

Oracle® Solaris Cluster 4.3 システム管理

ORACLE®

Part No: E62254
2016 年 7 月

Part No: E62254

Copyright © 2000, 2016, Oracle and/or its affiliates. All rights reserved.

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクルまでご連絡ください。

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

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

このソフトウェアまたはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアまたはハードウェアは、危険が伴うアプリケーション(人的傷害を発生させる可能性があるアプリケーションを含む)への用途を目的として開発されていません。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用する場合、安全に使用するために、適切な安全装置、バックアップ、冗長性(redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、Oracle Corporationおよびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはオラクル およびその関連会社の登録商標です。その他の社名、商品名等は各社の商標または登録商標である場合があります。

Intel, Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD, Opteron, AMDロゴ, AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。適用されるお客様とOracle Corporationとの間の契約に別段の定めがある場合を除いて、Oracle Corporationおよびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。適用されるお客様とOracle Corporationとの間の契約に定めがある場合を除いて、Oracle Corporationおよびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

ドキュメントのアクセシビリティについて

オラクルのアクセシビリティについての詳細情報は、Oracle Accessibility ProgramのWeb サイト(<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>)を参照してください。

Oracle Supportへのアクセス

サポートをご契約のお客様には、My Oracle Supportを通して電子支援サービスを提供しています。詳細情報は(<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>)か、聴覚に障害のあるお客様は (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>)を参照してください。

目次

このドキュメントの使用方法	21
1 Oracle Solaris Cluster の管理の概要	23
Oracle Solaris Cluster の管理の概要	24
ゾーンクラスタに関する作業	24
Oracle Solaris OS の機能制限	25
管理ツール	26
Oracle Solaris Cluster Manager ブラウザインタフェース	26
コマンド行インタフェース	27
クラスタ管理の準備	29
Oracle Solaris Cluster ハードウェア構成の記録	29
管理コンソールの使用	29
クラスタのバックアップ	30
クラスタの管理	30
リモートからクラスタにログインする	31
クラスタコンソールに安全に接続する方法	32
▼ クラスタ構成ユーティリティにアクセスする方法	32
▼ Oracle Solaris Cluster のリリース情報とバージョン情報を表示する方 法	33
▼ 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方 法	35
▼ クラスタコンポーネントのステータスを確認する方法	38
▼ パブリックネットワークのステータスを確認する方法	40
▼ クラスタ構成を表示する方法	41
▼ 基本的なクラスタ構成を検証する方法	50
▼ グローバルマウントポイントを確認する方法	56
▼ Oracle Solaris Cluster のコマンドログの内容を表示する方法	57
2 Oracle Solaris Cluster と RBAC	61
RBAC の設定と Oracle Solaris Cluster での使用	61
Oracle Solaris Cluster RBAC の権利プロファイル	61

Oracle Solaris Cluster 管理権利プロファイルによる RBAC 役割の作成と割り当て	63
▼ コマンド行から役割を作成する方法	63
ユーザーの RBAC プロパティの変更	64
▼ 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	65
▼ 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	65
3 クラスタの停止とブート	67
クラスタの停止とブートの概要	67
▼ クラスタを停止する方法	69
▼ クラスタをブートする方法	71
▼ クラスタをリブートする方法	75
クラスタ内の 1 つのノードの停止とブート	82
▼ ノードを停止する方法	83
▼ ノードをブートする方法	86
▼ ノードをリブートする方法	90
▼ 非クラスタモードでノードをブートする方法	93
満杯の /var ファイルシステムを修復する	96
▼ 満杯の /var ファイルシステムを修復する方法	96
4 データレプリケーションのアプローチ	97
データレプリケーションについての理解	98
サポートされるデータレプリケーション方式	98
キャンパスクラスタ内でのストレージベースのデータレプリケーションの使用	100
キャンパスクラスタ内でストレージベースのデータレプリケーションを使用する際の要件と制限	102
キャンパスクラスタ内でストレージベースのデータレプリケーションを使用する際の手動回復に関する懸念事項	103
ストレージベースのデータレプリケーションを使用する際のベストプラクティス	103
5 グローバルデバイス、ディスクパスモニタリング、およびクラスタファイルシステムの管理	105
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	105
Solaris Volume Manager のグローバルデバイスのアクセス権	106
グローバルデバイスでの動的再構成	106
ストレージベースのレプリケートされたデバイスの構成および管理	107
EMC Symmetrix Remote Data Facility でレプリケートされたデバイスの管理	108

クラスタファイルシステムの管理の概要	120
クラスタファイルシステムの制限事項	120
デバイスグループの管理	120
▼ グローバルデバイス名前空間を更新する方法	122
▼ グローバルデバイス名前空間で使用する lofi デバイスのサイズを変更する 方法	123
グローバルデバイス名前空間を移行する	124
▼ 専用パーティションから lofi デバイスにグローバルデバイス名前空間を移 行する方法	124
▼ lofi デバイスから専用パーティションにグローバルデバイス名前空間を移 行する方法	125
デバイスグループを追加および登録する	127
▼ デバイスグループを追加および登録する方法 (Solaris Volume Manager)	127
▼ デバイスグループ (raw ディスク) を追加および登録する方法	129
▼ レプリケートデバイスグループ(ZFS)の追加と登録方法	130
▼ HAStoragePlus を使用せずにローカル ZFS ストレージプールを構成する 方法	131
デバイスグループの保守	133
デバイスグループを削除して登録を解除する方法 (Solaris Volume Manager)	133
▼ すべてのデバイスグループからノードを削除する方法	133
▼ デバイスグループからノードを削除する方法 (Solaris Volume Manager)	134
▼ raw ディスクデバイスグループからノードを削除する方法	136
▼ デバイスグループのプロパティを変更する方法	138
▼ デバイスグループのセカンダリノードの希望数を設定する方法	140
▼ デバイスグループ構成の一覧を表示する方法	142
▼ デバイスグループのプライマリノードを切り替える	143
▼ デバイスグループを保守状態にする方法	144
ストレージデバイス用の SCSI プロトコル設定の管理	146
▼ 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 設定を表示する方法	147
▼ 単一ストレージデバイスの SCSI プロトコルを表示する方法	147
▼ すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロ トコル設定を変更する方法	148
▼ 単一ストレージデバイスのフェンシングプロトコルを変更する方法	150
クラスタファイルシステムの管理	151
▼ クラスタファイルシステムを追加する方法	152
▼ クラスタファイルシステムを削除する方法	154
▼ クラスタ内のグローバルマウントを確認する方法	156

ディスクパスモニタリングの管理	157
▼ ディスクパスをモニターする方法	158
▼ ディスクパスのモニターを解除する方法	159
▼ 障害のあるディスクパスを出力する方法	160
▼ ディスクパスのステータスエラーを解決する方法	161
▼ ファイルからディスクパスをモニターする方法	161
▼ モニターしているすべての共有ディスクパスが失敗したときのノードの自動 レポートを有効にする方法	163
▼ すべてのモニター共有ディスクパスが失敗した場合にノードの自動レポート を無効にする方法	164
6 定足数の管理	165
定足数デバイスの管理	165
定足数デバイスへの動的再構成	167
定足数デバイスの追加	167
定足数デバイスの削除または交換	173
定足数デバイスの保守	177
定足数のデフォルトのタイムアウトの変更	183
Oracle Solaris Cluster 定足数サーバーの管理	184
Quorum Server Software の起動および停止	184
▼ 定足数サーバーを起動する方法	184
▼ 定足数サーバーを停止する方法	185
定足数サーバーに関する情報の表示	186
期限切れの定足数サーバークラスタ情報のクリーンアップ	187
7 クラスタインターコネクトとパブリックネットワークの管理	189
クラスタインターコネクトの管理	190
クラスタインターコネクトでの動的再構成	191
▼ クラスタインターコネクトのステータスを確認する方法	191
▼ 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 スイッチを追加する方法	192
▼ 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、およびトランスポート スイッチを削除する方法	194
▼ クラスタトランスポートケーブルを有効にする方法	197
▼ クラスタトランスポートケーブルを無効にする方法	198
▼ 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方法	199
▼ 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 更する方法	200
クラスタインターコネクトのトラブルシューティング	202
パブリックネットワークの管理	204

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方法	204
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	206
8 クラスタノードの管理	209
クラスタまたはゾーンクラスタへのノードの追加	209
▼ 既存のクラスタまたはゾーンクラスタにノードを追加する方法	210
クラスタノードの復元	212
▼ 統合アーカイブからノードを復元する方法	212
クラスタからのノードの削除	216
▼ ゾーンクラスタからノードを削除する方法	217
▼ クラスタソフトウェア構成からノードを削除する方法	219
▼ 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 方法	221
▼ エラーメッセージを修正する方法	223
9 クラスタの管理	225
クラスタの管理の概要	225
▼ クラスタ名を変更する方法	226
▼ ノード ID をノード名にマップする方法	227
▼ 新しいクラスタノードの認証を操作する方法	228
▼ クラスタの時間をリセットする方法	230
▼ SPARC: ノードで OpenBoot PROM (OBP) を表示する方法	232
▼ ノードのプライベートホスト名を変更する	232
▼ ノード名を変更する	235
▼ 既存の Oracle Solaris Cluster の論理ホスト名リソースで使用されてい る論理ホスト名を変更する	236
▼ ノードを保守状態にする	237
▼ ノードを保守状態から戻す	239
▼ クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストー ルする方法	241
ノードのアンインストールのトラブルシューティング	244
Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管 理	245
負荷制限の構成	251
ゾーンクラスタ管理タスクの実行	254
▼ 統合アーカイブからゾーンクラスタを構成する方法	255
▼ 統合アーカイブからゾーンクラスタをインストールする方法	256
▼ ゾーンクラスタにネットワークアドレスを追加する方法	257
▼ ゾーンクラスタを削除する	259
▼ ゾーンクラスタからファイルシステムを削除する	260

▼ ゾーンクラスタからストレージデバイスを削除する	262
テスト目的のトラブルシューティング手順	264
グローバルクラスタ外でのアプリケーションの実行	265
破損したディスクセットの復元	267
10 CPU 使用率の制御の構成	271
CPU 制御の概要	271
シナリオの選択	271
公平配分スケジューラ	272
CPU 制御の構成	272
▼ グローバルクラスタノードで CPU 使用率を制御する方法	272
11 ソフトウェアの更新	275
Oracle Solaris Cluster ソフトウェアの更新の概要	275
Oracle Solaris Cluster ソフトウェアの更新	276
更新のヒント	277
特定のパッケージの更新	277
ゾーンクラスタの更新またはパッチ	279
定足数サーバーまたは AI インストールサーバーの更新	281
パッケージのアンインストール	282
▼ パッケージのアンインストール	282
▼ 定足数サーバーまたは AI インストールサーバーパッケージのアンインストール	282
12 クラスタのバックアップと復元	283
クラスタのバックアップ	283
▼ ミラーのオンラインバックアップを実行する方法 (Solaris Volume Manager)	283
▼ クラスタ構成をバックアップする方法	285
クラスタファイルの復元	286
▼ ZFS ルート (/) ファイルシステム (Solaris Volume Manager) を復元する方法	286
13 Oracle Solaris Cluster Manager ブラウザインタフェースの使用	289
Oracle Solaris Cluster Manager の概要	289
Oracle Solaris Cluster Manager ソフトウェアへのアクセス	290
▼ Oracle Solaris Cluster Manager にアクセスする方法	290
Oracle Solaris Cluster Manager のトラブルシューティング	291
Oracle Solaris Cluster Manager のアクセシビリティのサポートの構成	293

トポロジを使用したクラスタのモニタリング	294
▼ トポロジを使用してクラスタのモニタリングと更新を行う方法	294
A デプロイメントの例: Availability Suite ソフトウェアによるクラスタ間のホストベースの データレプリケーションの構成	297
クラスタにおける Availability Suite ソフトウェアの理解	297
Availability Suite ソフトウェアが使用するデータレプリケーション方式	298
リモートミラーレプリケーション	298
ポイントインタイムスナップショット	299
構成例でのレプリケーション	300
クラスタ間でホストベースのデータレプリケーションを構成するためのガイドライン ...	301
レプリケーションリソースグループの構成	302
アプリケーションリソースグループの構成	302
テイクオーバーの管理のガイドライン	306
タスクマップ: データレプリケーションの構成例	307
クラスタの接続とインストール	307
デバイスグループとリソースグループの構成例	309
▼ プライマリクラスタでデバイスグループを構成する方法	310
▼ セカンダリクラスタでデバイスグループを構成する方法	311
▼ プライマリ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 する方法	313
▼ セカンダリ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 する方法	314
▼ プライマリクラスタでレプリケーションリソースグループを作成する方法	315
▼ セカンダリクラスタでレプリケーションリソースグループを作成する方法	317
▼ プライマリクラスタで NFS アプリケーションリソースグループを作成する方 法	319
▼ セカンダリクラスタで NFS アプリケーションリソースグループを作成する方 法	322
データレプリケーションの有効化例	324
▼ プライマリクラスタでレプリケーションを有効にする方法	324
▼ セカンダリクラスタでレプリケーションを有効にする方法	326
データレプリケーションの実行例	327
▼ リモートミラーレプリケーションを実行する方法	327
▼ ポイントインタイムスナップショットを実行する方法	329
▼ レプリケーションが正しく構成されていることを確認する方法	330
テイクオーバーの管理の例	333
▼ DNS エントリを更新する方法	333
索引	335

図目次

図 1	ストレージベースのデータレプリケーションを装備した 2 ルーム構成	101
図 2	リモートミラーレプリケーション	299
図 3	ポイントインタイムスナップショット	300
図 4	構成例でのレプリケーション	301
図 5	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	304
図 6	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	305
図 7	クライアントからクラスタへの DNS マッピング	306
図 8	クラスタ構成例	308

表目次

表 1	Oracle Solaris Cluster サービス	25
表 2	Oracle Solaris Cluster 管理ツール	30
表 3	タスクリスト: クラスタの停止とブート	68
表 4	タスクマップ: ノードの停止とブート	83
表 5	タスクマップ: ディスクデバイスとテープデバイスでの動的再構成	107
表 6	タスクマップ: EMC SRDF ストレージベースのレプリケートされたデバイスの管理	108
表 7	タスクマップ: デバイスグループの管理	121
表 8	タスクリスト: クラスタファイルシステムの管理	151
表 9	タスクマップ: ディスクパスモニタリングの管理	157
表 10	タスクリスト: 定足数の管理	166
表 11	タスクマップ: 定足数デバイスへの動的再構成	167
表 12	タスクリスト: クラスタインターコネクトの管理	190
表 13	タスクマップ: 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	191
表 14	タスクマップ: 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	206
表 15	タスクマップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加	210
表 16	タスクマップ: ノードの削除	217
表 17	タスクリスト: クラスタの管理	225
表 18	タスクマップ: Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理	245
表 19	その他のゾーンクラスタのタスク	254
表 20	Oracle Solaris Cluster ソフトウェアの更新	276
表 21	タスクリスト: クラスタファイルのバックアップ	283
表 22	タスクリスト: クラスタファイルの復元	286
表 23	タスクマップ: データレプリケーションの構成例	307
表 24	必要なハードウェアとソフトウェア	308
表 25	構成例内のグループとリソースのサマリー	310

例目次

例 1	Oracle Solaris Cluster のリリース情報およびバージョン情報の表示	34
例 2	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	36
例 3	クラスタコンポーネントのステータス確認	38
例 4	パブリックネットワークのステータスを調べる	41
例 5	グローバルクラスタの構成を表示する	42
例 6	ゾーンクラスタの構成を表示する	49
例 7	グローバルクラスタ構成の基本検証 (エラーがない場合)	53
例 8	インタラクティブな妥当性検査のリスト	54
例 9	機能の妥当性検査の実行	54
例 10	グローバルクラスタ構成の検証 (エラーがある場合)	55
例 11	グローバルマウントポイントの確認	56
例 12	Oracle Solaris Cluster のコマンドログの内容の表示	59
例 13	smrole コマンドを使用してカスタムの Operator 役割を作成する	64
例 14	ゾーンクラスタの停止	70
例 15	SPARC: グローバルクラスタの停止	70
例 16	x86: グローバルクラスタの停止	71
例 17	SPARC: グローバルクラスタのブート	72
例 18	x86: クラスタのブート	73
例 19	ゾーンクラスタのリポート	77
例 20	SPARC: グローバルクラスタのリポート	78
例 21	x86: クラスタのリポート	79
例 22	SPARC: グローバルクラスタノードの停止	84
例 23	x86: グローバルクラスタノードの停止	85
例 24	ゾーンクラスタノードの停止	85
例 25	SPARC: グローバルクラスタノードのブート	88
例 26	x86: クラスタノードのブート	89
例 27	SPARC: グローバルクラスタノードのリポート	92
例 28	ゾーンクラスタノードのリポート	93
例 29	SPARC: 非クラスタモードでグローバルクラスタノードをブートする	95
例 30	レプリカペアの作成	113

例 31	データレプリケーション設定の確認	114
例 32	使用されているディスクに対応する DID の表示	115
例 33	DID インスタンスの結合	117
例 34	結合された DID の表示	117
例 35	プライマリサイトフェイルオーバー後の EMC SRDF データの手動回復	119
例 36	グローバルデバイス名前空間を更新する	122
例 37	Solaris Volume Manager デバイスグループの追加	129
例 38	デバイスグループからのノードの削除 (Solaris Volume Manager)	135
例 39	raw デバイスグループからノードを削除する	137
例 40	デバイスグループのプロパティの変更	139
例 41	セカンダリの希望数の変更 (Solaris Volume Manager)	141
例 42	セカンダリノードの希望数のデフォルト値への設定	142
例 43	すべてのデバイスグループのステータスの一覧表示	143
例 44	特定のデバイスグループの構成の一覧表示	143
例 45	デバイスグループのプライマリノードの切り替え	144
例 46	デバイスグループを保守状態にする	146
例 47	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	147
例 48	単一デバイスの SCSI プロトコルの表示	148
例 49	すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定の設定	150
例 50	単一デバイスのフェンシングプロトコルの設定	151
例 51	クラスタファイルシステムの削除	156
例 52	単一ノードのディスクパスをモニタリング	158
例 53	すべてのノードのディスクパスをモニタリング	159
例 54	CCR からディスク構成を読み直す	159
例 55	ディスクパスのモニタリング解除	160
例 56	障害のあるディスクパスを出力する	160
例 57	ファイルからのディスクパスのモニタリング	162
例 58	最後の定足数デバイスの削除	176
例 59	定足数デバイスを保守状態にする	179
例 60	定足数投票数 (定足数デバイス) のリセット	181
例 61	定足数構成の一覧表示	182
例 62	すべての構成済み定足数サーバーの起動	185
例 63	特定の定足数サーバーの起動	185
例 64	すべての構成済み定足数サーバーの停止	185
例 65	特定の定足数サーバーの停止	185
例 66	1 つの定足数サーバーの構成の表示	186

例 67	複数の定足数サーバーの構成の表示	187
例 68	動作しているすべての定足数サーバーの構成の表示	187
例 69	定足数サーバー構成からの期限切れのクラスタ情報のクリーンアップ	188
例 70	クラスタインターコネクトのステータスを確認する	192
例 71	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の追加の確認	194
例 72	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の削除の確認	196
例 73	クラスタソフトウェア構成からのノードの削除	221
例 74	クラスタ名の変更	227
例 75	ノード名のノードID へのマップ	228
例 76	新しいマシンがグローバルクラスタに追加されないようにする	229
例 77	すべての新しいマシンがグローバルクラスに追加されることを許可する	229
例 78	グローバルクラスタに追加される新しいマシンを指定する	229
例 79	認証を標準 UNIX に設定する	229
例 80	認証を DES に設定する	229
例 81	プライベートホスト名を変更する	234
例 82	グローバルクラスタノードを保守状態にする	238
例 83	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	240
例 84	グローバルクラスタからのゾーンクラスタの削除	259
例 85	ゾーンクラスタ内の高可用性ローカルファイルシステムの削除	262
例 86	ゾーンクラスタ内の高可用性 ZFS ファイルシステムの削除	262
例 87	ゾーンクラスタからの Solaris Volume Manager ディスクセットの削除	263
例 88	DID デバイスをゾーンクラスタから削除する	264
例 89	ZFS ルート (/) ファイルシステム (Solaris Volume Manager) の復元	288

このドキュメントの使用方法

『Oracle Solaris Cluster システム管理』では、SPARC および x86 ベースのシステムで Oracle Solaris Cluster の構成を管理する手順について説明します。

- 概要 – Oracle Solaris Cluster を構成する方法について説明します。
- 対象読者 – 技術者、システム管理者、および認定サービスプロバイダ
- 前提知識 – ハードウェアのトラブルシューティングや交換に関する豊富な経験

製品ドキュメントライブラリ

この製品および関連製品のドキュメントとリソースは <http://www.oracle.com/pls/topic/lookup?ctx=E62281> で入手可能です。

フィードバック

このドキュメントに関するフィードバックを <http://www.oracle.com/goto/docfeedback> からお聞かせください。

◆◆◆ 第 1 章

Oracle Solaris Cluster の管理の概要

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

- 24 ページの「Oracle Solaris Cluster の管理の概要」
- 24 ページの「ゾーンクラスタに関する作業」
- 25 ページの「Oracle Solaris OS の機能制限」
- 26 ページの「管理ツール」
- 29 ページの「クラスタ管理の準備」
- 30 ページの「クラスタの管理」

このガイドのすべての手順は、Oracle Solaris 11 オペレーティングシステムで使用するためのものです。

グローバルクラスタは、1 つ以上のグローバルクラスタノードで構成されます。グローバルクラスタには、ノードではなく、HA for Zones データサービスで構成された solaris または solaris10 ブランドの非大域ゾーンを含めることもできます。

ゾーンクラスタは、cluster 属性で設定された solaris、solaris10、または labeled ブランドの 1 つ以上の非帯域ゾーンで構成されます。ゾーンクラスタでは、その他のブランドタイプは許可されていません。labeled ブランドゾーンクラスタは、Oracle Solaris ソフトウェアの Trusted Extensions 機能でのみ使用します。ゾーンクラスタは、clzonecluster コマンド、clsetup ユーティリティ、または Oracle Solaris Cluster Manager ブラウザインタフェースを使用して作成します。

Oracle Solaris ゾーンで提供される分離を含めて、グローバルクラスタと同様にゾーンクラスタでサポートされるサービスを実行できます。ゾーンクラスタは、グローバルクラスタに依存しており、したがって、グローバルクラスタを必要とします。グローバルクラスタはゾーンクラスタを含みません。ゾーンクラスタは 1 つのマシン上に最大で 1 つのゾーンクラスタノードを持ちます。ゾーンクラスタノードは、同じマシン上のグローバルクラスタノードが動作している場合にのみ動作します。あるマシンのグローバルクラスタノードで障害が発生すると、そのマシン上のすべてのゾーンクラスタノードも動作しなくなります。ゾーンクラスタに関する一般的な情報については、『Oracle Solaris Cluster 4.3 Concepts Guide』を参照してください。

Oracle Solaris Cluster の管理の概要

Oracle Solaris Cluster の高可用性環境によって、エンドユーザーはクリティカルなアプリケーションを確実に使用できます。システム管理者の業務は、Oracle Solaris Cluster 構成の安定した動作を保証することです。

管理タスクを始める前に、次のマニュアルの計画情報をよく理解しておきます。

- 『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の第 1 章、「Oracle Solaris Cluster 構成の計画」
- 『Oracle Solaris Cluster 4.3 Concepts Guide』

Oracle Solaris Cluster の管理は、次のマニュアルの中のタスクに構成されます。

注記 - これらのタスクの一部は Oracle Solaris Cluster Manager ブラウザインタフェースを使用して実行できます。これについては個々のタスクの手順に記載しています。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

- グローバルクラスタまたはゾーンクラスタを定期的に (または毎日) 管理および維持するための標準的なタスク。これらのタスクは、このガイド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タスク。これらのタスクは、『Oracle Solaris Cluster 4.3 データサービス計画および管理ガイド』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タスク。これらのタスクは、『Oracle Solaris Cluster Hardware Administration Manual』で説明されています。

一般的に、Oracle Solaris Cluster の管理タスクはクラスタの動作中に実行できます。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。明記されていないかぎり、Oracle Solaris Cluster の管理タスクはグローバルクラスタノードで実行するようにしてください。クラスタ全体を停止する必要がある手順については、ダウンタイムのスケジュールを通常の業務時間外に設定してシステムへの影響を最小限に抑えてください。クラスタまたはクラスタノードを停止する予定があるときは、あらかじめユーザーに通知しておいてください。

ゾーンクラスタに関する作業

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

clzonecluster コマンドをグローバルクラスタノードで使用すると、グローバルクラスタ内のすべてのゾーンクラスタが対象になります。ゾーンクラスタコマンドはまた、コマンド発行時にゾーンクラスタノードが停止していても、ゾーンクラスタ上のすべてのノードを対象とします。

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

次の一覧に、ゾーンクラスタで実行する主な管理タスクを示します。

- **ゾーンクラスタの起動とリポート** - 第3章「クラスタの停止とブート」を参照してください。Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタをブートおよびリポートすることもできます。Oracle Solaris Cluster Manager のログイン手順については、290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」を参照してください。
- **ゾーンクラスタへのノードの追加** - 第8章「クラスタノードの管理」を参照してください。
- **ゾーンクラスタからのノードの削除** - 217 ページの「ゾーンクラスタからノードを削除する方法」を参照してください。また、Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタノードからソフトウェアをアンインストールすることもできます。Oracle Solaris Cluster Manager のログイン手順については、290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」を参照してください。
- **ゾーンクラスタの構成の表示** - 41 ページの「クラスタ構成を表示する方法」を参照してください。Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタの構成を表示することもできます。Oracle Solaris Cluster Manager のログイン手順については、290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」を参照してください。
- **ゾーンクラスタの構成の検証** - 50 ページの「基本的なクラスタ構成を検証する方法」を参照してください。
- **ゾーンクラスタの停止** - 第3章「クラスタの停止とブート」を参照してください。Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタを停止することもできます。Oracle Solaris Cluster Manager のログイン手順については、290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」を参照してください。

Oracle Solaris OS の機能制限

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

表 1 Oracle Solaris Cluster サービス

Oracle Solaris Cluster サービス	FMRI
pnm	svc:/system/cluster/pnm:default

Oracle Solaris Cluster サービス	FMRI
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

コマンド行または Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、Oracle Solaris Cluster 構成の管理タスクを実行できます。次のセクションでは、Oracle Solaris Cluster Manager とコマンド行ツールの概要を示します。

Oracle Solaris Cluster Manager ブラウザインタフェース

Oracle Solaris Cluster ソフトウェアは、クラスタ上でさまざまな管理タスクを実行するために使用できるブラウザインタフェース Oracle Solaris Cluster Manager をサポートしています。詳細は、[第13章「Oracle Solaris Cluster Manager ブラウザインタフェースの使用」](#)を参照してください。また、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)で Oracle Solaris Cluster Manager ログイン手順を確認することもできます。

Oracle Solaris Cluster Manager で実行できるタスクのいくつかを次に示します。

- ゾーンクラスタを作成および更新する
- リソースとリソースグループを作成する
- ゾーンクラスタにファイルシステム、論理ホスト、または共有ストレージを追加する
- Oracle Database データサービスを作成する
- グローバルクラスタまたはゾーンクラスタ内のノードを管理する
- 定足数デバイスとサーバーを追加および管理する
- NAS ストレージデバイスを追加および管理したり、ディスクとデバイスグループを管理したりする

- Geographic Edition パートナーシップを管理する

コマンド行インタフェース

Oracle Solaris Cluster のほとんどの管理タスクは、`clsetup` 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」メニュー内の次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
- ゾーンクラスタ
- そのほかのクラスタタスク

Oracle Solaris Cluster 構成を管理するため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。

`if_mpadm(1M)`

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。

`claccess(1CL)`

ノードを追加するために Oracle Solaris Cluster アクセスポリシーを管理します。

`cldevice(1CL)`

Oracle Solaris Cluster デバイスを管理します。

`cldevicegroup(1CL)`

Oracle Solaris Cluster デバイスグループを管理します。

`clinterconnect(1CL)`

Oracle Solaris Cluster インターコネクトを管理します。

`clnasdevice(1CL)`

Oracle Solaris Cluster 構成の NAS デバイスへのアクセスを管理します。

`clnode(1CL)`

Oracle Solaris Cluster ノードを管理します。

`clquorum(1CL)`

Oracle Solaris Cluster 定足数を管理します。

`clreslogicalhostname(1CL)`

論理ホスト名の Oracle Solaris Cluster リソースを管理します。

`clresource(1CL)`

Oracle Solaris Cluster データサービスのリソースを管理します。

`clresourcegroup(1CL)`

Oracle Solaris Cluster データサービスのリソースを管理します。

`clresourcetype(1CL)`

Oracle Solaris Cluster データサービスのリソースを管理します。

`clressharedaddress(1CL)`

共有アドレスのために Oracle Solaris Cluster リソースを管理します。

`clsetup(1CL)`

ゾーンクラスタを作成し、Oracle Solaris Cluster 構成を対話形式で構成します。

`clsnmphost(1CL)`

Oracle Solaris Cluster SNMP ホストを管理します。

`clsnmpmib(1CL)`

Oracle Solaris Cluster SNMP MIB を管理します。

`clsnmpuser(1CL)`

Oracle Solaris Cluster SNMP ユーザーを管理します。

`cltelemetryattribute(1CL)`

システムリソースモニタリングを構成します。

`cluster(1CL)`

Oracle Solaris Cluster 構成のグローバル構成とグローバルステータスを管理します。

`clzonecluster(1CL)`

ゾーンクラスタの作成と変更を行います。

さらに、コマンドを使用して Oracle Solaris Cluster 構成のボリュームマネージャーソフトウェアを管理することもできます。これらのコマンドは、クラスタで使用されている特定のボリュームマネージャーに依存します。

クラスタ管理の準備

ここでは、クラスタの管理を準備する方法について説明します。

Oracle Solaris Cluster ハードウェア構成の記録

Oracle Solaris Cluster ハードウェア構成は時とともに変化していくので、サイトに固有のハードウェアの特徴を記録しておきます。クラスタを変更または更新するときには、このハードウェアの記録を参照することで管理作業を少なくすることができます。また、さまざまなクラスタコンポーネント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ことで、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

専用のワークステーションまたは管理ネットワーク経由で接続されているワークステーションを管理コンソールとして使用して、アクティブなクラスタを管理できます。

`pconsole` ユーティリティを使用して、クラスタノードごとの端末ウィンドウに加えて、入力したコマンドを同時にすべてのノードに対して発行するマスターウィンドウを実行できます。管理コンソールへの `pconsole` ソフトウェアのインストールについては、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[pconsole ソフトウェアを管理コンソールにインストールする方法](#)」を参照してください。

Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、クラスタとクラスタコンポーネントを構成、モニター、および管理できます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)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リモートアクセスするために使用します。

Oracle Solaris Cluster には、専用の管理コンソールは必要ありませんが、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。Oracle Solaris Cluster ソフトウェアは高可用性環境を備えており、データのミラー化されたコピーがストレージデバイスに存在しますが、Oracle Solaris Cluster ソフトウェアは定期的なバックアップの代わりにはなりません。Oracle Solaris Cluster 構成は、複数の障害に耐えることができますが、ユーザーやプログラムのエラー、または致命的な障害からは保護されません。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

クラスタの管理

表2「Oracle Solaris Cluster 管理ツール」に、クラスタ管理の出発点を示します。

表 2 Oracle Solaris Cluster 管理ツール

タスク	ツール	手順
クラスタへのリモートログイン	コマンド行から Oracle Solaris <code>pconsole</code> ユーティリティを使用し、リモートからクラスタにログインします。	31 ページの「リモートからクラスタにログインする」 32 ページの「クラスタコンソールに安全に接続する方法」
対話形式でのクラスタの構成	<code>clzonecluster</code> コマンドまたは <code>clsetup</code> ユーティリティを使用します。	32 ページの「クラスタ構成ユーティリティにアクセスする方法」
Oracle Solaris Cluster のリリース番号とバージョン情報の表示	<code>clnode</code> コマンドに <code>show-<i>rev</i> -v -node</code> サブコマンドとオプションを付けて使用します。	33 ページの「Oracle Solaris Cluster のリリース情報とバージョン情報を表示する方法」
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</code> ■ <code>clresourcegroup</code> ■ <code>clresourcetype</code> 	35 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方法」
クラスタコンポーネントのグラフィカルなモニター	Oracle Solaris 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。
一部のクラスタコンポーネントのグラフィカルな管理	Oracle Solaris 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。
クラスタコンポーネントのステータスを確認します。	<code>cluster</code> コマンドに <code>status</code> サブコマンドを付けて使用します。	38 ページの「クラスタコンポーネントのステータスを確認する方法」

タスク	ツール	手順
パブリックネットワーク上の IPMP のステータス確認	グローバルクラスタの場合は、 <code>clnode status</code> コマンドに <code>-m</code> オプションを付けて使用します。	40 ページの「パブリックネットワークのステータスを確認する方法」
クラスタ構成を表示します。	ゾーンクラスタの場合は、 <code>clzonecluster</code> コマンドに <code>show</code> サブコマンドを付けて使用します。 グローバルクラスタの場合は、 <code>cluster</code> コマンドに <code>show</code> サブコマンドを付けて使用します。	41 ページの「クラスタ構成を表示する方法」
構成済み NAS デバイスの表示	グローバルクラスタまたはゾーンクラスタの場合は、 <code>clzonecluster</code> コマンドに <code>show</code> サブコマンドを付けて使用します。	<code>clnasdevice(1CL)</code>
グローバルマウントポイントの確認またはクラスタ構成の検証	グローバルクラスタの場合は、 <code>cluster</code> コマンドに <code>check</code> サブコマンドを付けて使用します。 ゾーンクラスタの場合は、 <code>clzonecluster verify</code> コマンドを使用します。	50 ページの「基本的なクラスタ構成を検証する方法」
Oracle Solaris 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。	57 ページの「Oracle Solaris Cluster のコマンドログの内容を表示する方法」
Oracle Solaris 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。	『Oracle Solaris 11.3 でのシステム管理のトラブルシューティング』の「システムメッセージの形式」 Oracle Solaris Cluster Manager ブラウザインタフェースでノードのシステムメッセージを表示することもできます。Oracle Solaris Cluster Manager のログイン手順については、290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」を参照してください。
Solaris Volume Manager のステータスのモニター	<code>metastat</code> コマンドを使用します。	『Solaris Volume Manager 管理ガイド』

リモートからクラスタにログインする

コマンド行からパラレルコンソールアクセス (`pconsole`) ユーティリティを使用して、リモートでクラスタにログインできます。`pconsole` ユーティリティは、Oracle Solaris `terminal/pconsole` パッケージの一部です。パッケージをインストールするには、`pkg install terminal/`

`pconsole` を実行します。`pconsole` ユーティリティは、コマンド行で指定した各リモートホストに対して 1 つのホスト端末ウィンドウを作成します。また、このユーティリティは、入力された内容を開いた各接続に伝播する、中央 (またはマスター) コンソールウィンドウを開きます。

`pconsole` ユーティリティは、X ウィンドウまたはコンソールモード内から実行できません。`pconsole` は、クラスタの管理コンソールとして使用するマシンにインストールします。サーバーの IP アドレスの特定のポート番号に接続できる端末サーバーがある場合は、ホスト名または IP アドレスに加えてポート番号を `terminal-server:portnumber` として指定できます。

詳細は、`pconsole(1)` のマニュアルページを参照してください。

クラスタコンソールに安全に接続する方法

端末集配信装置またはシステムコントローラが `ssh` をサポートする場合は、`pconsole` ユーティリティを使用してそれらのシステムのコンソールに接続できます。`pconsole` ユーティリティは、Oracle Solaris `terminal/pconsole` パッケージの一部であり、このパッケージのインストール時にインストールされます。`pconsole` ユーティリティは、コマンド行で指定した各リモートホストに対して 1 つのホスト端末ウィンドウを作成します。また、このユーティリティは、入力された内容を開いた各接続に伝播する、中央 (またはマスター) コンソールウィンドウを開きます。詳細は、`pconsole(1)` のマニュアルページを参照してください。

▼ クラスタ構成ユーティリティにアクセスする方法

`clsetup` ユーティリティでは、ゾーンクラスタを対話形式で作成し、グローバルクラスタの定足数、リソースグループ、クラスタトランスポート、プライベートホスト名、デバイスグループ、および新しいノードオプションを構成できます。`clzonecluster` ユーティリティは、同様な構成タスクをゾーンクラスタに対して実行します。詳細は、`clsetup(1CL)` と `clzonecluster(1CL)` のマニュアルページを参照してください。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用してこの手順を実行することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. **グローバルクラスタのアクティブメンバーノードで root 役割になります。**
グローバルクラスタのノードから、次の手順のステップをすべて実行します。

2. 構成ユーティリティを起動します。

```
phys-schost# clsetup
```

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

```
phys-schost# clsetup
```

q が表示されます。

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

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

ユーティリティで実行可能な操作は、次のオプションで確認できます。

```
clzc:sczone> ?
```

対話式の `clsetup` ユーティリティまたは Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタを作成したり、ファイルシステムまたはストレージデバイスをクラスタスコープに追加したりすることもできます。その他のすべてのゾーンクラスタ構成タスクは、`clzonecluster configure` コマンドで実行されます。`clsetup` ユーティリティまたは Oracle Solaris Cluster Manager を使用して、ゾーンクラスタを構成する手順については、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』を参照してください。

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、タスクを完了します。詳細は、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の「ゾーンクラスタの作成および構成」の手順を参照してください。

参照 詳細は、`clsetup` または `clzonecluster` のオンラインヘルプマニュアルページを参照してください。

▼ Oracle Solaris Cluster のリリース情報とバージョン情報を表示する方法

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

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

- Oracle Solaris Cluster のリリース情報とバージョン情報を表示します。

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

このコマンドは、すべての Oracle Solaris Cluster パッケージについて Oracle Solaris 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 Oracle Solaris Cluster のリリース情報およびバージョン情報の表示

次に、Oracle Solaris Cluster 4.2 に付属しているパッケージのクラスタのリリース情報とバージョン情報の例を示します。

```
phys-schost# clnode show-rev
4.2
```

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

```
Oracle Solaris Cluster 4.2 for Oracle Solaris 11 sparc
ha-cluster/data-service/apache           :4.2-0.30
ha-cluster/data-service/dhcp             :4.2-0.30
ha-cluster/data-service/dns              :4.2-0.30
ha-cluster/data-service/goldengate       :4.2-0.30
ha-cluster/data-service/glassfish-message-queue :4.2-0.30
ha-cluster/data-service/ha-ldom          :4.2-0.30
ha-cluster/data-service/ha-zones         :4.2-0.30
ha-cluster/data-service/iplanet-web-server :4.2-0.30
ha-cluster/data-service/jd-edwards-enterpriseone :4.2-0.30
ha-cluster/data-service/mysql            :4.2-0.30
ha-cluster/data-service/nfs               :4.2-0.30
ha-cluster/data-service/obiee            :4.2-0.30
ha-cluster/data-service/oracle-database  :4.2-0.30
ha-cluster/data-service/oracle-ebs       :4.2-0.30
ha-cluster/data-service/oracle-external-proxy :4.2-0.30
ha-cluster/data-service/oracle-http-server :4.2-0.30
ha-cluster/data-service/oracle-pmn-server :4.2-0.30
ha-cluster/data-service/oracle-traffic-director :4.2-0.30
ha-cluster/data-service/peoplesoft       :4.2-0.30
ha-cluster/data-service/postgresql       :4.2-0.30
ha-cluster/data-service/samba             :4.2-0.30
ha-cluster/data-service/sap-livecache    :4.2-0.30
ha-cluster/data-service/sapdb            :4.2-0.30
ha-cluster/data-service/sapnetweaver     :4.2-0.30
ha-cluster/data-service/siebel           :4.2-0.30
ha-cluster/data-service/sybase           :4.2-0.30
ha-cluster/data-service/timesten         :4.2-0.30
ha-cluster/data-service/tomcat           :4.2-0.30
ha-cluster/data-service/weblogic         :4.2-0.30
ha-cluster/developer/agent-builder       :4.2-0.30
ha-cluster/developer/api                 :4.2-0.30
ha-cluster/geo/geo-framework             :4.2-0.30
ha-cluster/geo/manual                    :4.2-0.30
ha-cluster/geo/replication/availability-suite :4.2-0.30
ha-cluster/geo/replication/data-guard    :4.2-0.30
```

```
ha-cluster/geo/replication/sbp :4.2-0.30
ha-cluster/geo/replication/srdf :4.2-0.30
ha-cluster/geo/replication/zfs-sa :4.2-0.30
ha-cluster/group-package/ha-cluster-data-services-full :4.2-0.30
ha-cluster/group-package/ha-cluster-framework-full :4.2-0.30
ha-cluster/group-package/ha-cluster-framework-l10n :4.2-0.30
ha-cluster/group-package/ha-cluster-framework-minimal :4.2-0.30
ha-cluster/group-package/ha-cluster-framework-scm :4.2-0.30
ha-cluster/group-package/ha-cluster-framework-slm :4.2-0.30
ha-cluster/group-package/ha-cluster-full :4.2-0.30
ha-cluster/group-package/ha-cluster-geo-full :4.2-0.30
ha-cluster/group-package/ha-cluster-geo-incorporation :4.2-0.30
ha-cluster/group-package/ha-cluster-incorporation :4.2-0.30
ha-cluster/group-package/ha-cluster-minimal :4.2-0.30
ha-cluster/group-package/ha-cluster-quorum-server-full :4.2-0.30
ha-cluster/group-package/ha-cluster-quorum-server-l10n :4.2-0.30
ha-cluster/ha-service/derby :4.2-0.30
ha-cluster/ha-service/gds :4.2-0.30
ha-cluster/ha-service/gds2 :4.2-0.30
ha-cluster/ha-service/logical-hostname :4.2-0.30
ha-cluster/ha-service/smf-proxy :4.2-0.30
ha-cluster/ha-service/telemetry :4.2-0.30
ha-cluster/library/cacao :4.2-0.30
ha-cluster/library/ucmm :4.2-0.30
ha-cluster/locale :4.2-0.30
ha-cluster/release/name :4.2-0.30
ha-cluster/service/management :4.2-0.30
ha-cluster/service/management/slm :4.2-0.30
ha-cluster/service/quorum-server :4.2-0.30
ha-cluster/service/quorum-server/locale :4.2-0.30
ha-cluster/service/quorum-server/manual/locale :4.2-0.30
ha-cluster/storage/svm-mediator :4.2-0.30
ha-cluster/system/cfgchk :4.2-0.30
ha-cluster/system/core :4.2-0.30
ha-cluster/system/dsconfig-wizard :4.2-0.30
ha-cluster/system/install :4.2-0.30
ha-cluster/system/manual :4.2-0.30
ha-cluster/system/manual/data-services :4.2-0.30
ha-cluster/system/manual/locale :4.2-0.30
ha-cluster/system/manual/manager :4.2-0.30
ha-cluster/system/manual/manager-glassfish3:4.2-0.30
```

