1(R) c5t50060E8000000000000004E600000003Ad0s2 20064 58..P-VOL PAIR DATA ,61114 29 -
VG01 pair2(L) c6t500060E80000000000000000E8BA0000001Ed0s2 61114 30..S-VOL PAIR DATA ,----- 59 -
VG01 pair2(R) c5t50060E8000000000000004E600000003Bd0s2 0064 59..P-VOL PAIR DATA ,61114 30 -
VG01 pair3(L) c6t500060E80000000000000000E8BA0000001Fd0s2 61114 31..S-VOL PAIR DATA ,----- 60 -
VG01 pair3(R) c5t50060E8000000000000004E600000003Cd0s2 20064 60..P-VOL PAIR DATA ,61114 31 -
```


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/rdisk/d1
2 node-1:/dev/rdisk/c0t6d0 /dev/did/rdisk/d2
11 node-1:/dev/rdisk/c6t500060E8000000000000E8BA00000020d0 /dev/did/rdisk/d11
11 node-2:/dev/rdisk/c5t500060E8000000000000E8BA00000020d0 /dev/did/rdisk/d11
12 node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Fd0 /dev/did/rdisk/d12
12 node-2:/dev/rdisk/c5t500060E8000000000000E8BA0000001Fd0 /dev/did/rdisk/d12
13 node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Ed0 /dev/did/rdisk/d13
13 node-2:/dev/rdisk/c5t500060E8000000000000E8BA0000001Ed0 /dev/did/rdisk/d13
14 node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Dd0 /dev/did/rdisk/d14
14 node-2:/dev/rdisk/c5t500060E8000000000000E8BA0000001Dd0 /dev/did/rdisk/d14
18 node-3:/dev/rdisk/c0t0d0 /dev/did/rdisk/d18
19 node-3:/dev/rdisk/c0t6d0 /dev/did/rdisk/d19
20 node-3:/dev/rdisk/c5t50060E800000000000004E6000000013d0 /dev/did/rdisk/d20
21 node-3:/dev/rdisk/c5t50060E800000000000004E600000003Dd0 /dev/did/rdisk/d21
22 node-3:/dev/rdisk/c5t50060E800000000000004E600000003Cd0 /dev/did/rdisk/d2223
23 node-3:/dev/rdisk/c5t50060E800000000000004E600000003Bd0 /dev/did/rdisk/d23
24 node-3:/dev/rdisk/c5t50060E800000000000004E600000003Ad0 /dev/did/rdisk/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/c5t50060E800000000000004E600000003Ad0
Combining instance 14 with 24
VG01 pair2 L node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Ed0
VG01 pair2 R node-3:/dev/rdisk/c5t50060E800000000000004E600000003Bd0
Combining instance 13 with 23
VG01 pair3 L node-1:/dev/rdisk/c6t500060E8000000000000E8BA0000001Fd0
VG01 pair3 R node-3:/dev/rdisk/c5t50060E800000000000004E600000003Cd0
```

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/c6t500060E8000000000000E6000000020d0 /dev/did/rdsk/d11
11 node-2:/dev/rdisk/c5t500060E8000000000000E6000000020d0 /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/c5t50060E80000000000004E6000000013d0 /dev/did/rdsk/d20
21 node-3:/dev/rdisk/c5t50060E80000000000004E600000003Dd0 /dev/did/rdsk/d21
22 node-1:/dev/rdisk/c6t500060E8000000000000E60000001Fd0 /dev/did/rdsk/d1222
22 node-2:/dev/rdisk/c5t500060E8000000000000E60000001Fd0 /dev/did/rdsk/d12
22 node-3:/dev/rdisk/c5t50060E8000000000004E600000003Cd0 /dev/did/rdsk/d22
23 node-1:/dev/rdisk/c6t500060E8000000000000E60000001Ed0 /dev/did/rdsk/d13
23 node-2:/dev/rdisk/c5t500060E8000000000000E60000001Ed0 /dev/did/rdsk/d13
23 node-3:/dev/rdisk/c5t50060E8000000000004E600000003Bd0 /dev/did/rdsk/d23
24 node-1:/dev/rdisk/c6t500060E8000000000000E60000001Dd0 /dev/did/rdsk/d24
24 node-2:/dev/rdisk/c5t500060E8000000000000E60000001Dd0 /dev/did/rdsk/d24
24 node-3:/dev/rdisk/c5t50060E8000000000004E600000003Ad0 /dev/did/rdsk/d24
```

次に、ボリュームマネージャ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、133 ページの「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。

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、他の 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` や `newfs` など) 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大域ゾーンからのみ認識できません。クラスタファイルシステムのデータを非大域ゾーンからアクセス可能にする必要がある場合は、`zoneadm(1M)` または `HASStoragePlus` を使用して非大域ゾーン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。対処方法として、`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。
- `directioioctl` 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。

SPARC: 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 (VxVM クラスタ機能 & VERITAS クラスタサーバーが必要)

キャッシュアダプザリは使用可能、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あります。

- 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後で、このクラスタファイルシステムの状態を調べる必要があります。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 – 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 5-3 作業マップ: デバイスグループの管理

作業	参照先
cldevice populate 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	145 ページの「グローバルデバイス名前空間を更新する」
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	146 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)」
metaset 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	148 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	148 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	149 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 5-3 作業マップ: デバイスグループの管理 (続き)

作業	参照先
SPARC:VxVM のコマンドおよび <code>clsetup</code> を使用することにより、VERITAS Volume Manager のディスクグループをデバイスグループとして追加する	153 ページの「SPARC: 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)」
	154 ページの「SPARC: 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)」
	155 ページの「SPARC: 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)」
	157 ページの「SPARC: 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)」
	157 ページの「SPARC: 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)」
	158 ページの「SPARC: 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)」
	162 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)」
SPARC: <code>clsetup</code> コマンドを使用することにより (<code>cldevicegroup</code> を生成して) 構成から VERITAS Volume Manager デバイスグループを削除する	163 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)」
	161 ページの「SPARC: ディスクグループの構成変更を登録する (VERITAS Volume Manager)」
SPARC: <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	164 ページの「SPARC: デバイスグループからボリュームを削除する (VERITAS Volume Manager)」
	165 ページの「SPARC: デバイスグループを削除して登録を解除する (VERITAS Volume Manager)」
SPARC: <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	166 ページの「SPARC: デバイスグループにノードを追加する (VERITAS Volume Manager)」
	167 ページの「SPARC: 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	169 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	171 ページの「デバイスグループのプロパティを変更する」

表 5-3 作業マップ: デバイスグループの管理 (続き)

作業	参照先
cldevicegroup show 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	176 ページの「デバイスグループ構成の一覧を表示する」
clsetup を使用して cldevicegroup 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	173 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」
cldevicegroup switch 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	177 ページの「デバイスグループの主ノードを切り替える」
metaset または vxvg 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	178 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作業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。/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は、scgdevs コマンドを実行しても無効です。

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形式もあります。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形式については、[付録 A](#) 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 名前空間を再構成します。
cldevice populate

例 5-15 グローバルデバイス名前空間の更新

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。

```
# cldevice populate
Configuring the /dev/global directory (global devices)...
```

obtaining access to all attached disks
reservation program successfully exiting

▼ 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、このディスクセットを Sun Cluster ディスク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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- 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 Volume Manager 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 metaset(1M) 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、[173 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/rdisk/dN` という形式の完全な DID デバイス名を使用します。

次の例では、DID デバイス `/dev/did/rdisk/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/rdisk/d1
  Full Device Path:              phys-schost-1:/dev/rdisk/c0t0d0
DID Device Name:                /dev/did/rdisk/d2
  Full Device Path:              phys-schost-1:/dev/rdisk/c0t6d0
DID Device Name:                /dev/did/rdisk/d3
  Full Device Path:              phys-schost-1:/dev/rdisk/c1t1d0
  Full Device Path:              phys-schost-2:/dev/rdisk/c1t1d0
...
```

- 7 ディスクセットにドライブを追加します。
完全な DID バス名を使用します。

```
# metaset -s setname -a /dev/did/rdisk/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 (`cNtXdY`) 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、クラスタ全体で一貫ではないため、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 5-16 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/rdisk/d1 および /dev/did/rdisk/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/rdisk/d1 /dev/did/rdisk/d2
```

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 に登録している 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。

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形式もあります。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形式については、[付録 A](#) 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149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167ページの「SPARC: 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、169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。

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形式もあります。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形式については、付録A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*           削除されるノード (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 5-17 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 を実行中である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。

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形式もあります。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形式については、[付録 A](#) 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 ベースのシステムでは、次の操作を実行します。

```
ok boot
```

- x86 ベースのシステムでは、次の操作を実行します。

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 `cldevice 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
```

## ▼ SPARC: 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、154 ページの「SPARC: ディスクをカプセル化の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。

---

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 Parallel Server または 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。  
詳細は、158 ページの「SPARC: 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 Parallel Server または 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。

## ▼ SPARC: 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は、153 ページの「SPARC: 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 Parallel Server または 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。

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形式もあります。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形式については、[付録 A](#) 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に、

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、すべての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` 上の非大域ゾーンの名前。リソースグループを作成した際に非大域ゾーン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。

詳細は、158 ページの「SPARC: 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 Parallel Server または 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。

## ▼ SPARC: 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、161 ページの「SPARC: 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。

---

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形式もあります。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形式については、付録 A 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。
- 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 status
```
- 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。  
DPM  
161 ページの「SPARC: ディスクグループの構成変更を登録する (VERITAS Volume Manager)」。

## ▼ SPARC: 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。  
詳細は、158 ページの「SPARC: 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。

## ▼ SPARC: 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 reminor 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。  
詳細は、158 ページの「SPARC: 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 5-18 SPARC: 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 から 16002 と 4000 から 4001 が使用されていることを示しています。ここでは、`vx dg remenor` コマンドを使用して新しいデバイスグループにベースとなるマイナー番号 5000 を割り当てています。

```
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 remenor dg3 5000
```

## ▼ SPARC: 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` ユーティリティーを使用して、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、161 ページの「[SPARC: 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。

---

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形式もあります。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形式については、[付録 A](#) 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の詳細については、`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。173 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コントローラ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、157 ページの「SPARC: 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。

- 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
```

- 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。145 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 5-19 SPARC: 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、185 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、157 ページの「SPARC: 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。

## ▼ SPARC: 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。

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形式もあります。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形式については、付録 A 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 5-20 SPARC: 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 ユーティリティーを終了します。
- 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
```

▼ SPARC: 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、[161 ページの「SPARC: 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。

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形式もあります。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形式については、[付録 A](#) 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。
161 ページの「SPARC: 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。

▼ SPARC: 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。

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形式もあります。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形式については、[付録 A](#) 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 5-21 SPARC: 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
```

▼ SPARC: デバイスグループにノードを追加する (VERITAS Volume Manager)

次の手順では、`clsetup` ユーティリティ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

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形式もあります。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形式については、[付録 A](#) 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 **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 5-22 SPARC: 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
```

▼ SPARC: 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 (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。

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形式もあります。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形式については、[付録 A](#) 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、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 5-23 SPARC: 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。

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形式もあります。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形式については、[付録 A](#) 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 プロパティの詳細については、`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 5-24 SPARC: 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もとづきます。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 ボ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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

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。

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形式もあります。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形式については、[付録 A](#) 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 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、**preferenced/failback** 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 5-25 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 preferred=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。詳細は、『Sun Cluster の概念 (Solaris OS 版)』の第3章「重要な概念 - システム管理者とアプリケーション開発者」を参照してください。

`numsecondaries` プロパティを変更すると、二次ノードの実際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。

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形式もあります。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形式については、[付録 A](#) 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。[145 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 5-26 二次ノードの希望数の変更 (Solstice DiskSuite または Solaris ポリ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ポリリュームは以前に作成されているものと想定しています。

```
# 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 5-27 SPARC: 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 に設定するときに、`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、[173 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 5-28 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。

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形式もあります。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形式については、[付録 A](#) 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 <i>devicegroup</i></code>	<code>cldevicegroup show <i>devicegroup</i></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 <i>devicegroup</i></code>	<code>cldevicegroup status <i>devicegroup</i></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 5-29 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 5-30 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。

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形式もあります。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形式については、[付録 A](#) 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 5-31 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。

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形式もあります。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形式については、[付録 A](#) 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:

```
# vxdbg 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 5-32 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:
# vxdbg -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:
# vxdbg 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。

- 181 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」
- 182 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」
- 182 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 プロトコル設定を変更する」
- 183 ページの「単一ストレージデバイスの SCSI 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

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形式もあります。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形式については、[付録 A](#) 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 5-33 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:              scsi3
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

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形式もあります。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形式については、[付録 A](#) 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(1CL)` のマニュアルページを参照してください。

例 5-34 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/rdsk/c4t8d0` の SCSI プロトコルを示します。

```
# cldevice show /dev/rdsk/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 SCSI プロトコル設定を変更する

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` または `scsi3` 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、この手順を使用してグローバルな SCSI プロトコル設定を `scsi3` 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、[183 ページ](#) の「[単一ストレージデバイスの SCSI プロトコルを変更する](#)」の手順を使用します。

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、設定を変更して、定足数デバイスを再構成します。

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形式もあります。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形式については、[付録 A](#) 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.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 SCSI プロトコルを設定します。

```
cluster set -p global_fencing={scsi3 | pathcount}
```

`-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。

`scsi3` 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。

例 5-35 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 SCSI プロトコルを、SCSI-3 プロトコルに設定します。

```
# cluster set -p global_fencing=scsi3
```

▼ 単一ストレージデバイスの SCSI プロトコルを変更する

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形式もあります。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形式については、[付録 A](#) 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.modify` を提供する役割になります。

2 ストレージデバイスの SCSI プロトコルを設定します。

```
# cldevice set -p default_fencing ={pathcount | scsi3 | global} device
```

- p デバイスの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。
- 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。
- 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 5-36 単一デバイスの SCSI 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 11 を SCSI-3 プロトコルに設定します。

```
# cldevice set -p default_fencing=scsi3 5
```

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 5-4 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
最初の Sun Cluster のインストールのあと、newfs(1M) および mkdir を使用してクラスタファイルシステムを追加する	185 ページの「クラスタファイルシステムを追加する」
fuser(1M) および umount(1M) を使用してクラスタファイルシステムを削除する	189 ページの「クラスタファイルシステムを削除する」
ノードにまたがる一貫性を確保するには、sccheck(1M) を使用してクラスタ内のグローバルマウントポイントを確認する	191 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 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)。

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形式もあります。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形式については、[付録 A](#) 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。

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 `newfs` コマンドを使用してファイルシステムを作成します。

注-`newfs` コマンドは、新しい UFS ファイルシステムを作成するときだけ有効です。新しい VxFS ファイルシステムを作成する場合は、VxFS マニュアルの手順に従ってください。

```
# newfs raw-disk-device
```

下の表に、引数 `raw-disk-device` の名前の例を挙げます。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	説明
Solaris ボリュームマネージャー	/dev/md/oracle/rdisk/d1	oracle メタセット内部の raw ディスクデバイス d1
SPARC:VERITAS Volume Manager	/dev/vx/rdisk/oradg/vol01	oradg ディスクグループ内部の raw ディスクデバイス vol01
なし	/dev/global/rdisk/d1s3	ブロックスライス d1s3 の raw ディスクデバイス

- 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ディレクトリを作成します。
- クラスタファイルシステムにアクセスしないノードがある場合でも、マウントポイントは各ノードごとに必要です。

ヒント-管理しやすくするために、マウントポイントは `/global/devicegroup` ディレクトリに作成します。これを使用することによって、グローバルに利用できるクラスタファイルシステムを、ローカルファイルシステムから簡単に判別できるようになります。

```
# mkdir -p /global/devicegroup mountpoint
```

devicegroup 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。

- a. 以下の必須マウントオプションを使用します。

注-ロギングはすべてのクラスタファイルシステムに必要です。

- **Solaris UFS** ロギング - `global, logging` マウントオプションを使用します。UFS マウントポイントの詳細については、`mount_ufs(1M)` のマニュアルページを参照してください。

注 - `syncdir` マウントオプションは UFS クラスタファイルシステムには必要ありません。`syncdir` を指定すると、POSIX に準拠したファイルシステムの動作が保証されます。指定しない場合は、UFS ファイルシステムと同じ動作になります。`syncdir` を指定しない場合、ディスクブロックを割り当てる (つまり、データをファイルに追加するような) 書き込みの性能が大幅に向上します。ただし、場合によっては `syncdir` を指定しないと、ファイルを閉じるまで容量不足の状態を検出できません。`syncdir` を指定しないことで生じる問題はほとんどありません。`syncdir` (つまり、POSIX の動作) を指定した場合、空間不足状態はファイルを閉じる前に見つかります。

- **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 - `global` マウントオプションを使用します (`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を設定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。

注 - 将来の Solaris ソフトウェアのリリースでは、トランザクションボリュームは Solaris OS から削除される予定です。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。

- **VxFS** ロギング - `global` および `log` マウントオプションを使用します。詳細は、VxFS ソフトウェアに付属の `mount_vxfs` のマニュアルページを参照してください。
- 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` が起動して `/global/oracle` をマウントしたあとにのみ `phys-schost-2` が起動して `/global/oracle/logs` をマウントできます。

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。

- 5 クラスタ内にある任意のノード上で、マウントポイントが存在し、クラスタ内にあるすべてのノード上で `/etc/vfstab` ファイルのエントリが正しいことを確認します。

```
# sccheck
```

エラーが発生していない場合は、何も戻されません。

- 6 クラスタ内にある任意のノードから、クラスタファイルシステムをマウントします。

```
# mount /global/devicegroup mountpoint
```

- 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` または `mount` のいずれかのコマンドを使用し、マウントされたファイルシステムの一覧を表示します。

Sun Cluster 環境で VxFS クラスタファイルシステムを管理するには、管理コマンドは 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から実行する必要があります。

例 5-37 クラスタファイルシステムの追加

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/rdsk/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。

```
# newfs /dev/md/oracle/rdsk/d1
```

```
...
```

```
[on each node:]
```

```
# mkdir -p /global/oracle/d1
```

```
# 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/rdsk/d1 /global/oracle/d1 ufs 2 yes global,logging
```

```
[save and exit]
```

```
[on one node:]
```

```
# sccheck
```

```
# mount /dev/md/oracle/dsk/d1 /global/oracle/d1
```

```
# mount
```

```
...
```

```
/global/oracle/d1 on /dev/md/oracle/dsk/d1 read/write/setuid/global/logging/largefiles
on Sun Oct 3 08:56:16 2001
```

▼ 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す。`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と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表示される。
-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せん。
- 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 5-38 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/rdsk/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

sccheck(1M) ユーティリティを使用して、/etc/vfstab ファイル内のクラスタファイルシステムのエントリの構文を確認します。エラーが発生していない場合は、何も戻されません。

注 - sccheck 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(クラスタファイルシステムの削除など)をクラスタ構成に加えたあとで実行します。

- 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。

```
# sccheck
```

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 の概念 (Solaris OS 版)』の第3章「重要な概念 - システム管理者とアプリケーション開発者」を参照してください。scdpm コマンドのオプションと関連するコマンドについては、cldevice(1CL)のマニュアルページを参照してください。デーモンが報告するロギングされたエラーについては、syslogd(1M)のマニュアルページを参照してください。

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パ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、ディスクパスは自動的に監視から除外されます。

表 5-5 作業マップ: ディスクパス監視の管理

作業	参照先
cldevice monitor コマンドを使用してディスクパスを監視する	192 ページの「ディスクパスを監視する」
cldevice unmonitor コマンドを使用してディスクパスの監視を解除する	194 ページの「ディスクパスの監視を解除する方法」
cldevice status コマンドを使用して、あるノードの、障害が発生したディスクパスのステータスを出力する	195 ページの「障害のあるディスクパスを表示する」
cldevice コマンドを使用してファイルからディスクパスを監視する	195 ページの「ファイルからディスクパスを監視する」
clnode set コマンドを使用して、すべての監視対象ディスクパスに障害が発生した場合、ノードの自動再起動を有効または無効にする	198 ページの「すべての監視対象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有効にする」 198 ページの「すべての監視対象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パス引数が含まれます。ディスクパ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。

▼ ディスクパスを監視する

この作業は、クラスタのディスクパ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。

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形式もあります。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形式については、[付録 A](#) 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 5-39 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 5-40 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 5-41 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パ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

▼ ディスクパスの監視を解除する方法

ディスクパ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。

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形式もあります。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形式については、[付録 A](#) 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パ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 5-42 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 5-43 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

▼ 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。

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形式もあります。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形式については、[付録 A](#) 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 5-44 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` の間のパ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>

```

そのパスを監視するには、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については、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。

▼ すべての監視対象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。

- 該当ノード上で監視対象のすべての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。

- 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、RBACの承認 `solaris.cluster.modify` 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対象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有効にします。

```
# clnode set -p reboot_on_path_failure=enabled +
```

▼ すべての監視対象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対象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、RBACの承認 `solaris.cluster.modify` 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対象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 内の定足数 (quorum) 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 の概念 (Solaris OS 版)』の「定足数と定足数デバイス」を参照してください。

- 199 ページの「定足数の管理の概要」
- 202 ページの「定足数デバイスの追加」
- 211 ページの「定足数デバイスの削除または交換」
- 214 ページの「定足数デバイスの保守」

この章で示す例は、主に 3 ノードクラスタです。

定足数の管理の概要

`clquorum(1CL)` コマンドを使用すると、定足数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。この章の管理手順は、可能なかぎり `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大域ゾーン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、ディスクの構成を解除したり、ディスクをストレージプールに追加し、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。

表 6-1 作業リスト: 定足数の管理

作業	参照先
<code>clsetup(1CL)</code> 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	202 ページの「定足数デバイスの追加」
<code>clsetup</code> を使用する (<code>clquorum</code> 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)	211 ページの「定足数デバイスを削除する」
<code>clsetup</code> を使用する (<code>clquorum</code> 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)	212 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	214 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	215 ページの「定足数デバイスのノードリストを変更する」
<code>clsetup</code> を使用する (<code>clquorum</code> 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)	217 ページの「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。)	

表 6-1 作業リスト:定足数の管理 (続き)

作業	参照先
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)	219 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	220 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 Detach 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	参照先
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	202 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	211 ページの「定足数デバイスを削除する」
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 の概念 (Solaris OS 版)』の「定足数と定足数デバイス」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。または、ディスクの構成を解除したり、ディスクをストレージプールに追加し、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せん。

Sun Cluster は、次の種類の定足数デバイス、SCSI、Network Appliance (NetApp) NAS、および Sun Cluster 定足数サーバーをサポートしています。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。

- 202 ページの「SCSI 定足数デバイスを追加する」
- 204 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」
- 207 ページの「定足数サーバーを定足数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、次のエラーメッセージが表示され、その後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.
```

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の詳細については、`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。

▼ SCSI 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 (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。

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形式もあります。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形式については、[付録 A](#) 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 **SCSI** 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、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 SCSI 定足数デバイスの追加

次の例は、SCSI 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:]
  [SCSI device           scsi]
  [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
```

```
Quorums      Type
-----
d20           scsi
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

次に、Network Appliance (NetApp) ネットワーク接続ストレージ (NAS) デバイスを定足数デバイスとして使用する場合の要件を示します。

- 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 ユニットを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。サービスが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す。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。

Network Appliance NAS デバイスと LUN の構築と設定については、以下の Network Appliance NAS マニュアルを参照してください。これらのマニュアルは、<http://now.netapp.com> で利用できます。

作業	Network Appliance のマニュアル
NAS デバイスの設定	『System Administration File Access Management Guide』
LUN の設定	『Host Cluster Tool for Unix Installation Guide』
ONTAP ソフトウェアのインストール	『Software Setup Guide』、 『Upgrade Guide』
クラスタのボリュームのエクスポート	『Data ONTAP Storage Management Guide』
クラスタノードへの NAS サポートソフトウェアパッケージのインストール	http://now.netapp.com にログインし、「Software Download」ページから『Host Cluster Tool for Unix Installation Guide』をダウンロードしてください。

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』参照してください。

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形式もあります。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形式については、[付録 A](#) 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メッセージが表示されます。
- 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メッセージが表示されます。
- 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 を指定するようにメッセージが表示されます。
- 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、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 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
```

```
Quorums      Type
-----
qd1          netapp_nas
scphyshost-1 node
scphyshost-2 node
```