▼ 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方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用してリソースとリソースグループを表示することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

始める前に root 役割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。

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

グローバルクラスタのノードから、次の手順のステップをすべて実行します。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 2 構成されているリソースタイプ、リソースグループ、リソースの表示

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

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

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

```
Resource Type:                SUNW.sctelemetry
RT_description:                sctelemetry service for Oracle Solaris 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:                       <NULL>
RT_system:                     True
Global_zone:                   True
```

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

```
Resource Group:                tel-rg
RG_description:                 <NULL>
RG_mode:                        Failover
```

```
RG_state: Managed
Failback: False
Nodelist: phys-schost-2 phys-schost-1

--- Resources for Group tel-rg ---

Resource: tel-res
Type: SUNW.sctelemetry
Type_version: 4.0
Group: tel-rg
R_description:
Resource_project_name: default
Enabled{phys-schost-2}: True
Enabled{phys-schost-1}: True
Monitored{phys-schost-2}: True
Monitored{phys-schost-1}: True

Resource Type: SUNW.qfs
RT_description: SAM-QFS Agent on Oracle Solaris Cluster
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
Global_zone: True

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

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

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

▼ クラスタコンポーネントのステータスを確認する方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用してクラスタコンポーネントのステータスをチェックすることもできます。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

Oracle Solaris Cluster Manager と `cluster status` コマンドでもゾーンクラスタのステータスが表示されます。

始める前に root 役割以外のユーザーが `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ステータスを確認します。

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

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

例 3 クラスタコンポーネントのステータス確認

次に、`cluster status` コマンドによって返されるクラスタコンポーネントのステータス情報の例を示します。

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

--- Node Status ---

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

=== Cluster Transport Paths ===

Endpoint1                Endpoint2                Status
-----                -
phys-schost-1:nge1       phys-schost-4:nge1       Path online
phys-schost-1:e1000g1    phys-schost-4:e1000g1    Path online

=== Cluster Quorum ===
```

--- Quorum Votes Summary ---

Needed	Present	Possible
3	3	4

--- Quorum Votes by Node ---

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

--- Quorum Votes by Device ---

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

=== Cluster Device Groups ===

--- Device Group Status ---

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

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

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

=== Cluster Resource Groups ===

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

```

=== Cluster Resources ===

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

test_1             phys-schost-1     Offline           Offline
                  phys-schost-2     Stop failed       Faulted

test_1             phys-schost-1     Online            Online
                  phys-schost-2     Online            Online

Device Instance    Node               Status
-----
/dev/did/rdisk/d2  phys-schost-1     Ok

/dev/did/rdisk/d3  phys-schost-1     Ok
                  phys-schost-2     Ok

/dev/did/rdisk/d4  phys-schost-1     Ok
                  phys-schost-2     Ok

/dev/did/rdisk/d6  phys-schost-2     Ok

=== Zone Clusters ===

--- Zone Cluster Status ---

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

```

▼ パブリックネットワークのステータスを確認する方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

IP ネットワークマルチパスグループのステータスを確認するには、このコマンドとともに `clnode status` コマンドを使用します。

始める前に root 役割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。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ノードのステータスをチェックすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

- クラスタコンポーネントのステータスを確認します。

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

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

例 4 パブリックネットワークのステータスを調べる

次に、`clnode status` コマンドによって返されるクラスタコンポーネントのステータス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 nge2      Online
phys-schost-2      test-rg Onlinenge3 Online
```

▼ クラスタ構成を表示する方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、クラスタの構成を表示することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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

- グローバルクラスタまたはゾーンクラスタの構成を表示します。

```
% cluster show
```

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

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

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

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

例 5 グローバルクラスタの構成を表示する

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

```
phys-schost# cluster show

=== Cluster ===

Cluster Name:                cluster-1
clusterid:                   0x4DA2C888
installmode:                 disabled
heartbeat_timeout:           10000
heartbeat_quantum:           1000
private_netaddr:             172.11.0.0
private_netmask:             255.255.248.0
max_nodes:                   64
max_privatenets:             10
num_zoneclusters:            12
udp_session_timeout:         480
concentrate_load:            False
global_fencing:              prefer3
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
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:         net1, net3

```

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

```

Transport Adapter:              net1
Adapter State:                 Enabled
Adapter Transport Type:        dlpi
Adapter Property(device_name): net
Adapter Property(device_instance): 1
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:              net3
Adapter State:                 Enabled
Adapter Transport Type:        dlpi
Adapter Property(device_name): net
Adapter Property(device_instance): 3
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:    e1000g1, nge1

```

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

```

Transport Adapter:        e1000g1
Adapter State:            Enabled
Adapter Transport Type:   dlpi
Adapter Property(device_name): e1000g
Adapter Property(device_instance): 2
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:        nge1
Adapter State:            Enabled
Adapter Transport Type:   dlpi
Adapter Property(device_name): nge
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:e1000g1,switch2@1
Cable Endpoint1:                phys-schost-1:e1000g1
Cable Endpoint2:                switch2@1
Cable State:                    Enabled

Transport Cable:                phys-schost-1:nge1,switch1@1
Cable Endpoint1:                phys-schost-1:nge1
Cable Endpoint2:                switch1@1
Cable State:                    Enabled

Transport Cable:                phys-schost-2:nge1,switch1@2
Cable Endpoint1:                phys-schost-2:nge1
Cable Endpoint2:                switch1@2
Cable State:                    Enabled

Transport Cable:                phys-schost-2:e1000g1,switch2@2
Cable Endpoint1:                phys-schost-2:e1000g1
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:                            shared_disk
Access Mode:                    scsi3
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: 4
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: <NULL>
RT_system: True
Global_zone: 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: <NULL>
RT_system: True
Global_zone: 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:                 <NULL>
RT_system:               True
Global_zone:             True
Resource Type:           SUNW.haderby
RT_description:          haderby server for Oracle Solaris 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:                 <NULL>
RT_system:               True
Global_zone:             True
Resource Type:           SUNW.sctelemetry
RT_description:          sctelemetry service for Oracle Solaris 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:                 <NULL>
RT_system:               True
Global_zone:             True
=== 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.HASStoragePlus: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:                           default
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:                           default
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/rdisk/c0t2d0
Replication:                              none
default_fencing:                         global

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

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

```

```

default_fencing:                global

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

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

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

=== NAS Devices ===

Nas Device:                     nas_filer1
Type:                           sun_uss
nodeIPs{phys-schost-2}:         10.134.112.112
nodeIPs{phys-schost-1}:         10.134.112.113
User ID: root

```

例 6 ゾーンクラスタの構成を表示する

次の例では、RAC を使用したゾーンクラスタ構成のプロパティを一覧表示します。

```

% clzonecluster show
=== Zone Clusters ===

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

--- Solaris Resources for sczone ---

Resource Name:                  net
address:                       172.16.0.1
physical:                      auto

Resource Name:                  net
address:                       172.16.0.2
physical:                      auto

Resource Name:                  fs
dir:                           /local/ufs-1
special:                       /dev/md/ds1/dsk/d0

```

```

raw: /dev/md/ds1/rdisk/d0
type: ufs
options: [logging]

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

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

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

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

--- Zone Cluster Nodes for sczone ---

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

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

```

`clnasdevice show` サブコマンドまたは Oracle Solaris Cluster Manager を使用して、グローバルまたはゾーンクラスタ用に構成された NAS デバイスを表示することもできます。詳細は、[clnasdevice\(1CL\)](#) のマニュアルページを参照してください。

▼ 基本的なクラスタ構成を検証する方法

`cluster` コマンドは、`check` サブコマンドを使用して、グローバルクラスタが正しく機能するために必要な基本構成を検証します。チェックにエラーがない場合、`cluster check` はシェルブ

プロンプトに戻ります。チェックにエラーがある場合、`cluster check` が指定したディレクトリかデフォルトの出力ディレクトリにレポートを生成します。`cluster check` を複数のノードに対して実行すると、`cluster check` は、ノードごとのレポートと複数ノードチェックのレポートを生成します。`cluster list-checks` コマンドを使用して、使用可能なすべてのクラスタチェックの一覧を表示させることもできます。

コマンドを使用することにより、ユーザーの対話型操作なしに実行される基本検査だけでなく、対話型検査、および機能検査も実行することができます。基本検査は、`-keyword` オプションが指定されていない場合に実行されます。

- 対話型検査を実行する場合、ユーザーは検査で判定できない情報を入力しなければなりません。検査の実行時には、ファームウェアバージョン番号などの必要な情報を入力するように促されます。1 つ以上の対話型検査を指定するには、`-k interactive` キーワードを使用します。
- 機能検査では、クラスタの特定の機能または動作を検査します。検査の実行時には、フェイルオーバー先となるノードや検査の開始または続行の確認などの情報を入力するように促されます。機能検査を指定するには、`-k functional check-id` キーワードを使用します。機能検査は 1 回につき 1 つだけ実行できます。

注記 - 一部の機能検査ではクラスタサービスの中断が必要になるので、検査の詳細説明を読み、最初にクラスタの稼働を停止する必要があるかどうか判断したうえで、機能検査を開始してください。この情報を表示するには、次のコマンドを使用します。

```
% cluster list-checks -v -C checkID
```

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

注記 - `cluster check` は、デバイス、ボリューム管理コンポーネント、または Oracle Solaris Cluster 構成を変更するような管理手順を行なったあとに実行してください。

`clzonecluster(1CL)` コマンドをグローバルクラスタノードから実行すると、ゾーンクラスタが正しく機能するために必要な構成を検証する一連のチェックが実行されます。すべてのチェックでエラーがなかった場合、`clzonecluster verify` はシェルプロンプトに戻ります (その場合は、ゾーンクラスタを安全にインストールできます)。エラーがあった場合は、エラーがあったグローバルクラスタノードに関して `clzonecluster verify` から報告があります。`clzonecluster verify` を複数のノードに対して実行すると、ノードごとのレポートと、複数ノードチェックのレポートが生成されます。ゾーンクラスタ内では、`verify` サブコマンドは指定できません。

1. グローバルクラスタのアクティブメンバーノードで root 役割になります。

```
phys-schost# su
```

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

2. 最新のチェックがあることを確認します。

a. [My Oracle Support](#) の「パッチと更新」タブを参照します。

b. 詳細検索で、製品として「Solaris Cluster」を選択し、「説明」フィールドで「check」と入力します。

この検索によって、チェックを含む Oracle Solaris Cluster ソフトウェアアップデートが見つかります。

c. まだクラスタにインストールされていないソフトウェアアップデートをすべて適用します。

3. 基本の妥当性検査を実行します。

```
phys-schost# cluster check -v -o outputdir
```

-v 冗長モード。

-o outputdir *outputdir* サブディレクトリに出力をリダイレクトします。

このコマンドによって、すべての使用可能な基本検査が実行されます。クラスタ機能には影響はありません。

4. インタラクティブな妥当性検査を実行します。

```
phys-schost# cluster check -v -k interactive -o outputdir
```

-k interactive 実行可能なインタラクティブ妥当性検査を指定します。

このコマンドで、すべての使用可能なインタラクティブ検査が実行され、クラスタについて必要な情報の入力が求められます。クラスタ機能には影響はありません。

5. 機能の妥当性検査を実行します。

a. 非冗長モードですべての使用可能な機能検査一覧が表示されます。

```
phys-schost# cluster list-checks -k functional
```

b. どの機能検査が、本稼働環境でクラスタの可用性またはサービスを中断する可能性がある処理を実行するかを判断してください。

たとえば、機能検査によって、ノードパニックまたは他のノードへのフェイルオーバーがトリガーされる可能性があります。

```
phys-schost# cluster list-checks -v -c check-ID
```

-c *check-ID* 特定の検査を指定します。

- c. クラスタの機能を中断するような機能検査を実行する場合、クラスタが本稼働状態から除外されるようにします。

- d. 機能検査を開始します。

```
phys-schost# cluster check -v -k functional -C check-ID -o outputdir
```

-k functional 実行可能な機能妥当性検査を指定します。

検査の実行に必要な情報を確認し、実行に必要な情報または操作を求めるプロンプトに入力を行います。

- e. 実行する残りの機能検査ごとに、手順 c と手順 d を繰り返します。

注記 - 記録を保存するために、実行する検査ごとに固有の *outputdir* サブディレクトリ名を指定します。*outputdir* 名を再利用する場合、新しい検査の出力によって、再利用した *outputdir* サブディレクトリの既存の内容が上書きされます。

6. ゾーンクラスタが構成されている場合は、ゾーンクラスタの構成を検証して、ゾーンクラスタがインストール可能かどうかを確認します。

```
phys-schost# clzonecluster verify zone-cluster-name
```

7. 今後の診断に活用できるように、クラスタ構成を記録しておきます。

『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の「クラスタ構成の診断データを記録する方法」を参照してください。

例 7 グローバルクラスタ構成の基本検証 (エラーがない場合)

次の例は、`cluster check` が冗長モードで `phys-schost-1` および `phys-schost-2` ノードに対して実行され、すべての検査に合格した場合を示しています。

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

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

例 8 インタラクティブな妥当性検査のリスト

クラスタで実行するために使用できるすべてインタラクティブな妥当性検査の例を以下に示します。出力例に、使用できる検査の例を示します。実際に使用できる検査は、構成によって異なります。

```
# cluster list-checks -k interactive
Some checks might take a few moments to run (use -v to see progress)...
I6994574:(Moderate)Fix for GLDV3 interfaces on cluster transport vulnerability applied?
```

例 9 機能の妥当性検査の実行

まず、次の例は機能検査の詳細なリストを示します。検査 F6968101 の詳細な説明が表示されます。この説明で、検査によってクラスタサービスが中断されることがわかります。クラスタは稼働状態ではなくなります。機能検査が実行され、`funct.test.F6968101.12Jan2011` サブディレクトリに詳細な出力が記録されます。出力例に、使用できる検査の例を示します。実際に使用できる検査は、構成によって異なります。

```
# cluster list-checks -k functional
F6968101: (Critical) Perform resource group switchover
F6984120: (Critical) Induce cluster transport network failure - single adapter.
F6984121: (Critical) Perform cluster shutdown
F6984140: (Critical) Induce node panic
# cluster list-checks -v -C F6968101
F6968101: (Critical) Perform resource group switchover
Keywords: SolarisCluster3.x, functional
Applicability: Applicable if multi-node cluster running live.
Check Logic: Select a resource group and destination node. Perform
'clresourcegroup switch' on specified resource group
either to specified node or to all nodes in succession.
Version: 1.2
Revision Date: 12/10/10
```

最初にクラスタの稼働を停止します

```
# cluster list-checks -k functional -C F6968101 -o funct.test.F6968101.12Jan2011
F6968101
initializing...
initializing xml output...
loading auxiliary data...
starting check run...
pschost1, pschost2, pschost3, pschost4: F6968101.... starting:
Perform resource group switchover
```

=====

>>> Functional Check

'Functional' checks exercise cluster behavior. It is recommended that you do not run this check on a cluster in production mode.' It is recommended that you have access to the system console for each cluster node and

observe any output on the consoles while the check is executed.

If the node running this check is brought down during execution the check must be rerun from this same node after it is rebooted into the cluster in order for the check to be completed.

Select 'continue' for more details on this check.

```
1) continue
2) exit
```

```
choice: 1
```

```
=====
```

```
>>> Check Description <<<
```

画面の指示に従います

例 10 グローバルクラスタ構成の検証 (エラーがある場合)

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

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

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

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

```
-----
CHECK ID : 3065
SEVERITY : HIGH
FAILURE: Global filesystem /etc/vfstab entries are not consistent across
all Oracle SolarisCluster 4.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方法

cluster コマンドには、クラスタファイルシステムとそのグローバルマウントポイントに構成エラーがないか、/etc/vfstab ファイルを調べるチェックが含まれています。詳細は、[cluster\(1CL\)](#) のマニュアルページを参照してください。

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

1. **グローバルクラスタのアクティブメンバーノードで root 役割になります。**
グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
% su
```

2. **グローバルクラスタ構成を検証します。**

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

例 11 グローバルマウントポイントの確認

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

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

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

```

#
# cat /var/cluster/logs/cluster_check/Dec5/cluster_check-results.suncluster.txt
...
=====
= ANALYSIS DETAILS =
=====
-----
CHECK ID : 3065
SEVERITY : HIGH
FAILURE: Global filesystem /etc/vfstab entries are not consistent across
all Oracle Solaris Cluster 4.x nodes.
ANALYSIS : The global filesystem /etc/vfstab entries are not consistent across
all nodes in this cluster.
Analysis indicates:
FileSystem '/global/phys-schost-1' is on 'phys-schost-1' but missing from 'phys-schost-2'.
RECOMMEND: Ensure each node has the correct /etc/vfstab entry for the
filesystem(s) in question.
...
#
# cat /var/cluster/logs/cluster_check/Dec5/cluster_check-results.phys-schost-1.txt
...
=====
= ANALYSIS DETAILS =
=====
-----
CHECK ID : 1398
SEVERITY : HIGH
FAILURE: An unsupported server is being used as an Oracle Solaris Cluster 4.x node.
ANALYSIS : This server may not be qualified to be used as an Oracle Solaris Cluster 4.x
node.
Only servers that have been qualified with Oracle Solaris Cluster 4.0 are supported as
Oracle Solaris Cluster 4.x nodes.
RECOMMEND: Because the list of supported servers is always being updated, check with
your Oracle representative to get the latest information on what servers
are currently supported and only use a server that is supported with Oracle Solaris Cluster 4.
x.
...
#

```

▼ Oracle Solaris Cluster のコマンドログの内容を表示する 方法

/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 Oracle Solaris 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設定したときに自動的に開始され、ユーザーがクラスタをシャットダウンしたときに終了します。コマンドは、実行中およびクラスタモードでブートされたすべてのノード上でロギングされます。

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

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

- claccess
- cldevice
- cldevicegroup
- clinterconnect
- clnasdevice
- clnode
- clquorum
- clreslogicalhostname
- clresource
- clresourcegroup
- clresourcetype
- clressharedaddress
- clsetup
- clsnmpghost
- clsnmpmib
- clsnmpuser
- cltelemetryattribute
- cluster
- clzonecluster

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)

注記 - 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

注記 - あるコマンドが未知の結果を伴って異常終了した場合、Oracle Solaris Cluster ソフトウェアは `commandlog` ファイル内に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。

Oracle Solaris Cluster ソフトウェアは任意の時点で、以前にアーカイブされた `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。

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

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

例 12 Oracle Solaris 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
```


◆◆◆ 第 2 章

Oracle Solaris 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 Oracle Solaris Cluster に関連する範囲で説明します。取り上げるトピック:

- [61 ページの「RBAC の設定と Oracle Solaris Cluster での使用」](#)
- [61 ページの「Oracle Solaris Cluster RBAC の権利プロファイル」](#)
- [63 ページの「Oracle Solaris Cluster 管理権利プロファイルによる RBAC 役割の作成と割り当て」](#)
- [64 ページの「ユーザーの RBAC プロパティの変更」](#)

RBAC の設定と Oracle Solaris 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ドキュメントを選んでください。RBAC を設定して、Oracle Solaris Cluster ソフトウェアで使用するための具体的な手順については、この章で後述します。

タスク	手順
RBAC の詳細を調べる	『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護』の第 1 章, 「権利を使用したユーザーとプロセスの制御について」
RBAC の設定、要素の管理、RBAC の使用など	『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護』の第 3 章, 「Oracle Solaris での権利の割り当て」
RBAC の要素とツールの詳細を調べる	『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護』の第 8 章, 「Oracle Solaris 権利リファレンス」

Oracle Solaris Cluster RBAC の権利プロファイル

コマンド行で発行する一部の Oracle Solaris 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 Oracle Solaris 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。Oracle Solaris Cluster RBAC の権利プロファイルは、グローバルクラスタ内のノードに適用されます。

`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作成、削除、および変更するための承認。

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

RBAC の権利プロファイルには 1 つ以上の RBAC の承認が含まれます。これらの権利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、Oracle Solaris 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、Oracle Solaris Cluster ソフトウェアに含まれる権利プロファイルを示します。

権利プロファイル	承認およびセキュリティ属性	付与される権利
Oracle Solaris Cluster Commands	<code>eid=0</code> セキュリティー属性を使用して実行される Oracle Solaris Cluster コマンドのリスト。	すべての Oracle Solaris Cluster コマンドの次のサブコマンドを含めて、クラスタを構成および管理するために使用する一部の Oracle Solaris Cluster コマンドの実行。 <ul style="list-style-type: none"> ■ list ■ show ■ status <code>scha_control</code> <code>scha_resource_get</code> <code>scha_resource_setstatus</code> <code>scha_resourcegroup_get</code> <code>scha_resourcetype_get</code>
基本 Oracle Solaris ユーザー	この既存の Oracle Solaris 権利プロファイルには、Oracle Solaris の承認のほか、次の承認が含まれません。	<code>solaris.cluster.read</code> – Oracle Solaris Cluster コマンドの一覧表示、表示、およびその他の読み取り操作を実行するほか、Oracle Solaris Cluster Manager ブラウザインタフェースにアクセスします。
Cluster Operation	この権利プロファイルは Oracle Solaris Cluster に固有で、次の承認が含まれています。	<code>solaris.cluster.read</code> – 一覧表示、表示、エクスポート、ステータス、およびその他の読み取り操作を実行するほか、Oracle Solaris Cluster Manager ブラウザインタフェースにアクセスします。 <code>solaris.cluster.admin</code> – クラスタオブジェクトの状態を変更します。

権利プロファイル	承認およびセキュリティ属性	付与される権利
System Administrator	この既存の Oracle Solaris 権利プロファイルには、Cluster 管理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	この権利プロファイルには、Cluster Operation プロファイルに含まれるものと同じ承認のほか、solaris.cluster.modify 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。

Oracle Solaris Cluster 管理権利プロファイルによる RBAC 役割の作成と割り当て

このタスクでは、Oracle Solaris Cluster 管理権利プロファイルを使用して新しい RBAC 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。

▼ コマンド行から役割を作成する方法

1. 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の場合は、`roleadd` コマンドを使用して新しいローカル役割およびその属性を指定します。詳細は、[roleadd\(1M\)](#) のマニュアルページを参照してください。
- または、ローカルスコープの役割の場合、`user_attr` ファイルを編集して `type=role` でユーザーを追加します。詳細は、[user_attr\(4\)](#) のマニュアルページを参照してください。
この方法は緊急時にのみ使用します。
- ネームサービスでの役割の場合は、`roleadd` および `rolemod` コマンドを使用して新しい役割およびその属性を指定します。詳細は、[roleadd\(1M\)](#) および [rolemod\(1M\)](#) のマニュアルページを参照してください。
このコマンドは、その他の役割を作成できる `root` 役割による認証を必要とします。`roleadd` コマンドは、すべてのネームサービスに適用できます。このコマンドは、Solaris Management Console サーバーのクライアントとして動作します。

2. 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 13 smrole コマンドを使用してカスタムの Operator 役割を作成する

次のシーケンスに、smrole コマンドを使用して役割を作成する方法を示します。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 パスワードを入力します>

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 :: <oper2 パスワードを入力します>

# /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 パスワードを入力します>

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 プロパティーの変更

ユーザーアカウントツールとコマンド行のいずれかを使用することで、ユーザーの RBAC プロパティーを変更できます。ユーザーの RBAC プロパティーを変更する場合は、65 ページの「[コマンド行からユーザーの RBAC プロパティーを変更する方法](#)」を参照してください。次のいずれかの手順を選択してください。

- 65 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」
- 65 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、root ユーザーとしてユーザーツールコレクションを実行するか、System Administrator 権利プロファイルが割り当てられている役割になる必要があります。

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には、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護』の「割り当てられている管理権利の使用」の説明に従って Solaris Management Console を起動します。ユーザーツールコレクションを開き、「ユーザーアカウント」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ペインに表示されます。

2. 変更する「ユーザーアカウント」アイコンをクリックし、「アクション」メニューから「プロパティ」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. 次のように、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クリックします。

- ユーザーに割り当てられている役割を変更するには、「役割」タブをクリックして、変更する役割の割り当てを「使用可能な役割」または「割り当てられた役割」のうち適切な列に移動します。
- ユーザーに割り当てられている権利プロファイルを変更するには、「権利」タブをクリックして、それを有効な権利または割り当てられた権利のどちらか、適切な列に移動します。

注記 - 権利プロファイルは、ユーザーに直接割り当てないようにします。特権のあるアプリケーションを実行する場合はユーザーがその役割になることが必要になる方法をお勧めします。この方法では、ユーザーによる特権の濫用が防止されます。

▼ 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. 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。

- ローカルスコープまたは LDAP リポジトリに定義されているユーザーに割り当てられているユーザープロパティを変更するには、`usermod` コマンドを使用します。詳細は、[usermod\(1M\)](#) のマニュアルページを参照してください。
- また同じくローカルスコープに定義されたユーザーに割り当てられている承認、役割、または権利プロファイルを変更する場合は、`user_attr` ファイルを編集することもできます。
この方法は緊急時にのみ使用します。
- ローカルで、またはネームサービス (LDAP リポジトリなど) で役割を管理する場合は、`roleadd` または `rolemod` コマンドを使用します。詳細は、[roleadd\(1M\)](#) または [rolemod\(1M\)](#) のマニュアルページを参照してください。

これらのコマンドは、ユーザーファイルを変更できる `root` 役割による認証を必要とします。これらのコマンドは、すべてのネームサービスに適用できます。『Oracle Solaris 11.3 のユーザーアカウントとユーザー環境の管理』の「ユーザー、役割、およびグループの管理に使用されるコマンド」を参照してください。

Oracle Solaris 11 に付属している Forced Privilege および Stop Rights プロファイルは変更できません。

◆◆◆ 第 3 章

クラスタの停止とブート

この章では、グローバルクラスタ、ゾーンクラスタ、および個々のノードの停止方法とブート方法について説明します。

- [67 ページの「クラスタの停止とブートの概要」](#)
- [82 ページの「クラスタ内の 1 つのノードの停止とブート」](#)
- [96 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については、[93 ページの「非クラスタモードでノードをブートする方法」](#)および表4「[タスマップ: ノードの停止とブート](#)」を参照してください。

クラスタの停止とブートの概要

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

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

注記 - グローバルクラスタ全体を正しく停止するには、`cluster shutdown` コマンドを使用します。Oracle Solaris の `shutdown` コマンドは `clnode evacuate` コマンドとともに使用して、個々のノードをシャットダウンします。詳細は、[69 ページの「クラスタを停止する方法」](#)、[82 ページの「クラスタ内の 1 つのノードの停止とブート」](#)、または `clnode(1CL)` のマニュアルページを参照してください。

Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ノードを退避することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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

1. 実行中のすべてのリソースグループをオフラインにする。
2. グローバルクラスタまたはゾーンクラスタのすべてのクラスタファイルシステムをマウント解除する。
3. `cluster shutdown` コマンドが、グローバルクラスタまたはゾーンクラスタ上のアクティブなデバイスサービスを停止する。
4. `cluster shutdown` コマンドは `init 0` を実行して、クラスタ上のすべてのノードを OpenBoot PROM `ok` プロンプトの状態にする (SPARC ベースのシステムの場合) か、または GRUB メニューの「Press any key to continue」メッセージの状態にします (x86 ベースのシステムの場合)。GRUB ベースのブートの詳細は、『[Oracle Solaris 11.3 システムのブートとシャットダウン](#)』の「[システムのブート](#)」を参照してください。`clzonecluster halt` コマンドは `zoneadm -z zone-cluster-name halt` コマンドを実行して、ゾーンクラスタのゾーンを停止します (ただし、シャットダウンしません)。

注記 - 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) ブートできます。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、[93 ページの「非クラスタモードでノードをブートする方法」](#)を参照してください。

表 3 タスクリスト：クラスタの停止とブート

タスク	手順
クラスタを停止します。	69 ページの「クラスタを停止する方法」
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と動作中の接続が必要です。	71 ページの「クラスタをブートする方法」
クラスタをリポートします。	75 ページの「クラスタをリポートする方法」

▼ クラスタを停止する方法

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



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

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. (x86 のみ) グローバルクラスタまたはゾーンクラスタで Oracle Real Application Clusters (RAC) が実行されている場合は、停止するクラスタ上のデータベースのインスタンスをすべて停止します。

停止の手順については、Oracle RAC 製品のドキュメントを参照してください。

2. クラスタ内の任意のノードで、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

3. グローバルクラスタ、1 つのゾーンクラスタ、またはすべてのゾーンクラスタを停止します。

- グローバルクラスタを停止します。この操作を行うと、すべてのゾーンクラスタも停止します。

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

- 特定のゾーンクラスタを停止します。

```
phys-schost# clzonecluster halt zone-cluster-name
```

- すべてのゾーンクラスタを停止します。

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

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

4. SPARC ベースのシステムの場合は、グローバルクラスタまたはゾーンクラスタ上のすべてのノードが `ok` プロンプトの状態になったことを確認します。x86 ベースのシステムの場合は、すべてのノードが GRUB メニューの状態になったことを確認します。

SPARC ベースのシステムの場合はすべてのノードが `ok` プロンプトになるまで、x86 ベースのシステムの場合はすべてのノードが `Boot Subsystem` の状態になるまで、どのノードの電源も切らないでください。

- クラスタ内でまだ稼働および実行中の別のグローバルクラスタノードから、1 つ以上のグローバルクラスタノードのステータスを確認します。

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

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

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

5. 必要であれば、グローバルクラスタのノードの電源を切ります。

例 14 ゾーンクラスタの停止

次の例では、`sczone` というゾーンクラスタをシャットダウンしています。

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

例 15 SPARC: グローバルクラスタの停止

次に、正常なグローバルクラスタの動作を停止して、すべてのノードを停止し、`ok` プロンプトが表示されたときのコンソールの出力例を示します。`-g 0` オプションは停止の猶予期間をゼロに設定し、`-y` オプションは確認質問に自動で `yes` 応答を提供します。停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

```
phys-schost# cluster shutdown -g0 -y
Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
/etc/rc0.d/K05initrgrm: Calling clnode evacuate
The system is down.
syncing file systems... done
Program terminated
ok
```

例 16 x86: グローバルクラスタの停止

次に、正常なグローバルクラスタの動作を停止して、すべてのノードを停止したときのコンソールの出力例を示します。この例では、すべてのノードで `ok` プロンプトが表示されるわけではありません。`-g 0` オプションは停止の猶予期間をゼロに設定し、`-y` オプションは確認質問に自動で `yes` 応答を提供します。停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

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

参照 停止したグローバルクラスタまたはゾーンクラスタを再起動するには、[71 ページの「クラスタをブートする方法」](#)を参照してください。

▼ クラスタをブートする方法

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

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - ゾーンクラスタを作成するには、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[ゾーンクラスタの作成および構成](#)」の手順に従うか、または Oracle Solaris Cluster Manager ブラウザインタフェースを使用してゾーンクラスタを作成します。

1. 各ノードをクラスタモードでブートします。

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Oracle Solaris エントリを選択し、Enter キーを押します。

GRUB ベースのブートの詳細は、『[Oracle Solaris 11.3 システムのブートとシャットダウン](#)』の「[システムのブート](#)」を参照してください。

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

- ゾーンクラスタが 1 つの場合は、ゾーンクラスタ全体をブートできます。

```
phys-schost# clzonecluster boot zone-cluster-name
```

- ゾーンクラスタが複数ある場合は、すべてのゾーンクラスタをブートできます。*zone-cluster-name* の代わりにプラス記号 (+) を使用します。

2. ノードが問題なくブートし、オンラインであることを確認します。

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

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

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

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

注記 - 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 Oracle Solaris 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、[96 ページの「満杯の /var ファイルシステムを修復する方法」](#)を参照してください。詳細は、`clzonecluster(1CL)` のマニュアルページを参照してください。

例 17 SPARC: グローバルクラスタのブート

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

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

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

例 18 x86: クラスタのブート

次に、ノード `phys-schost-1` をブート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。クラスタ内のほかのノードのコンソールにも同様のメッセージが表示されます。

```
ATI RAGE SDRAM BIOS P/N GR-xlint.007-4.330
* BIOS Lan-Console 2.0
Copyright (C) 1999-2001 Intel Corporation
MAC ADDR: 00 02 47 31 38 3C
AMIBIOS (C)1985-2002 American Megatrends Inc.,
Copyright 1996-2002 Intel Corporation
SCB20.86B.1064.P18.0208191106
SCB2 Production BIOS Version 2.08
BIOS Build 1064
2 X Intel(R) Pentium(R) III CPU family      1400MHz
Testing system memory, memory size=2048MB
2048MB Extended Memory Passed
512K L2 Cache SRAM Passed
ATAPI CD-ROM SAMSUNG CD-ROM SN-124

Press <F2> to enter SETUP, <F12> Network

Adaptec AIC-7899 SCSI BIOS v2.57S4
(c) 2000 Adaptec, Inc. All Rights Reserved.
Press <Ctrl><A> for SCSISelect(TM) Utility!

Ch B, SCSI ID: 0 SEAGATE ST336605LC          160
SCSI ID: 1 SEAGATE ST336605LC              160
SCSI ID: 6 ESG-SHV SCA HSBP M18            ASYN
```

Ch A, SCSI ID: 2 SUN StorEdge 3310 160
SCSI ID: 3 SUN StorEdge 3310 160

AMIBIOS (C)1985-2002 American Megatrends Inc.,
Copyright 1996-2002 Intel Corporation
SCB20.86B.1064.P18.0208191106
SCB2 Production BIOS Version 2.08
BIOS Build 1064

2 X Intel(R) Pentium(R) III CPU family 1400MHz
Testing system memory, memory size=2048MB
2048MB Extended Memory Passed
512K L2 Cache SRAM Passed
ATAPI CD-ROM SAMSUNG CD-ROM SN-124

SunOS - Intel Platform Edition Primary Boot Subsystem, vsn 2.0

Current Disk Partition Information

Part#	Status	Type	Start	Length
1	Active	X86 BOOT	2428	21852
2		SOLARIS	24280	71662420
3		<unused>		
4		<unused>		

Please select the partition you wish to boot: * *

Solaris DCB

loading /solaris/boot.bin

SunOS Secondary Boot version 3.00

Solaris Intel Platform Edition Booting System

Autobooting from bootpath: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/
pci8086,341a@7,1/sd@0,0:a

If the system hardware has changed, or to boot from a different
device, interrupt the autoboot process by pressing ESC.

Press ESCape to interrupt autoboot in 2 seconds.

Initializing system

Please wait...

Warning: Resource Conflict - both devices are added

NON-ACPI device: ISY0050

Port: 3F0-3F5, 3F7; IRQ: 6; DMA: 2

ACPI device: ISY0050

Port: 3F2-3F3, 3F4-3F5, 3F7; IRQ: 6; DMA: 2

<<< 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:
Size: 275683 + 22092 + 150244 Bytes
/platform/i86pc/kernel/unix loaded - 0xac000 bytes used
SunOS Release 5.9 Version Generic_112234-07 32-bit
Copyright 1983-2003 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.
configuring IPv4 interfaces: e1000g2.
Hostname: phys-schost-1
Booting as part of a cluster
NOTICE: CMM: Node phys-schost-1 (nodeid = 1) with votecount = 1 added.
NOTICE: CMM: Node phys-schost-2 (nodeid = 2) with votecount = 1 added.
NOTICE: CMM: Quorum device 1 (/dev/did/rdisk/dls2) added; votecount = 1, bitmask
of nodes with configured paths = 0x3.
NOTICE: clcomm: Adapter e1000g3 constructed
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 being constructed
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 being initiated
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 online
NOTICE: clcomm: Adapter e1000g0 constructed
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 being constructed
NOTICE: CMM: Node phys-schost-1: attempting to join cluster.
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 being initiated
NOTICE: CMM: Quorum device /dev/did/rdisk/dls2: owner set to node 1.
NOTICE: CMM: Cluster has reached quorum.
NOTICE: CMM: Node phys-schost-1 (nodeid = 1) is up; new incarnation number = 1068496374.
NOTICE: CMM: Node phys-schost-2 (nodeid = 2) is up; new incarnation number = 1068496374.
NOTICE: CMM: Cluster members: phys-schost-1 phys-schost-2.
NOTICE: CMM: node reconfiguration #1 completed.
NOTICE: CMM: Node phys-schost-1: joined cluster.

```

▼ クラスタをリブートする方法

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

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

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

停止の手順については、Oracle RAC 製品のドキュメントを参照してください。

2. **クラスタ内の任意のノードで、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。**

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

3. **クラスタを停止します。**

- **グローバルクラスタを停止します。**

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

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

```
phys-schost# clzonecluster halt zone-cluster-name
```

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

注記 - 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。

4. **各ノードをブートします。**

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ブートしてもかまいません。停止中に構成を変更した場合は、最新の構成情報を持つノードを最初に起動する必要があります。

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

```
ok boot
```

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

GRUB メニューが表示された時点で、適切な Oracle Solaris OS エントリを選択し、Enter キーを押します。

GRUB ベースのブートの詳細は、『[Oracle Solaris 11.3 システムのブートとシャットダウン](#)』の「[システムのブート](#)」を参照してください。

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

```
phys-schost# clzonecluster boot zone-cluster-name
```

注記 - 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。

クラスタコンポーネントがブートすると、ブート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. ノードが問題なくブートし、オンラインであることを確認します。

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

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

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

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

ゾーンクラスタ内で `cluster status` コマンドを実行して、ノードのステータスを確認することもできます。

注記 - 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 Oracle Solaris 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、[96 ページの「満杯の /var ファイルシステムを修復する方法」](#)を参照してください。

例 19 ゾーンクラスタのリポート

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

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

```

Sep  5 19:18:23 schost-4 cl_runtime: NOTICE: Membership : Node 2 of cluster 'sparse-szone'
joined.
Sep  5 19:18:23 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-szone'
joined.
Sep  5 19:18:23 schost-4 cl_runtime: NOTICE: Membership : Node 4 of cluster 'sparse-szone'
joined.

phys-schost#
phys-schost# clzonecluster status

=== Zone Clusters ===

--- Zone Cluster Status ---

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

```

例 20 SPARC: グローバルクラスタのリポート

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

```

phys-schost# cluster shutdown -g0 -y
Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
...
The system is down.
syncing file systems... done
Program terminated
ok boot
Rebooting with command: boot
...
Hostname: phys-schost-1
Booting as part of a cluster
...
NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-2 (incarnation # 937690106) has become reachable.
NOTICE: Node phys-schost-3 (incarnation # 937690290) has become reachable.
NOTICE: cluster has reached quorum.
...
NOTICE: Cluster members: phys-schost-1 phys-schost-2 phys-schost-3.

```

```

...
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:

```

例 21 x86: クラスタのリブート

次に、正常なクラスタの動作が停止し、すべてのノードが停止して、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。`-g 0` オプションは停止の猶予期間をゼロに設定し、`-y` は確認質問に自動で `yes` 応答を提供します。停止メッセージは、クラスタ内のほかのノードのコンソールにも表示されます。

```

# cluster shutdown -g0 -y
May 2 10:32:57 phys-schost-1 cl_runtime:
WARNING: CMM: Monitoring disabled.
root@phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
/etc/rc0.d/K05initrgm: Calling clnode 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

ATI RAGE SDRAM BIOS P/N GR-xlint.007-4.330
*
BIOS Lan-Console 2.0
Copyright (C) 1999-2001 Intel Corporation
MAC ADDR: 00 02 47 31 38 3C
AMIBIOS (C)1985-2002 American Megatrends Inc.,
Copyright 1996-2002 Intel Corporation
SCB20.86B.1064.P18.0208191106
SCB2 Production BIOS Version 2.08
BIOS Build 1064

```

2 X Intel(R) Pentium(R) III CPU family 1400MHz
Testing system memory, memory size=2048MB
2048MB Extended Memory Passed
512K L2 Cache SRAM Passed
ATAPI CD-ROM SAMSUNG CD-ROM SN-124

Press <F2> to enter SETUP, <F12> Network

Adaptec AIC-7899 SCSI BIOS v2.5754
(c) 2000 Adaptec, Inc. All Rights Reserved.
Press <Ctrl><A> for SCSISelect(TM) Utility!

Ch B, SCSI ID: 0 SEAGATE ST336605LC 160
SCSI ID: 1 SEAGATE ST336605LC 160
SCSI ID: 6 ESG-SHV SCA HSBP M18 ASYN
Ch A, SCSI ID: 2 SUN StorEdge 3310 160
SCSI ID: 3 SUN StorEdge 3310 160

AMIBIOS (C)1985-2002 American Megatrends Inc.,
Copyright 1996-2002 Intel Corporation
SCB20.86B.1064.P18.0208191106
SCB2 Production BIOS Version 2.08
BIOS Build 1064

2 X Intel(R) Pentium(R) III CPU family 1400MHz
Testing system memory, memory size=2048MB
2048MB Extended Memory Passed
512K L2 Cache SRAM Passed
ATAPI CD-ROM SAMSUNG CD-ROM SN-124

SunOS - Intel Platform Edition Primary Boot Subsystem, vsn 2.0

Current Disk Partition Information

Part#	Status	Type	Start	Length
1	Active	X86 BOOT	2428	21852
2		SOLARIS	24280	71662420
3		<unused>		
4		<unused>		

Please select the partition you wish to boot: * *

Solaris DCB

loading /solaris/boot.bin

SunOS Secondary Boot version 3.00

Solaris Intel Platform Edition Booting System

Autobooting from bootpath: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/
pci8086,341a@7,1/sd@0,0:a

If the system hardware has changed, or to boot from a different

```

device, interrupt the autoboot process by pressing ESC.
Press ESCape to interrupt autoboot in 2 seconds.
Initializing system
Please wait...
Warning: Resource Conflict - both devices are added

NON-ACPI device: ISY0050
Port: 3F0-3F5, 3F7; IRQ: 6; DMA: 2
ACPI device: ISY0050
Port: 3F2-3F3, 3F4-3F5, 3F7; IRQ: 6; DMA: 2

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@d/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
Size: 275683 + 22092 + 150244 Bytes
/platform/i86pc/kernel/unix loaded - 0xac000 bytes used
SunOS Release 5.9 Version Generic_112234-07 32-bit
Copyright 1983-2003 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.
configuring IPv4 interfaces: e1000g2.
Hostname: phys-schost-1
Booting as part of a cluster
NOTICE: CMM: Node phys-schost-1 (nodeid = 1) with votecount = 1 added.
NOTICE: CMM: Node phys-schost-2 (nodeid = 2) with votecount = 1 added.
NOTICE: CMM: Quorum device 1 (/dev/did/rdisk/dls2) added; votecount = 1, bitmask
of nodes with configured paths = 0x3.
NOTICE: clcomm: Adapter e1000g3 constructed
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 being constructed
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 being initiated
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 online
NOTICE: clcomm: Adapter e1000g0 constructed
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 being constructed
NOTICE: CMM: Node phys-schost-1: attempting to join cluster.
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 being initiated
NOTICE: CMM: Quorum device /dev/did/rdisk/dls2: owner set to node 1.
NOTICE: CMM: Cluster has reached quorum.
NOTICE: CMM: Node phys-schost-1 (nodeid = 1) is up; new incarnation number = 1068496374.
NOTICE: CMM: Node phys-schost-2 (nodeid = 2) is up; new incarnation number = 1068496374.
NOTICE: CMM: Cluster members: phys-schost-1 phys-schost-2.
NOTICE: CMM: node reconfiguration #1 completed.
NOTICE: CMM: Node phys-schost-1: joined cluster.
WARNING: mod_installdrv: no major number for rsmrdt
ip: joining multicasts failed (18) on clprivnet0 - will use link layer
broadcasts for multicast
The system is coming up. Please wait.

```