▼ 定足数サーバーを定足数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と起動についての詳細は、『Sun Cluster 定足数サーバーユーザズガイド』を参照してください。

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形式もあります。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形式については、[付録 A](#) 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に示す条件のいずれかを確実に満たすようにしてください。
 - スイッチは Rapid Spanning Tree Protocol (RSTP) をサポートしています。
 - 高速ポートモードは、スイッチ上で有効です。

クラスタノードと定足数サーバー間で確実に即時通信できるようにするためにこれらの機能のいずれかが必要になります。スイッチによってこの通信に大幅な遅延が生じる場合、クラスタはこの通信妨害を定足数デバイスの損失と解釈します。

- b. パブリックネットワークで **CIDR (Classless Inter-Domain Routing)** と呼ばれる可変長のサブネットが使用されている場合、各ノードで次に示すファイルを変更します。

RFC 791 に定義されているクラスフルサブネットを使用する場合、次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、クラスタで使用する各パブリックサブネット用のエントリを追加します。

パブリックネットワーク 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. `netmask + broadcast +` を、各 `/etc/hostname.adapter` ファイルのホスト名エントリのあとに追加します。

```
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. ネーミングサービスを使用する場合、定足数サーバーのホスト名とアドレスのマッピング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メッセージが表示されます。
- 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メッセージが表示されます。
- 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、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 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
```

```
Quorums      Type
-----      ----
qd1          quorum_server
scphyshost-1 node
scphyshost-2 node
```

-- Quorum Summary --

```
Quorum votes possible: 5
Quorum votes needed:   3
Quorum votes present:  5
```

-- Quorum Votes by Node --

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

-- Quorum Votes by Device --

	Device Name	Present	Possible	Status
Device votes:	qd1	1	1	Online
Device votes:	/dev/did/rdisk/d3s2	1	1	Online
Device votes:	/dev/did/rdisk/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。

- 211 ページの「定足数デバイスを削除する」
- 212 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」
- 214 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)` は失敗してデバイスは構成から削除されません。

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、[212 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。

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形式もあります。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形式については、[付録 A](#) 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 6-4 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
```

```
Quorums      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上の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手順については、『Sun Cluster 定足数サーバーユーザズガイド』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。2つのノードがあるクラスタで、そのうち1つのノードを削除する場合にのみ、この手順が必要となります。削除しようとしているデバイスが、2つのノードがあるクラスタで最後の定足数デバイスではない場合、前の手順 211 ページの「定足数デバイスを削除する」を使用してください。

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は、`clquorum(1CL)`を使用して構成からデバイスを削除できるように、このクラスタをインストールモードにする必要があります。この手順は、クラスタからノードを削除する場合にだけ行います。

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形式もあります。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形式については、[付録 A](#) 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 削除するノードを保守状態に変更
255 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 クラスタをインストールモードにします。
`# cluster set -p installmode=enabled`
- 4 `clquorum` コマンドを使用して定足数デバイスを削除します。
クラスタがインストールモードである場合、`clsetup(1CL)` クラスタ管理メニューオプションは利用できません。
`# clquorum remove qd1`
- 5 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 最後の定足数デバイスの削除

次に、クラスタ構成の最後の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.]
[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
Quorums      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。

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形式もあります。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形式については、[付録 A](#) 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、[202 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、[211 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。

- [215 ページ](#)の「[定足数デバイスのノードリストを変更する](#)」
- [217 ページ](#)の「[定足数デバイスを保守状態にする](#)」
- [219 ページ](#)の「[定足数デバイスを保守状態から戻す](#)」
- [220 ページ](#)の「[クラスタ構成を一覧表示する](#)」
- [221 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。

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形式もあります。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形式については、[付録 A](#) 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。
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の物理的な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。
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 6-6 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
Quorums          Type
-----          -
d2                scsi
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.
```

```
[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

```
# clquorum list -v
```

```
Quorums      Type
-----      ----
sc-phys-schost-1  node
sc-phys-schost-2  node
sc-phys-schost-3  node
```

▼ 定足数デバイスを保守状態にする

定足数デバイスを保守状態にするには、`clquorum(1CL)` コマンドを使用します。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、255 ページの「ノードを保守状態にする」を参照してください。

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形式もあります。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形式については、付録 A 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 6-7 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、219 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、255 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クォーラム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、257 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は、『Sun Cluster の概念 (Solaris OS 版)』の「定足数投票数について」を参照してください。

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形式もあります。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形式については、付録 A 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 6-8 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:                     scsi
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。

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形式もあります。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形式については、[付録 A](#) 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 6-9 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:                      scsi
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。

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形式もあります。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形式については、[付録 A](#) 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、202 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、211 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、202 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、211 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。

クラスタインターコネク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ネットワーク構成を変更する必要がある場合は、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、239 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。

- 223 ページの「クラスタインターコネクの管理」
- 239 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概要は、表 7-1 と表 7-3 を参照してください。

クラスタインターコネク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 の概念 (Solaris OS 版)』を参照してください。

クラスタインターコネクの管理

この節では、クラスタトランスポートアダプタや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` コーティリティーを使用すると、クラスタインターコネク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、すべてのインターコネクコマンドは大域ゾーン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
clsetup(1CL) を使用することで、クラスタトランスポートを管理する	25 ページの「clsetup ユーティリティにアクセスする」
clinterconnect status 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	225 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」
clsetup 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	226 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」
clsetup 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	229 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」
clsetup 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	232 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」
clsetup 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	233 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	235 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	236 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。

- 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 Detach 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、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。
- DR のボード削除操作によって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に影響がある場合には、Sun Cluster は操作を拒否し、操作によって影響を受けるインタフェースを特定します。



注意 - Sun Cluster 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	241 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	『Sun Enterprise 10000 DR 構成マニュアル』 および「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 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。

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形式もあります。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形式については、付録 A 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があります。

- 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。

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形式もあります。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形式については、付録 A 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。
- 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 SState (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、[225 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。

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形式もあります。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形式については、[付録 A](#) 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
Adapter 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 SState (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。

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形式もあります。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形式については、[付録 A](#) 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** ユーティリティーを起動します。

```
# 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。

- 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。

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形式もあります。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形式については、[付録 A](#) 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
                        |  |
      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/pci11c8,0@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
        pci11c8,0, instance #0
        pci11c8,0, 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。

- 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。

```
# 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。

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。

 - 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使用しない場合、`clsetup` ユーティリティーは、ユーザーがクラスタ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を入力するプロンプトを出します。
 - b. クラスタ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。
 - 第一のネットマスクは、ユーザーが指定したノードとプライベートネットワークの数をサポートする、最低限のネットマスクです。
 - 第二のネットマスクは、将来ありうる成長に対応するため、ユーザーが指定したノードとプライベートネットワークの数の2倍をサポートします。
 - 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 ベースのシステムでは、次の操作を実行します。

ok boot

- x86 ベースのシステムでは、次の操作を実行します。
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、および Sun Cluster 3.2 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 (アンプラム) 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、次のコマンドを使用します。`clinterconnect status` コマンド

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。

- 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 Detach 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 アドレスを切り換えます。
- 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	<p><code>if_mpadm(1M)</code> のマニュアルページ</p> <p>適切な SolarisOS のマニュアル:</p> <p>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)」</p> <p>Solaris 10: 『Solaris のシステム管理 (IP サービス)』のパート VI 「IPMP」</p>

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
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)」 <code>ifconfig(1M)</code> 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」
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『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』

クラスタの管理

この章では、クラスタ全体に影響を与える項目の管理手順について説明します。

この章の節の一覧は次のとおりです。

- 243 ページの「クラスタの管理の概要」
- 260 ページの「クラスタノードの追加」
- 262 ページの「ノード上での非大域ゾーンの管理」
- 264 ページの「クラスタノードの削除」
- 275 ページの「Sun Cluster SNMP イベント MIB の作成、設定、および管理」

クラスタの管理の概要

この節では、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、これらすべての管理作業と、関連する手順を示します。Solaris 10 OS では、明記しないかぎり、クラスタ管理作業は大域ゾーンで行います。

表 8-1 作業リスト:クラスタの管理

作業	参照先
クラスタ名を変更	244 ページの「クラスタ名を変更する」
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	245 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	245 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	247 ページの「クラスタの時刻をリセットする」

作業	参照先
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	249 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変更	250 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	255 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	257 ページの「ノードを保守状態から戻す」
ノードをクラスタに追加	260 ページの「クラスタノードの追加」
ノードをクラスタから削除	264 ページの「クラスタノードの削除」
ノードからの Sun Cluster ソフトウェアのアンインストール	271 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」
エラーメッセージの修正	273 ページの「エラーメッセージを修正する」

▼ 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。

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形式もあります。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形式については、[付録 A](#) 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。

例 8-1 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、`clsetup(1CL)` ユーティリティから生成される `cluster(1CL)` コマンドを示します。

```
# cluster -c dromedary
```

▼ ノード ID をノード名にマップする

Sun Cluster のインストール時に、各ノードには、自動的に一意のノード ID 番号が割り当てられます。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。

構成情報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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- `clnode(1CL)` コマンドを使用して、クラスタ構成情報を一覧表示します。

```
% clnode show | grep Node
```

例 8-2 ノード名のノード ID へのマップ

次に、ノード ID の割り当て例を示します。

```
% 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 により、新しいノードをクラスタに追加できるか、およびどのタイプの認証で追加するかを判別できます。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、ノードが加わるには、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)` および `publickey(4)` のマニュアルページを参照してください。

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形式もあります。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形式については、[付録 A](#) 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(1CL)` 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 8-3 新しいマシンがクラスタに追加されないようにする

次に、新しいマシンがクラスタに追加されないようにする、`clsetup` ユーティリティから生成された `claccess` コマンドの例を示します。

```
# claccess deny -h hostname
```

例 8-4 すべての新しいマシンがクラスタに追加されるように許可する

次に、すべての新しいマシンがクラスタに追加されるようにする、`clsetup` ユーティリティから生成された `claccess` コマンドの例を示します。

```
# claccess allow-all
```

例 8-5 クラスタに追加される新しいマシンを指定する

次に、単一の新しいマシンがクラスタに追加されるようにする、`clsetup` ユーティリティから生成された `claccess` コマンドの例を示します。

```
# claccess allow -h hostname
```

例 8-6 認証を標準 UNIX に設定する

次に、クラスタに加わる新しいノードの認証を標準 UNIX 認証にリセットする、`clsetup` ユーティリティから生成された `claccess` コマンドの例を示します。

```
# claccess set -p protocol=sys
```

例 8-7 認証を DES に設定する

次に、クラスタに加わる新しいノードで DES 認証を使用する、`clsetup` ユーティリティから生成された `claccess` コマンドの例を示します。

```
# 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。詳細については、`keyserv(1M)` および `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 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。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 の概念 (Solaris OS 版)』と『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。

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形式もあります。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形式については、[付録 A](#) 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 クラスタを停止します。

```
# 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、以下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、以下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適切な Solaris エントリを選択し、e を入力してその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)』の第 11 章「GRUB ベースのブート(手順)」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 -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**を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。
`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。
`rdate hostname`
- 7 各ノードを起動し、クラスタを再起動します。
`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。
`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。

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形式もあります。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形式については、[付録 A](#) 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、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。

```
# 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

インストール完了後、クラスタノードのプライベートホスト名を変更するには、次のようにします。

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)。`clusternode3-priv` 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。

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形式もあります。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形式については、[付録 A](#) 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。

```
# 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ようにカスタマイズしているアプリケーション
- 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 コマンドの使用法については、clresource(1CL) のマニュアルページと、『Sun Cluster データサービスの計画と管理 (Solaris OS 版)』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 デーモンの詳細については、xntpd (1M) のマニュアルページを参照してください。

```
# /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 デーモンの詳細については、svcadm(1M) のマニュアルページを参照してください。

```
# svcadm disable ntp
```