```
checking ufs filesystems
/dev/rdisk/clt0d0s5: is clean.
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 online
NIS domain name is dev.eng.mycompany.com
starting rpc services: rpcbind keyserv ypbind done.
Setting netmask of e1000g2 to 192.168.255.0
Setting netmask of e1000g3 to 192.168.255.128
Setting netmask of e1000g0 to 192.168.255.128
Setting netmask of clprivnet0 to 192.168.255.0
Setting default IPv4 interface for multicast: add net 224.0/4: gateway phys-schost-1
syslog service starting.
obtaining access to all attached disks

*****
*
* The X-server can not be started on display :0...
*
*****
volume management starting.
Starting Fault Injection Server...
The system is ready.

phys-schost-1 console login:
```

クラスタ内の 1 つのノードの停止とブート

グローバルクラスタノードまたはゾーンクラスタノードをシャットダウンできます。ここでは、グローバルクラスタノードとゾーンクラスタノードを停止する手順を説明します。

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

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

詳細は、[clnode\(1CL\)](#)、[shutdown\(1M\)](#)、および [clzonecluster\(1CL\)](#) のマニュアルページを参照してください。

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

表 4 タスクマップ: ノードの停止とブート

タスク	ツール	手順
ノードを停止します。	グローバルクラスタノードの場合は、 <code>clnode evacuate</code> および <code>shutdown</code> コマンドを使用します。 ゾーンクラスタノードの場合は、 <code>clzonecluster halt</code> コマンドを使用します。	83 ページの「ノードを停止する方法」
ノードを起動します。 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と動作中の接続が必要です。	グローバルクラスタノードの場合は、 <code>boot</code> または <code>b</code> コマンドを使用します。 ゾーンクラスタノードの場合は、 <code>clzonecluster boot</code> コマンドを使用します。	86 ページの「ノードをブートする方法」
クラスタ上のノードをいったん停止してから再起動。 クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクと動作中の接続が必要です。	グローバルクラスタノードの場合は、 <code>clnode evacuate</code> および <code>shutdown</code> コマンドを使用してから、 <code>boot</code> または <code>b</code> を使用します。 ゾーンクラスタノードの場合は、 <code>clzonecluster reboot</code> コマンドを使用。	90 ページの「ノードをリブートする方法」
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ブート。	グローバルクラスタノードの場合は、 <code>clnode evacuate</code> および <code>shutdown</code> コマンドを使用してから、 <code>boot -x</code> を使用します (SPARC または x86 の GRUB メニューエントリ編集で)。 基になるグローバルクラスタが非クラスタモードでブートされる場合は、ゾーンクラスタノードも自動的に非クラスタモードになります。	93 ページの「非クラスタモードでノードをブートする方法」

▼ ノードを停止する方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。



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

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、グローバルクラスタノードを退避し、すべてのリソースグループおよびデバイスグループを、次に優先されるノードに切り替えることもできます。また、ゾーンクラスタノードを停止することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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

停止の手順については、Oracle RAC 製品のドキュメントを参照してください。

2. 停止するクラスタノードで、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

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

```
phys-schost# clzonecluster halt -n physical-name zone-cluster-name
```

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

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

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

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

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

5. ノードを停止します。

停止するグローバルクラスタノードで shutdown コマンドを実行します。

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

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

6. 必要であればノードの電源を切ります。

例 22 SPARC: グローバルクラスタノードの停止

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

```
phys-schost# clnode evacuate phys-schost-1
phys-schost# shutdown -g0 -y -i0
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.pmfed 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 23 x86: グローバルクラスタノードの停止

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

```

phys-schost# clnode evacuate phys-schost-1
phys-schost# shutdown -g0 -y -i0
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 clnode evacuate
failfasts disabled on node 1
Print services already stopped.
Mar 10 13:47:44 phys-schost-1 syslogd: going down on signal 15
umount: /global/.devices/node@2 busy
umount: /global/.devices/node@1 busy
The system is down.
syncing file systems... done
WARNING: CMM: Node being shut down.
Type any key to continue

```

例 24 ゾーンクラスタノードの停止

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

```

phys-schost# clzonecluster status

```

```

=== Zone Clusters ===

--- Zone Cluster Status ---

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

phys-schost#
phys-schost# clzonecluster halt -n schost-4 sparse-sczone
Waiting for zone halt commands to complete on all the nodes of the zone cluster "sparse-
sczone"...
Sep  5 19:24:00 schost-4 cl_runtime: NOTICE: Membership : Node 3 of cluster 'sparse-sczone'
died.
phys-host#
phys-host# clzonecluster status

```

```

=== Zone Clusters ===

--- Zone Cluster Status ---

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

phys-schost#

```

参照 停止したグローバルクラスタノードを再起動するには、[86 ページの「ノードをブートする方法」](#)を参照してください。

▼ ノードをブートする方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタノードをブートすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

グローバルクラスタまたはゾーンクラスタの他のアクティブなノードを停止またはリブートする場合は、ブートするノードのマルチユーザーサーバーのマイルストーンがオンラインになるまで待ちます。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、停止またはリブートするクラスタ内の他のノードからサービスを引き継げません。

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

Oracle Solaris Cluster をゲストドメインで実行している場合は、制御ドメインまたは I/O ドメインをリブートすると、ドメインが停止するなど、実行中のゲストドメインが影響を受ける場合があります。制御ドメインまたは I/O ドメインをリブートする前に、ほかのノードとの負荷のバランスを取り直し、Oracle Solaris Cluster を実行しているゲストドメインを停止するようにしてください。

制御ドメインまたは I/O ドメインがリブートされると、ハートビートはゲストドメインによって送受信されません。これにより、スプリットブレインとクラスタ再構成が発生します。制御ドメインまたは I/O ドメインはリブート中のため、ゲストドメインは共有デバイスにアクセスできません。その他のクラスタノードでは、このゲストドメインは共有デバイスから切り離されます。制御ドメインまたは I/O ドメインのリブートが終了すると、ゲストドメインで I/O が再開されますが、ゲストドメインはクラスタ再構成の一部として共有ディスクから切り離されているため、共有ストレージとの間の I/O により、ゲストドメインでパニックが発生します。冗長性のためにゲストが 2 つの I/O ドメインを使用している場合、I/O ドメインを 1 つずつリブートすると、この問題を解決できます。

注記 - 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。

1. **停止したグローバルクラスタノードやゾーンクラスタノードを起動するために、そのノードを起動します。**

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

`ok boot`

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Oracle Solaris エントリを選択し、Enter キーを押します。

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

- ゾーンクラスタがある場合は、ブートするノードを指定できます。

```
phys-schost# clzonecluster boot -n node zone-cluster-name
```

2. ノードが問題なくブートし、オンラインであることを確認します。

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

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

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

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

ホストのノードがクラスタモードでブートされる場合は、ゾーンクラスタノードもクラスタモードのみでブートできます。

注記 - 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 Oracle Solaris 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、[96 ページの「満杯の /var ファイルシステムを修復する方法」](#)を参照してください。

例 25 SPARC: グローバルクラスタノードのブート

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

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

例 26 x86: クラスタノードのブート

次に、ノード phys-schost-1 をブート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。

```

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@d/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: Size: 276915 + 22156 + 150372 Bytes
/platform/i86pc/kernel/unix loaded - 0xac000 bytes used
SunOS Release 5.9 Version on81-feature-patch:08/30/2003 32-bit
Copyright 1983-2003 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.
configuring IPv4 interfaces: e1000g2.
Hostname: phys-schost-1
Booting as part of a cluster
NOTICE: CMM: Node phys-schost-1 (nodeid = 1) with votecount = 1 added.
NOTICE: CMM: Node phys-schost-2 (nodeid = 2) with votecount = 1 added.
NOTICE: CMM: Quorum device 1 (/dev/did/rdisk/dls2) added; votecount = 1, bitmask
of nodes with configured paths = 0x3.
WARNING: CMM: Initialization for quorum device /dev/did/rdisk/dls2 failed with
error EACCES. Will retry later.
NOTICE: clcomm: Adapter e1000g3 constructed
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 being constructed
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 being initiated
NOTICE: clcomm: Path phys-schost-1:e1000g3 - phys-schost-2:e1000g3 online
NOTICE: clcomm: Adapter e1000g0 constructed
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 being constructed
NOTICE: CMM: Node phys-schost-1: attempting to join cluster.
WARNING: CMM: Reading reservation keys from quorum device /dev/did/rdisk/dls2
failed with error 2.
NOTICE: CMM: Cluster has reached quorum.
NOTICE: CMM: Node phys-schost-1 (nodeid = 1) is up; new incarnation number =
1068503958.
NOTICE: CMM: Node phys-schost-2 (nodeid = 2) is up; new incarnation number =
1068496374.
NOTICE: CMM: Cluster members: phys-schost-1 phys-schost-2.
NOTICE: CMM: node reconfiguration #3 completed.
NOTICE: CMM: Node phys-schost-1: joined cluster.
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 being initiated
NOTICE: clcomm: Path phys-schost-1:e1000g0 - phys-schost-2:e1000g0 online
NOTICE: CMM: Retry of initialization for quorum device /dev/did/rdisk/dls2 was
successful.
WARNING: mod_installdr: no major number for rsmrdt
ip: joining multicasts failed (18) on clprivnet0 - will use link layer
broadcasts for multicast
The system is coming up. Please wait.

```

```
checking ufs filesystems
/dev/rdisk/clt0d0s5: is clean.
NIS domain name is dev.eng.mycompany.com
starting rpc services: rpcbind keyserv ypbind done.
Setting netmask of e1000g2 to 192.168.255.0
Setting netmask of e1000g3 to 192.168.255.128
Setting netmask of e1000g0 to 192.168.255.128
Setting netmask of clprivnet0 to 192.168.255.0
Setting default IPv4 interface for multicast: add net 224.0/4: gateway phys-schost-1
syslog service starting.
obtaining access to all attached disks

*****
*
* The X-server can not be started on display :0...
*
*****
volume management starting.
Starting Fault Injection Server...
The system is ready.

phys-schost-1 console login:
```

▼ ノードをリブートする方法

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

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用してゾーンクラスタノードをリブートすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。



注意 - リソースのメソッドのいずれかがタイムアウトして強制終了できなかった場合、リソースの Failover_mode プロパティが HARD に設定されているときに限り、ノードがリブートされません。Failover_mode プロパティがそれ以外の値に設定されている場合、ノードはリブートされません。

グローバルクラスタまたはゾーンクラスタの他のアクティブなノードを停止またはリブートするには、リブートするノードのマルチユーザーサーバーのマイルストーンがオンラインになるまで待ちます。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、停止またはリブート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。

1. グローバルクラスタノードまたはゾーンクラスタノードで Oracle RAC を実行している場合は、停止するノード上のデータベースのすべてのインスタンスを停止します。
停止の手順については、Oracle RAC 製品のドキュメントを参照してください。
2. 停止するノードで、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
グローバルクラスタのノードから、次の手順のステップをすべて実行します。
3. `clnode evacuate` および `shutdown` コマンドを使用して、グローバルクラスタノードを停止します。
グローバルクラスタのノード上で実行する `clzonecluster halt` コマンドで、ゾーンクラスタを停止します。(`clnode evacuate` コマンドと `shutdown` コマンドもゾーンクラスタ内で動作します。)
グローバルクラスタの場合は、停止するノードで次のコマンドを入力します。 `clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定のノード上の大域ゾーンから、ほかのノード上の次に優先される大域ゾーンへ、すべてのリソースグループを切り替えます。

注記 - ノードを個別にシャットダウンするには、`shutdown -g0 -y -i6` コマンドを使用します。複数のノードを一括してシャットダウンするには、`shutdown -g0 -y -i0` コマンドを使用してノードを停止します。すべてのノードが停止した後は、すべてのノードに対して `boot` コマンドを使用して、すべてのノードをブートしてクラスタに戻します。

- SPARC ベースのシステムの場合、ノードを個別にリポートするには、次のコマンドを実行します。

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

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

- x86 ベースのシステムの場合、ノードを個別にリポートするには、次のコマンドを実行します。

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

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

GRUB メニューが表示された時点で、適切な Oracle Solaris エントリを選択し、Enter キーを押します。

- 停止し、リポートするゾーンクラスタノードを指定します。

```
phys-schost# clzonecluster reboot - node zone-cluster-name
```

注記 - 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。

4. ノードが問題なくブートし、オンラインであることを確認します。

- グローバルクラスタノードがオンラインであることを確認します。

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

- ゾーンクラスタノードがオンラインであることを確認します。

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

例 27 SPARC: グローバルクラスタノードのリポート

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

```
phys-schost# clnode evacuate phys-schost-1
phys-schost# shutdown -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 28 ゾーンクラスタノードのリブート

次の例は、ゾーンクラスタ上のノードをリブートする方法を示しています。

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

phys-schost#
phys-schost# clzonecluster status

=== Zone Clusters ===

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

phys-schost#
```

▼ 非クラスタモードでノードをブートする方法

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

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 非クラスタモードで起動するクラスタで、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
グローバルクラスタのノードから、次の手順のステップをすべて実行します。
2. ゾーンクラスタノードまたはグローバルクラスタノードをシャットダウンします。

`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ゾーンから、ほかのノード上の次に優先される大域ゾーンへ、すべてのリソースグループを切り替えます。

■ 特定のグローバルクラスタノードをシャットダウンします。

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

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

■ グローバルクラスタノードから特定のゾーンクラスタノードを停止します。

```
phys-schost# clzonecluster halt -n node zone-cluster-name
```

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

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

4. 非クラスタモードでグローバルクラスタノードをブートします。

■ SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -xs
```

■ x86 ベースのシステム上で、次のコマンドを実行します。

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

GRUB メニューが表示されます。

GRUB ベースのブートの詳細は、『[Oracle Solaris 11.3 システムのブートとシャットダウン](#)』の「[システムのブート](#)」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 ブートパラメータ画面が表示されます。

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

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

```
grub edit> kernel$ /platform/i86pc/kernel/$ISADIR/unix -B $ZFS-BOOTFS -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。

e. **b** と入力して、ノードを非クラスタモードでブートします。

注記 - カーネルブートパラメータコマンドへのこの変更は、システムをブートすると無効になります。次にノードをリブートする際には、ノードはクラスタモードでブートします。クラスタモードではなく、非クラスタモードでブートするには、これらの手順を再度実行して、カーネルブートパラメータコマンドに `-x` オプションを追加します。

例 29 SPARC: 非クラスタモードでグローバルクラスタノードをブートする

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。`-g0` オプションは猶予期間をゼロに設定し、`-y` オプションは確認質問に自動で `yes` 応答を提供し、`-i0` オプションは実行レベル 0 (ゼロ) を呼び出します。このノードの停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

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

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

```
ok boot -x
...
Not booting as part of cluster
...
The system is ready.
phys-schost-1 console login:
```

満杯の /var ファイルシステムを修復する

Oracle Solaris ソフトウェアと Oracle Solaris 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。このため時間が経過すると、`/var` ファイルシステムがいっぱいになる可能性があります。クラスタノードの `/var` ファイルシステムが満杯になると、次のブート時にそのノード上で Oracle Solaris Cluster が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方法

`/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、Oracle Solaris Cluster サービスが引き続き実行されているときは、この手順を使用して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、『[Oracle Solaris 11.3 でのシステム管理のトラブルシューティング](#)』の「システムメッセージの形式」を参照してください。

1. 満杯の `/var` ファイルシステムのクラスタのノードで、`root` 役割になります。
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。

◆◆◆ 第 4 章

データレプリケーションのアプローチ

この章では、Oracle Solaris Cluster ソフトウェアで使用できるデータレプリケーション技術について説明します。データレプリケーションは、プライマリストレージデバイスからバックアップデバイス (セカンダリデバイス) へのデータのコピーとして定義されます。プライマリデバイスに障害が発生した場合も、セカンダリデバイスからデータを使用できます。データレプリケーションを使用すると、クラスタの高可用性と耐障害性を確保できます。

Oracle Solaris Cluster ソフトウェアは、次のデータレプリケーションタイプをサポートします。

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

データレプリケーションを実行するには、レプリケーションするオブジェクトと同じ名前のデバイスグループが必要です。デバイスは、一度に 1 つのデバイスグループのみに属することができるため、デバイスを含む Oracle Solaris Cluster デバイスグループがすでにある場合、そのデバイスを新しいデバイスグループに追加する前にそのグループを削除する必要があります。Solaris Volume Manager, ZFS, または raw ディスクデバイスグループの作成および管理については、[120 ページの「デバイスグループの管理」](#)を参照してください。

クラスタに最適なサービスを提供するレプリケーションアプローチを選択するには、ホストベースとストレージベースのデータレプリケーションを両方とも理解しておく必要があります。障害回復でデータレプリケーションを管理するための Oracle Solaris Cluster Geographic Edition フレームワークの使用の詳細は、『[Oracle Solaris Cluster 4.3 Geographic Edition Overview](#)』を参照してください。

この章の内容は次のとおりです。

- [98 ページの「データレプリケーションについての理解」](#)
- [100 ページの「キャンパスクラスタ内でのストレージベースのデータレプリケーションの使用」](#)

データレプリケーションについての理解

Oracle Solaris Cluster ソフトウェアは、ホストベースとストレージベースのデータレプリケーションをサポートしています。

- **ホストベースのデータレプリケーション**は、ソフトウェアを使用して、地理的に離れたクラスタ間でディスクボリュームをリアルタイムでレプリケーションします。リモートミラーレプリケーションを使用すると、プライマリクラスタのマスターボリュームのデータを、地理的に離れたセカンダリクラスタのマスターボリュームにレプリケーションできます。リモートミラービットマップは、プライマリディスク上のマスターボリュームと、セカンダリディスク上のマスターボリュームの差分を追跡します。クラスタ間またはクラスタとクラスタ内にないホストとの間のレプリケーションに使用されるホストベースのレプリケーションソフトウェアの例には、Oracle Solaris の Availability Suite 機能があります。

ホストベースのデータ複製は、特別なストレージレイではなくホストリソースを使用するため、より費用のかからないデータ複製ソリューションです。Oracle Solaris OS を実行する複数のホストが共有ボリュームにデータを書き込むことができるように構成されているデータベース、アプリケーション、またはファイルシステムはサポートされていません (Oracle RAC など)。

- 2 つのクラスタ間でのホストベースのデータレプリケーションの使用に関する詳細は、『[Oracle Solaris Cluster Geographic Edition Data Replication Guide for Oracle Solaris Availability Suite](#)』を参照してください。
- Geographic Edition フレームワークを使用しないホストベースのレプリケーションの例については、付録 A の [付録A デプロイメントの例: Availability Suite ソフトウェアによるクラスタ間のホストベースのデータレプリケーションの構成](#)を参照してください。
- **ストレージベースのデータレプリケーション**では、ストレージコントローラ上でソフトウェアを使用することで、データレプリケーション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します。このソフトウェアはノードの処理能力を一部解放し、クラスタのリクエストに対応します。クラスタ内またはクラスタ間でデータを複製できるストレージベースのソフトウェアには、EMC SRDF などがあります。ストレージベースのデータレプリケーションは、キャンパスクラスタ構成において特に重要になる場合があります、必要なインフラストラクチャーを簡素化できます。
 - キャンパスクラスタ環境でストレージベースのデータレプリケーションを使用する方法の詳細は、[100 ページの「キャンパスクラスタ内でのストレージベースのデータレプリケーションの使用」](#)を参照してください。
 - 複数のクラスタ間でのストレージベースのレプリケーションと、そのプロセスを自動化する Oracle Solaris Cluster Geographic Edition 製品の使用の詳細は、『[Oracle Solaris Cluster 4.3 Geographic Edition Overview](#)』を参照してください。

サポートされるデータレプリケーション方式

Oracle Solaris Cluster ソフトウェアは、クラスタ間またはクラスタ内のデータレプリケーション方式として次をサポートしています。

- クラスタ間のレプリケーション – Oracle Solaris Cluster Geographic Edition ソフトウェアを使用する障害回復の場合、クラスタ間のデータレプリケーションを実行するためにホストベースまたはストレージベースのレプリケーションを使用できます。通常、ホストベースのレプリケーションとストレージベースのレプリケーションのいずれかを選択し、両方を組み合わせることはありません。Geographic Edition ソフトウェアで両方のタイプのレプリケーションを管理できます。

レプリケーションのタイプ	データレプリケーション製品
ホストベースのレプリケーション	Oracle Solaris の Availability Suite 機能。
ストレージベースのレプリケーション	EMC Symmetrix Remote Data Facility (SRDF)。 Hitachi TrueCopy および Hitachi Universal Replicator、 Oracle Solaris Cluster Geographic Edition フレームワーク経由。 Oracle ZFS Storage Appliance。

詳細は、『[Oracle Solaris Cluster 4.3 Geographic Edition Overview](#)』の「[Data Replication](#)」を参照してください。

Oracle Solaris Cluster Geographic Edition ソフトウェアを使用せずにホストベースのレプリケーションを使用する場合は、[付録A デプロイメントの例: Availability Suite ソフトウェアによるクラスタ間のホストベースのデータレプリケーションの構成](#)の手順を参照してください。

Oracle Solaris Cluster Geographic Edition ソフトウェアを使用せずにストレージベースのレプリケーションを使用する場合は、レプリケーションソフトウェアのドキュメントを参照してください。

- キャンパスクラスタ内のレプリケーション - この方式は、ホストベースのミラー化の代替として使用されます。

レプリケーションのタイプ	データレプリケーション製品
ストレージベースのレプリケーション	EMC Symmetrix Remote Data Facility (SRDF)。 Hitachi TrueCopy および Hitachi Universal Replicator、 Oracle Solaris Cluster Geographic Edition フレームワーク経由。

- アプリケーションベースのレプリケーション – Oracle Data Guard はアプリケーションベースのレプリケーションソフトウェアの 1 つの例です。このタイプのソフトウェアは、Geographic Edition フレームワークによる障害回復で、単一インスタンス Oracle データベースまたは RAC データベースをレプリケートするためにのみ使用されます。

詳細は、『[Oracle Solaris Cluster Geographic Edition Data Replication Guide for Oracle Data Guard](#)』を参照してください。

キャンパスクラスタ内でのストレージベースのデータレプリケーションの使用

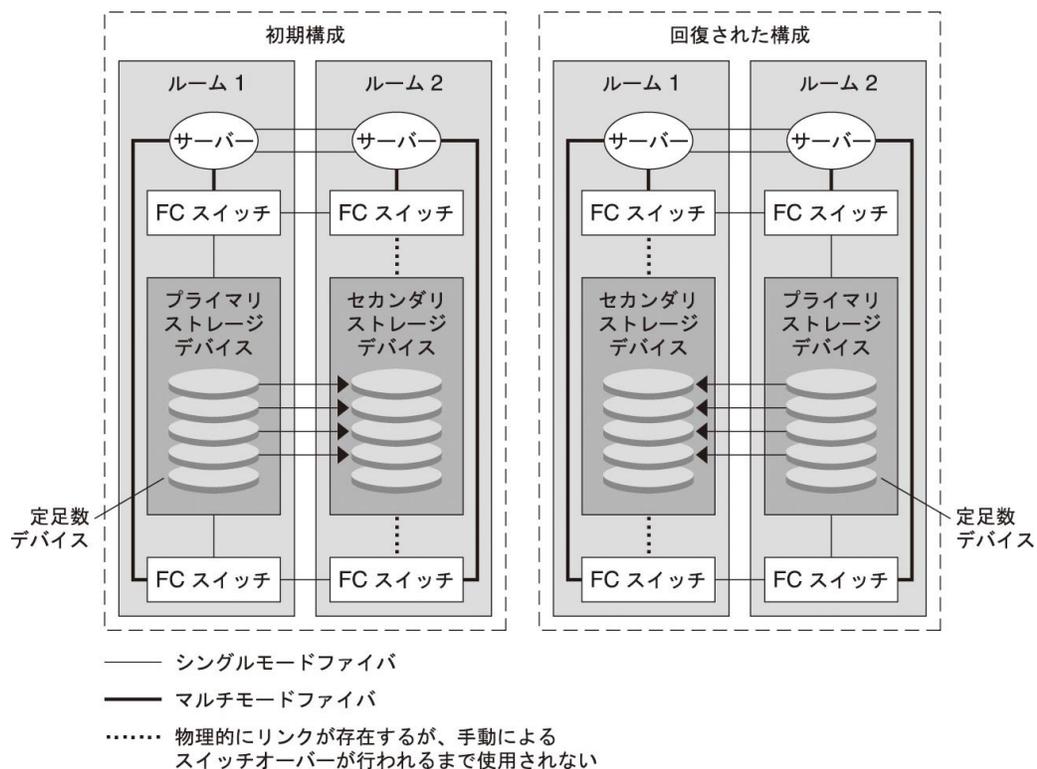
ストレージベースのデータレプリケーションは、ストレージデバイスにインストールされているソフトウェアを使用して、キャンパスクラスタと呼ばれるクラスタ内のレプリケーションを管理します。このようなソフトウェアは、特定のストレージデバイスに固有で、障害回復には使用されません。ストレージベースのデータレプリケーションを構成する際には、ストレージデバイスに付属するドキュメントを参照してください。

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レプリケーションで自動と手動のいずれかのフェイルオーバーを使用できます。Oracle Solaris Cluster では、EMC SRDF ソフトウェアでのレプリケーションで手動と自動の両方のフェイルオーバーをサポートしています。

このセクションでは、キャンパスクラスタで使用されるストレージベースのデータレプリケーションについて説明します。[図1](#)に、2つのストレージアレイ間でデータがレプリケートされる、2ルーム構成の例を示します。この構成では、最初のルームにプライマリストレージアレイがあり、これが両方のルームのノードにデータを提供します。また、プライマリストレージアレイは、複製するデータをセカンダリストレージアレイに提供します。

注記 - [図1](#)は、レプリケートされていないボリュームに定足数デバイスがあることを示しています。レプリケートされたボリュームを定足数デバイスとして使用することはできません。

図 1 ストレージベースのデータレプリケーションを装備した 2 ルーム構成



Oracle Solaris Cluster では、EMC SRDF を使用したストレージベースの同期レプリケーションがサポートされています。EMC SRDF では、非同期レプリケーションはサポートされていません。

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

リモートストレージデバイスとの通信が失われた場合は、never または async の Fence_level を指定して、プライマリクラスタ上で実行されているアプリケーションがブロックされないようにしてください。data または status の Fence_level を指定すると、リモートストレージデバイスに更新がコピーできない場合に、プライマリストレージデバイスが更新を拒否します。

キャンパスクラスタ内でストレージベースのデータレプリケーションを使用する際の要件と制限

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

- クラスタを自動フェイルオーバー用に構成する場合は、同期レプリケーションを使用します。レプリケート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、107 ページの「[ストレージベースのレプリケートされたデバイスの構成および管理](#)」を参照してください。キャンパスクラスタを設計するための要件の詳細は、『[Oracle Solaris Cluster Hardware Administration Manual](#)』の「[Shared Data Storage](#)」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、非同期データレプリケーション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使って、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レプリケートする最善の方法を決定します。
- ノード間の距離は、Oracle Solaris 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、Oracle のサービスプロバイダにお問い合わせください。
- レプリケートされたボリュームを、定足数デバイスとして構成しないでください。共有のレプリケートされていないボリュームにある定足数デバイスを見つけるか、定足数サーバーを使用します。
- データのプライマリ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。それ以外の場合、ボリュームマネージャーはデータのプライマリコピーとセカンダリコピーの両方に同時にアクセスしようとする場合があります。データコピーの可視性の制御については、ストレージアレイに付属するドキュメントを参照してください。
- EMC SRDF では、ユーザーがレプリケートされるデバイスのグループを定義できます。各レプリケーションデバイスグループには、同じ名前の Oracle Solaris Cluster デバイスグループが必要です。
- EMC SRDF を使用し、並列または直列接続された RDF デバイスを備える、3 つのサイトまたは 3 つのデータセンター構成の場合、参加するすべてのクラスタノードで、Solutions Enabler の SYMCLI オプションファイルに次のエントリを追加する必要があります。

```
SYMAPI_2SITE_CLUSTER_DG=device-group:rdf-group-number
```

このエントリにより、クラスタソフトウェアで、2 つの SRDF 同期サイト間でのアプリケーションの移動を自動化できます。エントリ内の *rdf-group-number* は、ホストのローカル Symmetrix を 2 番目のサイトの Symmetrix に接続する RDF グループを表します。

3 つのデータセンター構成の詳細については、『[Oracle Solaris Cluster 4.3 Geographic Edition Overview](#)』の「[Three-Data-Center \(3DC\) Topologies](#)」を参照してください。

- Oracle Real Application Clusters (Oracle RAC) は、クラスタ内でレプリケートする場合、SRDF ではサポートされません。現在プライマリレプリカではないレプリカに接続されたノードには、書き込みアクセス権はありません。クラスタのすべてのノードからの直接書き込

みアクセス権が必要なスケラブルアプリケーションは、レプリケートされるデバイスでサポートできません。

- Solaris Volume Manager for Sun Cluster 複数所有者ディスクセットはサポートされていません。
- EMC SRDF でドミノモードまたは適応型コピーモードを使用しないでください。詳細は、100 ページの「[キャンパスクラスタ内でのストレージベースのデータレプリケーションの使用](#)」を参照してください。

キャンパスクラスタ内でストレージベースのデータレプリケーションを使用する際の手動回復に関する懸念事項

すべてのキャンパスクラスタと同じように、ストレージベースのデータレプリケーションを使用するクラスタでは、通常、1 つの障害が発生した場合は介入の必要はありません。ただし、手動フェイルオーバーを使用しており、プライマリストレージデバイスを保持するルームが失われている場合 (図1を参照)、2 ノードクラスタで問題が発生します。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ブートできません。このような状況では、クラスタで次の手動介入が必要になります。

1. クラスタメンバーとしてブートするよう、Oracle 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 Oracle のサービスプロバイダが、セカンダリストレージデバイスのレプリケートされていない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. セカンダリストレージデバイスをプライマリストレージとして使用できるよう、ユーザーまたは Oracle 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レプリケーション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レプリケーションに EMC SRDF ソフトウェアを使用する場合は、静的デバイスではなく、動的デバイスを使用してください。静的デバイスではレプリケーションプライマリを変更するのに数分かかり、フェイルオーバー時間に影響を与えることがあります。

◆◆◆ 第 5 章

グローバルデバイス、ディスクパスモニタリング、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、ディスクパスモニタリング、およびクラスタファイルシステムの管理手順について説明します。

- 105 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」
- 107 ページの「ストレージベースのレプリケートされたデバイスの構成および管理」
- 120 ページの「クラスタファイルシステムの管理の概要」
- 120 ページの「デバイスグループの管理」
- 146 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」
- 151 ページの「クラスタファイルシステムの管理」
- 157 ページの「ディスクパスモニタリングの管理」

この章の関連手順の概要については、表7「タスクマップ: デバイスグループの管理」を参照してください。

グローバルデバイス、グローバル名前空間、デバイスグループ、ディスクパスモニタリング、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、『Oracle Solaris Cluster 4.3 Concepts Guide』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

Oracle Solaris Cluster デバイスグループの管理は、クラスタにインストールされているボリュームマネージャーによって異なります。Solaris Volume Manager は「クラスタ対応」であるため、Solaris Volume Manager の `metaset` コマンドを使用してデバイスグループを追加、登録、および削除します。詳細は、[metaset\(1M\)](#) のマニュアルページを参照してください。

Oracle Solaris 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、グループのプライマリ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、Oracle Solaris OS のリブート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を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、今後クラスタに参加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 Volume Manager のグローバルデバイスのアクセス権

グローバルデバイスのアクセス権に加えた変更は、Solaris Volume Manager 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播されません。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する必要があります。

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

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操作を実行する場合は、次の問題を考慮する必要があります。

- Oracle Solaris 動的再構成機能に関して記載されている要件、手順、および制限のすべてが、Oracle Solaris Cluster 動的再構成のサポートにも適用されます。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。そのため、Oracle Solaris Cluster ソフトウェアで動的再構成機能を使用する前に、Oracle Solaris 動的再構成機能のドキュメントを確認してください。特に、動的再構成の切り離し操作中に、ネットワークに接続されていない入力デバイスに影響する問題について確認してください。
- Oracle Solaris Cluster は、プライマリノード内のアクティブなデバイス上での動的再構成ボード削除操作を拒否します。動的再構成操作は、プライマリノード内のアクティブでないデバイスと、セカンダリノード内の任意のデバイス上で実行できます。
- 動的再構成操作が完了すると、クラスタデータへのアクセスが以前と同様に続行されます。
- Oracle Solaris Cluster は、定足数デバイスの可用性に影響を与える動的再構成操作を拒否します。詳細については、[167 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 - セカンダリノード上で動的再構成操作を実行中に現在のプライマリノードに障害が発生した場合は、クラスタの可用性に影響を受けます。新しいセカンダリノードが提供されるまで、プライマリ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動的再構成操作を実行するには、次の手順を示されている順序で完了します。

表 5 タスクマップ: ディスクデバイスとテープデバイスでの動的再構成

タスク	説明
1. 現在のプライマリノード上で、アクティブなデバイスグループに影響を与える動的再構成操作を実行する必要がある場合は、そのデバイス上で動的再構成削除操作を実行する前に、プライマリノードとセカンダリノードを切り替えます	143 ページの「デバイスグループのプライマリノードを切り替える」
2. 削除されるデバイス上で動的再構成削除操作を実行します	システムに付属しているドキュメントを確認してください。

ストレージベースのレプリケートされたデバイスの構成および管理

ストレージベースのレプリケーションを使用してレプリケートされるデバイスが含まれるように、Oracle Solaris Cluster デバイスグループを構成できます。Oracle Solaris Cluster ソフトウェアは、ストレージベースのレプリケート用に EMC Symmetrix Remote Data Facility ソフトウェアをサポートしています。

EMC Symmetrix Remote Data Facility ソフトウェアでデータをレプリケーションする前に、ストレージベースのレプリケーションのドキュメントによく目を通し、ストレージベースのレプリケーション製品と最新の更新をシステムにインストールしておく必要があります。ストレージベースのレプリケーションソフトウェアのインストールについては、製品のドキュメントを参照してください。

ストレージベースのレプリケーションソフトウェアは、デバイスのペアをレプリケーションとして構成する際、一方のデバイスをプライマリレプリケーション、もう一方のデバイスをセカンダリレプリケーションとします。一方のノードのセットに接続されたデバイスが、常にプライマリレプリカになります。もう一方のノードのセットに接続されたデバイスは、セカンダリレプリカになります。

Oracle Solaris Cluster 構成では、レプリカが属する Oracle Solaris Cluster デバイスグループが移動されると常に、プライマリレプリカが自動的に移動されます。そのため、Oracle Solaris Cluster 構成では、プライマリレプリカを直接移動しないでください。その代わりに、テイクオーバーは関連する Oracle Solaris Cluster デバイスグループを移動することによって行うようにしてください。



注意 - 作成する Oracle Solaris Cluster デバイスグループ (Solaris Volume Manager または raw ディスク) の名前は、レプリケートされたデバイスグループと同じ名前にする必要があります。

EMC Symmetrix Remote Data Facility でレプリケートされたデバイスの管理

次の表に、EMC Symmetrix Remote Data Facility (SRDF) ストレージベースのレプリケートされたデバイスを設定および管理するために実行する必要のあるタスクを示します。

表 6 タスクマップ: EMC SRDF ストレージベースのレプリケートされたデバイスの管理

タスク	手順
ストレージデバイスとノードに SRDF ソフトウェアをインストールする	EMC ストレージデバイスに付属するドキュメント。
EMC レプリケーショングループを構成する	108 ページの「EMC SRDF レプリケーショングループを構成する方法」
DID デバイスを構成する	110 ページの「EMC SRDF を使用して DID デバイスをレプリケーション用に構成する方法」
レプリケートされたグループを登録する	127 ページの「デバイスグループを追加および登録する方法 (Solaris Volume Manager)」
構成を確認する	112 ページの「EMC SRDF でレプリケートされたグローバルデバイスグループ構成を確認する方法」
キャンパスクラスタのプライマリルームが完全に失敗したあとに手動でデータを回復する	117 ページの「プライマリルームの完全な失敗後に EMC SRDF データを回復する方法」

▼ EMC SRDF レプリケーショングループを構成する方法

- 始める前に
- EMC Symmetrix Remote Data Facility (SRDF) レプリケーショングループを構成する前に、すべてのクラスタノードに EMC Solutions Enabler ソフトウェアをインストールする必要があります。まず、クラスタの共有ディスクに EMC SRDF デバイスグループを構成します。EMC SRDF デバイスグループを構成する方法の詳細は、EMC SRDF 製品のドキュメントを参照してください。
 - EMC SRDF を使用する場合は、静的デバイスではなく、動的デバイスを使用します。静的デバイスではレプリケーションプライマリを変更するのに数分かかり、フェイルオーバー時間に影響を与えることがあります。



注意 - 作成する Oracle Solaris Cluster デバイスグループ (Solaris Volume Manager または raw ディスク) の名前は、レプリケートされたデバイスグループと同じ名前にする必要があります。

1. ストレージアレイに接続されたすべてのノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. 並列 SRDF または直列接続されたデバイスを使用し、サイトが 3 つまたはデータセンターが 3 つある実装の場合、`SYMAPI_2SITE_CLUSTER_DG` パラメータを設定します。

参加しているすべてのクラスタノードで、Solutions Enabler のオプションファイルに次のエントリーを追加します。

```
SYMAPI_2SITE_CLUSTER_DG=device-group:rdf-group-number
```

device-group デバイスグループの名前を指定します。

rdf-group-number ホストのローカル Symmetrix を 2 番目のサイトの Symmetrix に接続する RDF グループを指定します。

このエントリーにより、クラスタソフトウェアで、2 つの SRDF 同期サイト間でのアプリケーションの移動を自動化できます。

3 つのデータセンター構成の詳細については、『[Oracle Solaris Cluster 4.3 Geographic Edition Overview](#)』の「[Three-Data-Center \(3DC\) Topologies](#)」を参照してください。

3. レプリケートデータで構成された各ノードで、シンメトリックスデバイス構成を検出します。

これには数分かかることがあります。

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

4. まだレプリカペアを作成していない場合は、この時点で作成します。

レプリカペアを作成するには、`symrdf` コマンドを使用します。レプリカペアの作成方法の手順については、SRDF のドキュメントを参照してください。

注記 - サイトが 3 つまたはデータセンターが 3 つある実装で並列 RDF デバイスを使用する場合は、すべての `symrdf` コマンドに次のパラメータを追加します。

```
-rdfg rdf-group-number
```

`symrdf` コマンドに RDF グループ番号を指定することで、`symrdf` の操作が正しい RDF グループに向けられるようになります。

5. レプリケーションされたデバイスによって構成された各ノードで、データのレプリケーションが正しく設定されていることを確認します。

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

6. デバイスグループのスワップを実行します。

a. プライマリレプリカとセカンダリレプリカが同期していることを確認します。

```
# /usr/symcli/bin/symrdf -g group-name verify -synchronized
```

b. どのノードにプライマリレプリカが含まれ、どのノードにセカンダリレプリカが含まれているかを判別するには、`symsg show` コマンドを使用します。

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

RDF1 デバイスのノードにはプライマリレプリカが含まれ、RDF2 デバイス状態のノードにはセカンダリレプリカが含まれます。

c. セカンダリレプリカを有効にします。

```
# /usr/symcli/bin/symrdf -g group-name failover
```

d. RDF1 デバイスと RDF2 デバイスをスワップします。

```
# /usr/symcli/bin/symrdf -g group-name swap -refresh R1
```

e. レプリカペアを有効にします。

```
# /usr/symcli/bin/symrdf -g group-name establish
```

f. プライマリノードとセカンダリレプリカが同期していることを確認します。

```
# /usr/symcli/bin/symrdf -g group-name verify -synchronized
```

7. 最初にプライマリレプリカがあったノードで、手順 5 のすべてを繰り返します。

次の手順 EMC SRDF でレプリケートされたデバイス用にデバイスグループを構成したら、レプリケートされたデバイスが使用するデバイス識別子 (DID) ドライバを構成する必要があります。

▼ EMC SRDF を使用して DID デバイスをレプリケーション用に構成する方法

この手順では、レプリケートされたデバイスが使用するデバイス識別名 (DID) ドライバを構成します。指定された DID デバイスインスタンスが相互のレプリケーションであること、およびそれらが指定されたレプリケーショングループに属していることを確認します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタの任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. 構成される RDF1 デバイスおよび RDF2 デバイスに対応する DID デバイスを判別します。

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

注記 - システムに Oracle Solaris デバイスのパッチ全体が表示されない場合は、環境変数 `SYMCLI_FULL_PDEVNAME` を 1 に設定して、`symdg -show` コマンドを再入力します。

- Oracle Solaris デバイスに対応する DID デバイスを判別します。

```
# cldevice list -v
```

- 一致した DID デバイスのペアごとに、レプリケートされた 1 つの DID デバイスにインスタンスを結合します。RDF2/セカンダリ側から次のコマンドを実行します。

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

注記 - SRDF データレプリケーションデバイスでは、`-t` オプションはサポートされていません。

`-t replication-type`

レプリケーションタイプを指定します。EMC SRDF の場合、**SRDF** を入力します。

`-g replication-device-group`

`symdg show` コマンドで表示されるデバイスグループの名前を指定します。

`-d destination-instance`

RDF1 デバイスに対応する DID インスタンスを指定します。

`source-instance`

RDF2 デバイスに対応する DID インスタンスを指定します。

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

```
# scdidadm -b device
```

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

- レプリケーションデバイスグループの名前が変更されている場合は、次の追加の手順を実行します。

レプリケーションデバイスグループと、対応するグローバルデバイスグループの名前が変更された場合は、最初に、`scdidadm -b` コマンドを使用して既存の情報を削除し、レプリケーションされたデバイス情報を更新する必要があります。最後の手順では、`cldevice combine` コマンドを使用して、更新された新しいデバイスを作成します。

- DID インスタンスが結合されていることを確認します。

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

7. SRDF レプリケーションが設定されていることを確認します。

```
# cldevice show device
```

8. すべてのノード上で、結合されたすべての 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 レプリケートされたデバイスが使用するデバイス識別名 (DID) ドライバを構成したら、EMC SRDF でレプリケートされたグローバルデバイスグループ構成を確認する必要があります。

▼ EMC SRDF でレプリケートされたグローバルデバイスグループ構成を確認する方法

始める前に グローバルデバイスグループを確認する前に、まずそれを作成する必要があります。Solaris Volume Manager、ZFS、または raw ディスクからデバイスグループを使用できます。詳細は、次を参照してください。

- [127 ページの「デバイスグループを追加および登録する方法 \(Solaris Volume Manager\)」](#)
- [129 ページの「デバイスグループ \(raw ディスク\) を追加および登録する方法」](#)
- [130 ページの「レプリケートデバイスグループ \(ZFS\) の追加と登録方法」](#)



注意 - 作成した Oracle Solaris Cluster デバイスグループ (Solaris Volume Manager または raw ディスク) の名前は、レプリケートされたデバイスグループと同じ名前である必要があります。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. プライマリデバイスグループが、プライマリレプリカが含まれるノードと同じノードに対応することを確認します。

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

2. 試験的にスイッチオーバーを実行し、デバイスグループが正しく構成され、レプリカ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場合は、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プライマリになります。

3. 次のコマンドの出力を比較し、スイッチオーバーが成功したことを確認します。

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

例: Oracle Solaris Cluster の SRDF レプリケーショングループの構成

この例では、クラスタの SRDF レプリケーションを設定するために必要な Oracle Solaris Cluster 固有の手順を実行します。この例では、次のタスクが実行済みであることを前提とします。

- アレイ間のレプリケーションの LUN のペア作成が完了している。
- ストレージデバイスとクラスタノードに SRDF ソフトウェアがインストール済みである。

例 30 レプリカペアの作成

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

すべてのノードで次のコマンドを実行します

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

```
# symdev list pd
```

```
Symmetrix ID: 000187990182
```

Device Name		Directors			Device		
-----		-----			-----		
Sym	Physical	SA	:P DA	:IT Config	Attribute	Sts	Cap (MB)

0067	c5t600604800001879901*	16D:0	02A:C1	RDF2+Mir	N/Grp'd	RW	4315
0068	c5t600604800001879901*	16D:0	16B:C0	RDF1+Mir	N/Grp'd	RW	4315
0069	c5t600604800001879901*	16D:0	01A:C0	RDF1+Mir	N/Grp'd	RW	4315

...