- 3 clsetup(1CL) ユーティリティを実行して、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 つのノードからのみユーティリティを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。ここでは、プライベートホスト名を変更するノードの名前 (`clusternode<nodeid>-priv`) と新しいプライベートホスト名を問い合わせられます。

- 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することを防ぎます。

```
# 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** を実行して応答を確認します。

- 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他のアプリケーションをすべて有効にします。

```
# clresource disable resource[,...]
```

scswitch コマンドの使用方法については、clresource(1CL) のマニュアルページと、『Sun Cluster データサービスの計画と管理 (Solaris OS 版)』を参照してください。

例 8-8 プライベートホスト名の変更

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.]
```

▼ 非大域ゾーン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、クラスタ非大域ゾーン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。

- 1 clsetup(1CL) ユーティリティを実行して、適切なゾーンでプライベートホスト名を追加します。
- 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。非大域ゾーンのプライベートホスト名にはデフォルトはありません。ホスト名を入力する必要があります。

▼ 非大域ゾーン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、クラスタの非大域ゾーン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`。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。

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形式もあります。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形式については、[付録 A](#) 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。

```
# 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ようにカスタマイズしているアプリケーション
- 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` コマンドの使用法については、`clresource(1CL)` のマニュアルページと、『Sun Cluster データサービスの計画と管理 (Solaris OS 版)』を参照してください。

- 2 `clsetup(1CL)` ユーティリティを実行して、適切な非大域ゾーンのプライベートホスト名を変更します。

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。
非大域ゾーンのプライベートホスト名にはデフォルトはありません。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大域ゾーンの名前 (clusternode<nodeid>-priv) と新しいプライベートホスト名を問い合わせられます。
- 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することを防ぎます。

```
# 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他のアプリケーションをすべて有効にします。

▼ 非大域ゾーンのプライベートホスト名を削除する

クラスタ非大域ゾーン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。

- 1 `clsetup(1CL)` ユーティリティを実行して、適切なゾーンでプライベートホスト名を削除します。
- 2 ゾーン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 削除する非大域ゾーン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、`clnode(1CL) disable` コマンドを使用します。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- 1 保守状態にするノード上で、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 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、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。

```
# clnode evacuate node
```

- 3 退避させたノードを停止します。

```
# 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。

```
# clquorum disable type [node]
```

type 定足数デバイスタイプを指定します。タイプには `node`、`scsi`、`netapp_nas`、または `quorum_server` を指定できます。

node 保守モードにするノードの名前を指定します。

- 5 クラスタノードが保守状態にあることを確認します。

```
# clquorum status -t node
```

保守状態にしたノードの状態はオフラインであり、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 8-9 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` のノードの投票数は 0 (ゼロ) で、その状態はオフラインです。定足数の概要では、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 `offline` 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 -t node phys-schost-1
phys-schost-2# clquorum status -t node
```

-- 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、257 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、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。

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形式もあります。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形式については、付録 A を参照してください。

- 1 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 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してハングアップすることがあります。

```
# 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。

```
# 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 8-10 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が。

```
phys-schost-2# clquorum reset
```

- SPARC ベースのシステムでは、次の操作を実行します。

ok boot

- x86 ベースのシステムでは、次の操作を実行します。
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

クラスタノードの追加

この節では、クラスタノードを追加する手順を説明します。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。これらの作業を正常に行うには、示された順番に従う必要があります。

表 8-2 作業マップ: 既存のクラスタへのクラスタノードの追加

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』
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	260 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「クラスタへのソフトウェアのインストール」

▼ ノードを認証ノードリストに追加する

既存のクラスタにマシンを追加する前に、プライベートクラスタインターコネクとの物理的な接続が正確に行われているかなど、必要なハードウェアがすべて正確にノードに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の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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- 1 現在のクラスタメンバー上でスーパーユーザーになります。

- 2 260 ページの「クラスタノードの追加」の作業マップに記載されている必要不可欠なハードウェアのインストール作業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。

```
# clsetup
```

メインメニューが表示されます。
- 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るのを防ぐために、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、**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、新しいマシンが自分自身をクラスタに追加しようとする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 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。
『Sun Cluster ソフトウェアのインストール (Solaris OS 版)』で説明されているように、`scinstall` または `JumpStart™` 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。

例 8-11 クラスタノードの認証ノードリストへの追加

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.]  
# clsetup  
[Select New nodes>Specify the name of a machine which may add itself.]  
[Answer the questions when prompted.]
```

[Verify that the `scconf` 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 クラスタノードを追加する作業の概要については、表 8-2 の「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 データサービスの計画と管理 (Solaris OS 版)』を参照してください。

ノード上での非大域ゾーンの管理

この節では、クラスタノード上で(単に「ゾーン」と呼ばれる)非大域ゾーンを作成するための次の情報と手順を説明します。

▼ ノード上で非大域ゾーンを作成する

- 1 作成する非大域ゾーンのあるノードでスーパーユーザーになります。ユーザーは大域ゾーン内に存在する必要があります。
- 2 **Solaris 10 OS** の場合、各ノードで **Service Management Facility (SMF)** 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てください。
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なるまで待ちます。

```
phys-schost# svcs multi-user-server
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- 非大域ゾーン内でリソースグループの機能をサポートするには、`autoboot` プロパティを `true` に設定します。

次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 (省略可能) ゾーンに、プライベート IP アドレスとプライベートホスト名を割り当てます。

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>	非大域ゾーンの名前。

- 6 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。

- ゾーンにログインする
- `/etc/sysidcfg` ファイルを使用する

- 7 非大域ゾーンで、`nsswitch.conf` ファイルを変更します。

これらの変更は、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するため行います。

- 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。

```
phys-schost# 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` スイッチを追加します。

変更されたエントリは次のようになります。

```
...
hosts:      cluster files nis [NOTFOUND=return]
...
netmasks:  cluster files nis [NOTFOUND=return]
...
```

▼ ノード上で非大域ゾーンを削除する

- 1 削除する非大域ゾーンのあるノードでスーパーユーザーになります。
- 2 システムから非大域ゾーンを削除します。

『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。

クラスタノードの削除

この節では、クラスタノードを削除する方法を説明します。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。これらの作業を正常に行うには、示された順番に従う必要があります。



注意-OPS 構成を実行しているクラスタでは、この手順を実行してはいけません。現時点では、OPS 構成のノードを削除すると、他のノードがリブート時にパニックを起こす可能性があります。

表 8-3 作業リスト: クラスタノードの削除

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>
- <code>clnode(1CL) evacuate</code> を使用	
すべてのリソースグループからノードを削除	『Sun Cluster データサービスの計画と管理 (Solaris OS 版)』
- <code>clresourcegroup(1CL) remove-node</code> を使用	

表 8-3 作業リスト:クラスタノードの削除 (続き)

作業	参照先
すべてのデバイスグループからノードを削除 - cldevicegroup (1CL) remove node、metaset (1M)、および clsetup (1CL) を使用	149 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)」 167 ページの「SPARC: デバイスグループからノードを削除する (VERITAS Volume Manager)」 169 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」 注意:望ましい二次ノードの数を 2 つ以上に構成する場合は、1 に減らす必要があります。
完全に接続された定足数デバイスをすべて削除する - clsetup を使用	注意:2 ノードクラスタからノードを削除している場合、定足数デバイスを削除してはなりません。 211 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。
ノードから完全に接続されたストレージデバイスをすべて削除する - devfsadm (1M)、\cldevice (1CL) refresh を使用	注意:2 ノードクラスタからノードを削除している場合、定足数デバイスを削除してはなりません。269 ページの「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」
定足数デバイスを再び追加する (クラスタに残しておきたいノードのみ) - clsetup を使用	202 ページの「定足数デバイスの追加」
削除するノードを保守状態にする - clnode (1CL) evacuate、cluster (1CL) shutdown、および clquorum (1CL) disable を使用	255 ページの「ノードを保守状態にする」
削除するノードからすべての論理トランスポート接続 (トランスポートケーブルとアダプタ) を削除 - clsetup を使用	229 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」
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する - cluster set、clquorum remove を使用	212 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」
クラスタソフトウェア構成からノードを削除 - clnode remove を使用	266 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」

表 8-3 作業リスト:クラスタノードの削除 (続き)

作業	参照先
(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する	271 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」

▼ クラスタソフトウェア構成からノードを削除する

クラスタからノードを削除するは、次の手順を実行します。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- 1 [264 ページの「クラスタノードの削除」](#)の「作業マップ:クラスタノードの削除」に記載されている必要不可欠な作業をすべて正しく完了していることを確認します。

注-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、および定足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。
- 3 削除するノードを非クラスタモードで起動します。
 - SPARC ベースのシステム上で、以下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、以下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適切な Solaris エントリを選択し、e を入力してその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)』の第 11 章「GRUB ベースのブート (手順)」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 -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 を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。

- 4 クラスタからノードを削除します。

```
# clnode remove nodename
```

- 5 別のクラスタノードから、`scstat(1M)` を使用して、ノードが削除されていることを確認します。

```
# clnode status nodename
```

- 6 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、[271 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 8-12 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で実行します (`phys-schost-1`)。

```
[Remove the node from the cluster:]
phys-schost-1# clnode remove phys-schost-2
[Verify node removal:]
phys-schost-1# clnode status phys-schost-2
-- 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、[271 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 8-3](#)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、260 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。

▼ 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。

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形式もあります。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形式については、付録 A 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。


```
# clresourcegroup status
# 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。



注意 ((SPARC のみ)) - Oracle Parallel Server/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 Parallel Server/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。

```
# 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。

- 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、第 8 章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。

```
# 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、`node` 上の非大域ゾーンの名前。リソースグループを作成した際に非大域ゾーン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についての詳細は、『Sun Cluster データサービスの計画と管理 (Solaris OS 版)』を参照してください。

注 - `clresourcegroup` を実行するときには、リソスタイプ、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、この手順を飛ばします)。
- 8 切断するノードからホストアダプタを削除する場合、ノードを停止し電源を切ります。切断するノードからホストアダプタを削除する場合、**Step 11** に移動します。
- 9 ノードからホストアダプタを削除します。
ホストアダプタを削除する手順については、ノードに付属しているマニュアルを参照してください。
- 10 起動が行われないようにして、ノードに電源を入れます。
- 11 **SPARC:Oracle Parallel Server/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。

```
# pkgrm SUNWscucm
```



注意 ((SPARC のみ)) - 切断したノードから Oracle Parallel Server/Oracle RAC ソフトウェアを削除しなければ、そのノードをクラスタに導入し直したときに、ノードにパニックが発生して、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 ベースのシステムでは、次の操作を実行します。

```
ok boot
```

- x86 ベースのシステムでは、次の操作を実行します。

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。

```
# 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手順については、「デバイスグループを保守状態にする」の手順を参照してください。

▼ 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クラスタノードを切り離す前に、クラスタノードから 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。

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形式もあります。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形式については、[付録 A](#) を参照してください。

- 1 クラスタノードの削除に必要なすべての前提条件 (作業マップを参照) が完了しているか確認します。

[表 8-3](#) 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 アンインストールを行なうノードではなく、それ以外のアクティブなクラスタメンバー上で、スーパーユーザーになります。
- 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。

```
# 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)` ユーティリティーを使用できます。手順については、[260 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 ノードを非クラスタモードで再起動します。

- SPARC: 次のように入力します。

```
# shutdown -g0 -y -i0ok boot -x
```

- x86: 次のように入力します。

```
# 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
```

- 6 /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す。
- 7 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 コマンドの用法についての詳細は、『**Sun Java Enterprise System 5 Installation Guide for UNIX**』の第 8 章「Uninstalling」を参照してください。
- 8 このクラスタ上で **Sun Cluster** 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した後で、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。

▼ エラーメッセージを修正する

前の節の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 ノードのクラスタへの再結合を試みます。

```
# 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3](#)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。
 148 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、[271 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 [271 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。