RDF1 側のすべてのノードで次のコマンドを実行します

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

RDF2 側のすべてのノードで次のコマンドを実行します

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

例 31 データレプリケーション設定の確認

次のコマンドをクラスタの 1 つのノードで実行します。

```
# symdg show dg1
```

Group Name: dg1

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

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

Standard (STD) Devices (1):

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

Device Group RDF Information

...

```
# symrdf -g dg1 establish
```

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

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

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

```
Device: 0067 ..... Merged.
Merge device track tables between source and target.....Done.
Resume RDF link(s).....Started.
Resume RDF link(s).....Done.
```

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

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

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

Target (R2) View					Source (R1) View					MODES					
Standard	Logical	Device	Dev	E	R1 Inv	R2 Inv	Tracks	Tracks	S	Dev	E	Tracks	Tracks	MDA	STATE
DEV001	0067	WD		0		0	RW	0067	RW		0		0	S..	Synchronized
Total		-----		-----		-----		-----		-----		-----			
MB(s)				0.0		0.0				0.0		0.0			

Legend for MODES:

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

```
#
```

例 32 使用されているディスクに対応する DID の表示

RDF1 側と RDF2 側に同じ手順を適用します。

dymdg show dg コマンドの出力の PdevName フィールドの下に DID が表示されます。

RDF1 側でこれらのコマンドを実行します

```
# symdbg show dg1
```

```
Group Name: dg1
```

```
Group Type : RDF1 (RDFA)
```

```
...
```

```
Standard (STD) Devices (1):
```

```
{
```

```
-----
```

```

                                Sym          Cap
                                Dev  Att. Sts  (MB)
-----
DEV001          /dev/rdisk/c5t6006048000018790002353594D303637d0s2 0067  RW  4315
}

Device Group RDF Information
...
  対応する DID を取得します
# cldevice list | grep c5t6006048000018790002353594D303637d0
217    pmoney1:/dev/rdisk/c5t6006048000018790002353594D303637d0 /dev/did/rdsk/d217
217    pmoney2:/dev/rdisk/c5t6006048000018790002353594D303637d0 /dev/did/rdsk/d217

  対応する DID を一覧表示します
# cldevice show d217

=== DID Device Instances ===

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

#

  RDF2 側でこれらのコマンドを実行します
# symdg show dg1

Group Name:  dg1

Group Type                :  RDF2      (RDFA)
...
Standard (STD) Devices (1):
{
-----
LdevName          PdevName          Sym          Cap
Dev  Att. Sts  (MB)
-----
DEV001          /dev/rdisk/c5t6006048000018799018253594D303637d0s2 0067  WD  4315
}

Device Group RDF Information
...
  対応する DID を取得します
# cldevice list | grep c5t6006048000018799018253594D303637d0
108    pmoney4:/dev/rdisk/c5t6006048000018799018253594D303637d0 /dev/did/rdsk/d108
108    pmoney3:/dev/rdisk/c5t6006048000018799018253594D303637d0 /dev/did/rdsk/d108

  対応する DID を一覧表示します
# cldevice show d108

```

```

=== DID Device Instances ===

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

#

```

例 33 DID インスタンスの結合

RDF2 側から、次のように入力します。

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

例 34 結合された DID の表示

クラスタ内の任意のノードから、次のように入力します。

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

```

=== DID Device Instances ===

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

```

▼ プライマリルールの完全な失敗後に EMC SRDF データを回復する方法

この手順では、キャンパスクラスタのプライマリルールが完全に失敗し、プライマリルールがセカンダリルールにフェイルオーバーして、プライマリルールがオンラインに戻ったとき、データ回復を実行します。キャンパスクラスタのプライマリルールは、プライマリノードとストレージサイトです。ルールの完全な失敗には、そのルールのホストとストレージの両方の不具合が含まれます。プライマリルールが失敗した場合、Oracle Solaris Cluster は自動的にセカンダリルールにフェイルオーバーし、セカンダリルールのストレージデバイスを読み書き可能にし、対応するデバイスグループとリソースグループのフェイルオーバーを有効にします。

プライマリルールがオンラインに戻ったら、セカンダリルールに書き込まれた SRDF デバイスグループからデータを手動で回復し、データを再同期できます。この手順では、元のセカンダリル

ム (この手順では、セカンダリルームに `phys-campus-2` を使用) のデータを元のプライマリルーム (`phys-campus-1`) と同期して、SRDF デバイスグループを回復します。また、この手順では、SRDF デバイスグループタイプを、`phys-campus-2` では RDF1 に変更し、`phys-campus-1` では RDF2 に変更します。

始める前に 手動でフェイルオーバーを実行する前に、EMC レプリケーショングループおよび DID デバイスを構成し、EMC レプリケーショングループを登録する必要があります。Solaris Volume Manager デバイスグループの作成の詳細は、[127 ページの「デバイスグループを追加および登録する方法 \(Solaris Volume Manager\)」](#)を参照してください。

注記 - これらの説明は、プライマリルームが完全にフェイルオーバーしてからオンラインに戻ったあとに、SRDF データを手動で回復するための 1 つの方法を示しています。その他の方法については、EMC のドキュメントを確認してください。

これらの手順を実行するには、キャンパスクラスタのプライマリルームにログインします。次の手順では、`dg1` は SRDF デバイスグループ名です。失敗した時点では、この手順のプライマリルームは `phys-campus-1` で、セカンダリルームは `phys-campus-2` です。

1. キャンパスクラスタのプライマリルームにログインし、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. プライマリルームから、`symrdf` コマンドを使用して RDF デバイスのレプリケーションステータスに対するクエリーを実行し、これらのデバイスに関する情報を表示します。

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

ヒント - `split` 状態にあるデバイスグループは同期されません。

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

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

4. RDF デバイスのステータスを表示します。

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

5. フェイルオーバー後、フェイルオーバーした RDF デバイスのデータをスワップできます。

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

6. RDF デバイスに関するステータスおよびその他の情報を確認します。

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

7. プライマリルームで SRDF デバイスグループを確立します。

```
phys-campus-1# symrdf -g dgl establish
```

8. デバイスグループが同期状態にあり、デバイスグループタイプが RDF2 であることを確認します。

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

例 35 プライマリサイトフェイルオーバー後の EMC SRDF データの手動回復

この例では、キャンパスクラスタのプライマリルームがフェイルオーバーし、セカンダリルームがテイクオーバーしてデータを記録し、プライマリルームがオンラインに戻ったあとで、EMC SRDF データを手動で回復するために必要な Oracle Solaris Cluster 固有の手順が示されています。この例では、SRDF デバイスグループは *dgl* と呼ばれ、標準論理デバイスは DEV001 です。失敗した時点では、プライマリルームは *phys-campus-1* で、セカンダリルームは *phys-campus-2* です。キャンパスクラスタのプライマリルーム *phys-campus-1* から各手順を実行します。

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

```
phys-campus-1# syndg list | grep RDF
dgl RDF1 Yes 00187990182 1 0 0 0 0
```

```
phys-campus-1# symrdf -g dgl -force failover
...
```

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

```
phys-campus-1# syndg list | grep RDF
dgl RDF1 Yes 00187990182 1 0 0 0 0
```

```
phys-campus-1# symrdf -g dgl swap
...
```

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

```
phys-campus-1# syndg list | grep RDF
dgl RDF2 Yes 00187990182 1 0 0 0 0
```

```
phys-campus-1# symrdf -g dgl establish
...
```

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

```
phys-campus-1# syndg list | grep RDF
dgl RDF2 Yes 00187990182 1 0 0 0 0
```

クラスタファイルシステムの管理の概要

クラスタファイルシステムの管理には特別な Oracle Solaris 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 Oracle Solaris ファイルシステムを管理するときと同じように、Oracle Solaris の標準のファイルシステムコマンド (`mount` や `newfs` など) を使用します。クラスタファイルシステムをマウントするには、`mount` コマンドに `-g` オプションを指定します。クラスタファイルシステムは UFS を使用しており、ブート時に自動的にマウントすることもできます。クラスタファイルシステムは、グローバルクラスタ内のノードからのみ認識できます。

注記 - 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。

- `unlink` コマンドは、空ではないディレクトリではサポートされません。詳細は、[unlink\(1M\)](#) のマニュアルページを参照してください。
- `lockfs -d` コマンドはサポートされません。回避方法として、`lockfs -n` を使用してください。
- クラスタファイルシステムの再マウント時に `directio` マウントオプションを追加して再マウントすることはできません。

デバイスグループの管理

クラスタの要件の変更により、クラスタ上のデバイスグループの追加、削除、または変更が必要になる場合があります。Oracle Solaris Cluster には、このような変更を使用できる、`clsetup` と呼ばれる対話式インタフェースがあります。`clsetup` ユーティリティは、`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タスクを示し、またこのセクションの適切な手順へのリンクを示します。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、デバイスグループをオンラインにしたり、オフラインにしたりすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

Oracle Solaris 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。



注意 - ほかのノードが有効なクラスタメンバーであり、それらのノードの少なくとも 1 つがディスクセットを持つ場合は、クラスタの外部でブートされるクラスタノードで `metaset -s setname -f -t` を実行しないでください。

表 7 タスクマップ: デバイスグループの管理

タスク	手順
cldevice populate コマンドを使用することにより、再構成のリポートを行わずにグローバルデバイス名前空間を更新する	122 ページの「グローバルデバイス名前空間を更新する方法」
グローバルデバイス名前空間で使用する lofi デバイスのサイズを変更する	123 ページの「グローバルデバイス名前空間で使用する lofi デバイスのサイズを変更する方法」
既存のグローバルデバイス名前空間を移動する	124 ページの「専用パーティションから lofi デバイスにグローバルデバイス名前空間を移行する方法」 125 ページの「lofi デバイスから専用パーティションにグローバルデバイス名前空間を移行する方法」
metaset コマンドを使用して、Solaris Volume Manager 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	127 ページの「デバイスグループを追加および登録する方法 (Solaris Volume Manager)」
cldevicegroup コマンドを使用することにより、raw ディスクデバイスグループを追加および登録する	129 ページの「デバイスグループ (raw ディスク) を追加および登録する方法」
cldevicegroup コマンドを使用して、ZFS に名前付きデバイスグループを追加する	130 ページの「レプリケートデバイスグループ(ZFS)の追加と登録方法」
metaset コマンドおよび metaclear コマンドを使用することにより、構成から Solaris Volume Manager デバイスグループを削除する	133 ページの「デバイスグループを削除して登録を解除する方法 (Solaris Volume Manager)」
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	133 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方法」
metaset コマンドを使用して、Solaris Volume Manager デバイスグループからノードを削除する	134 ページの「デバイスグループからノードを削除する方法 (Solaris Volume Manager)」
cldevicegroup コマンドを使用することにより、raw ディスクデバイスグループからノードを削除する	136 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方法」
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プロパティを変更する	138 ページの「デバイスグループのプロパティを変更する方法」
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	142 ページの「デバイスグループ構成の一覧を表示する方法」
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セカンダリノードの希望数を変更する	140 ページの「デバイスグループのセカンダリノードの希望数を設定する方法」
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プライマリノードを切り替える	143 ページの「デバイスグループのプライマリノードを切り替える」

タスク	手順
metaset 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	144 ページの「デバイスグループを保守状態にする方法」

▼ グローバルデバイス名前空間を更新する方法

新しいグローバルデバイスを追加するときに、`cldevice populate` コマンドを実行して手動でグローバルデバイス名前空間を更新します。

注記 - コマンドを実行するノードがクラスタのメンバーでない場合は、`cldevice populate` コマンドを実行しても無効です。また、`/global/.devices/node@nodeID` ファイルシステムがマウントされていない場合も、コマンドは無効になります。

1. クラスタの任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. クラスタの各ノードで、`devfsadm` 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。詳細は、[devfsadm\(1M\)](#) のマニュアルページを参照してください。

3. 名前空間を再構成します。

```
# cldevice populate
```

4. ディスクセットを作成する前に、各ノードで `cldevice populate` コマンドが完了していることを確認します。

ノードの 1 つで `cldevice` コマンドが実行された場合でも、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 cldevice populate
```

例 36 グローバルデバイス名前空間を更新する

次の例に、`cldevice populate` コマンドを正しく実行することにより生成される出力を示します。

```
# devfsadm
cldevice populate
Configuring the /dev/global directory (global devices)...
obtaining access to all attached disks
reservation program successfully exiting
# ps -ef | grep cldevice populate
```

▼ グローバルデバイス名前空間で使用する lofi デバイスのサイズを変更する方法

グローバルクラスターの 1 つ以上のノードのグローバルデバイス名前空間で lofi デバイスを使用する場合は、次の手順を使用してデバイスのサイズを変更します。

1. サイズを変更するグローバルデバイス名前空間の lofi デバイスのノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. ノードからサービスを退避させ、ノードを非クラスタモードでリブートします。
これは、この手順の実行中にグローバルデバイスがこのノードからサービスを提供されないようにするために行います。手順については、[93 ページの「非クラスタモードでノードをブートする方法」](#)を参照してください。
3. グローバルデバイスのファイルシステムをマウント解除し、その lofi デバイスを切り離します。
グローバルデバイスファイルシステムはローカルにマウントされます。

```
phys-schost# umount /global/.devices/node@\`c\linfo -n` > /dev/null 2>&1
```

lofi デバイスが切り離されていることを確認します

```
phys-schost# lofiadm -d /.globaldevices
```

コマンドから何の出力も得られない場合、デバイスが切り離されています

注記 -m オプションを使用してファイルシステムがマウントされた場合、`mnttab` ファイルにエントリは追加されません。umount コマンドによって次のような警告が報告される場合があります。

```
umount: warning: /global/.devices/node@2 not in mnttab  =====>>>not mounted
```

この警告は無視してもかまいません。

4. `/.globaldevices` ファイルを削除し、必要なサイズで再作成します。
次の例は、サイズが 200M バイトの新しい `/.globaldevices` ファイルの作成を示しています。

```
phys-schost# rm /.globaldevices
phys-schost# mkfile 200M /.globaldevices
```

5. グローバルデバイス名前空間の新しいファイルシステムを作成します。

```
phys-schost# lofiadm -a /.globaldevices
phys-schost# newfs `lofiadm /.globaldevices` < /dev/null
```

6. クラスタモードでノードをブートします。
グローバルデバイスが新しいファイルシステムに生成されました。

```
phys-schost# reboot
```

7. サービスを実行するノードに移行します。

グローバルデバイス名前空間を移行する

専用パーティションでグローバルデバイス名前空間を作成するのではなく、ループバックファイルインタフェース (lofi) デバイス上に名前空間を作成できます。

注記 - ルートファイルシステム用に ZFS がサポートされていますが、1 つの重要な例外があります。グローバルデバイスファイルシステムにブートディスクの専用パーティションを使用する場合、そのファイルシステムとして UFS のみを使用する必要があります。グローバルデバイス名前空間には、UFS ファイルシステムで動作しているプロキシファイルシステム (PxFS) が必要です。

ただし、グローバルデバイス名前空間の UFS ファイルシステムは、ルート (/) ファイルシステムやほかのルートファイルシステム (/var や /home など) の ZFS ファイルシステムと共存できます。または、代わりに lofi デバイスを使用してグローバルデバイス名前空間をホストする場合、ルートファイルシステムに対する ZFS の使用に関する制限はありません。

次の手順は、既存のグローバルデバイス名前空間を専用パーティションから lofi デバイスまたはその逆に移行する方法を説明しています。

- [124 ページの「専用パーティションから lofi デバイスにグローバルデバイス名前空間を移行する方法」](#)
- [125 ページの「lofi デバイスから専用パーティションにグローバルデバイス名前空間を移行する方法」](#)

▼ 専用パーティションから lofi デバイスにグローバルデバイス名前空間を移行する方法

1. 名前空間の場所を変更するグローバルクラスタノードで、root 役割になります。
2. ノードからサービスを退避させ、ノードを非クラスタモードでリブートします。
これは、この手順の実行中にグローバルデバイスがこのノードからサービスを提供されないようにするために行います。手順については、[93 ページの「非クラスタモードでノードをブートする方法」](#)を参照してください。
3. /.globaldevices という名前のファイルがノードに存在しないことを確認します。
ファイルが存在する場合は、削除します。
4. lofi デバイスを作成します。

```
# mkfile 100m /.globaldevices# lofiadm -a /.globaldevices
```

```
# LOFI_DEV=`lofiadm /.globaldevices`
# newfs `echo ${LOFI_DEV} | sed -e 's/lofi/rlofi/g'` < /dev/null
# lofiadm -d /.globaldevices
```

5. `/etc/vfstab` ファイルで、グローバルデバイス名前空間エントリをコメントアウトします。
このエントリには、`/global/.devices/node@nodeID` で始まるマウントパスがあります。
6. グローバルデバイスパーティション `/global/.devices/node@nodeID` のマウントを解除します。
7. `globaldevices` および `scmountdev` SMF サービスを無効にし再度有効にします。

```
# svcadm disable globaldevices
# svcadm disable scmountdev
# svcadm enable scmountdev
# svcadm enable globaldevices
```

lofi デバイスは現在 `/.globaldevices` に作成され、グローバルデバイスファイルシステムとしてマウントされています。

8. パーティションから lofi デバイスへ移行したいグローバルデバイス名前空間のある他のノードでもこのステップを繰り返します。
9. 1 つのノードから、グローバルデバイス名前空間を生成します。

```
# cldevice populate
```

各ノードで、コマンドが処理を完了したことを確認してから、クラスタに対する以降の操作を実行してください。

```
# ps -ef | grep "cldevice populate"
```

グローバルデバイス名前空間は、現在 lofi デバイスにあります。

10. サービスを実行するノードに移行します。

▼ lofi デバイスから専用パーティションにグローバルデバイス名前空間を移行する方法

1. 名前空間の場所を変更するグローバルクラスタノードで、`root` 役割になります。
2. ノードからサービスを退避させ、ノードを非クラスタモードでリブートします。
これは、この手順の実行中にグローバルデバイスがこのノードからサービスを提供されないようにするために行います。手順については、[93 ページの「非クラスタモードでノードをブートする方法」](#)を参照してください。

3. ノードのローカルディスクで、次の要件を満たす新しいパーティションを作成します。

- サイズが 512 M バイト以上
- UFS ファイルシステムの使用

4. グローバルデバイスファイルシステムとしてマウントする新しいパーティションに、`/etc/vfstab` ファイルへのエントリを追加します。

- 現在のノードのノード ID を指定します。

```
# /usr/sbin/clinfo -n node-ID
```

- 次の形式を使用して、`/etc/vfstab` ファイルに新しいエントリを作成します。

```
blockdevice rawdevice /global/.devices/node@nodeID ufs 2 no global
```

たとえば、使用するよう選択したパーティションが `/dev/did/rdisk/d5s3` である場合、`/etc/vfstab` ファイルに追加する新しいエントリは次のようになります。

```
/dev/did/dsk/d5s3 /dev/did/rdisk/d5s3 /global/.devices/node@3 ufs 2 no global
```

5. グローバルデバイスパーティション `/global/.devices/node@nodeID` のマウントを解除します。

6. `/.globaldevices` ファイルに関連付けられた lofi デバイスを削除します。

```
# lofiadm -d /.globaldevices
```

7. `/.globaldevices` ファイルを削除します。

```
# rm /.globaldevices
```

8. `globaldevices` および `scmountdev` SMF サービスを無効にし再度有効にします。

```
# svcadm disable globaldevices# svcadm disable scmountdev
# svcadm enable scmountdev
# svcadm enable globaldevices
```

パーティションは現在グローバルデバイス名前空間ファイルシステムとしてマウントされています。

9. lofi デバイスからパーティションへ移行したいグローバルデバイス名前空間のある他のノードでもこのステップを繰り返します。

10. クラスタモードでブートして、グローバルデバイス名前空間を生成します。

- a. クラスタの 1 つのノードから、グローバルデバイス名前空間を生成します。

```
# cldevice populate
```

- b. クラスタのすべてのノードで処理が完了したことを確認してから、ノードに対する作業を実行してください。

```
# ps -ef | grep cldevice populate
```

グローバルデバイス名前空間は、現在専用パーティションにあります。

11. サービスを実行するノードに移行します。

デバイスグループを追加および登録する

Solaris Volume Manager、ZFS、または raw ディスクのデバイスグループを追加および登録できます。

▼ デバイスグループを追加および登録する方法 (Solaris Volume Manager)

`metaset` コマンドを使用して Solaris Volume Manager ディスクセットを作成し、そのディスクセットを Oracle Solaris Cluster デバイスグループとして登録します。デバイスグループには、ディスクセットを登録するときにディスクセットに割り当てた名前が自動的に割り当てられます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。



注意 - 作成する Oracle Solaris Cluster デバイスグループ (Solaris Volume Manager、または raw ディスク) の名前は、レプリケートしたデバイスグループと同じ名前にする必要があります。

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、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. Solaris Volume Manager ディスクセットを追加し、これをデバイスグループとして Oracle Solaris 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。

注記 - `metaset` コマンドを実行して、クラスタに Solaris Volume Manager デバイスグループを設定すると、そのデバイスグループに含まれるノード数にかかわらず、デフォルトで 1 つのセカンダリになります。デバイスグループが作成されたあと、`clsetup` ユーティリティを使用することで、セカンダリ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、[140 ページの「デバイスグループのセカンダリノードの希望数を設定する方法」](#)を参照してください。

- レプリケーション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レプリケーションプロパティを設定します。

```
# cldevicegroup sync devicegroup
```

- 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
```

- 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` 形式の完全な DID デバイス名を使用してください。

次の例では、DID デバイス `/dev/did/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/clt1d0
Full Device Path:                   phys-schost-2:/dev/rdisk/clt1d0
...
```

- ディスクセットにドライブを追加します。

完全な DID パス名を使用します。

```
# metaset -s setname -a /dev/did/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。

注記 - 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 (cNt XdY) 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。

7. 新しいディスクセットとドライブのステータスを検査します。

```
# metaset -s setname
```

例 37 Solaris Volume Manager 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 と /dev/did/rdisk/d2 を含むディスクセットおよびデバイスグループの作成を示し、デバイスグループが作成されたことを確認します。

```
# 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 (raw ディスク) を追加および登録する方法

Oracle Solaris Cluster ソフトウェアでは、他のボリュームマネージャーに加え、raw ディスクデバイスグループを使用できます。Oracle Solaris Cluster を最初に構成する際、クラスタ内の raw デバイスごとにデバイスグループが自動的に構成されます。ここで説明する手順を使用して、これらの自動作成されたデバイスグループを Oracle Solaris Cluster ソフトウェアでできるように再構成します。

次の理由のため、raw ディスクタイプの新しいデバイスグループを作成します。

- 複数の DID をデバイスグループに追加したい
- デバイスグループの名前を変更する必要がある
- cldevicegroup コマンドの -v オプションを使用せずにデバイスグループのリストを作成したい



注意 - レプリケートしたデバイスにデバイスグループを作成する場合は、作成するデバイスグループ (Solaris Volume Manager または raw ディスク) の名前をレプリケートしたデバイスグループの名前と同じにする必要があります。

1. 使用する各デバイスを特定し、事前に規定されたデバイスグループの構成を解除します。
次のコマンドは、デバイス *dN* および *dX* に対する定義済みのデバイスグループを除去します。

```
phys-schost-1# cldevicegroup disable dsk/dN dsk/dX
phys-schost-1# cldevicegroup offline dsk/dN dsk/dX
phys-schost-1# cldevicegroup delete dsk/dN dsk/dX
```

2. 必要なデバイスを含む、新しい raw ディスクデバイスグループを作成します。
次のコマンドは、デバイス *dN* および *dX* が含まれるグローバルデバイスグループ *raw-disk-dg* を作成します。

```
phys-schost-1# cldevicegroup create -n phys-schost-1,phys-schost-2 \
-t rawdisk -d dN,dX raw-disk-dg
```

3. 作成された raw ディスクデバイスグループを確認します。

```
phys-schost-1# cldevicegroup show raw-disk-dg
```

▼ レプリケートデバイスグループ(ZFS)の追加と登録方法

この手順を使用して、HAStoragePlus によって管理されるレプリケートされた ZFS デバイスグループを作成します。

HAStoragePlus を使用しない ZFS ストレージプール (zpool) を作成するには、代わりに[131 ページの「HAStoragePlus を使用せずにローカル ZFS ストレージプールを構成する方法」](#)に進みます。

始める前に ZFS をレプリケートするには、名前付きデバイスグループを作成し、zpool に属するディスクを一覧表示する必要があります。デバイスは、一度に 1 つのデバイスグループのみに属することができるため、デバイスを含む Oracle Solaris Cluster デバイスグループがすでにある場合、そのデバイスを新しい ZFS デバイスグループに追加する前にそのグループを削除する必要があります。

作成する Oracle Solaris Cluster デバイスグループ (Solaris Volume Manager または raw ディスク) の名前は、レプリケートされたデバイスグループと同じ名前にする必要があります。

1. zpool のデバイスに対応するデフォルトデバイスグループを削除してください。
たとえば、2 つのデバイス */dev/did/dsk/d2* と */dev/did/dsk/d13* を含む *mypool* と呼ばれる zpool がある場合、*d2* と *d13* と呼ばれる 2 つのデフォルトデバイスグループを削除する必要があります。

```
# cldevicegroup offline dsk/d2 dsk/d13
# cldevicegroup delete dsk/d2 dsk/d13
```

2. **ステップ 1** で削除したデバイスグループの DID に対応する DID の名前付きデバイスグループを作成します。

```
# cldevicegroup create -n pnode1,pnode2 -d d2,d13 -t rawdisk mypool
```

このアクションでは、mypool(zpool と同じ名前)と呼ばれるデバイスグループが作成され、raw デバイス /dev/did/dsk/d2 と /dev/did/dsk/d13 を管理します。

3. それらのデバイスを含む zpool を作成します。

```
# zpool create mypool mirror /dev/did/dsk/d2 /dev/did/dsk/d13
```

4. リソースグループを作成し、ノードリストに唯一のグローバルゾーンのあるレプリケートしたデバイス (デバイスグループ内) の移行を管理します。

```
# clresourcegroup create -n pnode1,pnode2 migrate_srdfdg-rg
```

5. **ステップ 4** で作成したリソースグループに hasp-rs リソースを作成し、globaldevicepaths プロパティを raw ディスクタイプのデバイスグループに設定します。

このデバイスは**ステップ 2** で作成しました。

```
# clresource create -t HAStoragePlus -x globaldevicepaths=mypool \
-g migrate_srdfdg-rg hasp2migrate_mypool
```

6. このリソースグループの rg_affinities プロパティの +++ 値を、**ステップ 4** で作成したリソースグループに設定します。

```
# clresourcegroup create -n pnode1,pnode2 \
-p RG_affinities=+++migrate_srdfdg-rg oracle-rg
```

7. **ステップ 4** または**ステップ 6** のいずれかで作成したリソースグループ内に、**ステップ 3** で作成した zpool の HAStoragePlus リソース (hasp-rs) を作成します。

resource_dependencies プロパティを**ステップ 5** で作成した hasp-rs リソースに設定します。

```
# clresource create -g oracle-rg -t HAStoragePlus -p zpools=mypool \
-p resource_dependencies=hasp2migrate_mypool \
-p ZpoolsSearchDir=/dev/did/dsk hasp2import_mypool
```

8. デバイスグループ名が必要な場合には、この新しいリソースグループ名を使用します。

▼ HAStoragePlus を使用せずにローカル ZFS ストレージプールを構成する方法

この手順では、HAStoragePlus リソースを構成せずに、ローカルデバイスに ZFS ストレージプール (zpool) を構成する方法について説明します。

注記 - HAStoragePlus リソースを使用するローカル zpool を構成するには、代わりに [130 ページの「レプリケートデバイスグループ\(ZFS\)の追加と登録方法」](#)に進みます。

1. DID マッピングを一覧表示し、使用するローカルデバイスを識別します。

新しい zpool を使用するクラスタノードのみを一覧表示するデバイスを選択します。cNtXdY デバイス名と /dev/did/rdisk/dN DID デバイス名の両方を書きとめておきます。

```
phys-schost-1# cldevice show | grep Device
```

次の例で、DID デバイスのエントリ /dev/did/rdisk/d1 および /dev/did/rdisk/d2 は、それらのドライブが phys-schost-1 によってのみ使用されることを示します。この手順の例では、デバイス名 c0t6d0 の DID デバイス /dev/did/rdisk/d2 が使用され、クラスタノード phys-schost-1 用に構成されます。

```
=== 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
...
```

2. zpool に選択する DID デバイスのデバイスグループ名を特定します。

次の出力例では、disk/d2 が DID デバイス /dev/did/rdisk/d2s2 のデバイスグループ名であることを示しています。デバイスグループ名が DID デバイス名の一部である関係はよくありますが、常にではありません。このデバイスグループには、そのノードリストに 1 つだけのノード phys-schost-1 があります。

```
phys-schost-1# cldevicegroup show -v
...
Device Group Name:              disk/d2
  Type:                          Disk
  failback:                       false
  Node List:                       phys-schost-1
  preferred:                       false
  localonly:                       false
  autogen:                         true
  numsecondaries:                  1
  device names:                    /dev/did/rdisk/d2s2
...
```

3. DID デバイスに localonly プロパティを設定します。

[ステップ 2](#) で特定したデバイスグループ名を指定します。デバイスのフェンシングを無効にする場合は、コマンドに default_fencing=no fencing も含めます。

```
phys-schost-1# cldevicegroup set -p localonly=true \
```

```
-p autogen=true [-p default_fencing=nofencing] dsk/d2
```

cldevicegroup プロパティの詳細は、[cldevicegroup\(1CL\)](#) のマニュアルページを参照してください。

4. デバイスの設定を確認します。

```
phys-schost-1# cldevicegroup show dsk/d2
```

5. zpool を作成します。

```
phys-schost-1# zpool create localpool c0t6d0
```

6. (オプション) ZFS データセットを作成します。

```
phys-schost-1# zfs create localpool/data
```

7. 新しい zpool を確認します。

```
phys-schost-1# zpool list
```

デバイスグループの保守

デバイスグループに対して様々な管理タスクを実行することができます。これらのタスクの一部は Oracle Solaris Cluster Manager ブラウザインタフェースを使用して実行することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

デバイスグループを削除して登録を解除する方法 (Solaris Volume Manager)

デバイスグループは Oracle Solaris Cluster に登録されている Solaris Volume Manager ディスクセットです。Solaris Volume Manager 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、Oracle Solaris 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手順については、『[Solaris Volume Manager 管理ガイド](#)』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方法

あるクラスタノードを、潜在的なプライマリのリストにそのノードを含むすべてのデバイスグループから削除するには、この手順を使用します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. **すべてのデバイスグループの潜在的なプライマリとして削除するノードで、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. **ステップ 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134 ページの「[デバイスグループからノードを削除する方法 \(Solaris Volume Manager\)](#)」の手順を実行します。**
4. **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
```
5. **ステップ 4 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、136 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方法](#)」の手順を実行します。**
6. **すべてのデバイスグループの潜在的なプライマリ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プライマリ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方法 (Solaris Volume Manager)

Solaris Volume Manager デバイスグループの潜在的なプライマリノードのリストからクラスタノードを削除するには、次の手順を使用します。ノードを削除したいグループデバイスごとに `metaset` コマンドを繰り返します。



注意 - ほかのノードが有効なクラスタメンバーであり、それらのノードの少なくとも 1 つがディスクセットを持つ場合は、クラスタの外部でブートされるクラスタノードで `metaset -s setname -f -t` を実行しないでください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

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 Volume Manager 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 SDS/SVM は Solaris Volume Manager デバイスグループを示しています。

```
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プライマリノードであるかを特定します。

```
# cldevicegroup status devicegroup
```

3. 変更するデバイスグループを現在所有しているノードで、root 役割になります。

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 で特定されたノードをデバイスグループから削除します。

-h nodelist 削除するノード (複数可) のノード名を指定します。

注記 - 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. 潜在的なプライマリとしてノードを削除するデバイスグループごとに **ステップ 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 38 デバイスグループからのノードの削除 (Solaris Volume Manager)

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 を潜在的なプライマリとして削除しま

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。

ノードの Solaris Volume Manager デバイスグループを判断します

```
# 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
```

どのノードがデバイスグループの現在のプライマリノードであるかを特定します。

```
# 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現在所有しているノードで root 役割になります

デバイスグループからホスト名を削除します

```
# metaset -s dg-schost-1 -d -h phys-schost-2
```

ノードの削除を確認します

```
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
```

▼ raw ディスクデバイスグループからノードを削除する方法

raw ディスクデバイスグループの潜在的プライマリノードリストからクラスタノードを削除する場合は、この手順を使用します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタ内の削除するノード以外のノードで、RBAC の承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。
2. 削除されるノードに接続されたデバイスグループを特定し、どれが raw ディスクデバイスグループであるかを判別します。

```
# cldevicegroup show -n nodename -t rawdisk +
```
3. すべての Local_Disk raw ディスクデバイスグループの `localonly` プロパティを無効にします。

```
# cldevicegroup set -p localonly=false devicegroup
```

`localonly` プロパティについての詳細は、`cldevicegroup(1CL)` のマニュアルページを参照してください。
4. 削除するノードに接続されているすべての raw ディスクデバイスグループの `localonly` プロパティが無効になっていることを確認します。

デバイスグループタイプ `Disk` は、この raw ディスクデバイスグループの `localonly` プロパティが無効になっていることを表します。

```
# cldevicegroup show -n nodename -t rawdisk -v +
```
5. **ステップ 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 39 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。

*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断します*

```
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/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
      ノード上のローカルディスクごとに localonly フラグを無効にします
phys-schost-1# cldevicegroup set -p localonly=false dsk/d4

      localonly フラグが無効になっていることを確認します
phys-schost-1# cldevicegroup show -n phys-
schost-2 -t rawdisk +
(dsk/d4) Device group type:            Disk
(dsk/d8) Device group type:            Local_Disk

      すべての raw デバイスデバイスグループからノードを削除します
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方法

デバイスグループのプライマリ所有権を確立するための方法は、`preferenced` と呼ばれる所有権設定属性の設定に基づきます。この属性を設定していない場合は、ほかで所有されていないデバイスグループのプライマリ所有者が、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、プライマリ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。

この手順では、`clsetup` ユーティリティを使用して、Solaris Volume Manager デバイスグループの `preferenced` 属性と `failback` 属性を設定または設定解除します。

始める前に この手順を実行するには、属性値を変更するデバイスグループの名前が必要です。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタの任意のノードで、RBAC の承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。

2. `clsetup` ユーティリティーを起動します。

```
# clsetup
```

メインメニューが表示されます。

3. 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の番号を入力します。

「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、Solaris Volume Manager 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の番号を入力します。

「デバイスグループのプロパティ変更メニュー」が表示されます。

5. デバイスグループのプロパティを変更するには、`preference` または `failback` プロパティを変更するためのオプションの番号を入力します。

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 40 デバイスグループのプロパティの変更

次に、`clsetup` でデバイスグループ (`dg-schost-1`) の属性値を設定したときに生成される `cldevicegroup` コマンドの例を示します。

```
# cldevicegroup set -p preferenced=true -p failback=true -p numsecondaries=1 \
-p nodelist=phys-schost-1,phys-schost-2 dg-schost-1
# cldevicegroup show dg-schost-1
```

```
=== Device Groups ===
```

```
Device Group Name:          dg-schost-1
Type:                       SVM
failback:                   yes
Node List:                  phys-schost-1, phys-schost-2
preferenced:                yes
numsecondaries:             1
diskset names:              dg-schost-1
```

▼ デバイスグループのセカンダリノードの希望数を設定する方法

`numsecondaries` プロパティは、プライマリ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セカンダリ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プライマリ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、セカンダリ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、セカンダリノード数を増やすと、通常の動作中の性能が一律に下がります。通常、セカンダリ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、セカンダリ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、『[Oracle Solaris Cluster 4.3 Concepts Guide](#)』の第 3 章、「[Key Concepts for System Administrators and Application Developers](#)」を参照してください。

`numsecondaries` プロパティを変更すると、セカンダリノードの実際数と希望数の間に整合性がない場合、セカンダリ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場合のデバイスグループのオプションについては、[cldevicegroup\(1CL\)](#) のマニュアルページを参照してください。

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

1. クラスタの任意のノードで、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、「デバイスグループとボリューム (Device Groups and Volumes)」メニュー項目を選択します。
「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 (Change Key Properties of a Device Group)」メニュー項目を選択します。
「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. セカンダリ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の番号を入力します。

指示に従って、デバイスグループに構成したいセカンダリ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:                       Local_Disk
failback:                   yes
Node List:                   phys-schost-1, phys-schost-2, phys-schost-3
preferenced:                 yes
numsecondaries:              1
diskgroup names:            dg-schost-1
```

注記 - デバイスグループ構成の変更には、ボリュームの追加や削除のほか、既存のボリュームのグループ、所有者、またはアクセス権の変更が含まれます。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。[122 ページの「グローバルデバイス名前空間を更新する方法」](#)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 41 セカンダリの希望数の変更 (Solaris Volume Manager)

次に、デバイスグループ (`dg-schost-1`) のセカンダリノードの希望数を構成するときに、`clsetup` によって生成される `cldevicegroup` コマンドの例を示します。この例では、ディスクグループとボリュームは以前に作成されているものと想定しています。

```
# 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 42 セカンダリ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、セカンダリ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方法

構成の一覧を表示するために、root 役割になる必要はありません。ただし、`solaris.cluster.read` の権限は必要です。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

- 次に示されている方法のどれかを選択してください。

Oracle Solaris Cluster Manager ブラウザインタフェース

詳細は、[第13章「Oracle Solaris Cluster Manager ブラウザインタフェースの使用」](#)を参照してください。

`cldevicegroup show`

`cldevicegroup show` 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。

`cldevicegroup show devicegroup`

`cldevicegroup show devicegroup` 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。

`cldevicegroup status devicegroup`

`cldevicegroup status devicegroup` 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。

```
cldevicegroup status +
```

cldevicegroup status + 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 43 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 44 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プライマリ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。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用してアクティブでないデバイスグループをオンラインにすることもできます。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタの任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

2. `cldevicegroup switch` を使用して、デバイスグループのプライマリ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プライマリノードになります。

`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プライマリ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 45 デバイスグループのプライマリノードの切り替え

次に、デバイスグループのプライマリ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方法

デバイスグループを保守状態にすると、そのデバイスのいずれかにアクセスされるたびにそのデバイスグループが自動的にオンラインになることが防止されます。デバイスグループを保守状態にするべきなのは、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を修復していても、この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、データの損失を防ぐために役立ちます。

破損したディスクセットを復元する方法については、[267 ページの「破損したディスクセットの復元」](#)を参照してください。

注記 - デバイスグループを保守状態にする前に、そのデバイスへのすべてのアクセスを停止し、依存するすべてのファイルシステムをマウント解除する必要があります。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、アクティブなデバイスグループをオフラインにすることもできます。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)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、ディスクセットの所有権が必要な場合は、ディスクセットを手動でインポートします。

```
# metaset -C take -f -s diskset
```



注意 - Solaris Volume Manager 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。

3. 必要な修理手順を実行します。
4. ディスクセット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。

```
# metaset -C release -s diskset
```

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 46 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。

```
[デバイスグループを保守状態にします。]
# cldevicegroup disable dg-schost-1
# cldevicegroup offline dg-schost-1
[必要に応じて、手動でディスクセットをインポートします。]
For Solaris Volume Manager:
# metaset -C take -f -s dg-schost-1
[必要な修理手順をすべて実行します。]
[所有権をリリースします。]
Solaris Volume Manager の場合:
# metaset -C release -s dg-schost-1
[デバイスグループをオンラインにします。]
# 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

Oracle Solaris 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オーバーライドします。

- [147 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方法」](#)
- [147 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方法」](#)
- [148 ページの「すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定を変更する方法」](#)
- [150 ページの「単一ストレージデバイスのフェンシングプロトコルを変更する方法」](#)

▼ 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方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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 47 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
clusterid:                   0x4FES2C888
installmode:                  disabled
heartbeat_timeout:           10000
heartbeat_quantum:           1000
private_netaddr:              172.16.0.0
private_netmask:              255.255.111.0
max_nodes:                    64
max_privatenets:              10
udp_session_timeout:          480
concentrate_load:             False
global_fencing:               prefer3
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方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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 48 単一デバイスの SCSI プロトコルの表示

次の例に、デバイス `/dev/rdisk/c4t8d0` の SCSI プロトコルを示します。

```
# cldevice show /dev/rdisk/c4t8d0
```

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

```
DID Device Name:          /dev/did/rdisk/d3
Full Device Path:         phappy1:/dev/rdisk/c4t8d0
Full Device Path:         phappy2:/dev/rdisk/c4t8d0
Replication:              none
default_fencing: global
```

▼ すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定を変更する方法

フェンシングは、クラスタに接続されているすべてのストレージデバイスに対して、グローバルにオンまたはオフに設定できます。あるストレージデバイスのデフォルトのフェンシングが `pathcount`、`prefer3`、または `nofencing` に設定されている場合、そのデバイスのデフォルトのフェンシング設定は、グローバル設定をオーバーライドします。ストレージデバイスのデフォルトのフェンシング設定が `global` に設定されている場合、ストレージデバイスはグローバル設定を使用します。たとえば、ストレージデバイスのデフォルト設定が `pathcount` である場合、ここでの手順を使用してグローバルな SCSI プロトコル設定を `prefer3` に変更しても、設定は変更されません。単一デバイスのデフォルト設定を変更するには、[150 ページの「単一ストレージデバイスのフェンシングプロトコルを変更する方法」](#)の手順を使用します。