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。

削除されていないクラスタファイルシステムエントリがある場合

次のエラーメッセージは、削除したノードに、`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。

予期せぬグローバルマウントが `/etc/vfstab` に残っていないことを確認しています ... 失敗しました

scinstall: `global-mount1` はまだグローバルマウントとして構成されています

scinstall: `global-mount1` はまだグローバルマウントとして構成されています

scinstall: `/global/dg1` はまだグローバルマウントとして構成されています

scinstall: このようなエラーが出たままアンインストールするのは安全ではありません。

scinstall: 安全なアンインストール手順については、ドキュメントを参照してください。

scinstall: アンインストールが失敗しました。

このエラーを修正するためには、[271 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。`scinstall -r` を実行する前に、この[手順6](#)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。

```
このノードを参照しているデバイスサービスが存在しないことを確認しています ... 失敗しました
scinstall: このノードはまだ、デバイスサービス「service」をホストするように構成されています
scinstall: このノードはまだ、デバイスサービス「service2」をホストするように構成されています
scinstall: このノードはまだ、デバイスサービス「service3」をホストするように構成されています
scinstall: このノードはまだ、デバイスサービス「dg1」をホストするように構成されています

scinstall: このようなエラーが出たままアンインストールするのは安全ではありません。
scinstall: 安全なアンインストールの手順については、ドキュメントを参照してください。
scinstall: アンインストールが失敗しました。
```

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。またこの節では、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は、クラスタイベントをリアルタイムにトラップします。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 は、最新の 50 件のイベントの読み取り専用テーブルを維持しています。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属性ファイル `/etc/cacao/instances/default/private/cacao.properties` を変更することにより変更できます。

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 8-4 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	276 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	276 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	277 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	278 ページの「SNMP ホストを有効にしてノード上の SNMP トラップを受信する」
SNMP ホストの削除	278 ページの「SNMP ホストを無効にしてノード上の SNMP トラップを受信しない」
SNMP ユーザーの追加	279 ページの「ノード上に SNMP ユーザーを追加する」
SNMP ユーザーの削除	280 ページの「ノードから SNMP ユーザーを削除する」

▼ SNMP イベント MIB を有効にする

この手順では、SNMP イベント MIB を有効にする方法を示します。

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形式もあります。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形式については、[付録 A](#) 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.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。このケースでは、MIB の名前は `event` である必要があります。

▼ SNMP イベント MIB を無効にする

この手順では、SNMP イベント MIB を無効にする方法を示します。

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形式もあります。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形式については、[付録 A](#) 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.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。このケースでは、`event` を指定する必要があります。

▼ 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示します。

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形式もあります。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形式については、[付録 A](#) 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.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。このケースでは、`event` を指定する必要があります。

▼ SNMP ホストを有効にしてノード上の SNMP トラップを受信する

この手順では、別のノード上の 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。

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形式もあります。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形式については、[付録 A](#) 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.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# clnmphost 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 ホストを無効にしてノード上の SNMP トラップを受信しない

この手順では、MIB のトラップ通知を受信するホストのリストから、ノード上の SNMP ホストを削除する方法を説明します。

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形式もあります。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形式については、[付録 A](#) 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.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# clnmiphost 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* オプションを指定して *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。

▼ ノード上に SNMP ユーザーを追加する

この手順では、ノード上の SNMP ユーザー構成ファイルに SNMP ユーザーを追加する方法を示します。

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形式もあります。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形式については、[付録 A](#) 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.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。

▼ ノードから **SNMP ユーザー** を削除する

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示します。

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形式もあります。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形式については、[付録 A](#) 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.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。

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 制御機能の構成の詳細については、`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。

- 281 ページの「CPU 制御の概要」
- 283 ページの「CPU 制御の構成」

CPU 制御の概要

Sun Cluster を使用すると、CPU の使用率を制御できます。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の詳細は、『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 9-1 で、使用可能なさまざまな構成シナリオを説明します。

表 9-1 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>283 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 の大域ゾーンで動作する</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>非大域ゾーンが構成されているかどうかに関係なく、この手順を実行できます。</p>	<p>285 ページの「大域ゾーンで 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非大域ゾーンで動作する</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>287 ページの「デフォルトのプロセッサセットを使用して非大域ゾーンの CPU 使用率を制御する」</p>

表 9-1 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非大域ゾーンで動作する</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。専用のプロセッサセットを使用することにより、この制御は非大域ゾーンでのみ実行できます。</p>	<p>290 ページの「専用のプロセッサセットを使用して非大域ゾーンの 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

この節では次の作業について説明します。

- 283 ページの「SPARC: Solaris 9 OS で CPU 使用率を制御する」
- 285 ページの「大域ゾーンで CPU 使用率を制御する」
- 287 ページの「デフォルトのプロセッサセットを使用して非大域ゾーンの CPU 使用率を制御する」
- 290 ページの「専用のプロセッサセットを使用して非大域ゾーンの 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 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 制御機能の構成の詳細については、`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の詳細については、`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。

▼ 大域ゾーンで CPU 使用率を制御する

大域ゾーン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 は、大域ゾーン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。

- 大域ゾーン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)。
- 大域ゾーンに `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 制御機能の構成の詳細については、`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の詳細については、`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、大域ゾーンのシェア数と、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、大域ゾーンで動作中のプロセスを、非大域ゾーン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 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。

-p globalzonestshares=*integer* 大域ゾーン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、大域ゾーンのプロパティを設定します。これらのプロパティを設定しないと、非大域ゾーンで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 project.cpu-shares の数を指定し、大域ゾーンに割り当てられているCPUシェア zone.cpu-shares の数を判別します。

resource_group_name リソースグループの名前を指定します。

この手順では、`RG_SLM_PSET_TYPE` プロパティは設定しません。大域ゾーンでは、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非大域ゾーンの CPU 使用率を制御する

非大域ゾーンのリソースグループに対して CPU シェアを割り当てたいが、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 は、非大域ゾーンのその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_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。
- 非大域ゾーンを `SCSLM_poolzone_name` プールに動的にバインドします。
- 非大域ゾーン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)。
- 非大域ゾーンに `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 制御機能の構成の詳細については、`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の詳細については、`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、大域ゾーンのシェア数と、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、大域ゾーンで動作中のプロセスを、非大域ゾーン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
```

大域ゾーン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、大域ゾーン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 `project.cpu-shares` の数を指定し、非大域ゾーンに割り当てられている CPU シェア `zone.cpu-shares` の数を判別します。

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。

デフォルトのプール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のプール以外のプールに動的にバインドされている場合、非大域ゾーンで `RG_SLM_TYPE` を `automated` に設定することはできません。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` や `LogicalHostname` リソースなどのリソースは、非大域ゾーン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、大域ゾーンで起動されます。`RG_SLM_TYPE` プロパティを `automated` に設定した場合であっても、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、非大域ゾーン内の CPU 制御に対して設定されなくなった場合、非大域ゾーンの 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非大域ゾーンの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させる場合は、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 は、非大域ゾーンの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。
 - 非大域ゾーンを `SCSLM_pool_zone_name` プールに動的にバインドします。
 - 非大域ゾーン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)。
 - 非大域ゾーンに `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
```

prionctl 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の詳細については、dispadmin(1M) および prionctl(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、大域ゾーンのシェア数と、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、大域ゾーンで動作中のプロセスを、非大域ゾーン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 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 globalzonestshares=*integer* 大域ゾーン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、大域ゾーン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
```

<code>-p RG_SLM_TYPE=automated</code>	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。
<code>-p RG_SLM_CPU_SHARES=value</code>	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 <code>project.cpu-shares</code> の数を指定し、非大域ゾーンに割り当てられている CPU シェア <code>zone.cpu-shares</code> の数とプロセッサセット内のプロセッサの最大数を判別します。
<code>-p RG_SLM_PSET_TYPE=value</code>	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 <code>strong</code> または <code>weak</code> に設定します。値 <code>strong</code> と <code>weak</code> 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、 <code>strong</code> と <code>weak</code> が混在するように構成することはできません。
<code>-p RG_SLM_PSET_MIN=value</code>	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。
<code>resource_group_name</code>	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。

デフォルトのプール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のプール以外のプールに動的にバインドされている場合、非大域ゾーンで `RG_SLM_TYPE` を `automated` に設定することはできません。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` や `LogicalHostname` リソースなどのリソースは、非大域ゾーン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、大域ゾーンで起動されます。`RG_SLM_TYPE` プロパティを `automated` に設定した場合であっても、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、非大域ゾーン内の CPU 制御に対して設定されなくなった場合、非大域ゾーンの 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。

- 295 ページの「Sun Cluster へのパッチの適用の概要」
- 297 ページの「クラスタへのパッチの適用」

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。Sun Cluster 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 ファイルを確認してください。また、使用している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断してください。

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) パッチまたはファームウェアを適用するには、`boot-sx` または `shutdown -g -y -i0` コマンドを使用して、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。

このようにパッチを適用する間、個々のノードが一時的に停止しても、クラスタ自体は使用できます。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せん。

Rebooting patch (cluster)

ソフトウェアまたはファームウェアパッチを適用するには、クラスタを停止し、`boot -sx` または `shutdown -g -y -i0` コマンドを使用して、各ノードをシングルユーザーモードで起動します。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停止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。
- 使用している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断してください。
- 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。

クラスタへのパッチの適用

表 10-1 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	305 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	297 ページの「再起動パッチを適用する(ノード)」 301 ページの「再起動パッチを適用する(クラスタ)」
Apply a patch in single-user mode when your cluster contains failover zones	306 ページの「シングルユーザーモードでフェイルオーバーゾーンにパッチを適用する」
Sun Cluster パッチを削除	309 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。この手順では、まず、ノードを停止し、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この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。

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形式もあります。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形式については、[付録 A](#) 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` 指定したノードから、すべての非大域ゾーンを含む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、以下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、以下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適切な Solaris エントリを選択し、`e` を入力してその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)』の第 11 章「GRUB ベースのブート (手順)」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 -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 を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、*node* 上の非大域ゾーンの名前。リソースグループを作成した際に非大域ゾーンを指定した場合にかぎり、*zone* を指定します。

例 10-1 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

```

参照 パッチを取り消す必要がある場合は、[309 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)

この手順では、パッチを適用する前にまずクラスタを停止して、`boot -sx` または `shtudown -g -y -i0` コマンドを使用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、以下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、以下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適切な Solaris エントリを選択し、e を入力してその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 ベースの起動の詳細は、『Solaris のシステム管理 (基本編)』の第 11 章「GRUB ベースのブート (手順)」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 -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を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 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 10-2 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

```

参照 パッチを取り消す必要がある場合は、309 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。

例 10-3 非再起動 Sun Cluster パッチの適用

```

# patchadd -M /tmp/patches 234567-05
...
# showrev -p | grep 234567-05

```

参照 パッチを取り消す必要がある場合は、309 ページの「Sun Cluster パッチの変更」を参照してください。

▼ シングルユーザーモードでフェイルオーバーゾーンにパッチを適用する

シングルユーザーモードでフェイルオーバーゾーンにパッチを適用するには、この作業を実行します。このパッチ方法は、Sun Cluster とのフェイルオーバー構成で Sun Cluster データサービスを Solaris コンテナに使用する場合に必要です。

- 1 定足数デバイスが、この手順内で手動で採用されるゾーンパスがディスクセットの一部である共有ストレージの LUN 用に構成されていないことを確認してください。
 - a. 定足数デバイスがゾーンパスを含むディスクセット内で使用されるかどうか、および定足数デバイスが **scsi2** または **scsi3** の予約を使用するかどうかを判断します。

```
# 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、ゾーンパスを含まないディスクセットにも属さ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てください。

```
# clquorum remove old-didname
```
 - d. **scsi2** の予約が古い定足数デバイスに使用されている場合は、古い定足数から **scsi2** の予約を消し込み、**scsi2** の予約が残っていないことを確認してください。

```
# /usr/cluster/lib/sc/pgre -c pgre_scrub -d /dev/did/rdisk/old-didnames2
# /usr/cluster/lib/sc/pgre -c pgre_inkeys -d /dev/did/rdisk/old-didnames2
```

注-アクティブな定足数デバイスの予約キーを誤って消し込んだ場合は、定足数デバイスを削除してから再度追加し、定足数デバイスに新しい予約キーを付ける必要があります。

- 2 パッチを適用するノードを退避します。

```
# 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 リソースグループ内のオフラインにした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 クラスタからパッシブノードを起動します。
`# reboot -- -x`
- 9 次に進む前に、SMF 起動メソッドがパッシブノードで完了していることを確認してください。
`# svcs -x`
- 10 アクティブノード上の再構成プロセスがすべて完了していることを確認してください。
`# cluster status`
- 11 ディスクセット内のディスクに `scsi3` の予約があるかどうかを判断します。
 - a. ディスクセット内のすべてのディスクに対して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. キーが一覧表示される場合は、消し込みます。
`# /usr/cluster/lib/sc/scsi -c scrub -d /dev/did/rdisk/didnames2`
- 12 パッシブノード上のメタセットの所有権を取得します。
`# metaset -s disksetname -C take -f`
- 13 パッシブノード上のゾーンパスを含む1つまたは複数のファイルシステムをマウントします。
`# mount device mountpoint`
- 14 パッシブノード上でシングルユーザーモードに切り替えます。
`# init s`
- 15 起動されたゾーンのうち、Solaris コンテナ用 Sun Cluster データサービスによって制御されていないゾーンをすべて停止します。
`# zoneadm -z zonename halt`

- 16 (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、すべての構成された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
```
- 17 パッチを適用します。
- 18 ノードを再起動し、ノードのすべてのSMF起動メソッドが完了するまで待機します。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
```
- これで最初のノードの準備が整いました。
- 19 パッチを適用する2番目のノードを退避します。
- ```
# clresourcegroup evacuate -n node2
```
- 20 2番目のノードに対して、手順8から手順13を繰り返します。
- 21 パッチプロセスの速度を上げるため、パッチを適用済のゾーンを切り離します。
- ```
zoneadm -z zonename detach
```
- 22 パッシブノード上でシングルユーザーモードに切り替えます。
- ```
# init s
```
- 23 起動されたゾーンのうち、Solaris コンテナ用 Sun Cluster データサービスによって制御されていないゾーンをすべて停止します。
- ```
zoneadm -z zonename halt
```
- 24 (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、すべての構成された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
```
- 25 パッチを適用します。
- 26 切り離したゾーンを接続します。
- ```
zoneadm -z zonename attach -F
```
- 27 ノードをクラスタモードで再起動します。
- ```
# reboot
```

- 28 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。
- 29 リソースグループを起動します。

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 パッチを削除してから、以前のパッチを再適用するか、リリースを更新します。新しい Sun Cluster パッチを削除するには、次に示す手順を参照してください。以前の Sun Cluster パッチを再適用するには、次に示す手順のいずれかを参照してください。

- [297 ページの「再起動パッチを適用する\(ノード\)」](#)
- [301 ページの「再起動パッチを適用する\(クラスタ\)」](#)
- [305 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 ファイルを確認してください。

▼ 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。ノードの非クラスタモードでの起動についての詳細は、[73 ページの「非クラスタモードでクラスタノードを起動する」](#)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 クラスタノードをふたたびクラスタモードで再起動します。

```
# reboot
```

- 5 各クラスタノードに対して、手順2から手順4を繰り返します。

◆◆◆ 第 11 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。

- 311 ページの「クラスタのバックアップ」
- 324 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 11-1 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	312 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	312 ページの「完全バックアップに必要なテープ数を判別する」
ルートファイルシステムのバックアップの作成	313 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	316 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)」 319 ページの「SPARC: 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	323 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別します。

- 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 11-1 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    -
```

▼ 完全バックアップに必要なテープ数を判別する

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 11-2 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。

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形式もあります。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形式については、[付録 A](#) 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:次のように入力します。

```
ok boot -x
```

- x86 ベースのシステム上で、以下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適切な Solaris エントリを選択し、e を入力してその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)』の第 11 章「GRUB ベースのブート (手順)」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 -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**を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 ルート (/) ファイルシステムをバックアップします。
- ルートディスクがカプセル化されていない場合は、次のコマンドを使用します。

```
# ufsdump 0ucf dump-device /
```

- ルートディスクがカプセル化されている場合は、次のコマンドを使用します。

```
# ufsdump 0ucf dump-device /dev/vx/rdsk/rootvol
```

詳細については、`ufsdump(1M)` のマニュアルページを参照してください。

- 6 ノードをクラスタモードで再起動します。

```
# init 6
```

例 11-3 ルート (/) ファイルシステムのバックアップ

次に、ルート (/) ファイルシステムをテープデバイス `/dev/rmt/0` にバックアップする例を示します。

```
# ufsdump 0ucf /dev/rmt/0 /
DUMP: Writing 63 Kilobyte records
DUMP: Date of this level 0 dump: Tue Apr 18 18:06:15 2000
```

```
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c0t0d0s0 (phys-schost-1:/) to /dev/rmt/0
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Estimated 859086 blocks (419.48MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 859066 blocks (419.47MB) on 1 volume at 2495 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 0 dump on Tue Apr 18 18:06:15 2000
```

▼ 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 Volume Manager 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ため、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。

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形式もあります。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形式については、[付録 A](#) 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 ポリユーモマネージャ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確認します。

```
# 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、それ以外の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 11-4 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/rdisk/c1t9d0s0.
...
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)
...

```

▼ SPARC: 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。

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形式もあります。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形式については、[付録 A](#) 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 ポリユームのスナップショットを作成します。

```
# vxassist -g diskgroup snapstart volume
```

注-ポリユ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 新しいポリユ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ポリユームを作成し、それにスナップショットポリユームを添付します。

```
# 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、*node* 上の非大域ゾーンの名前。リソースグループを作成した際に非大域ゾーンを指定した場合にのみ、*zone* を指定します。

- 9 そのポリユームが新しいポリユーム 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/rdsk/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、それ以外の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 11-5 SPARC: 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 /vo101 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	clt2d0s2	-	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。Sun Cluster 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 構成ファイルの詳細については、`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)` コマンドを使うと、`ufsdump(1M)` コマンドで作成されたバックアップから現在の作業ディレクトリに対する相対パスで指定されるディスク上の位置にファイルがコピーされます。`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 11-2 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	325 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	325 ページの「ルート (/) ファイルシステムを復元する (/)」
	327 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」

表 11-2 作業リスト:クラスタファイルの復元 (続き)

作業	参照先
VERITAS Volume Manager の場合、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	333 ページの「SPARC: 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	335 ページの「SPARC: 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。復元するノードは起動しないでください。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。

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形式もあります。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形式については、[付録 A](#) 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、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: Rpcbind 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 27 章「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 ディスク 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 11-6 ルート (/) ファイルシステムの復元 (Solaris ボ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 node to be restored
```

```
.]
```

```
[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 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

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 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。復元するノードは起動しないでください。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。

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形式もあります。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形式については、[付録 A](#) 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
```

```
-s setname      メタ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            メタセットから削除します。
-h nodelist    メタセットから削除するノード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 Volume Manager** ボリュームからメタデバイスまたはボ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 ノードをマルチユーザーモードで再起動します。


```
# reboot
```
- 14 ディスク ID を交換します。


```
# cldevice repair rootdisk
```

- 15 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。
- 16 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 11-7 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/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 with access to the metaset, other than the node to be restored.]

[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、電源を切つ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 re-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
[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/rdsk/c0t0d0s0 /usr ufs 1 no -
[Unmount the temporary file system and check the raw disk device:]
# cd /
# umount /a

```

```
# fsck /dev/rdisk/c0t0d0s0
[Reboot:]
# reboot
[Replace the disk ID:]
# cldevice repair /dev/rdisk/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
```

▼ SPARC: 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、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。復元するノードは起動しないでください。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。

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形式もあります。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形式については、[付録 A](#) 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 ディスク 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 11-8 SPARC: 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
```

▼ SPARC: 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、カプセル化されたルート (/) ファイルシステムをノードに復元します。復元するノードは起動しないでください。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。

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形式もあります。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形式については、[付録 A](#) 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。
例:
変更前の—

```
/dev/vx/dsk/rootdg/rootvol /dev/vx/rdisk/rootdg/rootvol /      ufs  1    no  -
```


変更後の—

```
/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、ディスク 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、`vxinstall` 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。

```
# vxdg reminor 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 11-9 SPARC: 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, remminor the rootdg disk group
:]
# umount /global/.devices/node@nodeid
# vxdg remminor rootdg 100
# shutdown -g0 -i6 -y
```

参照 カプセル化されたルートディスクのミラーリングに関する指示については、『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 (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 に含まれているオンラインヘルプでは、Sun Cluster のさまざまな管理作業を行うための手順が示されています。

この章の内容は、次のとおりです。

- 341 ページの「Sun Cluster Manager の概要」
- 342 ページの「SPARC: Sun Management Center の概要」
- 343 ページの「Sun Cluster Manager の構成」
- 346 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** のインストールと起動: 『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。
- ポート番号、サーバーアドレス、セキュリティ証明書、ユーザーの構成: [343 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: [345 ページの「共通エージェントコンテナのセキュリティ鍵を再生成する」](#) 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 用の Sun Cluster モジュールのインストールと起動については、『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 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 を使用方法については、『Sun Management Center 3.5 ユーザーガイド』の「Sun Management Center 対応モジュール SNMP MIB」にあるそのほかの Sun Management Center MIB を使用するための指示を参照してください。

Sun Cluster Manager の構成

Sun Cluster Manager は、定数デバイス、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 を各自の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。

- [343 ページの「RBAC の役割の設定」](#)
- [344 ページの「Sun Cluster Manager のサーバーアドレスを変更する」](#)
- [345 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」](#)

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別します。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 の RBAC を設定および管理する方法の詳細は、[第 2 章](#)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/SUNWcacao/bin/cacaoadm stop
```