注意 - フェンシングを誤ってオフに設定すると、アプリケーションのフェイルオーバー時にデータの破損する可能性が生じやすくなります。フェンシングをオフに設定する場合は、そのような状況でもデータが破損しないかどうか十分に検査してください。共有ストレージデバイスが SCSI プロトコルをサポートしていない場合や、クラスタのストレージへのアクセスをクラスタ外のホストに対して許可する場合は、フェンシングをオフに設定できます。

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、フェンシング設定を変更して、定足数デバイスを再構成します。フェンシングをオフに構成したあとで、定足数デバイスを含むデバイスについては定期的にオンに戻す場合は、定足数サーバーサービスを利用して定足数を構成することを検討してください (そうすることで、定足数の動作を中断せずに済みます)。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. **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フェンシングプロトコルを設定します。**

```
cluster set -p global_fencing={pathcount | prefer3 | nofencing | nofencing-noscrub}
```

-p global_fencing

すべての共有デバイスの現在のグローバルなデフォルトフェンシングアルゴリズムを設定します。

prefer3

パスが 2 より多いデバイスに対して SCSI-3 プロトコルを使用します。

pathcount

共有デバイスに接続されている DID パスの数でフェンシングプロトコルを決定します。pathcount 設定は、定足数デバイスで使用されます。

nofencing

フェンシングをオフに設定します (すべてのストレージデバイスについてフェンシングステータスを設定します)。

nofencing-noscrub

ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスがクリアされ、クラスタの外側にあるシステムからストレージへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大な問題があるストレージデバイスに対してのみ使用してください。

例 49 すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定の設定

次の例では、クラスタ上のすべてのストレージデバイスのフェンシングプロトコルを、SCSI-3 プロトコルに設定します。

```
# cluster set -p global_fencing=prefer3
```

▼ 単一ストレージデバイスのフェンシングプロトコルを変更する方法

フェンシングプロトコルは、1 つのストレージデバイスに対して設定することもできます。

注記 - 定足数デバイスのデフォルトのフェンシング設定を変更するには、デバイスの構成を解除し、フェンシング設定を変更して、定足数デバイスを再構成します。フェンシングをオフに構成したあとで、定足数デバイスを含むデバイスについては定期的にオンに戻す場合は、定足数サーバーサービスを利用して定足数を構成することを検討してください (そうすることで、定足数の動作を中断せずに済みます)。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。



注意 - フェンシングを誤ってオフに設定すると、アプリケーションのフェイルオーバー時にデータの破損する可能性が生じやすくなります。フェンシングをオフに設定する場合は、そのような状況でもデータが破損しないかどうか十分に検査してください。共有ストレージデバイスが SCSI プロトコルをサポートしていない場合や、クラスタのストレージへのアクセスをクラスタ外のホストに対して許可する場合は、フェンシングをオフに設定できません。

1. 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フェンシングプロトコルを設定します。

```
cldevice set -p default_fencing ={pathcount | \  
scsi3 | global | nofencing | nofencing-noscrub} device
```

```
-p default_fencing
```

デバイスの `default_fencing` プロパティーを変更します。

```
pathcount
```

共有デバイスに接続されている DID パスの数でフェンシングプロトコルを決定します。

scsi3

SCSI-3 プロトコルを使用します。

大域 (global)

グローバルなデフォルトのフェンシング設定を使用します。global 設定は、定足数デバイス以外のデバイスで使用されます。

nofencing

指定された DID インスタンスのフェンシングステータスを設定することで、フェンシングをオフに設定します。

nofencing-noscrub

ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスがクリアされ、クラスタの外側にあるシステムからストレージデバイスへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大な問題があるストレージデバイスに対してのみ使用してください。

device

デバイスパスの名前またはデバイス名を指定します。

詳細は、[cluster\(1CL\)](#) のマニュアルページを参照してください。

例 50 単一デバイスのフェンシングプロトコルの設定

次の例では、(デバイス番号で指定される) デバイス d5 を SCSI-3 プロトコルに設定します。

```
# cldevice set -p default_fencing=prefer3 d5
```

次の例では、d11 デバイスのデフォルトフェンシングをオフに設定します。

```
#cldevice set -p default_fencing=nofencing d11
```

クラスタファイルシステムの管理

クラスタファイルシステムは、クラスタのどのノードからでも読み取りやアクセスが可能なグローバルなファイルシステムです。

表 8 タスクリスト：クラスタファイルシステムの管理

タスク	手順
Oracle Solaris Cluster の初期インストール後にクラスタファイルシステムを追加する	152 ページの「クラスタファイルシステムを追加する方法」
クラスタファイルシステムを削除する	154 ページの「クラスタファイルシステムを削除する方法」

タスク	手順
クラスタ内のグローバルマウントポイントをチェックして、ノード間の一貫性が保たれているかどうかを確認する	156 ページの「クラスタ内のグローバルマウントを確認する方法」

▼ クラスタファイルシステムを追加する方法

次のタスクは、Oracle Solaris 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で root 役割の特権が確立されています。
- 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ます。
- デバイスグループ (Solaris Volume Manager デバイスグループなど) またはブロックディスクスライスがクラスタファイルシステムの作成先に存在します。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタにクラスタファイルシステムを追加することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

Oracle Solaris 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に、クラスタファイルシステムを作成する十分な共有ディスクが存在していた場合は、1 つ以上のクラスタファイルシステムがすでに存在しています。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタの任意のノードで root 役割になります。

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プライマリノードで root 役割になります。

2. newfs コマンドを使用して UFS ファイルシステムを作成します。



注意 - ファイルシステムを作成するとき、ディスク上のデータは破壊されます。必ず、正しいディスクデバイス名を指定してください。間違ったデバイス名を指定した場合、削除するつもりのないデータが削除されてしまいます。

```
phys-schost# newfs raw-disk-device
```

次の表に、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。

ボリュームマネージャ	ディスクデバイス名の例	説明
Solaris Volume Manager	/dev/md/nfs/rdisk/d1	nfs ディスクセット内の raw ディスクデバイス d1
なし	/dev/global/rdisk/d1s3	raw ディスクデバイス d1s3

- クラスタ内の各ノードで、クラスタファイルシステムのマウントポイントのディレクトリを作成します。
そのノードからはクラスタファイルシステムにアクセスしない場合でも、マウントポイントはノードごとに必要です。

ヒント - 管理を容易にするには、マウントポイントを `/global/device-group/` ディレクトリに作成します。この場所を使用すると、グローバルに利用できるクラスタファイルシステムとローカルファイルシステムを区別しやすくなります。

```
phys-schost# mkdir -p /global/device-group/mount-point /
```

device-group

デバイスが含まれるデバイスグループ名に対応するディレクトリ名を指定します。

mount-point

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。

- クラスタ内の各ノードで、マウントポイント用の `/etc/vfstab` ファイルにエントリを追加します。詳細は、`vfstab(4)` のマニュアルページを参照してください。
 - 各エントリで、使用する種類のファイルシステムに必要なマウントオプションを指定します。
 - 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。
 - 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ブート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` がブートされ、`/global/oracle/` がマウントされたあとにのみ、`phys-schost-2` をブートし、`/global/oracle/logs/` をマウントできます。
5. クラスタの任意のノードで、構成確認ユーティリティーを実行します。

```
phys-schost# cluster check -k vfstab
```


構成確認ユーティリティーは、マウントポイントが存在することを確認します。また、`/etc/vfstab` ファイルのエントリが、クラスタのすべてのノードで正しいことを確認します。エラーが発生していない場合は、何も出力されません。
詳細は、[cluster\(1CL\)](#) のマニュアルページを参照してください。
 6. クラスタ内の任意のノードから、クラスタファイルシステムをマウントします。

```
phys-schost# mount /global/device-group/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` コマンドまたは `mount` コマンドのいずれかを使用し、マウントされたファイルシステムの一覧を表示します。詳細は、[df\(1M\)](#) のマニュアルページまたは [mount\(1M\)](#) のマニュアルページを参照してください。

▼ クラスタファイルシステムを削除する方法

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。

注記 - 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クラスタファイルシステム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で root 役割の特権が確立されています。
- ファイルシステムが使用中ではありません。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。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタファイルシステムを削除することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタの任意のノードで root 役割になります。
2. マウントされているクラスタファイルシステムを確認します。

```
# mount -v
```
3. 各ノードで、クラスタファイルシステムを使用中の全プロセスの一覧を表示し、停止するプロセスを判断します。

```
# fuser -c [ -u ] mountpoint
```

-c ファイルシステムのマウントポイントとなっているファイルと、マウントされているファイルシステム内のファイルがすべて表示されます。

-u (任意) 各プロセス ID のユーザーログイン名を表示します。

mountpoint プロセスを停止するクラスタファイルシステムの名前を指定します。
4. 各ノードで、クラスタファイルシステムのプロセスをすべて停止します。
プロセスは任意の方法で停止できます。必要であれば、次のコマンドを使用して、クラスタファイルシステムに関するプロセスを強制終了してください。

```
# fuser -c -k mountpoint
```

クラスタファイルシステムを使用している各ノードに SIGKILL が送信されます。
5. 各ノードで、ファイルシステムを使用しているプロセスがないことを確認します。

```
# fuser -c mountpoint
```
6. 1 つのノードからファイルシステムをマウント解除します。

```
# umount mountpoint
```

mountpoint マウント解除するクラスタファイルシステムの名前を指定します。クラスタファイルシステムがマウントされているディレクトリの名前や、ファイルシステムのデバイス名パスを指定できます。

7. (オプション) **/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がある各クラスタノードで実行してください。
8. (オプション) 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ドキュメントを参照してください。

例 51 クラスタファイルシステムの削除

次に、Solaris Volume Manager メタデバイスまたはボリューム **/dev/md/oracle/rdsk/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
```

各ノードで、強調表示されたエントリを削除します

```
# pfedit /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
```

保存して終了します

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ドキュメント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方法

cluster(1CL) ユーティリティーは、**/etc/vfstab** ファイル内のクラスタファイルシステムに対するエントリの構文を検証します。エラーがない場合は、何も戻されません。

注記 - クラスタファイルシステムの削除など、デバイスやボリューム管理コンポーネントに影響を及ぼすような変更をクラスタ構成に加えたあとに `cluster check` コマンドを実行します。

1. クラスタの任意のノードで `root` 役割になります。
2. クラスタのグローバルマウントを確認します。

```
# cluster check -k vfstab
```

ディスクパスモニタリングの管理

ディスクパスモニタリング (DPM) の管理コマンドを使用すれば、セカンダリディスクパス障害の通知を受け取ることができます。このセクションでは、ディスクパスのモニタリングに必要な管理タスクを行うための手順を説明します。

次の追加情報を参照できます。

トピック	情報
ディスクパスモニタリングデーモンに関する概念情報	『Oracle Solaris Cluster 4.3 Concepts Guide』の第3章、「Key Concepts for System Administrators and Application Developers」
<code>cldevice</code> コマンドオプションと関連するコマンドの説明	<code>cldevice(1CL)</code> のマニュアルページ
<code>sccdpm</code> デーモンの調整	<code>sccdpm.conf(4)</code> のマニュアルページ
<code>syslogd</code> デーモンが報告するログに記録されたエラー	<code>syslogd(1M)</code> のマニュアルページ

注記 - `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、モニタリングを行っていたモニタリングリストにディスクパスが自動的に追加されます。Oracle Solaris 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モニター対象から外されます。

表 9 タスクマップ: ディスクパスモニタリングの管理

タスク	手順
ディスクパスをモニターします。	158 ページの「ディスクパスをモニターする方法」
ディスクパスのモニターを解除します。	159 ページの「ディスクパスのモニターを解除する方法」
あるノードに対する障害のあるディスクパスのステータスを出力します。	160 ページの「障害のあるディスクパスを出力する方法」
ファイルからディスクパスをモニターします。	161 ページの「ファイルからディスクパスをモニターする方法」
モニターしているすべての共有ディスクパスが失敗したときのノードの自動リポートを有効化または無効化します。	163 ページの「モニターしているすべての共有ディスクパスが失敗したときのノードの自動リポートを有効にする方法」

タスク	手順
不正なディスクパスステータスを解決します。ブート時にモニター対象の DID デバイスを利用できず、DID インスタンスが DID ドライバにアップロードされない場合、不正なディスクパスステータスが報告されることがあります。	164 ページの「すべてのモニター共有ディスクパスが失敗した場合にノードの自動リブートを無効にする方法」 161 ページの「ディスクパスのステータスエラーを解決する方法」

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。

▼ ディスクパスをモニターする方法

このタスクは、クラスタのディスクパスをモニターするときに行います。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ディスクパスのモニタリングを有効にすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタ内の任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. ディスクパスをモニターします。

```
# cldevice monitor -n node disk
```
3. ディスクパスがモニターされているか確認します。

```
# cldevice status device
```

例 52 単一ノードのディスクパスをモニタリング

次の例では、単一ノードから `schost-1:/dev/did/rdisk/d1` ディスクパスをモニターします。ディスク `/dev/did/dsk/d1` へのパスをモニターするのは、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 53 すべてのノードのディスクパスをモニタリング

次の例では、すべてのノードから `schost-1:/dev/did/dsk/d1` ディスクパスをモニターします。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 54 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、モニター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
```

▼ ディスクパスのモニターを解除する方法

ディスクパスのモニターを解除する場合は、この手順を使用します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ディスクパスのモニタリングを無効にすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタ内の任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

2. モニター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モニターを解除します。

```
# cldevice unmonitor -n node disk
```

例 55 ディスクパスのモニタリング解除

次の例では、`schost-2:/dev/did/rdisk/d1` ディスクパスのモニターを解除し、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出力する方法

クラスタに障害のあるディスクパスを出力する場合は、次の手順を使用します。

1. クラスタの任意のノードで `root` 役割になります。
2. 全クラスタ内の障害のあるディスクパスを出力します。

```
# cldevice status -s fail
```

例 56 障害のあるディスクパスを出力する

次の例では、全クラスタ内の障害のあるディスクパスを出力します。

```
# cldevice status -s fail
```

Device Instance	Node	Status
-----	----	-----
dev/did/dsk/d4	phys-schost-1	fail

▼ ディスクパスのステータスエラーを解決する方法

次のイベントが発生すると、DPM が障害の発生したパスがオンラインになっても、そのパスのステータスを更新しない可能性があります。

- モニター対象パスの障害によって、ノードがリブートする。
- リブートしたノードがオンラインに戻るまで、モニター対象の DID パスの下のデバイスがオンラインに戻らない。

ブート時にモニター対象の DID デバイスを利用できず、このため DID インスタンスが DID ドライバにアップロードされないため、不正なディスクパスステータスが報告されます。このような状態が発生する場合は、手動で DID 情報を更新します。

1. 1 つのノードからグローバルデバイス名前空間を更新します。

```
# cldevice populate
```

2. 次の手順に進む前に、各ノードでコマンド処理が完了していることを確認します。

このコマンドは、1 つのノードからのみ実行されても、リモートからすべてのノードで実行されます。コマンドが処理を終了したかどうかを確認するには、クラスタの各ノードで次のコマンドを実行します。

```
# ps -ef | grep cldevice populate
```

3. DPM ポーリングタイムフレーム内で障害の発生したディスクパスのステータスが OK になっていることを確認します。

```
# cldevice status disk-device
```

Device Instance	Node	Status
-----	----	-----
dev/did/dsk/dN	phys-schost-1	Ok

▼ ファイルからディスクパスをモニターする方法

ファイルを使ってディスクパスをモニターしたり、そのモニターを解除する場合は、次の手順を使用します。

ファイルを使用してクラスタ構成を変更するには、まず現在の構成をエクスポートします。このエクスポート操作により XML ファイルが作成されます。このファイルは、変更する構成項目を設定するために修正できます。この手順では、このプロセス全体を説明します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタ内の任意のノードで、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. デバイスパスがモニターされるよう、構成ファイルを変更します。

モニターするデバイスパスを検索し、`monitored` 属性を `true` に設定します。

4. デバイスパスをモニターします。

```
# 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モニターされていることを確認します。

```
# cldevice status
```

例 57 ファイルからのディスクパスのモニタリング

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モニターされています。deviceconfig XML ファイルは、`phys-schost-2` と `d3` の間のパスが現在はモニターされていないことを示しています。

現在のクラスタ構成をエクスポートします

```
# cldevice export -o deviceconfig
<?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>

    monitored 属性を true に設定してパスをモニターします
<?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>

    ファイルを読み取り、モニタリングをオンにします
# cldevice monitor -i deviceconfig

    デバイスが現在モニターされていることを確認します
# cldevice status

```

参照 クラスタ構成のエクスポート、および結果の XML ファイルを使用したクラスタ構成の設定の詳細は、[cluster\(1CL\)](#) および [clconfiguration\(5CL\)](#) のマニュアルページを参照してください。

▼ モニターしているすべての共有ディスクパスが失敗したときのノードの自動リブートを有効にする方法

この機能を有効にすると、次の条件が満たされる場合、ノードは自動的にリブートします。

- ノード上ですべてのモニター対象の共有ディスクパスが失敗した。
- 少なくとも 1 つのディスクがクラスタ内の異なるノードからアクセス可能である。

ノードが再起動すると、そのノード上でマスターされているすべてのリソースグループとデバイスグループが別のノード上で再起動します。

ノードが自動リブートしたあと、ノード上のすべてのモニター対象共有ディスクパスがアクセス不能のままである場合、そのノードは再び自動リブートしません。しかし、ノードがリブート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リブートします。

`reboot_on_path_failure` プロパティを有効にすると、ローカルディスクパスの状態は、ノードのリブートが必要かどうか決定するときには考慮されません。モニターされた共有ディスクのみが影響を受けます。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用し、`reboot_on_path_failure` ノードプロパティを編集することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタ内の任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. クラスタのすべてのノードに対して、モニター共有ディスクパスがすべて失敗したときの、ノードの自動リブートを有効にします。

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

▼ すべてのモニター共有ディスクパスが失敗した場合にノードの自動リブートを無効にする方法

この機能を無効にすると、あるノード上のすべてのモニター共有ディスクパスに障害が発生しても、ノードは自動的にリポートしません。

1. クラスタ内の任意のノードで、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モニター共有ディスクパスに障害が発生した場合の、ノードの自動リブートを無効にします。

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

定足数の管理

この章では、Oracle Solaris Cluster および Oracle Solaris Cluster 定足数サーバー内の定足数デバイスの管理手順について説明します。定足数の概念については、『[Oracle Solaris Cluster 4.3 Concepts Guide](#)』の「[Quorum and Quorum Devices](#)」を参照してください。

- [165 ページの「定足数デバイスの管理」](#)
- [184 ページの「Oracle Solaris Cluster 定足数サーバーの管理」](#)

定足数デバイスの管理

定足数デバイスとは、複数のノードによって共有される共有ストレージデバイスまたは定足数サーバーで、定足数を確立するために使用される票を構成します。このセクションでは、定足数デバイスを管理するための手順について説明します。

`clquorum` コマンドを使用すると、定足数デバイスの管理手順をすべて実行できます。また、`clsetup` 対話型ユーティリティまたは Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、いくつかの手順を行うことができます。Oracle Solaris Cluster Manager のログイン手順については、『[Oracle Solaris Cluster 4.3 システム管理](#)』の「[Oracle Solaris Cluster Manager にアクセスする方法](#)」を参照してください。

このセクションの管理手順は、可能なかぎり `clsetup` ユーティリティを使用して説明してあります。Oracle Solaris Cluster Manager オンラインヘルプでは、Oracle Solaris Cluster Manager を使用して定足数の手順を実行する方法を説明しています。詳細は、[clquorum\(1CL\)](#) および [clsetup\(1CL\)](#) のマニュアルページ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。

- 定足数コマンドはすべて、グローバルクラスタノードから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、Oracle 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 ZFS ストレージプールに追加すると、ディスクは EFI ディスクとしてラベルが変更され、また定足数構成情報が失われ、ディスクはクラスタへの定足数投票を提供しなくなります。ディスクがストレージプール内に入ると、そのディスクは定足数デバイスとして構成できます。または、ディスクの定足数デバイス構成を解除し、ディスクをストレージプールに追加した後に、そのディスクを定足数デバイスとして再構成することができます。

注記 - `clsetup` コマンドは、ほかの Oracle Solaris 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、ステータスと投票数の情報を提供します。

このセクションで示す例は、主に 3 ノードクラスタです。

表 10 タスクリスト：定足数の管理

タスク	説明
<code>clsetup</code> ユーティリティーを使用して、クラスタに定足数デバイスを追加する	167 ページの「定足数デバイスの追加」
<code>clsetup</code> ユーティリティーを使用する (<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	174 ページの「定足数デバイスを削除する方法」
<code>clsetup</code> ユーティリティーを使用する (<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	175 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方法」
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	176 ページの「定足数デバイスを交換する方法」
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	177 ページの「定足数デバイスのノードリストを変更する方法」
<code>clsetup</code> ユーティリティーを使用する (<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 (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。)	178 ページの「定足数デバイスを保守状態にする方法」
<code>clsetup</code> ユーティリティーを使用して <code>clquorum</code> 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	180 ページの「定足数デバイスを保守状態から戻す」
<code>clquorum</code> 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	181 ページの「クラスタ構成を一覧表示する方法」

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操作を実行する場合は、いくつかの問題を考慮する必要があります。

- Oracle Solaris 動的再構成機能に関して記載されている要件、手順、および制限のすべてが、Oracle Solaris Cluster 動的再構成のサポートにも適用されます (オペレーティングシステムの休止操作を除く)。そのため、Oracle Solaris Cluster ソフトウェアで動的再構成機能を使用する前に、Oracle Solaris 動的再構成機能のドキュメントを確認してください。特に、動的再構成の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- Oracle Solaris 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ときに実行される動的再構成ボード削除操作を拒否します。
- 動的再構成操作がアクティブなデバイスに関連する場合、Oracle Solaris Cluster はその操作を拒否し、その操作の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、次の手順を示されている順番で実行する必要があります。

表 11 タスクマップ: 定足数デバイスへの動的再構成

タスク	説明
1. 削除する定足数デバイスと交換する、新しい定足数デバイスを有効に設定	167 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	174 ページの「定足数デバイスを削除する方法」
3. 削除されるデバイス上で動的再構成削除操作を実行します。	

定足数デバイスの追加

このセクションでは、定足数デバイスを追加する手順について説明します。クラスタのすべてのノードがオンラインか確認してから、新しい定足数デバイスを追加します。クラスタに必要な定足数投票数の決定、推奨される定足数構成、および障害フェンシングについては、『[Oracle Solaris Cluster 4.3 Concepts Guide](#)』の「[Quorum and Quorum Devices](#)」を参照してください。



注意 - 現在定足数デバイスとして構成されているディスクは、Solaris ZFS ストレージプールには追加しないでください。構成済みの定足数デバイスを Solaris ZFS ストレージプールに追加すると、ディスクは EFI ディスクとしてラベルが変更され、また定足数構成情報が失われ、ディスクはクラスタへの定足数投票を提供しなくなります。ディスクがストレージプール内に入ると、そのディスクは定足数デバイスとして構成できます。ディスクの定足数デバイス構成を解除し、ディスクをストレージプールに追加したあとに、そのディスクを定足数デバイスとして再構成することもできます。

Oracle Solaris Cluster ソフトウェアは、次の種類の定足数デバイスをサポートしています。

- 以下の共有 LUN
 - 共有 SCSI ディスク
 - Serial Attached Technology Attachment (SATA) ストレージ
 - OracleZFS Storage Appliance
 - サポートされている NAS デバイス
- Oracle Solaris Cluster Quorum Server

これらのデバイスを追加する方法については、次のセクションで説明しています。

- [169 ページの「共有ディスク定足数デバイスを追加する方法」](#)
- [171 ページの「定足数サーバー定足数をデバイスとして追加する方法」](#)

注記 - レプリケート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。レプリケートされたディスクを定足数デバイスとして追加しようとすると、次のエラーメッセージが表示され、コマンドはエラーコードとともに終了します。

Disk-name is a replicated device. Replicated devices cannot be configured as quorum devices.

共有ディスク定足数デバイスは、Oracle Solaris Cluster ソフトウェアがサポートする任意の接続済みストレージデバイスです。共有ディスクは、クラスタの複数のノードに接続されます。フェンシングをオンに構成すると、デュアルポートのディスクを定足数デバイスとして構成して、SCSI-2 または SCSI-3 (デフォルトは SCSI-2) を使用できます。フェンシングがオンに構成され、共有デバイスが 3 つ以上のノードに接続されている場合は、SCSI-3 プロトコル (2 ノードを超える場合のデフォルトのプロトコル) を使用する定足数デバイスとして共有ディスクを構成できます。SCSI オーバーライドフラグを使用すると、デュアルポートの共有ディスクで SCSI-3 プロトコルを使用するように Oracle Solaris Cluster ソフトウェアに対して指示できます。

共有ディスクのフェンシングをオフに構成した場合は、ソフトウェア定足数プロトコルを使用する定足数デバイスとしてディスクを構成できます。これは、そのディスクが SCSI-2 と SCSI-3 のどちらのプロトコルをサポートしている場合でも有効です。ソフトウェアの定足数は、SCSI Persistent Group Reservations (PGR) のフォームをエミュレートする、Oracle のプロトコルです。



注意 - 使用するディスクが SCSI (SATA など) をサポートしていない場合は、SCSI フェンシングをオフにするようにしてください。

定足数デバイスには、ユーザーデータが含まれているディスク、またはデバイスグループのメンバーであるディスクを使用できます。共有ディスクがある定足数サブシステムで使用されているプロトコルは、`cluster show` コマンドの出力の、共有ディスクの `access-mode` 値で確認します。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、定足数サーバーデバイスまたは共有ディスク定足数デバイスを作成することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

次の手順で使用されるコマンドについては、[clsetup\(1CL\)](#) および [clquorum\(1CL\)](#) のマニュアルページを参照してください。

▼ 共有ディスク定足数デバイスを追加する方法

Oracle Solaris Cluster ソフトウェアでは、共有ディスク (SCSI と SATA の両方) デバイスを定足数デバイスとして使用できます。SATA デバイスは SCSI 予約をサポートしていないため、その種類のディスクを定足数デバイスとして構成するには、SCSI 予約フェンシングフラグをオフに構成し、ソフトウェア定足数プロトコルを使用します。

この手順を完了するには、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特定します。`cldevice show` コマンドを使用して、DID 名の一覧を参照します。詳細は、[cldevice\(1CL\)](#) のマニュアルページを参照してください。クラスタのすべてのノードがオンラインか確認してから、新しい定足数デバイスを追加します。

次の手順を実行して、SCSI または SATA デバイスを構成します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタの任意のノードで、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の番号を入力します。
「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の番号を入力し、追加する定足数デバイスを確認する `clsetup` ユーティリティのプロンプトが表示されたら `yes` と入力します。
追加する定足数デバイスの種類を確認するメッセージが表示されます。
5. 共有ディスク定足数デバイスのオプションの番号を入力します。
どのグローバルデバイスを使用するかを確認するメッセージが表示されます。
6. 使用しているグローバルデバイスを入力します。
指定したグローバルデバイスに新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。

7. 「yes」と入力し、新しい定足数デバイスの追加を続行します。
新しい定足数デバイスが正常に追加されると、clsetup ユーティリティーではその旨のメッセージが表示されます。
8. 定足数デバイスが追加されていることを確認します。

```
# clquorum list -v
```

▼ Oracle ZFS Storage Appliance NAS の定足数デバイスを追加する方法

クラスタ内のすべてのノードがオンラインであることを確認してから、新しい定足数デバイスを追加します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して Oracle ZFS Storage Appliance NAS デバイスを追加することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. iSCSI デバイスのセットアップ手順については、OracleZFS Storage Appliance に付属のインストールドキュメントまたはアプライアンスのオンラインヘルプを参照してください。
2. 各クラスタノードで、iSCSI LUN を検出して、iSCSI アクセスリストを静的構成に設定します。

```
# iscsiadm modify discovery -s enable
```

```
# iscsiadm list discovery
```

```
Discovery:
```

```
Static: enabled
```

```
Send Targets: disabled
```

```
iSNS: disabled
```

```
# iscsiadm add static-config iqn.LUN-name,IP-address-of-NASdevice
```

```
# devfsadm -i iscsi
```

```
# cldevice refresh
```

3. 1 つのクラスタノードから DID を iSCSI LUN 用に構成します。

cldevice populate
4. iSCSI を使用するクラスタに構成した NAS デバイス LUN を表す DID デバイスを特定します。

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。

5. クラスタの任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
6. `clquorum` コマンドを使用して、[ステップ 4](#) で特定した DID デバイスを使用する定足数デバイスとして NAS デバイスを追加します。

```
# clquorum add d20
```

クラスタには、scsi-2、scsi-3、またはソフトウェア定足数プロトコルのどれを使用するかを判断するためのデフォルトのルールがあります。詳細は、[clquorum\(1CL\)](#) のマニュアルページを参照してください。

▼ 定足数サーバー定足数をデバイスとして追加する方法

始める前に

Oracle Solaris Cluster 定足数サーバーを定足数デバイスとして追加する前に、Oracle Solaris Cluster 定足数サーバーソフトウェアがホストマシンにインストールされていて、定足数サーバーが起動および実行されている必要があります。定足数サーバーのインストールについては、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[Oracle Solaris Cluster Quorum Server ソフトウェアをインストールおよび構成する方法](#)」を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、定足数サーバーデバイスを作成することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページ](#)の「[Oracle Solaris Cluster Manager にアクセスする方法](#)」を参照してください。

1. クラスタの任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. すべての Oracle Solaris Cluster ノードがオンラインであり、Oracle Solaris Cluster 定足数サーバーと通信できることを確認します。
 - a. クラスタノードに直接接続されているネットワークスイッチが次の基準のいずれかを満たすことを確認します。
 - スイッチは RSTP (Rapid Spanning Tree Protocol) をサポートしています。
 - スイッチ上で高速ポートモードが有効になっています。

クラスタノードと定足数サーバー間ですぐに通信できるようにするには、これらの機能の 1 つが必要です。この通信がスイッチによって大幅に遅延すると、クラスタはこの通信の中断を定足数デバイスが失われたものと解釈します。

- b. パブリックネットワークで可変長サブネット化 (CIDR (Classless Inter-Domain Routing) とも呼ばれる) を使用している場合は、各ノードで次のファイルを変更します。クラスフルサブネットを使用する場合は、これらの手順を実行する必要はありません。

- i. `/etc/inet/netmasks` ファイルにクラスタが使用する各パブリックサブネットのエントリを追加します。

パブリックネットワークの IP アドレスとネットマスクを含むエントリの例を次に示します。

```
10.11.30.0 255.255.255.0
```

- ii. それぞれの `/etc/hostname.adapter` ファイルに `netmask + broadcast +` を追加します。

```
nodename netmask + broadcast +
```

- c. クラスタ内の各ノード上で、定足数サーバーのホスト名を `/etc/inet/hosts` ファイルまたは `/etc/inet/ipnodes` ファイルに追加します。

次のように、ホスト名とアドレスのマッピングをファイルに追加します。

```
ipaddress qshost1
```

`ipaddress` 定足数サーバーが実行中であるコンピュータの IP アドレス。

`qshost1` 定足数サーバーが実行中であるコンピュータのホスト名。

- d. ネームサービスを使用する場合、定足数サーバーホストの名前とアドレスの対応付けをネームサービスデータベースに追加します。

3. `clsetup` ユーティリティを起動します。

```
# 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の番号を入力します。

「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の番号を入力します。

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. **定足数サーバー上の定足数デバイスのオプションの番号を入力したあと、「yes」と入力して、定足数サーバー上の定足数デバイスを追加することを確認します。**

新しい定足数デバイスの名前を入力するように、`clsetup` ユーティリティのプロンプトが表示されます。

7. **追加する定足数デバイスの名前を入力します。**

定足数デバイスの名前は任意に選択できます。この名前は、今後の管理コマンドの処理だけに使用されるものです。

定足数サーバーのホスト名を入力するように、`clsetup` ユーティリティのプロンプトが表示されます。

8. **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。

注記 - 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、`clsetup` ユーティリティではその旨のメッセージが表示されます。

11. **定足数デバイスが追加されていることを確認します。**

```
# clquorum list -v
```

定足数デバイスの削除または交換

このセクションでは、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。

- [174 ページの「定足数デバイスを削除する方法」](#)

- [175 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方法」](#)
- [176 ページの「定足数デバイスを交換する方法」](#)

▼ 定足数デバイスを削除する方法

定足数デバイスが削除されると、そのデバイスは定足数を確立するための投票に参加しなくなります。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。これがクラスタの最後の定足数デバイスである場合、`clquorum(1CL)` はそのデバイスを構成から削除できません。ノードを削除する場合は、そのノードに接続されている定足数デバイスをすべて削除してください。

注記 - 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、[175 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方法」](#)の手順を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、定足数デバイスを削除することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタ内の任意のノードで、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` 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の番号を入力します。
5. 定足数デバイスを削除するためのオプションの番号を入力します。
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
```

注意事項 定足数サーバー定足数デバイスの削除中に、クラスタと定足数サーバーホストの間の通信が失われた場合、定足数サーバーホストに関する無効な構成情報をクリーンアップする必要があります。このクリーンアップの実行に関する説明は、[187 ページの「期限切れの定足数サーバークラスタ情報のクリーンアップ」](#)を参照してください。

▼ クラスタから最後の定足数デバイスを削除する方法

この手順では、`clquorum force` のオプション `-F` を使用して、2 ノードクラスタから最後の定足数デバイスを削除します。通常、不具合が起きたデバイスをまず削除し、代替りの定足数デバイスを追加します。これが 2 ノードクラスタの最後の定足数デバイスでない場合は、[174 ページの「定足数デバイスを削除する方法」](#)の手順に従ってください。

定足数デバイスを追加する処理では、ノードが再構成されるため、障害のあった定足数デバイスに影響が及び、マシンでパニックが発生します。`F` (強制) オプションを使用すると、マシンでパニックを発生させることなく、障害があった定足数デバイスを削除できます。`clquorum` コマンドでは、構成からデバイスを削除できます。詳細は、[clquorum\(1CL\)](#) のマニュアルページを参照してください。不具合が発生した定足数デバイスを削除したあと、`clquorum add` コマンドで新しいデバイスを追加することができます。[167 ページの「定足数デバイスの追加」](#)を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタ内の任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

2. `clquorum` コマンドを使用して定足数デバイスを削除します。

定足数デバイスに障害が発生した場合は、`-F` (強制) オプションを使用して、障害が発生したデバイスを削除します。

```
# clquorum remove -F qd1
```

注記 - また、削除するノードを保守状態とし、定足数デバイスを `clquorum remove quorum` コマンドを使用して削除することもできます。`clsetup` クラスタ管理メニューオプションは、クラスタがインストールモードのときは使用できません。詳細は、[237 ページの「ノードを保守状態にする」](#)および [clsetup\(1CL\)](#) のマニュアルページを参照してください。

3. 定足数デバイスが削除されたことを確認します。

```
# clquorum list -v
```

4. 定足数デバイスを削除する理由に応じて、次のいずれかの手順に進みます。

■ 削除された定足数デバイスを交換する場合は、次のサブステップを完了します。

a. 新しい定足数デバイスを追加します。

新しい定足数デバイスの追加方法については、[167 ページの「定足数デバイスの追加」](#)を参照してください。

b. クラスタをインストールモードから削除します。

```
# cluster set -p installmode=disabled
```

■ クラスタを単一ノードクラスタに減らす場合は、クラスタをインストールモードから削除します。

```
# cluster set -p installmode=disabled
```

例 58 最後の定足数デバイスの削除

この例では、クラスタを保持モードにし、クラスタ構成で最後の定足数デバイスを削除する方法を示しています。

クラスタをインストールモードにします

```
# cluster set -p installmode=enabled
```

定足数デバイスを削除します

```
# clquorum remove d3
```

定足数デバイスが削除されたことを確認します。

```
# clquorum list -v
Quorum      Type
-----
scphyshost-1 node
scphyshost-2 node
scphyshost-3 node
```

▼ 定足数デバイスを交換する方法

この作業は、既存の定足数デバイスをほかの定足数デバイスに交換する場合に行います。定足数デバイスは、類似したデバイスタイプに交換することも (例: NAS デバイスをほかの NAS デバイスに置き換える)、あるいは類似点がないデバイスに交換することも (例: NAS デバイスを共有ディスクに置き換える) こともできます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

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、[167 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、[174 ページの「定足数デバイスを削除する方法」](#)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。**
ディスク格納装置のハードウェアマニュアルのハードウェア手順を参照してください。『[Oracle Solaris Cluster Hardware Administration Manual](#)』も参照してください。

定足数デバイスの保守

このセクションでは、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。

- [177 ページの「定足数デバイスのノードリストを変更する方法」](#)
- [178 ページの「定足数デバイスを保守状態にする方法」](#)
- [180 ページの「定足数デバイスを保守状態から戻す」](#)
- [181 ページの「クラスタ構成を一覧表示する方法」](#)
- [182 ページの「定足数デバイスを修復する方法」](#)
- [183 ページの「定足数のデフォルトのタイムアウトの変更」](#)

▼ 定足数デバイスのノードリストを変更する方法

`clsetup` コーティリティーを使用すると、既存の定足数デバイスのノードリストにノードを追加したり、そこからノードを削除したりできます。定足数デバイスのノードリストを変更するには、定足数デバイスを削除し、削除した定足数デバイスへのノードの物理的な接続を変更して、定足数デバイスをクラスタ構成に追加し直す必要があります。定足数デバイスが追加されると、`clquorum` コマンドによって、そのディスクに接続されているすべてのノードに対応する、ノードからディスクへのパスが自動的に構成されます。詳細は、[clquorum\(1CL\)](#) のマニュアルページを参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタの任意のノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

2. 変更したい定足数デバイスの名前を判別します。

```
# clquorum list -v
```

3. `clsetup` ユーティリティーを起動します。

```
# clsetup
```

メインメニューが表示されます。

4. 定足数のオプションの番号を入力します。

「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ためのオプションの番号を入力します。

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ためのオプションの番号を入力します。

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
```

▼ 定足数デバイスを保守状態にする方法

定足数デバイスを保守状態にするには、`clquorum` コマンドを使用します。詳細は、[clquorum\(1CL\)](#) のマニュアルページを参照してください。現在、`clsetup` ユーティリティーにこの機能はありません。

定足数デバイスを長期間にわたって非稼働状態にする場合は、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。

注記 - 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` 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、[237 ページの「ノードを保守状態にする」](#)を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、定足数デバイスを無効にし、保守状態にすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

クラスタがインストールモードにある場合は、「定足数デバイスのリセット」をクリックしてインストールモードを終了します。

1. クラスタの任意のノードで、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 59 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、[180 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、[237 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

定足数デバイスが保守状態にあるときに、その定足数デバイスを保守状態から戻し、定足数投票数をデフォルト値にリセットする場合は常にこの手順を実行します。



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、Oracle Solaris Cluster ソフトウェアは定足数デバイスに投票数として $N-1$ を割り当てます (N は定足数デバイスに結合された投票の数)。たとえば、2つのノードに接続された、投票数がゼロ以外の定足数デバイスの投票数は 1 (2マイナス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、[239 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、『[Oracle Solaris Cluster 4.3 Concepts Guide](#)』の「[About Quorum Vote Counts](#)」を参照してください。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、定足数デバイスを有効にし、保守状態から戻すこともできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタの任意のノードで、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、このノードをリポートします。
4. 定足数投票数を確認します。

```
# clquorum show +
```

例 60 定足数投票数 (定足数デバイス) のリセット

次に、定足数デバイスの投票数をリセットしてデフォルト設定に戻し、結果を検証する例を示します。

```
# clquorum enable d20
# clquorum show +
```

```
=== Cluster Nodes ===
```

```
Node Name:                phys-schost-2
Node ID:                   1
Quorum Vote Count:        1
Reservation Key:          0x43BAC41300000001
```

```
Node Name:                phys-schost-3
Node ID:                   2
Quorum Vote Count:        1
Reservation Key:          0x43BAC41300000002
```

```
=== Quorum Devices ===
```

```
Quorum Device Name:      d3
Enabled:                 yes
Votes:                   1
Global Name:             /dev/did/rdisk/d20s2
Type:                    shared_disk
Access Mode:             scsi3
Hosts (enabled):        phys-schost-2, phys-schost-3
```

▼ クラスタ構成を一覧表示する方法

定足数構成の一覧を表示するには、root 役割であ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。

注記 - 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、正しい定足数が再構成されます。2 ノードクラスタの場合、定足数デバイスを取り外して、もとの定足数デバイスに戻す前に一時的に新しい定足数デバイスを追加します。次に一時的に追加した定足数デバイスを取り外します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して定足数構成を表示することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

- **clquorum** コマンドを使用して、定足数の構成を一覧表示します。

```
% clquorum show +
```

例 61 定足数構成の一覧表示

```
% clquorum show +
```

```
=== Cluster Nodes ===
```

```
Node Name:                phys-schost-2
Node ID:                   1
Quorum Vote Count:        1
Reservation Key:           0x43BAC41300000001
```

```
Node Name:                phys-schost-3
Node ID:                   2
Quorum Vote Count:        1
Reservation Key:           0x43BAC41300000002
```