- 2 Sun Java Web Console を停止します。

```
# /usr/sbin/sunmcwebserver 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/SUNWcacao/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/SUNWcacao/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/sunmcwebserver 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/SUNWcacao/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もと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/SUNWcacao/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/SUNWcacao/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/SUNWcacao/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/SUNWcacao/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ーフェース (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。

▼ 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。
 - 使用する場合は、[手順 5](#)に進みます。
 - そうでない場合は、[手順 3](#)に進んで、Sun Cluster Manager の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) コマンドを使って、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、51 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 **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。
- 9 **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。
- 10 「ログイン」 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。
- 11 「システム」 カテゴリの下にある「Sun Cluster Manager」 リンクをクリックします。
- 12 **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。
- 13 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/`

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、その短形式、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、短形式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 A-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
<code>claccess</code>	なし	Sun Cluster のアクセスポリシーの管理
<code>cldevice</code>	<code>cldev</code>	Sun Cluster デバイスの管理
<code>cldevicegroup</code>	<code>cldg</code>	Sun Cluster デバイスグループの管理
<code>clinterconnect</code>	<code>clintr</code>	Sun Cluster インターコネクットの管理
<code>clnasdevice</code>	<code>clnas</code>	Sun Cluster の NAS デバイスへのアクセスの管理
<code>clnode</code>	なし	Sun Cluster ノードの管理
<code>clquorum</code>	<code>clq</code>	Sun Cluster 定足数の管理
<code>clquorumserver</code>	<code>clqs</code>	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
<code>clreslogicalhostname</code>	<code>clrslh</code>	論理ホスト名のための Sun Cluster リソースの管理
<code>clresource</code>	<code>clrs</code>	Sun Cluster データサービスのリソースの管理

表 A-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 のグローバルな構成とステータスの管理
clvxvm	なし	Veritas Volume Manager for Sun Cluster の構成

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 A-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 A-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	グローバル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ステータスを表示します。
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 A-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 A-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、あるプライマリ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 A-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 A-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 A-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
<code>export</code>	クラスタ NAS デバイス構成情報をエクスポートします。
<code>list</code>	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。
<code>remove</code>	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。
<code>remove-dir</code>	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。
<code>set</code>	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。
<code>show</code>	クラスタ内の NAS デバイスの構成情報を表示します。

表 A-7 `clnode`: Sun Cluster ノードの管理

サブコマンド	目的
<code>add</code>	ノードをクラスタに構成および追加します。
<code>add-farm</code>	ファームノードをクラスタに追加します。
<code>clear</code>	Sun Cluster ソフトウェア構成からノードを削除します。
<code>evacuate</code>	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。
<code>export</code>	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。
<code>list</code>	クラスタまたはファームに構成されているノードの名前を表示します。
<code>remove</code>	ノードをクラスタから削除します。
<code>remove-farm</code>	ファームノードをクラスタから削除します。
<code>set</code>	指定したノードに関連付けられたプロパティを設定します。
<code>show</code>	指定されたノード (1 つまたは複数) の構成を表示します。
<code>show-rev</code>	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。
<code>status</code>	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 A-8 `clquorum`、`clq`: Sun Cluster の定足数構成の管理

サブコマンド	目的
<code>add</code>	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。
<code>disable</code>	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 A-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	定足数デバイスのステータスと投票数を表示します。

表 A-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 A-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	論理ホスト名の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付けられた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 A-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
status	論理ホスト名リソースのステータスを表示します。
unmonitor	論理ホスト名リソースに対する監視をオフにします。

表 A-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
create	コマンドに対するオペランドとして指定されたリソースを作成します。
delete	コマンドに対するオペランドとして指定されたリソースを削除します。
disable	リソースを無効にします。
enable	リソースを有効にします。
export	クラスタリソース構成をエクスポートします。
list	クラスタリソースのリストを表示します。
list-props	リソースプロパティのリストを表示します。
monitor	リソースの監視をオンにします。
reset	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。
set	リソースプロパティを設定します。
show	リソース構成を表示します。
status	リソースのステータスを表示します。
unmonitor	リソースの監視をオフにします。

表 A-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
add-node	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。
create	新しいリソースグループを作成します。
delete	リソースグループを削除します。
evacuate	-n オプションで指定する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。
export	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。
list	リソースグループのリストを表示します。

表 A-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	指定するリソースグループを、もっと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 NodeList プロパティから削除します。
restart	元々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	指定する保存停止されたリソースグループの保存停止状態をクリアします。
set	指定する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	指定するリソースグループの構成レポートを生成します。
status	指定する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	指定する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、Resource Group Manager (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 A-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 A-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 A-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 A-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 A-16 `clsnmpmib`: Sun Cluster SNMP MIB の管理

サブコマンド	目的
<code>disable</code>	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。
<code>enable</code>	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。
<code>export</code>	クラスタの MIB の構成情報をエクスポートします。
<code>list</code>	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。
<code>set</code>	1つまたは複数の MIB で使用される SNMP プロトコルを設定します。
<code>show</code>	指定されたノード上の MIB の構成情報を表示します。

表 A-17 `clsnmpuser`: Sun Cluster SNMP ユーザーの管理

サブコマンド	目的
<code>create</code>	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。
<code>delete</code>	SNMPv3 ユーザーを指定されたノードから削除します。
<code>export</code>	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。
<code>list</code>	指定されたノードで設定されている SNMPv3 ユーザーのリストを出力します。
<code>set</code>	指定されたノード上のユーザーの構成を設定します。
<code>set-default</code>	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。
<code>show</code>	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 A-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
<code>disable</code>	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。
<code>enable</code>	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。
<code>export</code>	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 <code>stdout</code> にエクスポートします。
<code>list</code>	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。
<code>print</code>	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 A-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 A-19 cluster: クラスタのグローバルな構成とステータスの管理

サブコマンド	目的
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実行対象であるクラスタの名前を表示します。
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実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実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実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実行対象であるクラスタを順序正しく停止します。
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 A-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。

索引

A

Add Administrative Role ウィザード, 説明, 51
administering, cluster file systems, 140
autoboot プロパティ, 262
Availability Suite, データ複製用に使用, 83

B

boot コマンド, 60-62

C

cconsole コマンド, 21, 24
ccp コマンド, 19, 24
claccess, 350
claccess コマンド, 19
cldev, 351
cldevice, 351
cldevicegroup, 352
cldevicegroup コマンド, 19
cldevice コマンド, 19
cldg, 352
clinterconnect, 352
clinterconnect コマンド, 19
clintr, 352
clnas, 353
clnasdevice, 353
clnasdevice コマンド, 19
clnode, 353
clnode check コマンド, 19

clq, 354
clqs, 354
clquorum, 354
clquorumserver, 354
clquorum コマンド, 19
clreslogicalhostname, 355
clreslogicalhostname コマンド, 19
clresource, 355
clresourcegroup, 356
clresourcegroup コマンド, 19
clresourcetype, 357
clresourcetype コマンド, 19
clresource コマンド, 19
clressharedaddress, 357
clressharedaddress コマンド, 19
clrg, 356
clrs, 355
clrslh, 355
clrssa, 357
clrt, 357
clsetup
 command, 19
 アクセス, 25
 管理ツール, 18
clsnmphost, 357
clsnmphost コマンド, 19
clsnmpmib, 358
clsnmpmib コマンド, 19
clsnmpuser, 358
clsnmpuser コマンド, 19
clta, 359
cltelemattribute コマンド, 19

cltelemetryattribute, 359
cluster, 359
cluster file system, administration overview, 140
cluster check コマンド, 19
cluster shutdown コマンド, 57
clvsvm, 359
commands, clnasdevice, 19
CPU シェア
 構成, 281
 制御, 281
 大域ゾーン, 285
 非大域ゾーン, 287
 専用のプロセッサセット, 290
CPU 制御, 概要, 281
crlogin コマンド, 24
cssh コマンド, 24
ctelnet コマンド, 24

D

DR, 動的再構成, 201

E

/etc/nsswitch.conf ファイル, 非大域ゾーンの変更, 263
/etc/vfstab ファイル, 44

F

failback プロパティ, 171

G

GUI, グラフィカルユーザーインターフェース, 341

I

IPMP
 管理, 239

IPMP (続き)
 ステータス, 32

K

/kernel/drv/, md.conf ファイル, 146

M

md.tab ファイル, 22
metaset コマンド, 125-128
MIB
 SNMP イベントの無効化, 276
 SNMP イベントの有効化, 276
MIB (管理情報ベース), SNMP イベントプロトコルの変更, 277

N

NAS, ネットワーク接続ストレージ, 204
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, 21
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
NFS アプリケーション用のファイルシステム, データ複製用の構成, 99-100
nsswitch.conf ファイル, 非大域ゾーンの変更, 263