```
=== Quorum Devices ===
```

```
Quorum Device Name:      d3
Enabled:                  yes
Votes:                    1
Global Name:              /dev/did/rdisk/d20s2
Type:                     shared_disk
Access Mode:              scsi3
Hosts (enabled):         phys-schost-2, phys-schost-3
```

▼ 定足数デバイスを修復する方法

この手順は、動作が不正な定足数デバイスを交換する場合に使用します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

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。

注記 - 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、[167 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、[174 ページの「定足数デバイスを削除する方法」](#)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場合は、ハードウェアガイドのディスク格納装置の手順を参照してください。『[Oracle Solaris Cluster Hardware Administration Manual](#)』も参照してください。

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、[167 ページの「定足数デバイスの追加」](#)を参照してください。

注記 - [ステップ 1](#) で追加の定足数デバイスを追加した場合は、それ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、[174 ページの「定足数デバイスを削除する方法」](#)を参照してください。

定足数のデフォルトのタイムアウトの変更

クラスタ再構成時の定足数の操作を完了するまでのタイムアウトは、デフォルトで 25 秒に構成されています。定足数タイムアウト値は、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[定足数デバイスを構成する方法](#)」の手順に従って増やすことができます。タイムアウト値を増分する代わりに、別の定足数デバイスに切り替えるという方法もあります。

その他のトラブルシューティング情報については、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[定足数デバイスを構成する方法](#)」を参照してください。

注記 - Oracle Real Application Clusters (Oracle RAC) では、デフォルトの定足数タイムアウトである 25 秒を変更しないでください。一部のスプリットプレーンシナリオでは、タイムアウト時間を長くすると、VIP リソースのタイムアウトが原因で Oracle RAC VIP フェイルオーバーが失敗する可能性があります。

使用している定足数デバイスがデフォルトの 25 秒のタイムアウトに適合しない場合は、別の定足数デバイスを使用してください。

Oracle Solaris Cluster 定足数サーバーの管理

Oracle Solaris Cluster Quorum Server は、共有ストレージデバイスではない、定足数デバイスを提供します。このセクションでは、Oracle Solaris Cluster 定足数サーバーを管理するための次のような手順について説明します。

- [184 ページの「Quorum Server Software の起動および停止」](#)
- [184 ページの「定足数サーバーを起動する方法」](#)
- [185 ページの「定足数サーバーを停止する方法」](#)
- [186 ページの「定足数サーバーに関する情報の表示」](#)
- [187 ページの「期限切れの定足数サーバークラスタ情報のクリーンアップ」](#)

Oracle Solaris Cluster 定足数サーバーのインストールおよび構成については、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[Oracle Solaris Cluster Quorum Server ソフトウェアをインストールおよび構成する方法](#)」を参照してください。

Quorum Server Software の起動および停止

次の手順では、Oracle Solaris Cluster ソフトウェアを起動および停止する方法を説明します。

デフォルトでは、次の手順は、定足数サーバー構成ファイル `/etc/scqsd/scqsd.conf` の内容をカスタマイズしていない場合の、1 つのデフォルト定足数サーバーを起動および停止します。デフォルトの定足数サーバーはポート 9000 上にバインドされ、定足数情報には `/var/scqsd` ディレクトリを使用します。

定足数サーバーソフトウェアのインストールについては、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[Oracle Solaris Cluster Quorum Server ソフトウェアをインストールおよび構成する方法](#)」を参照してください。定足数タイムアウトの値の変更については、[183 ページの「定足数のデフォルトのタイムアウトの変更」](#)を参照してください。

▼ 定足数サーバーを起動する方法

1. Oracle Solaris Cluster ソフトウェアを起動するホスト上で `root` 役割になります。
2. ソフトウェアを起動するには、`clquorumserver start` コマンドを使用します。

```
# clquorumserver start quorumserver
```

`quorumserver` 定足数サーバーを識別します。定足数サーバーが待機するポート番号を使用できます。構成ファイルでインスタンス名を指定した場合は、代わりにその名前を使用できます。

1 台の定足数サーバーを起動するには、インスタンス名とポート番号のいずれかを指定します。複数の定足数サーバーを構成している場合、すべての定足数サーバーを起動するには、+ オペランドを使用します。

例 62 すべての構成済み定足数サーバーの起動

次の例では、構成されているすべての定足数サーバーを起動します。

```
# clquorumserver start +
```

例 63 特定の定足数サーバーの起動

次の例では、ポート番号 2000 で待機している定足数サーバーを起動します。

```
# clquorumserver start 2000
```

▼ 定足数サーバーを停止する方法

1. Oracle Solaris Cluster ソフトウェアを起動するホスト上で root 役割になります。
2. ソフトウェアを停止するには、`clquorumserver stop` コマンドを使用します。

```
# clquorumserver stop [-d] quorumserver
```

`-d` マシンを次回起動したときに、定足数サーバーを起動するかどうかを制御します。`-d` オプションを指定すると、次のマシン起動時に定足数サーバーは起動しません。

`quorumserver` 定足数サーバーを識別します。定足数サーバーが待機するポート番号を使用できます。構成ファイルでインスタンス名を指定した場合は、代わりにその名前を使用できます。

1 台の定足数サーバーを停止するには、インスタンス名とポート番号のいずれかを指定します。複数の定足数サーバーを構成している場合、すべての定足数サーバーを停止するには、+ オペランドを使用します。

例 64 すべての構成済み定足数サーバーの停止

次の例では、構成されているすべての定足数サーバーを停止します。

```
# clquorumserver stop +
```

例 65 特定の定足数サーバーの停止

次の例では、ポート番号 2000 で待機している定足数サーバーを停止します。

```
# clquorumserver stop 2000
```

定足数サーバーに関する情報の表示

定足数サーバーについての構成情報を表示することができます。このコマンドは、定足数サーバーを定足数デバイスとして構成しているすべてのクラスタごとに、対応するクラスタ名、クラスタ ID、予約鍵のリスト、および登録鍵のリストを表示します。

▼ 定足数サーバーに関する方法情報を表示する方法

1. 定足数サーバーの情報を表示するホスト上で **root** 役割になります。

root 役割以外のユーザーには、役割に基づくアクセス制御 (RBAC) の承認 `solaris.cluster.read` が必要です。RBAC 権利プロファイルの詳細は、[rbac\(5\)](#) のマニュアルページを参照してください。

2. `clquorumserver` コマンドを使用することで、定足数サーバーの構成情報を表示します。

```
# clquorumserver show quorumserver
```

```
quorumserver      1 つまたは複数の定足数サーバーを識別します。インスタンス名または
                  ポート番号で定足数サーバーを指定できます。すべての定足数サーバーの
                  構成情報を表示するには、+ オペランドを使用します。
```

例 66 1 つの定足数サーバーの構成の表示

次の例では、ポート 9000 を使用する定足数サーバーの構成情報を表示します。次のコマンドは、定足数サーバーが定足数デバイスとして構成されているすべてのクラスタの情報を表示します。この情報にはクラスタの名前と ID、およびデバイスの予約鍵と登録鍵のリストが含まれます。

次の例では、クラスタ `bastille` の ID が 1、2、3、および 4 であるノードが、定足数サーバー上に鍵を登録しています。また、ノード 4 は定足数デバイスの予約を所有しているため、その鍵は予約リストに表示されます。

```
# clquorumserver show 9000
```

```
=== Quorum Server on port 9000 ===
```

```
--- Cluster bastille (id 0x439A2EFB) Reservation ---
```

```
Node ID:                4
Reservation key:         0x439a2efb00000004
```

```
--- Cluster bastille (id 0x439A2EFB) Registrations ---
```

```
Node ID:                1
```

```

Registration key:      0x439a2efb00000001

Node ID:               2
Registration key:      0x439a2efb00000002

Node ID:               3
Registration key:      0x439a2efb00000003

Node ID:               4
Registration key:      0x439a2efb00000004

```

例 67 複数の定足数サーバーの構成の表示

次の例では、3 つの定足数サーバー qs1、qs2、および qs3 の構成情報を表示します。

```
# clquorumserver show qs1 qs2 qs3
```

例 68 動作しているすべての定足数サーバーの構成の表示

次の例では、動作しているすべての定足数サーバーの構成情報を表示します。

```
# clquorumserver show +
```

期限切れの定足数サーバークラスタ情報のクリーンアップ

quorumserver のタイプの定足数デバイスを削除するには、How to Remove a Quorum Device で説明されているように、174 ページの「定足数デバイスを削除する方法」コマンドを使用します。通常の動作では、このコマンドは定足数サーバーホストに関する定足数サーバーの情報も削除します。ただし、クラスタが定足数サーバーホストとの通信を失うと、定足数デバイスを削除しても、この情報がクリーンアップされません。

定足数サーバークラスタ情報は、次の状況で無効になります。

- clquorum remove コマンドを使用してクラスタ定足数デバイスを削除せずに、クラスタの運用を停止した場合。
- 定足数サーバーホストが停止している間に、quorum_server タイプの定足数デバイスをクラスタから削除した場合。



注意 - タイプ quorumserver の定足数デバイスがまだクラスタから削除されていない場合、この手順を使用して無効な定足数サーバーを削除すると、クラスタ定足数に障害が発生する可能性があります。

▼ 定足数サーバーの構成情報をクリーンアップする方法

始める前に 174 ページの「定足数デバイスを削除する方法」で説明されているとおりに、定足数サーバーの定足数デバイスを削除します。



注意 - クラスタがまだこの定足数サーバーを使用している場合、この手順を実行するとクラスタ定足数に障害が発生します。

1. 定足数サーバーホスト上で **root** 役割になります。
2. **clquorumserver clear** コマンドを使用して、構成ファイルをクリーンアップします。

```
# clquorumserver clear -c clustername -I clusterID quorumserver [-y]
```

-c clustername 以前に定足数サーバーを定足数デバイスとして使用していたクラスタの名前です。
クラスタ名を取得するには、クラスタノード上で **cluster show** を実行します。

-I clusterID クラスタ ID です。
クラスタ ID は 8 桁の 16 進数です。クラスタ ID を取得するには、クラスタノード上で **cluster show** を実行します。

quorumserver 1 つまたは複数の定足数サーバーの識別子です。
定足数サーバーは、ポート番号かインスタンス名で識別できます。ポート番号は、クラスタノードが定足数サーバーと通信するために使用されます。インスタンス名は、定足数サーバーの構成ファイル `/etc/scqsd/scqsd.conf` で指定されます。

-y 実行前に確認のプロンプトを表示することなく、**clquorumserver clear** コマンドに、構成ファイルからクラスタ情報をクリーン アップさせます。
期限切れのクラスタ情報を定足数サーバーから削除したいことが確かである場合のみ、このオプションを使用します。

3. (オプション) このサーバーインスタンスでほかに定足数デバイスが構成されていない場合は、定足数サーバーを停止します。

例 69 定足数サーバー構成からの期限切れのクラスタ情報のクリーンアップ

この例は、`sc-cluster` という名前のクラスタについての情報を、ポート 9000 を使用する定足数サーバーから削除します。

```
# clquorumserver clear -c sc-cluster -I 0x4308D2CF 9000
The quorum server to be unconfigured must have been removed from the cluster.
Unconfiguring a valid quorum server could compromise the cluster quorum. Do you
want to continue? (yes or no) y
```

クラスタインターコネクとパブリックネットワークの管理

この章では、Oracle Solaris Cluster インターコネクとパブリックネットワークを管理するためのソフトウェア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、はじめてクラスタをインストールおよび構成するときには、すべてのパブリックネットワーク管理 (PNM) オブジェクト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。

PNM オブジェクトには、インターネットプロトコルネットワークマルチパス (IPMP) グループ、トランクおよびデータリンクマルチパス (DLMP) リンクアグリゲーション、およびリンクアグリゲーションに直接基づく VNIC が含まれます。マルチパスは Oracle Solaris 11 OS で自動的にインストールされますが、使用するには有効にする必要があります。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、[204 ページの「パブリックネットワークの管理」](#)のセクションを参照してください。IPMP については、『[Oracle Solaris 11.3 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理](#)』の第 3 章、「[IPMP の管理](#)」を参照してください。リンクアグリゲーションについては、『[Oracle Solaris 11.3 でのネットワークデータリンクの管理](#)』の第 2 章、「[リンクアグリゲーションを使用した高可用性の構成](#)」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。

- [190 ページの「クラスタインターコネクの管理」](#)
- [204 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については、[表12「タスクリスト: クラスタインターコネクの管理」](#)を参照してください。

クラスタインターコネクおよびパブリックネットワークの背景情報および概要情報については、『[Oracle Solaris Cluster 4.3 Concepts Guide](#)』を参照してください。

クラスタインターコネクットの管理

このセクションでは、クラスタトランスポートアダプタ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ットを再構成する手順について説明します。これらの手順では、Oracle Solaris Cluster ソフトウェアがインストールされている必要があります。

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ットのクラスタトランスポートを管理できます。詳細は、[clsetup\(1CL\)](#) のマニュアルページを参照してください。クラスタインターコネクットコマンドはすべて、グローバルクラスタノードから実行する必要があります。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用してこれらのタスクの一部を実行することもできます。ログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』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の保守手順については、『[Oracle Solaris Cluster Hardware Administration Manual](#)』を参照してください。

注記 - 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。

表 12 タスクリスト: クラスタインターコネクットの管理

タスク	手順
<code>clsetup(1CL)</code> を使用することで、クラスタトランスポートを管理する	32 ページの「クラスタ構成コーティリティーにアクセスする方法」
<code>clinterconnect status</code> を使用することで、クラスタインターコネクットのステータスを確認する	191 ページの「クラスタインターコネクットのステータスを確認する方法」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブル、トランスポートアダプタ、またはスイッチを追加する	192 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、またはトランスポートスイッチを追加する方法」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブル、トランスポートアダプタ、またはトランスポートスイッチを削除する	194 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、およびトランスポートスイッチを削除する方法」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブルを有効にする	197 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方法」
<code>clsetup</code> を使用することで、クラスタトランスポートケーブルを無効にする	198 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方法」
トランスポートアダプタのインスタンス番号の確認	199 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方法」
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	200 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方法」

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) を完了するときには、いくつかの問題を考慮する必要があります。

- Oracle Solaris 動的再構成機能に関して記載されている要件、手順、および制限のすべてが、Oracle Solaris Cluster 動的再構成のサポートにも適用されます (オペレーティングシステムの休止操作を除く)。そのため、Oracle Solaris Cluster ソフトウェアで動的再構成機能を使用する前に、Oracle Solaris 動的再構成機能のドキュメントを確認してください。特に、動的再構成の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- Oracle Solaris Cluster ソフトウェアは、アクティブなプライベートインターコネクットインタフェース上で実行された動的再構成ボード削除操作を拒否します。
- アクティブなクラスタインターコネクット上で動的再構成を実行するには、クラスタからアクティブなアダプタを完全に削除する必要があります。clsetup メニューまたは該当するコマンドを使用します。



注意 - Oracle Solaris Cluster ソフトウェアの個々のクラスタノードは、ほかのすべてのクラスタノードに対する有効なパスを少なくとも 1 つは持っている必要があります。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動的再構成操作を実行する場合は、次の手順を示されている順序で完了します。

表 13 タスクマップ: 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

タスク	手順
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	206 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動的再構成操作を実行します。	

▼ クラスタインターコネクットのステータスを確認する方法

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

この手順を実行するために、root 役割としてログインする必要はありません。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、クラスタインターコネクトのステータスをチェックすることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタインターコネクトのステータスを確認します。

```
% clinterconnect status
```
2. 一般的なステータスメッセージについては、以下の表を参照してください。

ステータスメッセージ	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 <code>clinterconnect status</code> を実行してもメッセージが繰り返される場合は、適切な処置を行なってください。

例 70 クラスタインターコネクトのステータスを確認する

次に、正常に機能しているクラスタインターコネクトのステータスの例を示します。

```
% clinterconnect status
-- Cluster Transport Paths --
      Endpoint                Endpoint                Status
      -----                -
Transport path: phys-schost-1:net0 phys-schost-2:net0 Path online
Transport path: phys-schost-1:net4 phys-schost-2:net4 Path online
Transport path: phys-schost-1:net0 phys-schost-3:net0 Path online
Transport path: phys-schost-1:net4 phys-schost-3:net4 Path online
Transport path: phys-schost-2:net0 phys-schost-3:net0 Path online
Transport path: phys-schost-2:net4 phys-schost-3:net4 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、またはトランスポートスイッチを追加する方法

クラスタのプライベートトランスポートの要件については、『[Oracle Solaris Cluster Hardware Administration Manual](#)』の「[Interconnect Requirements and Restrictions](#)」を参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、クラスタにケーブル、トランスポートアダプタ、およびプライベートアダプタを追加することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. **物理クラスタトランスポートケーブルが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取り付け手順については、『[Oracle Solaris Cluster Hardware Administration Manual](#)』を参照してください。

2. **クラスタの任意のノードで root 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. **クラスタインターコネクトメニューを表示するためのオプションの番号を入力します。**

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の番号を入力します。**

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の番号を入力します。**

指示に従い、必要な情報を入力します。

クラスタインターコネクトで次のアダプタのいずれかを使用する予定の場合、関連するエントリを各クラスタノードの `/etc/system` ファイルに追加します。このエントリは、次のシステム再ブート後に有効になります。

アダプタ	エントリ
nge	set nge:nge_taskq_disable=1
e1000g	set e1000g:e1000g_taskq_disable=1

7. **トランスポートスイッチを追加するためのオプションの番号を入力します。**

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 71 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の追加の確認

次の例は、ノードへの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の追加を確認する方法を示しています。この例には、Data Link Provider Interface (DLPI) トランスポートタイプの設定が含まれています。

```
# clinterconnect show phys-schost-1:net5,hub2
===Transport Cables ===
Transport Cable:          phys-schost-1:net5@0,hub2
Endpoint1:                phys-schost-2:net4@0
Endpoint2:                hub2@2
State:                    Enabled

# clinterconnect show phys-schost-1:net5
=== Transport Adepters for net5
Transport Adapter:        net5
Adapter State:           Enabled
Adapter Transport Type:  dlpi
Adapter Property (device_name): net6
Adapter Property (device_instance): 0
Adapter Property (lazy_free): 1
Adapter Property (dlpi_heartbeat_timeout): 10000
Adpater Property (dlpi_heartbeat_quantum): 1000
Adapter Property (nw_bandwidth): 80
Adapter Property (bandwidth): 70
Adapter Property (ip_address): 172.16.0.129
Adapter Property (netmask): 255.255.255.128
Adapter Port Names:      0
Adapter Port State (0):  Enabled

# clinterconnect show phys-schost-1:hub2
=== Transport Switches ===
Transport Switch:        hub2
Switch State:           Enabled
Switch Type:            switch
Switch Port Names:      1 2
Switch Port State(1):   Enabled
Switch Port State(2):   Enabled
```

次の手順 クラスタトランスポートケーブルのインターコネクトのステータスを確認するには、[191 ページの「クラスタインターコネクトのステータスを確認する方法」](#)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、およびトランスポートスイッチを削除する方法

ノードの構成からクラスタのトランスポートケーブル、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削除するには、次の手順を使用します。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 - 各クラスタードには、ほかのすべてのクラスタードに対する (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トのステータスを確認してください。状態が冗長な場合、つまり別の接続が使用できる場合だけ、ケーブル接続を無効にします。ノードの最後の機能しているケーブルを無効にすると、そのノードはクラスタメンバーシップから外れます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、クラスタからケーブル、トランスポートアダプタ、およびプライベートアダプタを削除することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタの任意のノードで `root` 役割になります。
2. 残りのクラストランスポートパスのステータスを確認します。

```
# 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、`path faulted` (パス障害) 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の番号を入力します。
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の番号を入力します。
 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の番号を入力します。

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。

注記 - 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の番号を入力します。

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。

物理アダプタをノードから取り外す場合のハードウェアサービス手順については、『[Oracle Solaris Cluster Hardware Administration Manual](#)』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の番号を入力します。

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。

注記 - 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 72 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の削除の確認

次の例は、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の削除を確認する方法を示しています。

```
# clinterconnect show phys-schost-1:net5,hub2@0
===Transport Cables ===
Transport Cable:                phys-schost-1:net5,hub2@0
Endpoint1:                      phys-schost-1:net5
Endpoint2:                      hub2@0
State:                          Enabled

# clinterconnect show phys-schost-1:net5
=== Transport Adepters for net5
Transport Adapter:              net5
Adapter State:                 Enabled
Adapter Transport Type:        dlpi
Adapter Property (device_name): net6
```

```

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 State (0):                       Enabled

# clinterconnect show hub2
=== Transport Switches ===
Transport Switch:                             hub2
State:                                         Enabled
Type:                                         switch
Port Names:                                  1 2
Port State(1):                                Enabled
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方法

このオプションは、既存のクラスタトランスポートケーブルを有効にするために使用されます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ケーブルを有効にすることもできます。「プライベートインターコネクト」をクリックし、ケーブルの番号をクリックして強調表示し、「有効」をクリックします。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタの任意のノードで root 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. 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の番号を入力します。
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の番号を入力します。

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
```

次に出力例を示します。

```
# clinterconnect show phys-schost-1:net5,hub2
Transport cable:  phys-schost-2:net0@0 ethernet-1@2  Enabled
Transport cable:  phys-schost-3:net5@1 ethernet-1@3  Enabled
Transport cable:  phys-schost-1:net5@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方法

クラスタインターコネクトパスを一時的に停止するために、クラスタトランスポートケーブルを無効にする必要がある場合があります。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、ほか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トのステータスを確認してください。状態が冗長な場合、つまり別の接続が使用できる場合だけ、ケーブル接続を無効にします。ノードの最後の機能しているケーブルを無効にすると、そのノードはクラスタメンバーシップから外れます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ケーブルを無効にすることもできます。「プライベートインターコネクト」をクリックし、ケーブルをクリックし、ケーブルの番号をクリックして強調表示し、「無効」をクリックします。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタの任意のノードで root 役割になります。
2. ケーブルを無効にする前に、クラスタインターコネクトのステータスを確認します。

```
# 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の番号を入力します。**

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の番号を入力します。**

プロンプトに従って、要求された情報を指定します。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
```

次に出力例を示します。

```
# clinterconnect show -p phys-schost-1:net5,hub2
Transport cable:  phys-schost-2:net0@0 ethernet-1@2  Disabled
Transport cable:  phys-schost-3:net5@1 ethernet-1@3  Enabled
Transport cable:  phys-schost-1:net5@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方法

clsetup コマンドを使用して正しいトランスポートアダプタを追加および削除するには、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。

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
```

```
-----
XYZ  8   B   2   33  33  2,0  ok   xyz11c8,0-xyz11c8,d665.11c8.0.0
XYZ  8   B   3   33  33  3,0  ok   xyz11c8,0-xyz11c8,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
"/xyz@1f,4000/pci11c8,o@2" 0 "ttt"
"/xyz@1f,4000.pci11c8,o@4 "ttt"
```

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
...
xyz, instance #0
xyz11c8,0, instance #0
xyz11c8,0, 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方法

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。コマンド行を使用してこのタスクを実行するには、[cluster\(1CL\)](#) のマニュアルページを参照してください。

始める前に root 役割のリモートシェル ([rsh\(1M\)](#)) または Secure Shell ([ssh\(1\)](#)) アクセスが、すべてのクラスタノードで有効になっていることを確認します。

1. 各クラスタノード上で次のサブステップを実行することで、すべてのクラスタノードをリポートし、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で、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、指定されたノードから次に優先されるノードへ、すべてのリソースグループを切り替えます。

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

2. 1つのノードから、`clsetup` ユーティリティーを起動します。
非クラスタモードで動作している場合、`clsetup` ユーティリティー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. 「Cluster トランスポート」メニュー項目の「ネットワークアドレス指定と範囲の変更」を選択します。
`clsetup` ユーティリティーは現在のプライベートネットワーク構成を表示し、この構成を変更するかどうかを尋ねます。
4. プライベートネットワーク IP アドレスか IP アドレス範囲のいずれかを変更するには、`yes` と入力し、Return キーを押します。
`clsetup` ユーティリティーはデフォルトのプライベートネットワーク IP アドレスである `172.16.0.0` を表示し、このデフォルトをそのまま使用してもよいかどうかを尋ねます。
5. プライベートネットワーク IP アドレスを変更するか、そのまま使用します。
 - デフォルトのプライベートネットワーク IP アドレスをそのまま使用し、IP アドレス範囲の変更に進むには、`yes` と入力し、Return キーを押します。
 - 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
 - a. `clsetup` ユーティリティーの、デフォルトのアドレスをそのまま使用してもよいかどうかに関する質問に対しては「`no`」と入力し、Return キーを押します。
`clsetup` ユーティリティーは、新しいプライベートネットワーク IP アドレスを入力するプロンプトを表示します。
 - b. 新しい IP アドレスを入力し、Return キーを押します。
`clsetup` ユーティリティーはデフォルトのネットマスクを表示し、デフォルトのネットマスクをそのまま使用してもよいかどうかを尋ねます。
6. デフォルトのプライベートネットワーク IP アドレス範囲を変更するか、そのまま使用します。
デフォルトのネットマスクは `255.255.240.0` です。このデフォルトの IP アドレス範囲は、クラスター内で最大 64 のノード、最大 12 のゾーンクラスタ、および最大 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。
 - IP アドレス範囲を変更するには
 - 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` ユーティリティは、クラスタで構成する予定のノード、プライベートネットワーク、およびゾーンクラスタの数を入力するプロンプトを表示します。

- b. クラスタで構成する予定のノード、プライベートネットワーク、およびゾーンクラスタの数を入力します。

これらの数から、`clsetup` ユーティリティは 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、プライベートネットワーク、およびゾーンクラスタ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、プライベートネットワーク、およびゾーンクラスタの数の 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、プライベートネットワーク、およびゾーンクラスタの予定数をサポートする別のネットマスクを指定します。

7. 更新の継続に関する `clsetup` ユーティリティの質問に対しては、`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リブートし、クラスタモードに戻します。

- a. ノードをブートします。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Oracle Solaris エントリを選択し、Enter キーを押します。

10. ノードが問題なくブートし、オンラインであることを確認します。

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

クラスタインターコネクトのトラブルシューティング

このセクションでは、クラスタインターコネクト (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) を無効にしてから有効にするためのトラブルシューティング手順について説明します。

ipadm コマンドを使用してクラスタトランスポートアダプタを管理しないでください。ipadm disable-if コマンドを使用してトランスポートアダプタを無効にした場合は、clinterconnect コマンドを使用してトランスポートパスを無効にしてから有効にする必要があります。

この手順には、Oracle Solaris Cluster ソフトウェアがインストールされている必要があります。これらのコマンドは、グローバルクラスタノードから実行する必要があります。

▼ クラスタインターコネクトを有効にする方法

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、クラスタインターコネクトを有効にすることもできます。「プライベートインターコネクト」をクリックし、ケーブルの番号をクリックして強調表示し、「有効」をクリックします。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. クラスタインターコネクトのステータスを確認します。

```
% clinterconnect status

=== Cluster Transport Paths===
Endpoint1          Endpoint2          Status
-----
pnode1:net1        pnode2:net1        waiting
pnode1:net5        pnode2:net5        Path online
```

2. クラスタインターコネクトパスを無効にします。

a. クラスタインターコネクトパスを確認します。

```
% clinterconnect show | egrep -ie "cable.*pnode1"
Transport Cable: pnode1:net5,switch2@1
Transport Cable: pnode1:net1,switch1@1
```

b. クラスタインターコネクトパスを無効にします。

```
% clinterconnect disable pnode1:net1,switch1@1
```

3. クラスタインターコネクトパスを有効にします。

```
% clinterconnect enable pnode1:net1,switch1@1
```

4. クラスタインターコネクトが有効になっていることを確認します。

```
% clinterconnect status

=== Cluster Transport Paths===
Endpoint1          Endpoint2          Status
-----
pnode1:net1        pnode2:net1        Path online
```

pnode1:net5

pnode2:net5

Path online

パブリックネットワークの管理

Oracle Solaris Cluster ソフトウェアは、パブリックネットワークのための IPMP、リンクアグリゲーション、および VNIC の Oracle Solaris ソフトウェアの実装をサポートします。基本的なパブリックネットワークの管理は、クラスタ環境でも非クラスタ環境でも同じです。

マルチパスは Oracle Solaris 11 OS をインストールすると自動的にインストールされますが、使用するには有効にする必要があります。マルチパスの管理については、適切な Oracle Solaris OS のドキュメントを参照してください。ただし、Oracle Solaris Cluster 環境で IPMP、リンクアグリゲーション、および VNIC を管理する前に、従うガイドラインを確認してください。

IPMP については、『[Oracle Solaris 11.3 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理](#)』の第 3 章、「[IPMP の管理](#)」を参照してください。リンクアグリゲーションについては、『[Oracle Solaris 11.3 でのネットワークデータリンクの管理](#)』の第 2 章、「[リンクアグリゲーションを使用した高可用性の構成](#)」を参照してください。

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方法

IPMP 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。

- SUNW.SharedAddress ネットワークリソースを使用するスケラブルサービスリソース (リソースタイプのリソースタイプ登録ファイルで SCALABLE=TRUE) を構成している場合は、SUNW.SharedAddress の構成に使用されているものに加えて、クラスタノード上のすべての IPMP グループの IPMP グループステータスをモニターするように PNM を構成できます。この構成により、クラスタノード上のいずれかの IPMP グループが失敗した場合に、クラスタノードと同じサブネット上で共有されているネットワーククライアントのサービス可用性を最大限にするため、サービスを再起動してフェイルオーバーできます。例を示します。

```
# echo ssm_monitor_all > /etc/cluster/pnm/pnm.conf
```

ノードをリブートします。

- 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。
- クラスタでは、プローブベースの IPMP グループ、またはリンクベースの IPMP グループを使用できます。プローブベースの IPMP グループは、ターゲットの IP アドレスをテストし、可用性が損なわれる可能性がある条件をより多く認識することで最大限の保護を提供します。iSCSI ストレージを定数デバイスとして使用している場合は、プローブベースの IPMP デバイスが正しく構成されていることを確認してください。iSCSI ネットワークがクラスタノードと iSCSI ストレージデバイスのみを含むプライベートネットワークであり、iSCSI ネットワーク

上にほかのホストが存在しない場合、1つを除くすべてのクラスタノードがダウンすると、プローブベースの IPMP 機構が壊れる可能性があります。この問題は、IPMP によるプローブの対象となるほかのホストが iSCSI ネットワーク上に存在しないために発生するため、クラスタ内にノードが 1 つしか残っていない場合、IPMP はこれをネットワーク障害として処理します。IPMP は iSCSI ネットワークアダプタをオフラインにし、残りのノードは iSCSI ストレージ (つまり、定足数デバイス) へのアクセスを失います。この問題を解決するために、iSCSI ネットワークにルーターを追加することで、クラスタの外部にあるほかのホストがプローブにตอบสนองし、IPMP はネットワークアダプタをオフラインにしないようにできます。または、プローブベースのフェイルオーバーの代わりにリンクベースのフェイルオーバーを使用するように IPMP を構成することもできます。

- リンクローカルでない IPv6 パブリックネットワークインタフェースがパブリックネットワーク構成内に 1 つ以上存在していないかぎり、`scinstall` ユーティリティは自動的に、同じサブネットを使用しているクラスタ内の一連のパブリックネットワークアダプタごとに、複数のアダプタを含む IPMP グループを 1 つずつ構成します。これらのグループはリンクベースであり、推移的プローブを備えています。プローブベースの障害検出が必要な場合は、テストアドレスを追加できます。
- 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` の名前は `sc_ipmp0@1` または `sc_ipmp0@phys-schost-1` のどちらでもかまいません。
- 最初に `if_mpadm(1M)` コマンドを使用して、IP アドレスを削除対象のアダプタからグループ内の代替アダプタに切り替えることなく、IP ネットワークマルチパスグループのアダプタを構成解除 (`unplumb`) したり、停止したりしないでください。
- Oracle Solaris Cluster の HA IP アドレスが `plumb` されている IPMP グループからネットワークインタフェースを `unplumb` または削除しないでください。この IP アドレスは、論理ホスト名リソースまたは共有アドレスリソースに属することがあります。ただし、`ifconfig` コマンドを使用してアクティブなインタフェースを `unplumb` した場合、Oracle Solaris Cluster はこのイベントを認識するようになりました。プロセスで IPMP グループが使用できなくなった場合は、リソースグループをほかの正常なノードにフェイルオーバーします。Oracle Solaris Cluster は IPMP グループが有効だが、HA IP アドレスがない場合に、同じノード上でリソースグループを再起動することもできます。IPMP グループは、いくつかの理由 (IPv4 接続の損失、IPv6 接続の損失、またはその両方) で使用できなくなります。詳細は、`if_mpadm(1M)` のマニュアルページを参照してください。
- 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モニタリング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ステータスを表示するには、`ipmpstat -g` コマンドを使用します。詳細は、『Oracle Solaris 11.3 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理』の第 3 章、「IPMP の管理」を参照してください。

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、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』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の保守手順については、『Oracle Solaris Cluster Hardware Administration Manual』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で動的再構成 (DR) 操作を実行するときは、いくつかの問題を考える必要があります。

- Oracle Solaris 動的再構成機能に関して記載されている要件、手順、および制限のすべてが、Oracle Solaris Cluster 動的再構成のサポートにも適用されます (オペレーティングシステムの休止操作を除く)。そのため、Oracle Solaris Cluster ソフトウェアで動的再構成機能を使用する前に、Oracle Solaris 動的再構成機能のドキュメントを確認してください。特に、動的再構成の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- 動的再構成ボード削除操作が成功するのは、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場合だけです。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` コマンドを使用して、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に IP アドレスを切り換えます。詳細は、`if_mpadm(1M)` のマニュアルページを参照してください。
- アクティブネットワークインタフェースとして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、Oracle Solaris 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 つのアダプタを含むマルチパスグループの場合は、無効になっているネットワークアダプタ上の動的再構成削除操作を実行中に残りのネットワークアダプタに障害が発生すると、可用性が影響を受けます。動的再構成操作の期間中、残りのアダプタのフェイルオーバー先はありません。

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操作を実行する場合は、次の手順を示されている順序で完了します。

表 14 タスクマップ: 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

タスク	手順
1. <code>if_mpadm</code> コマンドを使用して、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 IP アドレスの切り換えを実行	<code>if_mpadm(1M)</code> のマニュアルページ 『Oracle Solaris 11.3 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理』の「インタフェースを 1 つの IPMP グループから別の IPMP グループに移動する方法」

タスク	手順
2. ipadm 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	ipadm(1M) のマニュアルページ 『Oracle Solaris 11.3 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理』の「IPMP グループからインタフェースを削除する方法」
3. 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操作を実行します。	

クラスタノードの管理

この章では、クラスタへのノードの追加とノードの削除方法を説明します。

- 209 ページの「クラスタまたはゾーンクラスタへのノードの追加」
- 212 ページの「クラスタノードの復元」
- 216 ページの「クラスタからのノードの削除」

クラスタのメンテナンスタスクについては、[第9章「クラスタの管理」](#)を参照してください。

クラスタまたはゾーンクラスタへのノードの追加

このセクションでは、グローバルクラスタまたはゾーンクラスタにノードを追加する方法を説明します。新しいゾーンクラスタノードは、そのゾーンクラスタをホストするグローバルクラスタのノード上に作成できますが、それは、グローバルクラスタノードが、そのゾーンクラスタのノードをまだホストしていない場合に限られます。

注記 - 追加するノードは、それが参加するクラスタと同じバージョンの Oracle Solaris Cluster ソフトウェアを実行している必要があります。

各ゾーンクラスタノードの IP アドレスおよび NIC の指定は任意です。

注記 - 各ゾーンクラスタノードで IP アドレスを構成しない場合、次の 2 つのことが発生します。

1. その特定のゾーンクラスタでは、ゾーンクラスタで使用するための NAS デバイスを構成することができません。NAS デバイスと通信する際にはゾーンクラスタノードの IP アドレスを使用するため、IP アドレスを持たないクラスタは、NAS デバイスのフェンシングをサポートできません。
 2. クラスタソフトウェアによって、NIC の論理ホスト IP アドレスが有効化されます。
-

元のゾーンクラスタノード用 IP アドレスまたは NIC を指定しなかった場合、新しいゾーンクラスタノード用にその情報を指定する必要はありません。

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

次の表に、ノードを既存のクラスタに追加するときに行うタスクを示します。タスクは、示されている順に実行してください。

表 15 タスクマップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加

タスク	手順
ホストアダプタのノードへの取り付けと、既存のクラスタインターコネクトが新しいノードをサポートできることの確認	『Oracle Solaris Cluster Hardware Administration Manual』
共有ストレージの追加	『Oracle Solaris Cluster Hardware Administration Manual』の手順に従って、共有ストレージを手動で追加します。 Oracle Solaris Cluster Manager を使用して、ゾーンクラスタに共有ストレージデバイスを追加することもできます。Oracle Solaris Cluster Manager のゾーンクラスタのページに移動して、「Solaris リソース」タブをクリックします。Oracle Solaris Cluster Manager のログイン手順については、290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」を参照してください。
認証ノードリストへのノードの追加	<code>claccess allow -h node-being-added</code>
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	『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の第 2 章、「グローバルクラスタノードへのソフトウェアのインストール」
既存のクラスタに新しいノードを追加する	210 ページの「既存のクラスタまたはゾーンクラスタにノードを追加する方法」
クラスタが Oracle Solaris Cluster Geographic Edition のパートナーシップで構成されている場合、構成内のアクティブな参加メンバーとして新しいノードを構成する	『Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide』の「How to Add a New Node to a Cluster in a Partnership」

▼ 既存のクラスタまたはゾーンクラスタにノードを追加する方法

既存のグローバルクラスタまたはゾーンクラスタに Oracle Solaris ホストまたは仮想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トとの使用可能な物理的接続も含め、ノードに必要なすべてのハードウェアが正しくインストールされ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、『Oracle Solaris Cluster Hardware Administration Manual』またはサーバーに付属するハードウェアのドキュメントを参照してください。

この手順によって、マシンは自分自身をクラスタ内にインストールします。つまり、自分のノード名を当該クラスタの認証ノードリストに追加します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 現在のグローバルクラスタメンバーで、現在のクラスタメンバー上の root 役割になります。次の手順は、グローバルクラスタのノードから実行します。
2. [表15「タスマップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加」](#)のタスマップに記載されている前提条件のハードウェアのインストールと構成タスクをすべて正しく完了していることを確認します。
3. 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。
『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の説明に従って、scinstall ユーティリティーを使用して、新しいノードのインストールと構成を完了します。
4. 新しいノードで scinstall ユーティリティーを使用して、クラスタ内のそのノードを構成します。
5. ノードをゾーンクラスタに手動で追加するには、Oracle Solaris ホストおよび仮想ノード名を指定してください。

また、各ノードでパブリックネットワーク通信に使用するネットワークリソースも指定してください。次の例では、ゾーン名は sczone で、sc_ipmp0 は IPMP グループ名です。

```
clzc:sczone>add node
clzc:sczone:node>set physical-host=phys-cluster-3
clzc:sczone:node>set hostname=hostname3
clzc:sczone:node>add net
clzc:sczone:node:net>set address=hostname3
clzc:sczone:node:net>set physical=sc_ipmp0
clzc:sczone:node:net>end
clzc:sczone:node>end
clzc:sczone>exit
```

ノードを構成する手順の詳細は、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の「[ゾーンクラスタの作成および構成](#)」を参照してください。

6. 新しいゾーンクラスタノードが solaris10 ブランドタイプになり、Oracle Solaris Cluster ソフトウェアがゾーンクラスタにインストールされていない場合は、DVD イメージへのパスを指定してソフトウェアをインストールします。

```
# clzonecluster install-cluster -d dvd-image zone-cluster-name
```

7. ノードの構成後、ノードをクラスタモードでリブートし、そのノードにゾーンクラスタをインストールします。

```
# clzonecluster install zone-cluster-name
```

8. 新しいマシンがクラスタに追加されないようにするために、新しいマシンを追加する要求を無視するようクラスタに指示するオプションの番号を `clsetup` ユーティリティから入力します。

Return キーを押します。

`clsetup` のプロンプトに従います。このオプションを設定すると、クラスタは、自分自身をクラスタに追加しようとする新しいマシンからのパブリックネットワーク経由の要求をすべて無視します。

9. `clsetup` ユーティリティを終了します。

参照 `clsetup(1CL)` のマニュアルページ。

クラスタノードを追加するすべてのタスクの一覧については、[表15「タスクマップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加」](#)、「タスクマップ: 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には、『[Oracle Solaris Cluster 4.3 データサービス計画および管理ガイド](#)』を参照してください。

クラスタノードの復元

統合アーカイブを使用して、クラスタノードをアーカイブとまったく同じ状態になるように復元できます。ノードを復元する前に、まずクラスタノード上に復旧用のアーカイブを作成する必要があります。使用できるのは、復旧用のアーカイブだけです。クラスタノードの復元にクローンアーカイブは使用できません。回復用のアーカイブを作成する手順については、下記の[ステップ 1](#)を参照してください。

この手順では、クラスタ名、ノード名とそれらの MAC アドレス、および統合アーカイブへのパスを入力するように求められます。`scinstall` ユーティリティは、指定するアーカイブごとに、アーカイブのソースノード名が復元するノードと同じであることを確認します。統合アーカイブからクラスタ内のノードを復元する手順については、[212 ページの「統合アーカイブからノードを復元する方法」](#)を参照してください。

▼ 統合アーカイブからノードを復元する方法

この手順では、Automated Installer サーバー上で対話型の `scinstall` ユーティリティを使用します。前もって AI サーバーを設定し、Oracle Solaris Cluster リポジトリから `ha-cluster/system/install` パッケージをインストールしておく必要があります。アーカイブのノード名は、復元するノードと同じである必要があります。

これらのガイドラインに従い、次に示す手順で対話式の `scinstall` ユーティリティーを使用します。

- 対話式 `scinstall` を使用すると、先行入力が可能になります。したがって、次のメニュー画面がすぐに表示されなくても、Return キーを押すのは一度だけにしてください。
- 特に指定のある場合を除いて、Control-D キーを押すと、関連する一連の質問の最初に戻るか、メインメニューに戻ります。
- 前のセッションのデフォルトの解凍が、質問の最後に角かっこ ([]) で囲まれて表示されます。入力せずに角かっこ内の回答を入力するには、Return キーを押します。

1. **グローバルクラスタのノード上で root 役割になり、復旧用のアーカイブを作成します。**

```
phys-schost# archiveadm create -r archive-location
```

アーカイブを作成する場合は、共有ストレージ上に存在する ZFS データセットを除外します。共有ストレージ上のデータを復元する予定がある場合は、従来の方法を使用してください。

`archiveadm` コマンドの使用の詳細は、[archiveadm\(1M\)](#) のマニュアルページを参照してください。

2. **Automated Installer サーバーにログインし、root 役割になります。**

3. **`scinstall` ユーティリティーを起動します。**

```
phys-schost# scinstall
```

4. **クラスタを復元するオプションの番号を入力します。**

```
*** Main Menu ***
```

```
Please select from one of the following (*) options:
```

- * 1) Install, restore, or replicate a cluster from this Automated Installer server
- * 2) Securely install, restore, or replicate a cluster from this Automated Installer server
- * 3) Print release information for this Automated Installer install server

- * ?) Help with menu options
- * q) Quit

```
Option: 2
```

セキュアでない AI サーバーインストールを使用してクラスタノードを復元するには、オプション 1 を選択します。セキュアな AI サーバーインストールを使用してクラスタノードを復元するには、オプション 2 を選択します。

カスタム Automated Installer メニューまたはカスタムセキュア Automated Installer メニューが表示されます。

5. **統合アーカイブからクラスタノードを復元するオプションの番号を入力します。**

「クラスタ名」画面が表示されます。

6. 復元するノードを含むクラスタ名を入力します。

「クラスタノード」画面が表示されます。

7. 統合アーカイブから復元するクラスタノードの名前を入力します。

1 行に 1 つのノード名を入力します。終了したら Control-D を押し、リストを確認するために yes を入力して Return キーを押します。クラスタ内のすべてのノードを復元する場合は、すべてのノードを指定します。

scinstall ユーティリティーがノードの MAC アドレスを見つけられない場合は、プロンプトが表示されたときに各アドレスを入力します。

8. 回復用のアーカイブのフルパスを入力します。

ノードを復元するために使用されるアーカイブは、復旧用のアーカイブである必要があります。特定のノードを復元するために使用するアーカイブファイルは、同じノード上に作成されている必要があります。復元するクラスタノードごとに、これを繰り返します。

9. この AI サーバーからクラスタノードをインストールするために必要な構成が scinstall ユーティリティーによって実行されるように、各ノードに対して選択したオプションを確認します。