ntp.conf.cluster ファイル, 252
numsecondaries プロパティ, 173

O

OpenBoot PROM (OBP), 249

P

patching Sun Cluster, 295-297

R

- RBAC, 49-56
 - 権限プロファイル (説明), 50-51
 - 作業
 - カスタム役割を追加, 54
 - 使用, 49
 - 設定, 49
 - 役割の追加, 51
 - ユーザーの変更, 55
 - 大域ゾーンの, 50
 - 非大域ゾーンの, 50
- RBAC の役割, Sun Cluster Manager, 343
- Role-Based Access Control, 「RBAC」を参照

S

- SCSI 定足数デバイス, 追加, 202
- Service Management Facility (SMF), オンラインサービスの確認, 262
- showrev -p コマンド, 26, 27
- SMF, オンラインサービスの確認, 262
- SnapShot, 「ストレージベースの複製」を参照
- SNMP イベント MIB
 - プロトコルの変更, 277
 - 無効化, 276
 - 有効化, 276
- SNMP イベント MIB の無効化, 276
- SNMP イベント MIB の有効化, 276
- SNMP イベント MIB プロトコルの変更, 277
- SNMP ホストの削除, 278
- SNMP ホストの追加, 278
- SNMP ユーザーの削除, 280
- SNMP ユーザーの追加, 279
- Solaris 9 OS, CPU シェアの制御, 283
- Solaris ゾーン, autoboot プロパティ, 262
- ssh, 「クラスタコンソールへのセキュリティー保護された接続」を参照
- Sun Cluster Manager, 19, 341
 - RBAC の役割, 設定, 343
 - 起動, 346
 - サーバーアドレスの変更, 344
- Sun Management Center, 19, 342
 - インストール, 21

- Sun StorageTek Availability Suite, データ複製用に使用, 83
- Sun Cluster Manager ソフトウェアの起動, 346
- Sun Cluster Manager の起動, 346
- Sun Cluster ソフトウェアのアンインストール, 271
- SunMC, 19
 - Sun Management Center, 342
- supported features, VxFS, 140
- System Service Processor (SSP), 21

T

- TrueCopy, ベストプラクティス, 83

V

- /var/adm/messages ファイル, 76
- VERITAS 管理, 127-128
- VxFS supported features, 140
- VxVM, 127-128

あ

- アクセス権, 広域デバイスの, 126
- アダプタ, トランスポート, 229
- アプリケーションリソースグループ
 - ガイドライン, 88
 - データ複製の構成, 105-107

い

- 一覧表示
 - 定足数構成, 220
 - デバイスグループ構成, 176
- イベント MIB
 - SNMP (簡易ネットワーク管理プロトコル) の無効化, 276
 - SNMP の有効化, 276
- イベント MIB (管理情報ベース) プロトコル, SNMP (簡易ネットワーク管理プロトコル) の変更, 277

え

エラーメッセージ, /var/adm/messages ファイル, 76

か

概要, 定足数, 199-202

確認

SMF サービス, 262

クラスタインターコネクトの状態, 225

広域マウントポイント, 44, 191

データ複製の構成, 118-121

ファイルシステムの名前, 312

監視, ディスクパス, 192-194

監視解除, ディスクパス, 194-195

管理

IPMP, 223-242

クラスタインターコネクトとパブリックネットワーク, 223-242

グラフィカルユーザーインタフェース (GUI) ツールによるクラスタの, 341-347

広域クラスタ設定, 243-280

大域ゾーン, 17

定足数, 199-202

非大域ゾーン, 17

管理コンソール, 21

き

起動

クラスタ, 57-76

ノード, 64-75

非クラスタモード, 73

非大域ゾーン, 64

共通エージェントコンテナ, ポート番号の変更, 343

共通エージェントコンテナのセキュリティー鍵, 再生成, 345

切り換え, デバイスグループの主ノード, 177-178

く

クラスタ

管理, 243-280

起動, 57-76

構成の検証, 42

構成の表示, 33

コンポーネントステータス, 30

再起動, 62

再起動パッチの適用, 301

時刻の設定, 247

停止, 57-76

名前の変更, 244-245

ノード認証, 245

バックアップ, 22, 311-324

ファイルの復元, 324

クラスタインターコネクト

管理, 223-242

状態の確認, 225

動的再構成, 224

クラスタコンソールへのセキュリティー保護された接続, 25

クラスタコントロールパネル (CCP), 21

クラスタの起動, 60-62

クラスタの時刻の設定, 247

クラスタファイルシステム, 125-198

削除, 189-191

大域ゾーン, 140

追加, 185-188

非大域ゾーン, 140

グラフィカルユーザーインタフェース (GUI) 管理ツール, 18, 341-347

Sun Cluster Manager, 341

Sun Management Center, 342

グローバル

デバイス

動的再構成, 126-127

名前空間, 125-128, 145

グローバルな名前空間の更新, 145

け

ケーブル, トランスポート, 229

権限プロファイル, RBAC, 50-51

検索, ノード ID, 245

検証、クラスタ構成の, 42

こ

広域

- デバイス, 125-198
 - アクセス権の設定, 126
- マウントポイント
 - 確認, 44, 191

構成

- データ複製, 83-124
 - デバイスグループのマイナー番号, 157

構成例 (構内クラスタ化)

- 2空間、ストレージベースのデータ複製, 80-83
- 2空間、ストレージベースの複製, 80-83

構内クラスタ化

- ストレージベースのデータ複製, 80-83
- ストレージベースのデータ複製を使用した復旧, 82-83

構内クラスタリング, 構成例, 79

公平配分スケジューラ

- CPU シェアの構成, 283
- 構成, 284

コマンド, 349-359

- boot, 60-62
- cconsole, 21, 24
- ccp, 19, 24
- claccess, 19
- cldevice, 19
- cldevicegroup, 19
- clinterconnect, 19
- clnode check, 19
- clquorum, 19
- clreslogicalhostname, 19
- clresource, 19
- clresourcegroup, 19
- clresourcetype, 19
- clressharedaddress, 19
- clsetup, 19
- clsnmpghost, 19
- clsnmpmib, 19
- clsnmpuser, 19
- cltelemetryattribute, 19
- cluster check, 19

コマンド (続き)

- crlogin, 24
- cssh, 24
- ctelnet, 24
- metaset, 125-128
- netcon, 21
- sccheck, 22, 42, 44
- scshutdown, 57

コマンド行管理ツール, 18

コンソール

- セキュリティー保護された接続, 25
- への接続, 24

さ

サービスから外す, 定足数デバイス, 217

再起動

- クラスタ, 62
- ノード, 70-72

最後の定足数デバイス, 削除, 212

再生成

- 共通エージェントコンテナのセキュリティー鍵, 345
- セキュリティー鍵, 345

削除

- Solstice DiskSuite デバイスグループ, 148
- クラスタファイルシステム, 189-191
- 最後の定足数デバイス, 212
- ストレージレイ, 269
- すべてのデバイスグループからノードを, 148
- 定足数デバイス, 201, 211
- デバイスグループ, 165
- デバイスグループからノードを, 167
- デバイスグループから、ボリューム, 164-165
- トランスポートケーブル、アダプタ、スイッチ, 229
- ノード, 266

作成, 新しいディスクグループ, 153-154

サブコマンド, 349-359

し

修復, 定足数デバイス, 221

修復、満杯の /var/adm/messages ファイル, 76
主所有者権、デバイスグループの, 171
使用, 役割 (RBAC), 49

す

スイッチ、トランスポート, 229
スイッチバック、データ複数での実行ガイドライン, 92
ステータス、クラスタコンポーネント, 30
ストレージアレイ、削除, 269
ストレージベースのデータ複製
 TrueCopy のベストプラクティス, 83
 定義, 78
ストレージベースの複製, 80-83
 制限, 82
 と定足数デバイス, 82-83
 復旧, 82-83
 要件, 81
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, 85

せ

セキユアシエル、「クラスタコンソールへのセキユリティー保護された接続」を参照
セキユアシエル接続, 24
セキユリティー鍵、再生成, 345
設定、役割 (RBAC), 49
専用のプロセッサセット、構成, 290

そ

属性、「プロパティー」を参照

た

大域ゾーン
 CPU シェア, 285
 管理, 17

大域ゾーン (続き)

 クラスタファイルシステムの管理, 140
耐障害性、定義, 84

つ

追加

Network-Attached Storage 定足数, 204
SCSI 定足数デバイス, 202
Solstice DiskSuite デバイスグループ, 148
新しいボリュームをデバイスグループに, 155
カスタム役割 (RBAC), 54
クラスタファイルシステム, 185-188
定足数サーバー定足数デバイス, 207
定足数デバイス, 202
デバイスグループ, 146
トランスポートケーブル、アダプタ、およびスイッチ, 226
ノード, 166, 260
役割 (RBAC), 51

て

停止

クラスタ, 57-76
ノード, 64-75
非大域ゾーン, 64
ディスクグループ
 構成変更の登録, 161
 作成, 153-154
 登録, 158
 変更, 157
ディスクのカプセル化, 154
ディスクパス
 監視, 192-194
 監視解除, 194-195
ディスクパス監視, 125-198
ディスクパスの監視, 191-198
 障害のあるディスクパスを表示, 195
定足数
 概要, 199-202
 管理, 199-202

- 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, 207
 - 削除のトラブルシューティング, 212
 - 追加, 207
- 定足数デバイス
 - 交換, 214
 - 構成の一覧表示, 220
 - 最後の定足数デバイスを削除, 212
 - 削除, 201, 211
 - 修復, 221
 - 追加, 202
 - SCSI 定足数デバイス, 202
 - 定足数サーバー定足数デバイス, 207
 - ネットワーク接続ストレージ定足数デバイス, 204
 - デバイスの動的再構成, 201
 - とストレージベースの複製, 82-83
 - ノードリストの変更, 215
 - 保守状態
 - デバイスを保守状態から戻す, 219
 - デバイスを保守状態にする, 217
- 定足数デバイスの交換, 214
- データ複製, 77-124
 - DNS エントリの更新, 123-124
 - ガイドライン
 - スイッチオーバーの管理, 92
 - フェイルオーバーの管理, 92
 - リソースグループの構成, 87
 - 概要, 84
 - 構成
 - NFS アプリケーション用のファイルシステム, 99-100
 - NFS アプリケーションリソースグループ, 105-107
 - アフィニティスイッチオーバー, 88, 102
 - デバイスグループ, 96
 - 構成の確認, 118-121
 - 構成例, 92
 - 実行, 114-121
 - ストレージベースの, 78, 80-83
 - 定義, 77-78
 - 同期, 85
 - データ複製 (続き)
 - 必要なハードウェアとソフトウェア, 94
 - 非同期, 85
 - フェイルオーバーの管理, 121-124
 - ポイントインタイムスナップショット, 85, 116-118
 - ホストベースの, 78, 79
 - 有効化, 109-114
 - リソースグループ
 - アプリケーション, 88
 - 共有アドレス, 91
 - 構成, 88
 - 作成, 102-103
 - スケーラブルアプリケーション, 90-91
 - フェイルオーバーアプリケーション, 89-90
 - 命名規則, 88
 - リモートミラー, 114-116
 - リモートミラー化, 84
 - データ複製のアフィニティスイッチオーバー, データ複製の拡張プロパティ, 88
 - 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, 106, 108
 - 複製リソース, 102, 104
 - データ複製のスイッチオーバー, アフィニティスイッチオーバー, 88
 - データ複製のためのスイッチオーバー, 管理のガイドライン, 92
 - データ複製用のアフィニティスイッチオーバー, データ複製用の構成, 102
 - データ複製用の共有アドレスリソースグループ, 91
 - データ複製用のスイッチオーバー, 実行, 121-124
 - データ複製用のスケーラブルアプリケーション, 90-91
 - データ複製用のフェイルオーバー, 管理, 121-124
 - 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, 88
 - ガイドライン
 - フェイルオーバーの管理, 92
 - リソースグループ, 89-90
- 適用
 - パッチ, 297
 - 非再起動パッチ, 305

デバイス, 広域, 125-198

デバイスグループ

新しいマイナー番号の割当て, 157

確認

登録, 163

管理の概要, 142

構成の表示, 176

削除と登録解除, 148, 165

主所有者権, 171

追加, 148

データ複製用の構成, 96

保守状態, 178

デバイスグループ, 追加, 146

デバイスグループの主ノードの切り替え, 177-178

デバイスグループのプロパティ, 変更, 171

と

同期データ複製, 85

動的再構成, 126-127

クラスタインターコネクト, 224

定足数デバイス, 201

パブリックネットワークインタフェース, 241

登録

ディスクグループ構成の変更, 161

ディスクグループをデバイスグループとして, 158

登録解除

Solstice DiskSuite デバイスグループ, 148

デバイスグループ, 165

ドメインネームシステム (DNS)

更新のガイドライン, 92

データ複製での更新, 123-124

トランスポートアダプタ, 追加, 226

トランスポートアダプタの追加, 229

トランスポートケーブル

追加, 226, 229

無効にする, 233

有効にする, 232

トランスポートスイッチ, 追加, 226

トランスポートスイッチの追加, 229

な

名前空間, グローバル, 125-128

に

二次ノード

希望数の設定, 173

デフォルト数, 171

ね

ネットワーク接続ストレージ, NAS, 204

ネットワーク接続ストレージ定足数デバイス
インストールの要件, 204

追加, 204

ネットワークファイルシステム (NFS), データ複製
用のアプリケーションファイルシステムの構
成, 99-100

の

ノード

ID を検索, 245

起動, 64-75

再起動, 70-72

再起動パッチの適用, 297

削除, 266

主ノード, 171

追加, 260

停止, 64-75

デバイスグループから削除, 148, 167

デバイスグループへの追加, 166

二次ノード, 171

認証, 245

プライマリ, 126-127

への接続, 24

保守状態にする, 255

ノードの再起動, 70-72

は

バックアップ

- クラスタ, 22, 311-324
- ファイルシステム, 312
- ボリュームをオンラインで, 319
- ミラーのオンライン, 316
- ルートファイルシステム, 313

パッチ

- クラスタとファームウェアに適用, 301
- 再起動パッチの適用, 297
- 注意事項, 296
- 非再起動パッチを適用, 305
- 非大域ゾーンでの, 300

パブリックネットワーク

- 管理, 223-242
- 動的再構成, 241

ひ

非クラスタノードの起動, 73

非大域ゾーン

- CPU シェア, 287
 - 専用のプロセッサセット, 290
- nsswitch.conf ファイルの変更, 263
- 管理, 17
- クラスタファイルシステムの管理, 140
- 停止と再起動, 64
- パッチの適用, 300
- プライベートホスト名
 - 削除, 255
 - プライベートホスト名の追加, 253
 - プライベートホスト名の変更, 254

ビットマップ

- ポイントインタイムスナップショット, 85
- リモートミラー複製, 84

非同期データ複製, 85

- 表示, 障害のあるディスクパス, 195
- 表示, クラスタ構成の, 33
- 表示, 構成済みタイプの, 28

ふ

ファイル

- /etc/vfstab, 44
- md.conf, 146
- md.tab, 22
- ntp.conf.cluster, 252
- 対話形式で1つずつ復元, 325

ファイルシステム

- カプセル化されたルートの, 復元, 335
- カプセル化されていないルートの, 復元, 333
- 名前の確認, 312
- バックアップ, 312
- ルートの, 復元, 325
- ルートを復元
 - ボリュームから, 327
 - メタデバイスから, 327

復元

- カプセル化されたルートファイルシステム, 335
- カプセル化されていないルートファイルシステム, 333
- クラスタファイル, 324
- 個々のファイルを対話形式で, 325
- ルートファイルシステム, 325
 - ボリュームから, 327
 - メタデバイスから, 327

複製, 「データ複製」を参照

複製, ストレージベースの, 80-83

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, 82-83

プライベートホスト名

- 削除
 - 非大域ゾーン, 255
 - 非大域ゾーン, 253, 254

プライベートホスト名の変更, 250

プロパティ

- failback, 171
- numsecondaries, 173
- preferenced, 171

プロファイル, RBAC 権限, 50-51

へ

別名, 349-359

変更

- numsecondaries プロパティ、173
- Sun Cluster Manager
 - サーバーアドレス、344
- クラスタ名、244-245
- 主ノード、177-178
- ディスクグループ、157
- 定足数デバイスのノードリスト、215
- プライベートホスト名、250
- プロパティ、171
- ポート番号
 - 共通エージェントコンテナの使用、343
- ユーザー (RBAC)、55

ほ

- ポイントインタイムスナップショット
 - 実行、116-118
 - 定義、85
- ポート番号、共通エージェントコンテナの使用の
変更、343
- 保守、定足数デバイス、217
- 保守状態
 - nodes、255
 - 定足数デバイスを保守状態から戻す、219
 - 定足数デバイスを保守状態にする、217
- ホスト
 - SNMP の削除、278
 - SNMP の追加、278
- ホストベースのデータ複製、79
 - 定義、78
 - 例、83-124
- 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、319
 - デバイスグループからの削除、164-165
 - デバイスグループに追加、155
- ボリュームマネージャ、VERITAS、127-128

ま

- マウントポイント、広域、44
- マルチユーザーサービス、確認、262

み

- 短いコマンド、349-359
- ミラー、オンラインバックアップ、316

む

- 無効にする、トランスポートケーブルを、233

め

- 命名規則、複製リソースグループ、88

や

- 役割
 - カスタム役割を追加、54
 - 設定、49
 - 役割の追加、51

ゆ

- 有効にする、トランスポートケーブルを、232
- ユーザー
 - SNMP の削除、280
 - SNMP の追加、279
 - プロパティの変更、55
- ユーザーアカウントツール、説明、55

り

- リソースグループ
 - データ複製
 - 構成、88
 - 構成のガイドライン、87
 - フェイルオーバーでの役割、88
 - リソース、構成済みタイプを表示、28
- リモート複製、「ストレージベースの複製」を参
照
- 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, 114-116
 定義, 84
リモートログイン, 24
リリース情報, 26, 27

ろ

ローカルゾーン, 「非大域ゾーン」を参照
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, リモート, 24
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, 88