このユーティリティーではまた、DHCP サーバー上で DHCP マクロを追加するための手順が出力されるほか、SPARC ノードのセキュリティー鍵も追加またはクリアされます (セキュアインストールを選択した場合)。これらの手順に従ってください。

10. (オプション) ターゲットデバイスをカスタマイズするには、各ノードの AI マニフェストを更新します。

AI マニフェストは、次のディレクトリにあります。

```
/var/cluster/logs/install/autosinstall.d/ \  
cluster-name/node-name/node-name_aimanifest.xml
```

a. ターゲットデバイスをカスタマイズするには、マニフェストファイル内の target 要素を更新します。

インストールのターゲットデバイスを見つけるためのサポートされる条件の使用方法に基づいて、マニフェストファイル内の target 要素を更新します。たとえば、disk_name サブ要素を指定できます。

注記 - scinstall は、マニフェストファイル内の既存のブートディスクをターゲットデバイスであるとみなします。ターゲットデバイスをカスタマイズするには、マニフェストファイル内の target 要素を更新します。詳細は、『Oracle Solaris 11.3 システムのインストール』の [パート III](#), 「インストールサーバーを使用したインストール」および [ai_manifest\(4\)](#) のマニュアルページを参照してください。

b. ノードごとに installadm コマンドを実行します。

```
# installadm update-manifest -n cluster-name-{sparc|i386} \  

```

```
-f /var/cluster/logs/install/autoscinstall.d/cluster-name/node-name/node-
name_aimanifest.xml \
-m node-name_manifest
```

クラスタノードのアーキテクチャーが SPARC と i386 であることに注意してください。

11. クラスタ管理コンソールを使用している場合、クラスタ内にある各ノードのコンソール画面を表示します。

- 管理コンソール上で `pconsole` ソフトウェアがインストールおよび構成されている場合は、`pconsole` ユーティリティを使用して個々のコンソール画面を表示します。

`root` 役割として、次のコマンドを使用して、`pconsole` ユーティリティを起動します。

```
adminconsole# pconsole host[:port] [...] &
```

また、`pconsole` ユーティリティを使用してマスターウィンドウを開くことができます。ここで
のを入力を、個々のすべてのコンソールウィンドウに同時に送信できます。

- `pconsole` ユーティリティを使用しない場合は、各ノードのコンソールに個別に接続します。

12. AI インストールを開始するために、各ノードをシャットダウンしてブートします。

Oracle Solaris ソフトウェアはデフォルトの構成でインストールされます。

注記 - Oracle Solaris のインストールをカスタマイズする必要がある場合は、この方法を使用
できません。Oracle Solaris の対話式インストールを選択した場合、Automated Installer は
バイパスされ、Oracle Solaris Cluster ソフトウェアのインストールや構成は行われません。

インストール中に Oracle Solaris をカスタマイズするには、代わりに『Oracle Solaris
Cluster 4.3 ソフトウェアのインストール』の「Oracle Solaris ソフトウェアをインストールす
る方法」の手順に従ったあと、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の
「Oracle Solaris Clusterソフトウェアパッケージをインストールする方法」の手順に従ってクラ
スタをインストールし、構成します。

- SPARC:

- a. 各ノードを停止します。

```
phys-schost# cluster shutdown -g 0 -y
```

- b. 次のコマンドでノードをブートします

```
ok boot net:dhcp - install
```

注記 - 上記コマンド内のダッシュ記号 (-) の両側は、空白文字で囲む必要があります。

■ x86

- a. ノードをリブートします。

```
# reboot -p
```

- b. PXE ブート時に Control-N キーを押します。

GRUB メニューが表示されます。

- c. すぐに「自動インストール」エントリを選択します。

注記 - 「自動インストール」エントリを 20 秒以内に選択しなかった場合は、デフォルトの対話式テキストインストーラ方式を使用してインストールが進みますが、その場合は Oracle Solaris Cluster ソフトウェアのインストールや構成は行われません。

各ノードが自動的にリブートされ、インストールが完了したらクラスタに参加するようになります。ノードは、アーカイブが作成されたときと同じ状態に復元されます。Oracle Solaris Cluster のインストール出力は、各ノード上の `/var/cluster/logs/install/sc_ai_config.log` ファイルに記録されます。

13. 1 つのノードから、すべてのノードがクラスタに参加したことを確認します。

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

出力は次のようになります。

```
=== Cluster Nodes ===
```

```
--- Node Status ---
```

Node Name	Status
phys-schost-1	Online
phys-schost-2	Online
phys-schost-3	Online

詳細は、[clnode\(1CL\)](#) のマニュアルページを参照してください。

クラスタからのノードの削除

このセクションでは、グローバルクラスタまたはゾーンクラスタ上のノードを削除する方法について説明します。グローバルクラスタから特定のゾーンクラスタを削除することもできます。次の表に、ノードを既存のクラスタから削除するときに行うタスクを示します。タスクは、示されている順に実行してください。



注意 - RAC 構成の場合、この手順のみを使用してノードを削除すると、リポート中のノードでパニックが発生する可能性があります。RAC 構成からノードを削除する方法については、『[Oracle Solaris Cluster Data Service for Oracle Real Application Clusters ガイド](#)』の「[選択したノードから Oracle RAC のサポートを削除する方法](#)」を参照してください。処理が完了したら、RAC 構成用のノードを削除して、次の該当する手順に従います。

表 16 タスクマップ: ノードの削除

タスク	手順
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する。ゾーンクラスタがある場合は、ゾーンクラスタにログインし、アンインストールされる物理ノード上にあるゾーンクラスタノードを退避させます。その後、物理ノードを停止する前に、ゾーンクラスタからそのノードを削除します。	<code>clnode evacuate node</code> 217 ページの「ゾーンクラスタからノードを削除する方法」
影響を受ける物理ノードにすでに障害が発生している場合は、単にそのノードをクラスタから削除します。	
許可されたホストをチェックして、ノードを削除できることを確認する。	<code>claccess show</code>
ノードが <code>claccess show</code> コマンドで一覧表示されない場合は、そのノードを削除できません。そのノードにクラスタ構成へのアクセスを許可してください。	<code>claccess allow -h node-to-remove</code>
すべてのデバイスグループからノードを削除する。	134 ページの「デバイスグループからノードを削除する方法 (Solaris Volume Manager)」
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。	174 ページの「定足数デバイスを削除する方法」
2 ノードクラスタのノードを削除する場合、この手順はオプションです。	175 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方法」
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。	
削除するノードを非クラスタモードにする。	237 ページの「ノードを保守状態にする」
クラスタソフトウェア構成からノードを削除する。	219 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方法」
(オプション) Oracle Solaris 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。	241 ページの「クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法」

▼ ゾーンクラスタからノードを削除する方法

ノードを停止してアンインストールし、構成からノードを削除することによって、ゾーンクラスタからノードを削除できます。あとでノードをゾーンクラスタに戻す場合は、[表15「タスクマップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加」](#)の手順に従います。ここからの手順のほとんどは、グローバルクラスタノードから実行します。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用してゾーンクラスタノードを停止することもできますが、ゾーンクラスタノードを削除することはできません。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. グローバルクラスタのノードで **root** 役割になります。
2. ノードとそのゾーンクラスタを指定して、削除するゾーンクラスタノードを停止します。

```
phys-schost# clzonecluster halt -n node zone-cluster-name
```

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

3. ゾーンクラスタ内のすべてのリソースグループからノードを削除します。

```
phys-schost# clresourcegroup remove-node -n zone-hostname -Z zone-cluster-name rg-name
```

段階 2 の注で説明されている手順を使用した場合は、リソースグループが自動的に削除されるため、この段階はスキップできます。

4. ゾーンクラスタノードをアンインストールします。

```
phys-schost# clzonecluster uninstall -n node zone-cluster-name
```

5. ゾーンクラスタノードを構成から削除します。

次のコマンドを使用します。

```
phys-schost# clzonecluster configure zone-cluster-name
clzc:sczone> remove node physical-host=node
clzc:sczone> exit
```

注記 - 削除するゾーンクラスタノードが、アクセスできないシステムまたはクラスタに参加できないシステム上に存在する場合は、`clzonecluster` 対話型シェルを使用してそのノードを削除してください。

```
clzc:sczone> remove -F node physical-host=node
```

この方法を使用して最後のゾーンクラスタノードを削除すると、そのゾーンクラスタを完全に削除するよう求められます。それに従わないことを選択した場合、最後のノードは削除されません。この削除は、`clzonecluster delete -F zone-cluster-name` と同じ効果があります。

6. ノードがゾーンクラスタから削除されたことを確認します。

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

▼ クラスタソフトウェア構成からノードを削除する方法

この手順を実行して、ノードをグローバルクラスタから削除します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. この手順を実行する前に、ノードをすべてのリソースグループ、デバイスグループ、および定数デバイス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、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
3. 削除するグローバルクラスタノードを非クラスタモードでブートします。
ゾーンクラスタノードの場合は、この手順を実行する前に、[217 ページの「ゾーンクラスタからノードを削除する方法」](#)の手順を実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
shutdown -g -y -i0
Press any key to continue
```

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

GRUB ベースのブートの詳細は、『[Oracle Solaris 11.3 システムのブートとシャットダウン](#)』の「[システムのブート](#)」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。

- 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ブートするように指定します。

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

```
grub edit> kernel$ /platform/i86pc/kernel/#ISADIR/unix -B $ZFS-BOOTFS -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。

e. **b** と入力して、ノードを非クラスタモードでブートします。

カーネルブートパラメータコマンドへのこの変更は、システムをブートすると無効になります。次にノードをリブートする際には、ノードはクラスタモードでブートします。クラスタモードではなく、非クラスタモードでブートするには、これらの手順を再度実行して、カーネルブート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。

a. アクティブなノードから次のコマンドを実行します。

```
phys-schost# clnode clear -F nodename
```

`rg_system=true` が設定されているリソースグループがある場合、`clnode clear -F` コマンドが成功するためには、それらを `rg_system=false` に変更する必要があります。`clnode clear -F` を実行したあとに、そのリソースグループを `rg_system=true` に戻します。

b. 削除するノードから次のコマンドを実行します。

```
phys-schost# clnode remove -F
```

注記 - 削除するノードが使用できないか、またはブートできない場合は、アクティブな任意のクラスタノードで次のコマンドを実行します。

```
# clnode clear -F node-to-be-removed
```

`clnode status nodename` を実行して、ノードが削除されていることを確認します。

クラスタ内の最後のノードを削除する場合は、そのノードがクラスタモードでないこと、およびクラスタ内にアクティブなノードがないことが必要です。

5. 別のクラスタノードから、ノードの削除を確認します。

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

6. ノードの削除を完了します。

- 削除するノードから Oracle Solaris Cluster ソフトウェアをアンインストールするつもりの場合は、[241 ページの「クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法」](#)に進んでください。

クラスタからのノードの削除および Oracle Solaris Cluster ソフトウェアのアンインストールを同時に行うように選択することもできます。Oracle Solaris Cluster ファイルが含まれていないディレクトリに移動し、`scinstall -r` と入力します。

- 削除するノードから Oracle Solaris Cluster ソフトウェアをアンインストールする予定がない場合は、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。

手順については、『[Oracle Solaris Cluster Hardware Administration Manual](#)』を参照してください。

例 73 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方法を示します。`clnode remove` コマンドは、クラスタから削除するノード (`phys-schost-2`) から非クラスタモードで実行されます。

```

      クラスタからノードを削除します。
phys-schost-2# clnode remove
phys-schost-1# clnode clear -F phys-schost-2
      ノードの削除を確認します。
phys-schost-1# clnode status
-- Cluster Nodes --
      Node name          Status
      -----          -
Cluster node:  phys-schost-1  Online

```

参照 削除するノードから Oracle Solaris 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、[241 ページの「クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法」](#)を参照してください。

ハードウェア手順については、『[Oracle Solaris Cluster Hardware Administration Manual](#)』を参照してください。

クラスタノードを削除するタスクの総合的な一覧については、[表16「タスクマップ: ノードの削除」](#)を参照してください。

既存のクラスタにノードを追加するには、[210 ページの「既存のクラスタまたはゾーンクラスタにノードを追加する方法」](#)を参照してください。

▼ 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方法

3 ノードまたは 4 ノード接続のクラスタでストレージアレイを単一クラスタノードから取り外すには、次の手順を使用します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 取り外す予定のストレージアレイに関連付けられているすべてのデータベーステーブル、データサービス、ボリュームのバックアップを作成します。
2. 切断する予定のノードで動作しているリソースグループとデバイスグループを判別します。

```
phys-schost# clresourcegroup status
phys-schost# cldevicegroup status
```

3. 必要であれば、切断する予定のノードからすべてのリソースグループとデバイスグループを移動します。



注意 (SPARC のみ) - 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、ノードで動作している Oracle RAC データベースのインスタンスを停止します。手順については、『[Oracle Database Administration Guide](#)』を参照してください。

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

clnode evacuate コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定されたノードから次に優先されるノードへ、すべてのリソースグループを切り替えます。

4. デバイスグループを保守状態にします。
デバイスグループを保守状態にする手順については、[237 ページの「ノードを保守状態にする」](#)を参照してください。
5. デバイスグループからノードを削除します。
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。
6. HAStoragePlus リソースが含まれる各リソースグループで、リソースグループのノードリストからノードを削除します。

```
phys-schost# clresourcegroup remove-node -n node + | resourcegroup
```

リソースグループのノードリストを変更する詳細については、『[Oracle Solaris Cluster 4.3 データサービス計画および管理ガイド](#)』を参照してください。

注記 - `clresourcegroup` を実行するときには、リソースタイプ、リソースグループ、およびリソースのプロパティ名には大文字と小文字の区別があります。

7. 削除する予定のストレージアレイがノードに接続されている最後のストレージアレイである場合、このストレージアレイに接続されているノードとハブまたはスイッチの間にある光ケーブルを取り外します。

それ以外の場合は、この手順をスキップします。

8. 切断するノードからホストアダプタを削除する場合、ノードの電源を切ります。
切断するノードからホストアダプタを削除する場合は、[ステップ 11](#)にスキップします。
9. ノードからホストアダプタを削除します。
ホストアダプタの削除手順については、ノード用ドキュメントを参照してください。
10. ブートが行われないうようにして、ノードに電源を入れます。
11. Oracle RAC ソフトウェアがインストールされている場合、切断する予定のノードから Oracle RAC ソフトウェアパッケージを削除します。

```
phys-schost# pkg uninstall /ha-cluster/library/ucmm
```



注意 (SPARC のみ) - 切断したノードから Oracle RAC ソフトウェアを削除しない場合、そのノードをクラスタに導入し直すときにノードでパニックが発生し、データの可用性が失われる可能性があります。

12. クラスタモードでノードをブートします。
 - SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```
 - x86 ベースのシステム上で、次のコマンドを実行します。
 GRUB メニューが表示された時点で、適切な Oracle Solaris エントリを選択し、Enter キーを押します。
13. ノードの `/devices` と `/dev` エントリを更新して、デバイスの名前空間を更新します。

```
phys-schost# devfsadm -C
phys-schost# cldevice refresh
```
14. デバイスグループをオンラインに戻します。
 デバイスグループをオンラインにする方法については、[239 ページの「ノードを保守状態から戻す」](#)を参照してください。

▼ エラーメッセージを修正する方法

クラスタノードの削除手順のいずれかを実行中に発生したエラーメッセージを修正するには、次の手順を実行します。

1. グローバルクラスタへのノードの再参加を試みます。

この手順は、グローバルクラスタ上のみで実行します。

```
phys-schost# boot
```

2. ノードがクラスタに正常に再参加したかどうかを確認します。
 - そうでない場合は、[ステップ 2b](#)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参加した場合は、残っているデバイスグループからノードを削除します。
[133 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方法」](#)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、[241 ページの「クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法」](#)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参加できなかった場合は、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. [241 ページの「クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法」](#)に戻り、その手順を繰り返します。

クラスタの管理

この章では、グローバルクラスタやゾーンクラスタ全体に影響する管理手順について説明します。

- 225 ページの「クラスタの管理の概要」
- 254 ページの「ゾーンクラスタ管理タスクの実行」
- 264 ページの「テスト目的のトラブルシューティング手順」

クラスタへのノードの追加または削除については、[第8章「クラスタノードの管理」](#)を参照してください。

クラスタの管理の概要

このセクションでは、グローバルクラスタやゾーンクラスタ全体の管理タスクを実行する方法を説明します。次の表に、これらの管理タスクと、関連する手順を示します。クラスタの管理タスクは通常は大域ゾーンで行います。ゾーンクラスタを管理するには、そのゾーンクラスタをホストするマシンが 1 台以上クラスタモードで起動していることが必要です。すべてのゾーンクラスタノードが起動し動作している必要はありません。現在クラスタ外にあるノードがクラスタに再度参加すると、Oracle Solaris Cluster はすべての構成変更を再実行します。

注記 - デフォルトで、電源管理は無効になっているため、クラスタに干渉しません。単一ノードクラスタの電源管理を有効にすると、クラスタは引き続き動作していますが、数秒間使用できなくなる場合があります。電源管理機能はノードを停止しようとしても、停止されません。

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

表 17 タスクリスト：クラスタの管理

タスク	手順
クラスタへのノードの追加または削除	第8章「クラスタノードの管理」
クラスタ名を変更	226 ページの「クラスタ名を変更する方法」
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	227 ページの「ノード ID をノード名にマップする方法」

タスク	手順
クラスタへの新しいノードの追加を許可または拒否	228 ページの「新しいクラスタノードの認証を操作する方法」
NTP を使用して、クラスタの時間を変更する	230 ページの「クラスタの時間をリセットする方法」
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	232 ページの「ノードで OpenBoot PROM (OBP) を表示する方法」
プライベートホスト名の追加または変更	232 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	Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、論理ホスト名をグローバルクラスタまたはゾーンクラスタに追加することもできます。「タスク」をクリックし、「論理ホスト名」をクリックしてウィザードを起動します。Oracle Solaris Cluster Manager のログイン手順については、 290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」 を参照してください。
ノード名を変更する	237 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	235 ページの「ノード名を変更する」
クラスタノードからソフトウェアをアンインストール	239 ページの「ノードを保守状態から戻す」
SNMP Event MIB の追加および管理	241 ページの「クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法」
各ノードの負荷制限の構成	246 ページの「SNMP イベント MIB を有効にする」
ゾーンクラスタの移動、アプリケーション用ゾーンクラスタの準備、ゾーンクラスタの削除	250 ページの「SNMP ユーザーをノードに追加する」
	252 ページの「ノードに負荷制限を構成する」
	254 ページの「ゾーンクラスタ管理タスクの実行」

▼ クラスタ名を変更する方法

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。



注意 - クラスタが Oracle Solaris Cluster Geographic Edition パートナーシップ関係にある場合は、この手順を実行しないでください。代わりに、『[Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide](#)』の「[Renaming a Cluster That Is in a Partnership](#)」の手順に従います。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. グローバルクラスタ内の任意のノードで root 役割になります。

2. **clsetup** ユーティリティーを起動します。

```
phys-schost# clsetup
```

メインメニューが表示されます。

3. クラスタ名を変更するには、クラスタその他のプロパティのオプションの番号を入力します。「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なって、画面の指示に従います。

5. Oracle Solaris Cluster のサービスタグに新しいクラスタ名を反映させる場合は、既存の Oracle Solaris Cluster タグを削除してクラスタを再起動します。

Oracle Solaris Cluster サービスタグインスタンスを削除するには、クラスタ内のすべてのノードで次のサブステップを完了します。

- a. すべてのサービスタグの一覧を表示します。

```
phys-schost# stclient -x
```

- b. Oracle Solaris Cluster サービスタグインスタンス番号を見つけて、次のコマンドを実行します。

```
phys-schost# stclient -d -i service_tag_instance_number
```

- c. クラスタ内のすべてのノードをリブートします。

```
phys-schost# reboot
```

例 74 クラスタ名の変更

次の例に、新しいクラスタ名 `dromedary` へ変更するために、`clsetup` ユーティリティーから生成される `cluster` コマンドを示します。

```
phys-schost# cluster rename -c dromedary
```

詳細は、[cluster\(1CL\)](#) および [clsetup\(1CL\)](#) のマニュアルページを参照してください。

▼ ノード ID をノード名にマップする方法

Oracle Solaris Cluster のインストール中、ノードにはそれぞれ一意のノード ID 番号が自動で割り当てられます。このノード ID 番号は、最初にクラスタに参加したときの順番でノードに割り当てられます。ノード ID 番号が割り当てられたあとでは、番号は変更できません。ノード ID 番号は、通常、エラーメッセージが発生したクラスタノードを識別するために、エラーメッセージで使用されます。この手順を使用し、ノード ID とノード名間のマッピングを判別します。

グローバルクラスタまたはゾーンクラスタ用の構成情報を一覧表示するために、root 役割になる必要はありません。グローバルクラスタのノードから、このプロシージャの 1 ステップが実行されます。他のステップはゾーンクラスタノードから実行されます。

1. **clnode** コマンドを使用して、グローバルクラスタに対するクラスタ構成情報を一覧表示します。

```
phys-schost# clnode show | grep Node
```

詳細は、[clnode\(1CL\)](#) のマニュアルページを参照してください。

2. (オプション) ゾーンクラスタのノード ID を一覧表示します。

ゾーンクラスタノードは、実行中のグローバルクラスタノードと同じノード ID を持っています。

```
phys-schost# zlogin szone clnode -v | grep Node
```

例 75 ノード名のノードID へのマップ

次の例は、グローバルクラスタに対するノード ID の割り当てを示しています。

```
phys-schost# clnode show | grep Node
=== Cluster Nodes ===
Node Name:   phys-schost1
Node ID:     1
Node Name:   phys-schost2
Node ID:     2
Node Name:   phys-schost3
Node ID:     3
```

▼ 新しいクラスタノードの認証を操作する方法

Oracle Solaris Cluster では、新しいノードがそれ自身をグローバルクラスタに追加できるようにするかどうかと、使用する認証の種類を指定できます。パブリックネットワーク上のクラスタに参加す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参加することを拒否したり、クラスタに参加す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、ノードが参加するには、すべての必要な暗号化鍵を構成する必要があります。詳細は、[keyserv\(1M\)](#) と [publickey\(4\)](#) のマニュアルページを参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. グローバルクラスタ内の任意のノードで `root` 役割になります。
2. `clsetup` ユーティリティーを起動します。

```
phys-schost# clsetup
```

メインメニューが表示されます。
3. クラスタ認証で作業するため、新規ノードのオプションの番号を入力します。「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なって、画面の指示に従います。

例 76 新しいマシンがグローバルクラスタに追加されないようにする

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、新しいマシンがクラスタに追加されないようにする `claccess` コマンドを示しています。

```
phys-schost# claccess deny -h hostname
```

例 77 すべての新しいマシンがグローバルクラスに追加されることを許可する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、すべての新しいマシンをクラスタに追加できるようにする `claccess` コマンドを示しています。

```
phys-schost# claccess allow-all
```

例 78 グローバルクラスタに追加される新しいマシンを指定する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、1 台の新しいマシンをクラスタに追加できるようにする `claccess` コマンドを示しています。

```
phys-schost# claccess allow -h hostname
```

例 79 認証を標準 UNIX に設定する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、クラスタに参加している新規ノードの標準 UNIX 認証に対し、リセットを行う `claccess` コマンドを示しています。

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

例 80 認証を DES に設定する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、クラスタに参加している新規ノードの DES 認証を使用する `claccess` コマンドを示しています。

```
phys-schost# claccess set -p protocol=des
```

DES 認証を使用する場合、クラスタにノードが参加するには、すべての必要な暗号化鍵を構成します。詳細は、[keyserv\(1M\)](#) と [publickey\(4\)](#) のマニュアルページを参照してください。

▼ クラスタの時間をリセットする方法

Oracle Solaris Cluster ソフトウェアは、NTP を使用して、クラスタノード間の時間同期を維持しています。グローバルクラスタの時間の調整は、ノードが時間を同期するときに、必要に応じて自動的に行われます。詳細は、『[Oracle Solaris Cluster 4.3 Concepts Guide](#)』および <http://download.oracle.com/docs/cd/E19065-01/servers.10k/> の時間情報プロトコルのユーザーガイド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間を調整しないでください。date、rdate、または svcadm コマンドを使用した対話形式で、または cron スクリプト内で、時間を調整しないでください。詳細は、[date\(1\)](#)、[rdate\(1M\)](#)、[svcadm\(1M\)](#)、または [cron\(1M\)](#) のマニュアルページを参照してください。ntpd(1M) のマニュアルページは、service/network/ntp の Oracle Solaris 11 パッケージで配布されています。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. グローバルクラスタ内の任意のノードで root 役割になります。
2. グローバルクラスタを停止します。

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

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ブートします。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
# shutdown -g -y -i0
```

```
Press any key to continue
```

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

GRUB メニューが表示されます。

GRUB ベースのブートの詳細は、『Oracle Solaris 11.3 システムのブートとシャットダウン』の「システムのブート」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 ブートパラメータ画面が表示されます。

- 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ブートするように指定します。

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

```
grub edit> kernel$ /platform/i86pc/kernel/$ISADIR/unix_B $ZFS-BOOTFS -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。

- e. **b** と入力して、ノードを非クラスタモードでブートします。

注記 - カーネルブートパラメータコマンドへのこの変更は、システムをブートすると無効になります。次にノードをリブートする際には、ノードはクラスタモードでブートします。クラスタモードではなく、非クラスタモードでブートするには、これらの手順を再度実行して、カーネルブート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間を設定します。

```
phys-schost# date HHMM.SS
```

6. ほかのマシンで、**rdate(1M)** コマンドを実行し、時間をこのノードに同期化します。

```
phys-schost# rdate hostname
```

7. 各ノードを起動し、クラスタを再起動します。

```
phys-schost# reboot
```

8. すべてのクラスタノードで変更が行われたことを確認します。

各ノードで、**date** コマンドを実行します。

```
phys-schost# date
```

▼ SPARC: ノードで OpenBoot PROM (OBP) を表示する方法

OpenBoot™ PROM 設定を構成または変更する必要がある場合は、この手順を使用します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

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、それから `shutdown` コマンドを使用することで、クラスタノードを正常に停止します。

`clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、グローバルクラスタの指定されたノードから次に優先されるノードへ、すべてのリソースグループを切り替えます。

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



注意 - クラスタコンソールで `send brk` を使用して、クラスタノードをシャットダウンしないでください。

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、この手順を使用します。

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、`clusternodenodeid-priv` です (`clusternode3-priv` な

ど)。名前がすでにドメイン内で使用されている場合にかぎり、プライベートホスト名を変更します。



注意 - 新しいプライベートホスト名には IP アドレスを割り当てないでください。クラスタソフトウェアがそれらを割り当てます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. クラスタ内のすべてのノード上で、プライベートホスト名をキャッシュする可能性があるデータサービスリソースやアプリケーションをすべて無効にします。

```
phys-schost# clresource disable resource[,...]
```

無効にするアプリケーションには次のようなものがあります。

- HA-DNS と HA-NFS サービス (構成している場合)
- プライベートホスト名を使用するためにカスタム構成されたアプリケーション
- クライアントがプライベートインターコネクト経由で使用しているアプリケーション

clresource コマンドの使用については、[clresource\(1CL\)](#) のマニュアルページおよび『[Oracle Solaris Cluster 4.3 データサービス計画および管理ガイド](#)』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 NTP デーモンを停止します。

svcadm コマンドを使用して、NTP デーモンをシャットダウンします。NTP デーモンについての詳細は、[svcadm\(1M\)](#) のマニュアルページを参照してください。

```
phys-schost# svcadm disable ntp
```

3. **clsetup** 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 つのノードでのみユーティリティを実行します。詳細は、[clsetup\(1CL\)](#) のマニュアルページを参照してください。

注記 - 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。

clsetup ユーティリティの代わりに clnode コマンドを実行して、プライベートホスト名を変更することもできます。次の例では、クラスタノード名は phys-schost-1 です。次の clnode コマンドを実行したら、[ステップ 6](#)に進みます。

```
phys-schost# clnode set -p privatehostname=New-private-nodename phys-schost-1
```

4. **clsetup** ユーティリティで、プライベートホスト名のオプションの番号を入力します。

5. **clsetup** ユーティリティーで、プライベートホスト名を変更するためのオプションの番号を入力します。

表示される質問に答えます。変更しようとしているプライベートホスト名のノード名 (clusternodeid-priv) および新しいプライベートホスト名を要求されます。

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しないようにします。

```
phys-schost# nscd -i hosts
```

7. **NTP 構成またはインクルードファイルでプライベートホスト名を変更した場合は、各ノードの NTP ファイルを更新します。**

NTP 構成ファイル (/etc/inet/ntp.conf) 内のプライベートホスト名を変更し、NTP 構成ファイル (/etc/inet/ntp.conf.include) にピアホストエントリまたはピアホストのインクルードファイルへのポインタがある場合は、各ノードのこのファイルを更新します。NTP インクルードファイルのプライベートホスト名を変更した場合は、各ノードの /etc/inet/ntp.conf.sc ファイルを更新します。

- 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。通常、ntp.conf.sc ファイルは各クラスタノード上で同じです。

- b. **すべてのクラスタノードから新しいプライベートホスト名に ping を実行できることを確認します。**

- c. **NTP デーモンを再起動します。**

クラスタの各ノードで次の手順を実行します。

NTP デーモンを再起動するには、svcadm コマンドを使用します。

```
# svcadm enable svc:network/ntp:default
```

8. **ステップ 1 で無効にしたデータサービスリソースとほかのアプリケーションをすべて有効にします。**

```
phys-schost# clresource enable resource[,...]
```

clresource コマンドの使用については、[clresource\(1CL\)](#) のマニュアルページおよび『[Oracle Solaris Cluster 4.3 データサービス計画および管理ガイド](#)』を参照してください。

例 81 プライベートホスト名を変更する

次に、ノード phys-schost-2 上のプライベートホスト名 clusternode2-priv を clusternode4-priv に変更する例を示します。各ノードでこの操作を実行します。

```

Disable all applications and data services as necessary
phys-schost-1# svcadm disable ntp
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# pfedit /etc/inet/ntp.conf.sc
...
peer clusternode1-priv
peer clusternode4-priv
peer clusternode3-priv
phys-schost-1# ping clusternode4-priv
phys-schost-1# svcadm enable ntp
Enable all applications and data services disabled at the beginning of the procedure

```

▼ ノード名を変更する

Oracle Solaris Cluster 構成の一部であるノードの名前を変更できます。ノード名を変更する前に Oracle Solaris ホスト名を変更する必要があります。ノード名を変更するには、`clnode rename` コマンドを使用します。

次の説明は、グローバルクラスタで動作しているすべてのアプリケーションに該当します。

1. **グローバルクラスタでは、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。**
2. **(オプション) パートナーシップ関係にある Oracle Solaris Cluster Geographic Edition クラスタ内のノードの名前を変更する場合は、保護グループを切り替えるかどうかを判定します。** 名前変更手順を実行しているクラスタが保護グループのプライマリで、保護グループのアプリケーションをオンラインにしておく場合は、名前変更手順を実行している間、保護グループをセカンダリクラスタに切り替えることができます。Geographic Edition クラスタおよびノードの詳細は、『[Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide](#)』の第 5 章、「[Administering Cluster Partnerships](#)」を参照してください。
3. **Oracle Solaris ホスト名を変更します。** 『[Oracle Solaris 11.3 でのシステム情報、プロセス、およびパフォーマンスの管理](#)』の「[システムのアイデンティティの変更方法](#)」の手順を完了します。ただし、手順の最後にあるレポートは実行しません。代わりに、この手順を完了したあと、クラスタの停止を実行します。
4. **すべてのクラスタノードを非クラスタモードでブートします。**

```
ok> boot -x
```

5. Oracle Solaris のホスト名を変更したノード上で、非クラスタモードでノードの名前を変更し、名前を変更したホストごとに `cmd` コマンドを実行します。
1 回につき 1 つのノード名を変更します。

```
# clnode rename -n new-node old-node
```

6. クラスタで実行されるアプリケーション内の以前のホスト名への既存の参照を更新します。
7. コマンドメッセージとログファイルをチェックして、ノード名が変更されたことを確認します。
8. すべてのノードをクラスタモードでリブートします。

```
# sync;sync;sync;reboot
```

9. ノードに新しい名前が表示されていることを確認します。

```
# clnode status -v
```

10. 新しいクラスタノード名を使用するように Geographic Edition 構成を更新します。
保護グループやデータレプリケーション製品によって使用されている構成情報で、このノード名が指定される可能性があります。
11. 論理ホスト名リソースの `hostnameList` プロパティを変更することを選択できます。
このオプションな手順については、[236 ページの「既存の Oracle Solaris Cluster の論理ホスト名リソースで使用されている論理ホスト名を変更する」](#)を参照してください。

▼ 既存の Oracle Solaris Cluster の論理ホスト名リソースで使用されている論理ホスト名を変更する

[235 ページの「ノード名を変更する」](#)の手順に従ってノード名を変更する前またはあとに、論理ホスト名リソースの `HostnameList` プロパティを変更することもできます。この手順はオプションです。

1. グローバルクラスタでは、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. 必要に応じて、既存の Oracle Solaris Cluster の論理ホスト名リソースのいずれかで使用されている論理ホスト名を変更できます。
次の手順は、新しい論理ホスト名と連動するように `apache-lh-res` リソースを構成する方法を示したもので、クラスタモードで実行する必要があります。
 - a. クラスタモードで、論理ホスト名を含む Apache リソースグループをオフラインにします。

```
# clresourcegroup offline apache-rg
```

- b. Apache 論理ホスト名リソースを無効にします。

```
# clresource disable apache-lh-res
```

- c. 新しいホスト名リストを指定します。

```
# clresource set -p HostnameList=test-2 apache-lh-res
```

- d. `hostnameList` プロパティの以前のエントリに対するアプリケーションの参照を、新しいエントリを参照するように変更します。

- e. 新しい Apache 論理ホスト名リソースを有効にします。

```
# clresource enable apache-lh-res
```

- f. Apache リソースグループをオンラインにします。

```
# clresourcegroup online -eM apache-rg
```

- g. 次のコマンドを実行してクライアントをチェックし、アプリケーションが正しく起動したことを確認します。

```
# clresource status apache-rs
```

▼ ノードを保守状態にする

グローバルクラスタノードを長期間にわたって非稼働状態にする場合は、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。ノードを保守状態にするには、そのノードを `clnode evacuate` および `shutdown` コマンドで停止する必要があります。詳細は、[clnode\(1CL\)](#) および [cluster\(1CL\)](#) のマニュアルページを参照してください。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ノードを退避し、すべてのリソースグループおよびデバイスグループを次に優先されるノードに切り替えることもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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。

注記 - Oracle Solaris の `shutdown` コマンドが 1 つのノードを停止するのに対して、`cluster shutdown` コマンドはクラスタ全体を停止します。

クラスタノードを保守状態にするには、まだクラスタメンバーになっている別のノードから `clquorum disable` コマンドを使用します。詳細は、[clquorum\(1CL\)](#) のマニュアルページを参照してください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. **保守状態にするグローバルクラスタノード上で、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。**
2. **ノードからすべてのリソースグループおよびデバイスグループを退避させます。**
`clnode evacuate` コマンドは、すべてのリソースグループおよびデバイスグループを、指定ノードから次に優先されるノードに切り替えます。

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

3. **退避させたノードをシャットダウンします。**

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

4. **クラスタ内の別のノード上で、RBAC の承認 `solaris.cluster.modify` を提供する役割になり、[ステップ 3](#) でシャットダウンしたノードを保守状態にします。**

```
phys-schost# clquorum disable node
```

5. **グローバルクラスタノードが保守状態にあることを確認します。**

```
phys-schost# clquorum status node
```

保守状態にしたノードの Status は `offline`、その Present と Possible の定足数投票数は 0 (ゼロ) にすることをお勧めします。

例 82 グローバルクラスタノードを保守状態にする

次に、クラスタノードを保守状態にして、その結果を確認する例を示します。`clnode status` の出力では、`phys-schost-1` のノードの `Node votes` は 0 (ゼロ) で、そのステータスは `Offline` です。Quorum Summary では、投票数も減っているはずですが、構成によって異なりますが、Quorum Votes by Device の出力では、いくつかの定足数ディスクデバイスもオフラインである可能性があります。

[On the node to be put into maintenance state:]

```
phys-schost-1# clnode evacuate phys-schost-1
phys-schost-1# shutdown -g0 -y -i0
```

```
[On another node in the cluster:]
phys-schost-2# clquorum disable phys-schost-1
phys-schost-2# clquorum status phys-schost-1
```

```
-- Quorum Votes by Node --
```

Node Name	Present	Possible	Status
phys-schost-1	0	0	Offline
phys-schost-2	1	1	Online
phys-schost-3	1	1	Online

参照 ノードをオンライン状態に戻す方法については、[239 ページの「ノードを保守状態から戻す」](#)を参照してください。

▼ ノードを保守状態から戻す

次の手順を使用して、グローバルクラスタノードをオンラインに戻し、定足数投票数をリセットしてデフォルト設定に戻します。クラスタノードのデフォルトの投票数は 1 です。定足数デバイスのデフォルトの定足数は $N-1$ です (N は、投票数が 0 以外で、定足数デバイスへのポートを持つノードの数を示します)。

ノードが保守状態になると、そのノードの投票数は 1 つ減ります。また、このノードのポートに構成されているすべての定足数デバイスの投票数も (1 つ) 減ります。投票数がリセットされ、ノードが保守状態から戻されると、ノードの投票数と定足数デバイスの投票数の両方が 1 つ増えます。

保守状態にしたグローバルクラスタノードを保守状態から戻した場合は、必ずこの手順を実行してください。



注意 - globaldev または node オプションのどちらも指定しない場合、定足数投票数はクラスタ全体でリセットされます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. グローバルクラスタの、保守状態のもの以外の任意のノード上で、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
2. グローバルクラスタ構成内にあるノードの数に応じて、次の手順のいずれかを実行します。

- クラスタ構成のノードが 2 つの場合は、[ステップ 4](#) に進みます。
 - クラスタ構成のノードが 2 つより多い場合は、[ステップ 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、そのノードをリブートする前である必要があります。そうしないと、定足数の確立を待機中にハングアップすることがあります。

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

4. 保守状態を解除するノードをブートします。
5. 定足数投票数を確認します。

```
phys-schost# clquorum status
```

保守状態を解除したノードのステータスは online、その Present と Possible の定足数投票数は適切な値にすることをお勧めします。

例 83 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。cluster status の出力は、phys-schost-1 の Node votes は 1 で、そのステータスは online であると表示します。Quorum Summary では投票数も増加しているはずですが。

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

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な Oracle Solaris エントリを選択し、Enter キーを押します。

```
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

▼ クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法

完全に確立されたクラスタ構成からグローバルクラスタノードを切り離す前に、この手順を実行して、そのノードから Oracle Solaris Cluster ソフトウェアを構成解除します。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。

注記 - クラスタにまだ参加していない、またはまだインストールモードであるノードから Oracle Solaris Cluster ソフトウェアをアンインストールする場合、この手順を実行してはいけません。その代わりに、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の「インストールの問題を修正する方法ために Oracle Solaris Cluster ソフトウェアを構成解除する方法」に進みます。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. **タスクマップにあるクラスタノードを削除するための前提タスクがすべて完了していることを確認します。**

[表16「タスクマップ: ノードの削除」](#)を参照してください。

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。その他の手順には、クラスタのノード認証リストへのアンインストール対象のノードの追加、ゾーンクラスタのアンインストールなどが含まれる場合があります。

注記 - ノードを構成解除して、Oracle Solaris Cluster ソフトウェアはノードにインストールしたままにするには、`clnode remove` コマンドを実行したあとに先に進まないでください。

2. **アンインストールするノードで root 役割になります。**

3. ノードにグローバルデバイス名前空間用の専用パーティションがある場合、グローバルクラスタノードを非クラスタモードでリブートします。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
# shutdown -g0 -y -i0 ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
# shutdown -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
```

4. /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 (/global/.devices グローバルマウントを除く)。

5. ノードをリブートして非クラスタモードにします。

- SPARC ベースのシステムで、次のコマンドを実行します。

```
ok boot -x
```

- x86 ベースのシステムで、次のコマンドを実行します。

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

GRUB ベースのブートの詳細は、『[Oracle Solaris 11.3 システムのブートとシャットダウン](#)』の「[システムのブート](#)」を参照してください。

- b. ブートパラメータ画面で矢印キーを使用して kernel エントリを選択し、e と入力してエントリを編集します。

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

- d. Enter キーを押して変更を承諾し、ブートパラメータ画面に戻ります。

画面には編集されたコマンドが表示されます。

e. **b** と入力して、ノードを非クラスタモードでブートします。

注記 - カーネルブートパラメータコマンドへのこの変更は、システムをブートすると無効になります。次にノードをリブートする際には、ノードはクラスタモードでブートします。非クラスタモードでブートするには、上記の手順を実行してもう一度カーネルのブートパラメータコマンドに `-x` オプションを追加してください。

6. Oracle Solaris Cluster パッケージのファイルが何も含まれていない、`root (/)` ディレクトリなどのディレクトリへ移動します。

```
phys-schost# cd /
```

7. ノードを構成解除し、Oracle Solaris Cluster ソフトウェアを削除するには、次のコマンドを実行します。

```
phys-schost# scinstall -r [-b bename]
```

`-r`

クラスタノードから、クラスタの構成情報を削除し、Oracle Solaris Cluster のフレームワークおよびデータサービスソフトウェアをアンインストールします。その後、このノードを再インストールしたり、クラスタから削除したりできます。

`-b bootenvironmentname`

アンインストール処理の完了後のブート先となる新しいブート環境の名前を指定します。名前の指定はオプションです。ブート環境の名前を指定しなかった場合は、名前が自動的に生成されます。

詳細は、[scinstall\(1M\)](#) のマニュアルページを参照してください。

8. アンインストールが完了したあとに、このノードに Oracle Solaris Cluster ソフトウェアを再インストールするつもりの場合は、ノードをリブートして新しいブート環境でブートします。
9. このクラスタ上で Oracle Solaris Cluster ソフトウェアを再インストールするつもりでない場合は、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. 切断の手順については、ホストアダプタおよびサーバーに付属しているドキュメントに従います。

ヒント - グローバルデバイス名前空間の `lofi` デバイスへの移行の詳細については、[124 ページの「グローバルデバイス名前空間を移行する」](#)を参照してください。

ノードのアンインストールのトラブルシューティング

ここでは、`clnode remove` コマンドを実行したときに出力される可能性があるエラーメッセージとその対処方法について説明します。

削除されないクラスタファイルシステムのエン트리

次のエラーメッセージは、削除したグローバルクラスタノードに、`vfstab` ファイルから参照されているクラスタファイルシステムがまだあることを示しています。

```
Verifying that no unexpected global mounts remain in /etc/vfstab ... failed
clnode: global-mount1 is still configured as a global mount.
clnode: global-mount1 is still configured as a global mount.
clnode: /global/dg1 is still configured as a global mount.

clnode: It is not safe to uninstall with these outstanding errors.
clnode: Refer to the documentation for complete uninstall instructions.
clnode: Uninstall failed.
```

このエラーを修正するためには、[241 ページの「クラスタノードから Oracle Solaris Cluster ソフトウェアをアンインストールする方法」](#)に戻って、その手順を繰り返す必要があります。`clnode remove` コマンドを再度実行する前に、手順の [ステップ 4](#) が正しく完了していることを確認してください。

デバイスグループに削除されていないリストがある場合

次のエラーメッセージは、削除したノードが依然としてデバイスグループにリストされていることを示しています。

```
Verifying that no device services still reference this node ... failed
clnode: This node is still configured to host device service "service".
clnode: This node is still configured to host device service "service2".
clnode: This node is still configured to host device service "service3".
clnode: This node is still configured to host device service "dg1".

clnode: It is not safe to uninstall with these outstanding errors.
```

clnode: Refer to the documentation for complete uninstall instructions.
 clnode: Uninstall failed.

Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理

ここでは、SNMP イベント管理情報ベース (MIB) を作成、設定、および管理する方法を説明します。またこのセクションでは、Oracle Solaris 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。

Oracle Solaris Cluster ソフトウェアでは現在、イベント MIB という MIB を 1 つサポートしています。SNMP マネージャーソフトウェアがクラスターイベントをリアルタイムでトラップします。有効な場合、SNMP マネージャー はトラップ通知を `clsnmphost` コマンドによって定義されているすべてのホストに自動的に送信します。クラスターは多数の通知を生成するため、`min_severity` 以上の重要度を持つイベントのみがトラップ通知として送信されます。デフォルトでは、`min_severity` 値は NOTICE に設定されています。`log_number` の値には、古いエントリを破棄するまでに MIB テーブルに記録するイベントの数を指定します。MIB は、トラップが送信された最新のイベントの読み取り専用テーブルを維持しています。イベントの数は、`log_number` 値によって制限されます。この情報は、リブ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。

Cluster Event SNMP のインタフェースは、共通のエージェントコンテナ (cacao) SNMP アダプタをその SNMP エージェントインフラストラクチャーとして使用します。デフォルトでは、SNMP のポート番号は 11161 で、SNMP トラップのデフォルトのポート番号は 11162 です。これらのポート番号は、`cacaoadm` コマンドを使用して変更できます。詳細は、`cacaoadm(1M)` のマニュアルページを参照してください。

Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理には、次のタスクが関係する可能性があります。

表 18 タスクマップ: Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理

タスク	手順
SNMP イベント MIB の有効化	246 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	246 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	247 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	248 ページの「SNMP ホストがノード上の SNMP トラップを受信できるようにする」
SNMP ホストの削除	249 ページの「SNMP ホストがノード上の SNMP トラップを受信できないようにする」
SNMP ユーザーの追加	250 ページの「SNMP ユーザーをノードに追加する」

タスク	手順
SNMP ユーザーの削除	251 ページの「SNMP ユーザーをノードから削除する」

▼ SNMP イベント MIB を有効にする

この手順では、SNMP イベント MIB を有効化する方法を説明します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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。この場合、MIB 名は `event` にしてください。

▼ SNMP イベント MIB を無効にする

この手順では、SNMP イベント MIB を無効化する方法を説明します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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。この場合、*event* を指定してください。

▼ SNMP イベント MIB を変更する

この手順では、SNMP イベント MIB のプロトコル、最小重要度値、およびイベントロギングを変更する方法を示しています。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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、重要度の最小値、およびイベントロギングを変更します。

```
phys-schost-1# clsnmpmib set -n node
-p version=SNMPv3 \
-p min_severity=WARNING \
-p log_number=100 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せん。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`

-p min_severity=value

MIB で使用する重要度の最小値を指定します。*value* は次のように指定します。

- `min_severity=NOTICE`
- `min_severity=WARNING`
- `min_severity=ERROR`

- `min_severity=CRITICAL`
- `min_severity=FATAL`

`-p log_number=number`

古いエントリを破棄するまでに MIB テーブルに記録するイベントの数を指定します。デフォルト値は 100 です。値は 100-500 の範囲である必要があります。*value* は次のように指定します: `log_number=100`。

MIB

サブコマンドが適用される単数または複数の MIB の名前を指定します。この場合、*event* を指定してください。このオペランドを指定しない場合は、サブコマンドが、すべての MIB を意味するデフォルトのプラス記号 (+) を使用します。*MIB* オペランドを使用する場合は、ほかのすべてのコマンド行オプションのあとで、MIB を空白区切りのリスト内に指定します。

詳細は、[clsnmpmib\(1CL\)](#) のマニュアルページを参照してください。

▼ SNMP ホストがノード上の SNMP トラップを受信できるようにする

この手順では、ノード上の SNMP ホストを、MIB のトラップ通知を受信するホストのリストに追加する方法を示します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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# clsnmphost 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。このホストは、トラップを受信するように構成できるネットワーク内のシステムです。

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が付与された SNMP ホストのクラスタ *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場合、デフォルトはコマンドが実行されるノードです。

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。これは、クラスタの外部にあるホスト、または SNMP トラップを取得しようとしているクラスタノード自体のどちらでもかまいません。

▼ SNMP ホストがノード上の SNMP トラップを受信できないようにする

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

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. **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# clsnmhost 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

SNMP ホストが構成から削除されるクラスタ *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場合、デフォルトはコマンドが実行されるノードです。

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。これは、クラスタの外部にあるホスト、または SNMP トラップを取得しようとしているクラスタノード自体のどちらでもかまいません。

指定の SNMP コミュニティー内のすべてのホストを削除するには、`-c` オプション付きの `host` に正符号 (+) を使用します。すべてのホストを削除するには、`host` に正符号 + を使用します。

▼ SNMP ユーザーをノードに追加する

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

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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# clsnmpuser 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せん。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`-a authentication`

ユーザーの承認に使用する認証プロトコルを指定します。認証プロトコルの値は、SHA または MD5 です。

`-f password`

SNMP ユーザーパスワードを含むファイルを指定します。新しいユーザーを作成する際にこのオプションを指定しないと、コマンドはパスワードを求めるプロンプトを表示します。このオプションは、`add` サブコマンドとだけ有効です。

ユーザーパスワードは、次の形式で、独立した行の上に指定します。

```
user:password
```

パスワードには次に示す文字または空白文字を含めることはできません。

- ; (セミコロン)
- : (コロン)
- \ (バックスラッシュ)
- \n (復帰改行)

`user`

追加する SNMP ユーザーの名前を指定します。

▼ SNMP ユーザーをノードから削除する

この手順では、ノード上の SNMP ユーザー構成から SNMP ユーザーを削除する方法を示します。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. 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。

負荷制限の構成

負荷制限を設定することによって、ノードをまたがるリソースグループ負荷の自動分散を有効にできます。一連の負荷制限はクラスタノードごとに設定できます。リソースグループに負荷係数を割り当てると、その負荷係数はノードの定義済み負荷制限に対応します。デフォルトの動作では、リソースグループの負荷がそのリソースグループのノードリスト内の使用可能なすべてのノードに均等に分散されます。

リソースグループは RGM によってリソースグループのノードリストのノード上で起動されるため、ノードの負荷制限を超えることはありません。RGM によってリソースグループがノードに割り当てられると、各ノードのリソースグループの負荷係数が合計され、合計負荷が算出されます。次に、合計負荷がそのノードの負荷制限と比較されます。

負荷制限は次の項目から構成されます。

- ユーザーが割り当てた名前。
- 弱い制限値 - 弱い負荷制限は一時的に超えることができます。
- 強い負荷制限 - 強い負荷制限は超えることはできず、厳格に適用されます。

1 つのコマンドで強い制限と弱い制限の両方を設定できます。いずれかの制限が明示的に設定されていない場合は、デフォルト値が使用されます。各ノードの強いおよび弱い負荷制限値は、`clnode create-loadlimit`、`clnode set-loadlimit`、および `clnode delete-loadlimit` コマンドで作成および変更されます。詳細は、[clnode\(1CL\)](#) のマニュアルページを参照してください。

高い優先度を持つようにリソースグループを構成すると、特定のノードから移動させられる可能性が低くなります。preemption_mode プロパティを設定して、ノードの過負荷が原因であるリソースグループが優先度の高いリソースグループによってノードから横取りされるかどうかを判定することもできます。concentrate_load プロパティを使用して、リソースグループの負荷をできるだけ少ないノードに集中させることもできます。concentrate_load プロパティは、デフォルトで FALSE です。

注記 - 負荷制限は、グローバルクラスタまたはゾーンクラスタのノード上で構成できます。負荷制限を構成するには、コマンド行、clsetup ユーティリティ、または Oracle Solaris Cluster Manager ブラウザインタフェースを使用できます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。次の手順は、コマンド行を使用して負荷制限を構成する方法を示しています。

▼ ノードに負荷制限を構成する

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、グローバルクラスタノードまたはゾーンクラスタノードの負荷制限を作成および構成したり、ノードの既存の負荷制限を編集または削除したりすることもできます。「ノード」または「ゾーンクラスタ」をクリックし、そのページにアクセスするノードの名前をクリックします。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. **グローバルクラスタの任意のノードで、RBAC の承認 solaris.cluster.modify を提供する役割になります。**
2. **負荷分散を使用するノードに対して、負荷制限を作成および設定します。**

次のコマンド例では、ゾーンクラスタ名は zc1 です。サンプルプロパティは mem_load と呼ばれ、弱い負荷制限は 11、強い負荷制限は 20 です。強い制限と弱い制限はオプションの引数で、特に定義しなかった場合、デフォルトは無制限です。詳細は、[clnode\(1CL\)](#) のマニュアルページを参照してください。

```
# clnode create-loadlimit -p limitname=mem_load -Z zc1 \  
-p softlimit=11 -p hardlimit=20 node1 node2 node3
```

3. **負荷係数値を各リソースグループに割り当てます。**

次のコマンド例では、rg1 および rg2 の 2 つのリソースグループに負荷係数が設定されます。負荷係数の設定は、ノードの定義済み負荷制限に対応します。

```
# clresourcegroup set -p load_factors=mem_load@50,factor2@1 rg1 rg2
```

この手順は、リソースグループの作成中に clresourcegroup create コマンドを使用して実行することもできます。詳細は、[clresourcegroup\(1CL\)](#) のマニュアルページを参照してください。

4. 必要に応じて、1 つ以上の追加のオプションの構成タスクを実行します。

■ 既存の負荷を再分散します。

```
# clresourcegroup remaster rg1 rg2
```

このコマンドにより、リソースグループを現在のマスターからほかのノードに移動し、均等な負荷分散を実現できます。

■ いくつかのリソースグループにほかより高い優先順位を割り当てます。

```
# clresourcegroup set -p priority=600 rg1
```

デフォルトの優先度は 500 です。優先度の値が高いリソースグループは、ノードの割り当てにおいて、優先度の値が低いリソースグループよりも優先されます。

■ Preemption_mode プロパティを設定します。

```
# clresourcegroup set -p Preemption_mode=No_cost rg1
```

HAS_COST、NO_COST、および NEVER オプションについては、[clresourcegroup\(1CL\)](#) のマニュアルページを参照してください。

■ Concentrate_load フラグを設定します。

```
# cluster set -p Concentrate_load=TRUE
```

■ リソースグループ間のアフィニティを指定します。

強い正または負のアフィニティは負荷分散より優先されます。強いアフィニティや強い負荷制限が無効になることはありません。強いアフィニティと強い負荷制限の両方を設定すると、両方の制限が満たされなかった場合に一部のリソースグループが強制的にオフラインのままになることがあります。

次の例では、ゾーンクラスタ `zc1` のリソースグループ `rg1` とゾーンクラスタ `zc2` のリソースグループ `rg2` の間の強い正のアフィニティを指定しています。

```
# clresourcegroup set -p RG_affinities=++zc2:rg2 zc1:rg1
```

5. クラスタ内のすべてのグローバルクラスタノードとゾーンクラスタノードのステータスを確認します。

```
# clnode status -Z all -v
```

出力には、ノードで定義された負荷制限設定がすべて含まれます。

ゾーンクラスタ管理タスクの実行

ゾーンパスの移動、アプリケーションを実行するためのゾーンクラスタの準備、ゾーンクラスタのクローニングなどの、ゾーンクラスタに関するその他の管理タスクを実行できます。これらのコマンドは、グローバルクラスタのノードから実行する必要があります。

`clsetup` ユーティリティを使用してゾーンクラスタの構成ウィザードを起動することによって、新しいゾーンクラスタを作成したり、既存のゾーンクラスタにファイルシステムまたはストレージデバイスを追加したりできます。`clzonecluster install -c` を実行してプロファイルを構成すると、ゾーンクラスタのゾーンが構成されます。`clsetup` ユーティリティまたは `-c config_profile` オプションの使用手順については、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[ゾーンクラスタの作成および構成](#)」を参照してください。

Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタを作成したり、ファイルシステムまたはストレージデバイスをそれに追加したりすることもできます。また、Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタのリソースセキュリティプロパティを編集することもできます。「ゾーンクラスタ」をクリックし、そのページに移動するゾーンクラスタの名前をクリックして、「Solaris リソース」タブをクリックし、ゾーンクラスタコンポーネントを管理するか、または「プロパティ」をクリックして、ゾーンクラスタのプロパティを管理します。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

注記 - グローバルクラスタ内のノードからのみ実行する Oracle Solaris Cluster コマンドは、ゾーンクラスタでは使用できません。ゾーンクラスタでのコマンドの有効な使用方法については、Oracle Solaris Cluster の該当するマニュアルページを参照してください。

表 19 その他のゾーンクラスタのタスク

タスク	手順
新規ゾーンパスへのゾーンパスの移動	<code>clzonecluster move -f zonepath zone-cluster-name</code>
アプリケーション実行用のゾーンクラスタの準備	<code>clzonecluster ready -n nodename zone-cluster-name</code>
統合アーカイブからノードを復元する	212 ページの「統合アーカイブからノードを復元する方法」
統合アーカイブからゾーンクラスタを構成またはインストールする	255 ページの「統合アーカイブからゾーンクラスタを構成する方法」 256 ページの「統合アーカイブからゾーンクラスタをインストールする方法」
	次のコマンドを使用します。
	<code>clzonecluster clone -Z target- zone-cluster-name [-m copymethod] source-zone-cluster-name</code>
	<code>clone</code> サブコマンドを使用する前に、ソースゾーンクラスタを停止してください。複製先のゾーンクラスタは、構成済みである必要があります。
ゾーンクラスタへのネットワークアドレスの追加	257 ページの「ゾーンクラスタにネットワークアドレスを追加する方法」

タスク	手順
ゾーンクラスタへのノードの追加	210 ページの「既存のクラスタまたはゾーンクラスタにノードを追加する方法」
ゾーンクラスタからノードを削除	217 ページの「ゾーンクラスタからノードを削除する方法」
ゾーンクラスタの削除	259 ページの「ゾーンクラスタを削除する」
ゾーンクラスタからファイルシステムを削除	260 ページの「ゾーンクラスタからファイルシステムを削除する」
ゾーンクラスタからストレージデバイスを削除	262 ページの「ゾーンクラスタからストレージデバイスを削除する」
統合アーカイブからゾーンクラスタノードを復元する	212 ページの「統合アーカイブからノードを復元する方法」
ノードのアンインストールに関するトラブルシューティング	244 ページの「ノードのアンインストールのトラブルシューティング」
Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理	245 ページの「Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理」

▼ 統合アーカイブからゾーンクラスタを構成する方法

統合アーカイブから solaris10 または labeled ブランドゾーンクラスタを構成するには、`clzonecluster` コマンドを使用して対話式ユーティリティーを起動します。`clzonecluster configure` ユーティリティーでは、復旧用のアーカイブまたはクローンアーカイブを指定できます。

対話型ユーティリティーではなく、コマンド行を使用してアーカイブからゾーンクラスタを構成する場合は、`clzonecluster configure -f command-file` コマンドを使用します。詳細は、[clzonecluster\(1CL\)](#) のマニュアルページを参照してください。

注記 - インストールしようとしているゾーンクラスタが、サポートされているほかの方法を使用しすでに構成されている場合は、統合アーカイブからゾーンクラスタを構成する必要はありません。

1. 復旧用のアーカイブまたはクローンアーカイブを作成します。

```
phys-schost# archiveadm create -r archive-location
```

`create` コマンドを使用してクローンアーカイブを作成するか、または `-r` オプションを使用して復旧用のアーカイブを作成します。`archiveadm` コマンドの使用の詳細は、[archiveadm\(1M\)](#) のマニュアルページを参照してください。

2. ゾーンクラスタをホストするグローバルクラスタのノード上で `root` 役割になります。
3. 統合アーカイブ内の復旧用のアーカイブまたはクローンアーカイブからゾーンクラスタを構成します。

```
phys-schost-1# clzonecluster configure zone-cluster-name
```

`clzonecluster configure zone-cluster-name` コマンドによって対話型ユーティリティーが起動され、そこで `create -a archive [other-options-such-as-"x"]` を指定できます。このアーカイブは、クローンアーカイブまたは復旧用のアーカイブのどちらでもかまいません。

注記 - ゾーンクラスタを作成するには、構成にゾーンクラスタメンバーを追加しておく必要があります。

`configure` サブコマンドは `zonecfg` コマンドを使用して、指定されたそれぞれのマシンでゾーンを構成します。`configure` サブコマンドにより、ゾーンクラスタの各ノードに適用するプロパティを指定できます。これらのプロパティは、個別ゾーンの `zonecfg` コマンドによって確立された場合と同じ意味を持ちます。`configure` サブコマンドは `zonecfg` コマンドには分からないプロパティの構成をサポートします。`-f` オプションを指定しない場合、`configure` サブコマンドは対話型シェルを起動します。`-f` オプションは、その引数としてコマンドファイルを取ります。`configure` サブコマンドはこのファイルを使用して、ゾーンクラスタを非対話型で作成または変更します。

▼ 統合アーカイブからゾーンクラスタをインストールする方法

統合アーカイブからゾーンクラスタをインストールできます。`clzonecluster install` ユーティリティーでは、アーカイブの絶対パスか、またはインストールに使用する Oracle Solaris 10 イメージアーカイブを指定できます。サポートされているアーカイブタイプの詳細は、[solaris10\(5\)](#) のマニュアルページを参照してください。アーカイブの絶対パスは、ゾーンクラスタがインストールされるクラスタのすべての物理ノードでアクセスできるようにしてください。統合アーカイブのインストールでは、復旧用のアーカイブまたはクローンアーカイブを使用できます。

対話型ユーティリティーではなく、コマンド行を使用してアーカイブからゾーンクラスタをインストールする場合は、`clzonecluster create -a archive -z archived-zone` コマンドを使用します。詳細は、[clzonecluster\(1CL\)](#) のマニュアルページを参照してください。

1. 復旧用のアーカイブまたはクローンアーカイブを作成します。

```
phys-schost# archiveadm create -r archive-location
```

`create` コマンドを使用してクローンアーカイブを作成するか、または `-r` オプションを使用して復旧用のアーカイブを作成します。`archiveadm` コマンドの使用の詳細は、[archiveadm\(1M\)](#) のマニュアルページを参照してください。

2. ゾーンクラスタをホストするグローバルクラスタのノード上で `root` 役割になります。
3. 統合アーカイブの復旧用のアーカイブまたはクローンアーカイブからゾーンクラスタをインストールします。

```
phys-schost-1# clzonecluster install -a absolute_path_to_archive zone-cluster-name
```

アーカイブの絶対パスは、ゾーンクラスタがインストールされるクラスタのすべての物理ノードでアクセスできるようにしてください。HTTPS 統合アーカイブの場所がある場合は、`-x cert|ca-cert|key=file` を使用して、SSL 証明書、認証局 (CA) 証明書、および鍵ファイルを指定します。

統合アーカイブにゾーンクラスタノードのリソースは含まれていません。ノードのリソースは、クラスタが構成される時に指定されます。統合アーカイブを使用して大域ゾーンからゾーンクラスタを構成する場合は、ゾーンパスを設定する必要があります。

統合アーカイブに複数のゾーンが含まれている場合は、`zone-cluster-name` を使用して、インストールのソースのゾーン名を指定します。詳細は、[clzonecluster\(1CL\)](#) のマニュアルページを参照してください。

注記 - 統合アーカイブの作成に使用したソースに Oracle Solaris Cluster パッケージが含まれていない場合は、`pkg install ha-cluster-packages` (`ha-cluster-minimal` や `ha-cluster-framework-full` などの特定のパッケージ名に置き換えてください) を実行する必要があります。ゾーンをブートし、`zlogin` コマンドを実行したあと、`pkg install` コマンドを実行する必要があります。このアクションによって、ターゲットゾーンクラスタにグローバルクラスタと同じパッケージがインストールされます。

4. 新しいゾーンクラスタをブートします。

```
phys-schost-1# clzonecluster boot zone-cluster-name
```

▼ ゾーンクラスタにネットワークアドレスを追加する方法

この手順では、既存のゾーンクラスタで使用するネットワークアドレスを追加します。ネットワークアドレスは、論理ホストまたは共有 IP アドレスリソースをゾーンクラスタに構成する場合に使用します。`clsetup` ユーティリティを複数回実行して、必要な数のネットワークアドレスを追加できます。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタにネットワークアドレスを追加することもできます。「ゾーンクラスタ」をクリックし、そのページに移動するゾーンクラスタの名前をクリックして、「Solaris リソース」タブをクリックし、ゾーンクラスタコンポーネントを管理します。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. そのゾーンクラスタをホストしているグローバルクラスタのノードで、`root` 役割になります。
2. グローバルクラスタ上で、ゾーンクラスタで使用するクラスタファイルシステムを構成します。
`clsetup` ユーティリティを起動します。

```
phys-schost# clsetup
```

メインメニューが表示されます。

3. 「ゾーンクラスタ」メニュー項目を選択します。
4. 「ゾーンクラスタにネットワークアドレスを追加」メニュー項目を選択します。
5. ネットワークアドレスを追加するゾーンクラスタを選択します。
6. 追加するネットワークアドレスを指定するためのプロパティを選択します。

```
address=value
```

論理ホストまたは共有 IP アドレスリソースをゾーンクラスタに構成する場合に使用されるネットワークアドレスを指定します。たとえば、192.168.100.101 などです。

次の種類のネットワークアドレスがサポートされています。

- 有効な IPv4 アドレス (オプションで / および接頭辞長が続く)。
- 有効な IPv6 アドレス (/ および接頭辞長が続く必要があります)。
- IPv4 アドレスに解決するホスト名。IPv6 アドレスに解決するホスト名はサポートされていません。

ネットワークアドレスの詳細は、[zonecfg\(1M\)](#) のマニュアルページを参照してください。

7. ネットワークアドレスを追加するには、**a** を入力します。
8. **c** を入力して、構成の変更を保存します。

構成の変更の結果が表示されます。例を示します。

```
>>> Result of Configuration Change to the Zone Cluster(sczone) <<<
```

```
Adding network address to the zone cluster...
```

```
The zone cluster is being created with the following configuration
```

```
/usr/cluster/bin/clzonecluster configure sczone
add net
set address=phys-schost-1
end
```

```
All network address added successfully to sczone.
```

9. 完了後 **clsetup** ユーティリティを終了します。

▼ ゾーンクラスタを削除する

グローバルクラスタ上に構成されているゾーンクラスタは、特定の 1 つのゾーンクラスタを削除することも、ワイルドカードを使用してすべてのゾーンクラスタを削除することもできます。構成されていないゾーンクラスタは、削除できません。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタを削除することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. グローバルクラスタのノードで、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

2. ゾーンクラスタからすべてのリソースグループとそれらのリソースを削除します。

```
phys-schost# clresourcegroup delete -F -Z zone-cluster-name +
```

注記 - この手順は、グローバルクラスタノードから実行されます。この手順をゾーンクラスタのノードから実行するには、ゾーンクラスタノードにログインし、コマンドの「-Z zonecluster」を省略します。

3. ゾーンクラスタを停止します。

```
phys-schost# clzonecluster halt zone-cluster-name
```

4. ゾーンクラスタをアンインストールします。

```
phys-schost# clzonecluster uninstall zone-cluster-name
```

5. ゾーンクラスタを構成解除します。

```
phys-schost# clzonecluster delete zone-cluster-name
```

例 84 グローバルクラスタからのゾーンクラスタの削除

```
phys-schost# clresourcegroup delete -F -Z sczone +
phys-schost# clzonecluster halt sczone
phys-schost# clzonecluster uninstall sczone
phys-schost# clzonecluster delete sczone
```

▼ ゾーンクラスタからファイルシステムを削除する

ファイルシステムをゾーンクラスタにエクスポートするには、直接マウントまたはループバックマウントを使用します。

ゾーンクラスタでは、次の直接マウントがサポートされます。

- UFS ローカルファイルシステム
- StorageTek QFS スタンドアロンファイルシステム
- StorageTek QFS 共有ファイルシステム (Oracle RAC のサポートに使用する場合)
- Oracle Solaris ZFS (データセットとしてエクスポート)
- サポートされている NAS デバイスの NFS

ゾーンクラスタでは、次のループバックマウントを管理できます。

- UFS ローカルファイルシステム
- StorageTek QFS スタンドアロンファイルシステム
- StorageTek QFS 共有ファイルシステム (Oracle RAC のサポートに使用する場合のみ)
- UFS クラスタファイルシステム

ファイルシステムのマウントを管理する `HAStoragePlus` または `ScalMountPoint` リソースを構成します。ファイルシステムをゾーンクラスタに追加する手順については、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の「[ゾーンクラスタにファイルシステムを追加する](#)」を参照してください。

ファイルシステムの `mountpoint` プロパティが `none` または `legacy` に設定されているか、その `canmount` プロパティが `off` に設定されている場合、`HAStoragePlus` リソースは ZFS ファイルシステムをモニターしません。ほかのすべての ZFS ファイルシステムでは、`HAStoragePlus` リソースの障害モニターが、システムがマウントされているかどうかを確認します。ファイルシステムがマウントされている場合、`HAStoragePlus` リソースは `ReadOnly/ReadWrite` と呼ばれる `I00option` プロパティの値に応じてファイルシステムの読み書きを行うことで、ファイルシステムのアクセシビリティをプローブします。

ZFS ファイルシステムがマウントされていないか、ファイルシステムのプローブが失敗した場合、リソース障害モニターは失敗し、リソースは `Faulted` に設定されます。RGM はファイルシステムを再起動しようとします (リソースの `retry_count` および `retry_interval` プロパティによって決定される)。先に説明した `mountpoint` プロパティと `canmount` プロパティの特定の設定が有効でない場合は、このアクションによってファイルシステムが再マウントされます。障害モニターが引き続き失敗し、`retry_interval` 内に `retry_count` を超えた場合、RGM はリソースを別のノードにフェイルオーバーします。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタからファイルシステムを削除することもできます。「ゾーンクラスタ」をクリックし、そのページに移動するゾーンクラスタの名前をクリックして、「Solaris リソース」タブをクリックし、ゾーンクラスタコンポーネントを管理します。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. そのゾーンクラスタをホストしているグローバルクラスタのノードで、**root** 役割になります。この手順のいくつかのステップはグローバルクラスタのノードから行います。他のステップは、ゾーンクラスタのノードから実行されます。
2. 削除するファイルシステムに関連するリソースを削除します。
 - a. 削除するゾーンクラスタのファイルシステム用に構成されている Oracle Solaris Cluster リソースタイプ (HASStoragePlus、SUNW.ScalMountPoint など) を特定し、削除します。

```
phys-schost# clresource delete -F -Z zone-cluster-name fs_zone_resources
```

- b. 削除するファイルシステム用のグローバルクラスタ内に構成されている SUNW.qfs タイプの Oracle Solaris Cluster リソースがあれば、そのリソースを特定し、削除します。

```
phys-schost# clresource delete -F fs_global_resources
```

-F オプションを指定すると、前もって無効にしているリソースも含め、指定したリソースがすべて強制的に削除されるため、このオプションは注意して使用してください。すべての指定リソースが、ほかのリソースのリソース関係設定から削除されるため、クラスタ内のサービスが失われることがあります。削除されていない依存リソースは、無効な状態やエラー状態になる可能性があります。詳細は、[clresource\(1CL\)](#) のマニュアルページを参照してください。

ヒント - 削除したリソースのリソースグループがあとで空になると、そのリソースグループを安全に削除できます。

3. ファイルシステムのマウントポイントディレクトリのパスを調べます。
例を示します。

```
phys-schost# clzonecluster configure zone-cluster-name
```

4. ファイルシステムをゾーンクラスタの構成から削除します。

```
phys-schost# clzonecluster configure zone-cluster-name
```

```
clzc:zone-cluster-name> remove fs dir=filesystemdirectory
```

```
clzc:zone-cluster-name> commit
```

ファイルシステムのマウントポイントは、**dir=** で指定します。

5. ファイルシステムが削除されたことを確認します。

```
phys-schost# clzonecluster show -v zone-cluster-name
```

例 85 ゾーンクラスタ内の高可用性ローカルファイルシステムの削除

この例は、**sczone** というゾーンクラスタ内に構成された、マウントポイントディレクトリ (`/local/ufs-1`) のあるファイルシステムを削除する方法を示しています。リソースは **hasp-rs** で、そのタイプは **HASStoragePlus** です。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name:                fs
dir:                          /local/ufs-1
special:                       /dev/md/ds1/dsk/d0
raw:                           /dev/md/ds1/rdisk/d0
type:                          ufs
options:                       [logging]
...
phys-schost# clresource delete -F -Z sczone hasp-rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove fs dir=/local/ufs-1
clzc:sczone> commit
phys-schost# clzonecluster show -v sczone
```

例 86 ゾーンクラスタ内の高可用性 ZFS ファイルシステムの削除

この例は、リソース **hasp-rs**、タイプ **SUNW.HASStoragePlus** の **sczone** ゾーンクラスタ内で構成された、**HAzpool** という ZFS プール内の ZFS ファイルシステムを削除する方法を示します。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name:                dataset
name:                         HAzpool
...
phys-schost# clresource delete -F -Z sczone hasp-rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove dataset name=HAzpool
clzc:sczone> commit
phys-schost# clzonecluster show -v sczone
```

▼ ゾーンクラスタからストレージデバイスを削除する

ゾーンクラスタから、Solaris Volume Manager ディスクセットや DID デバイスなどのストレージデバイスを削除できます。この手順は、ゾーンクラスタからストレージデバイスを削除する場合に実行します。

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタからストレージデバイスを削除することもできます。「ゾーンクラスタ」をクリックし、そのページに移動するゾーンクラスタの名前をクリックして、「Solaris リソース」タブをクリックし、ゾーンクラスタコンポーネントを管理します。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

1. そのゾーンクラスタをホストしているグローバルクラスタのノードで、**root** 役割になります。
この手順のいくつかのステップはグローバルクラスタのノードから行います。ほかのステップは、ゾーンクラスタのノードから実行することが可能です。

2. **削除するデバイスに関連するリソースを削除します。**

削除するゾーンクラスタのデバイス用に構成されている Oracle Solaris Cluster リソースタイプ (SUNW.HAStoragePlus, SUNW.ScalDeviceGroup など) を特定し、削除します。

```
phys-schost# clresource delete -F -Z zone-cluster dev_zone_resources
```

3. **削除するデバイスに対して一致するエントリを調べます。**

```
phys-schost# clzonecluster show -v zone-cluster
...
Resource Name:      device
match:              <device_match>
...
```

4. **デバイスをゾーンクラスタの構成から削除します。**

```
phys-schost# clzonecluster configure zone-cluster
clzc:zone-cluster-name> remove device match=devices-match
clzc:zone-cluster-name> commit
clzc:zone-cluster-name> end
```

5. **ゾーンクラスタをリブートします。**

```
phys-schost# clzonecluster reboot zone-cluster
```

6. **デバイスの削除を確認します。**

```
phys-schost# clzonecluster show -v zone-cluster
```

例 87 ゾーンクラスタからの Solaris Volume Manager ディスクセットの削除

この例は、sczone というゾーンクラスタ内に構成された apachedg という Solaris Volume Manager ディスクセットを削除する方法を示しています。apachedg ディスクセットのセット番号は 3 です。このデバイスは、クラスタに構成された zc_rs のリソースにより使用されます。

```
phys-schost# clzonecluster show -v sczone
```

```

...
Resource Name:    device
match:           /dev/md/apachedg/*dsk/*
Resource Name:    device
match:           /dev/md/shared/3/*dsk/*
...
phys-schost# clresource delete -F -Z sczone zc_rs

phys-schost# ls -l /dev/md/apachedg
lrwxrwxrwx 1 root root 8 Jul 22 23:11 /dev/md/apachedg -> shared/3
phys-schost# clzonecluster configure sczone
clzc:sczone> remove device match=/dev/md/apachedg/*dsk/*
clzc:sczone> remove device match=/dev/md/shared/3/*dsk/*
clzc:sczone> commit
clzc:sczone> end
phys-schost# clzonecluster reboot sczone
phys-schost# clzonecluster show -v sczone

```

例 88 DID デバイスをゾーンクラスタから削除する

この例は、DID デバイス d10 および d11 を削除する方法を示しています。このデバイスは、sczone というゾーンクラスタに構成されています。このデバイスは、クラスタに構成された zc_rs のリソースにより使用されます。

```

phys-schost# clzonecluster show -v sczone
...
Resource Name:    device
match:           /dev/did/*dsk/d10*
Resource Name:    device
match:           /dev/did/*dsk/d11*
...
phys-schost# clresource delete -F -Z sczone zc_rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove device match=/dev/did/*dsk/d10*
clzc:sczone> remove device match=/dev/did/*dsk/d11*
clzc:sczone> commit
clzc:sczone> end
phys-schost#
phys-schost# clzonecluster show -v sczone

```

テスト目的のトラブルシューティング手順

このセクションでは、テスト用に使用できるトラブルシューティング手順について説明します。

グローバルクラスタ外でのアプリケーションの実行

▼ 非クラスタモードでブートしたノードから Solaris Volume Manager メタセットを取得する方法

この手順を使用して、テスト用にグローバルクラスタ外でアプリケーションを実行します。

1. Solaris Volume Manager メタセットで定足数デバイスが使用されているかどうかを確認し、定足数デバイスが SCSI2 または SCSI3 予約を使用するかどうかを確認します。

```
phys-schost# clquorum show
```

- a. 定足数デバイスが Solaris Volume Manager メタセットにある場合は、あとで非クラスタモードで取得されるメタセットには含まれない、新しい定足数デバイスを追加します。

```
phys-schost# clquorum add did
```

- b. 古い定足数デバイスを削除します。

```
phys-schost# clquorum remove did
```

- c. 定足数デバイスが SCSI2 予約を使用する場合は、古い定足数からの SCSI2 予約をスクラブして、SCSI2 予約が残らないことを確認します。

次のコマンドは、PGRE (Persistent Group Reservation Emulation) 鍵を検索します。ディスク上に鍵が存在しない場合は、`errno=22` メッセージが表示されます。

```
# /usr/cluster/lib/sc/pgre -c pgre_inkeys -d /dev/did/rdisk/dids2
```

鍵が見つかったら、PGRE 鍵をスクラブします。

```
# /usr/cluster/lib/sc/pgre -c pgre_scrub -d /dev/did/rdisk/dids2
```



注意 - アクティブな定足数デバイス鍵をディスクからスクラブすると、次の再構成時にクラスタでパニックが発生し、「操作可能な定足数を失いました」というメッセージが表示されます。

2. 非クラスタモードでブートするグローバルクラスタノードを退避します。

```
phys-schost# clresourcegroup evacuate -n target-node
```

3. HAStorage または HAStoragePlus リソースを含み、あとで非クラスタモードで取得するメタセットの影響を受けるデバイスまたはファイルシステムを含む、1 つまたは複数のリソースグループをオフラインにします。

```
phys-schost# clresourcegroup offline resource-group
```

4. オフラインにしたリソースグループ内のすべてのリソースを無効にします。

```
phys-schost# clresource disable resource
```

5. リソースグループを非管理状態に切り替えます。

```
phys-schost# clresourcegroup unmanage resource-group
```

6. 対応する 1 つまたは複数のデバイスグループをオフラインにします。

```
phys-schost# cldevicegroup offline device-group
```

7. 1 つまたは複数のデバイスグループを無効にします。

```
phys-schost# cldevicegroup disable device-group
```

8. パッシブノードを非クラスタモードでブートします。

```
phys-schost# shutdown -g0 -i0 -y  
ok> boot -x
```

9. 続ける前にパッシブノードでブートプロセスが完了していることを確認します。

```
phys-schost# svcs -x
```

10. メタセット内のディスクに SCSI3 予約があるかどうかを調べます。
メタセットのすべてのディスクで次のコマンドを実行します。

```
phys-schost# /usr/cluster/lib/sc/scsi -c inkeys -d /dev/did/rdisk/dids2
```

11. ディスクに SCSI3 予約が存在する場合は、それらをスクラブします。

```
phys-schost# /usr/cluster/lib/sc/scsi -c scrub -d /dev/did/rdisk/dids2
```

12. 退避したノードでメタセットを取得します。

```
phys-schost# metaset -s name -C take -f
```

13. メタセットで定義されたデバイスが含まれている 1 つまたは複数のファイルシステムをマウントします。

```
phys-schost# mount device mountpoint
```

14. アプリケーションを起動し、目的のテストを行います。テストが終了したら、アプリケーションを停止します。

15. ノードをリブートし、ブートプロセスが終了するまで待ちます。

```
phys-schost# reboot
```

16. 1 つまたは複数のデバイスグループをオンラインにします。

```
phys-schost# cldevicegroup online -e device-group
```

- 1つまたは複数のリソースグループを起動します。

```
phys-schost# clresourcegroup online -eM resource-group
```

破損したディスクセットの復元

ディスクセットが破損しているか、またはクラスタのノードがディスクセットの所有権を取得できない状態の場合に、この手順を使用します。状態をクリアしようとして、失敗した場合は、ディスクセットを修正する最後の試みとして次の手順に従います。

これらの手順は、Solaris Volume Manager のメタセットおよび複数所有者 Solaris Volume Manager のメタセットに適用されます。

▼ Solaris Volume Manager ソフトウェア構成を保存する方法

最初からディスクセットを復元すると、時間がかかり、エラーが発生しやすくなります。適切な代替方法は、`metastat` コマンドを使用して定期的にレプリカをバックアップするか、Oracle Explorer (SUNWexplo) を使用してバックアップを作成することです。その後、保存された構成を使用して、ディスクセットを再作成できます。(prvtoc および `metastat` コマンドを使用して) 現在の構成をファイルに保存し、ディスクセットとそのコンポーネントを再作成してください。268 ページの「Solaris Volume Manager ソフトウェア構成を再作成する方法」を参照してください。

1. ディスクセット内の各ディスクのパーティションテーブルを保存します。

```
# /usr/sbin/prvtoc /dev/global/rdisk/disk-name > /etc/lvm/disk-name.vtoc
```

2. Solaris Volume Manager ソフトウェア構成を保存します。

```
# /bin/cp /etc/lvm/md.tab /etc/lvm/md.tab_ORIGINAL
# /usr/sbin/metastat -p -s set-name >> /etc/lvm/md.tab
```

注記 - /etc/vfstab ファイルなどのほかの構成ファイルが、Solaris Volume Manager ソフトウェアを参照する場合があります。この手順では、同一の Solaris Volume Manager ソフトウェア構成を再構築することを想定しているため、マウント情報は同じです。セットを所有するノード上で Oracle Explorer (SUNWexplo) を実行すると、prvtoc および `metaset -p` の情報が取得されます。

▼ 破損したディスクセットを削除する方法

1 つのノードまたはすべてのノードからセットを削除すると、構成が削除されます。ノードからディスクセットを削除するには、ノードがそのディスクセットの所有権を持ってはなりません。

1. すべてのノードで削除コマンドを実行します。

```
# /usr/sbin/metaset -s set-name -P
```

このコマンドを実行すると、データベースのレプリカから、ディスクセット情報と Oracle Solaris Cluster リポジトリが削除されます。`-P` および `-c` オプションを使用すると、Solaris Volume Manager 環境を完全に再構築しなくても、ディスクセットを削除できます。

注記 - クラスタモードからノードがブートしたときに複数所有者のディスクセットが削除された場合、DCS 構成ファイルから情報を削除する必要がある場合があります。

```
# /usr/cluster/lib/sc/dcs_config -c remove -s set-name
```

詳細は、[dcs_config\(1M\)](#) のマニュアルページを参照してください。

2. データベースのレプリカからディスクセット情報のみを削除する場合は、次のコマンドを使用します。

```
# /usr/sbin/metaset -s set-name -C purge
```

通常は、`-c` オプションではなく、`-P` オプションを使用するようにしてください。`-c` オプションを使用すると、Oracle Solaris Cluster ソフトウェアはまだディスクセットを認識するため、ディスクセットの再作成で問題が発生することがあります。

- a. `metaset` コマンドで `-c` オプションを使用した場合は、問題が発生しないかどうかを確認するために、まずディスクセットを作成します。
- b. 問題が発生した場合は、`dcs` 構成ファイルから情報を削除してください。

```
# /usr/cluster/lib/sc/dcs_config -c remove -s setname
```

`purge` オプションが失敗した場合は、最新のカーネルとメタデバイスの更新がインストールされていることを確認し、[My Oracle Support](#) にアクセスします。

▼ Solaris Volume Manager ソフトウェア構成を再作成する方法

この手順に従うのは、Solaris Volume Manager ソフトウェア構成が完全に失われた場合のみです。この手順では、現在の Solaris Volume Manager 構成とそのコンポーネントを保存しており、破損したディスクセットを削除しているものと想定しています。

注記 - メディエータは 2 ノードクラスタでのみ使用します。

1. 新しいディスクセットを作成します。

```
# /usr/sbin/metaset -s set-name -a -h node1 node2
```

これが複数所有者ディスクセットの場合は、次のコマンドを使用して新しいディスクセットを作成します。

```
/usr/sbin/metaset -s set-name -aM -h node1 node2
```

2. セットが作成されたのと同じホストで、必要に応じてメディアータホストを追加します (2 ノードのみ)。

```
/usr/sbin/metaset -s set-name -a -m node1 node2
```

3. この同じホストからディスクセットに同じディスクを再度追加します。

```
/usr/sbin/metaset -s set-name -a /dev/did/rdisk/disk-name /dev/did/rdisk/disk-name
```

4. ディスクセットを削除して、再作成する場合は、ボリュームの目次 (Volume Table of Contents、VTOC) がディスクに残っているため、この手順は省略できます。

ただし、回復するセットを再作成する場合は、`/etc/lvm/disk-name.vtoc` ファイルに保存されている構成に従ってディスクをフォーマットするようにしてください。例を示します。

```
# /usr/sbin/fmthard -s /etc/lvm/d4.vtoc /dev/global/rdsk/d4s2
```

```
# /usr/sbin/fmthard -s /etc/lvm/d8.vtoc /dev/global/rdsk/d8s2
```

このコマンドはどのノードでも実行できます。

5. メタデバイスごとに、既存の `/etc/lvm/md.tab` ファイルの構文を確認します。

```
# /usr/sbin/metainit -s set-name -n -a metadvice
```

6. 保存されている構成から各メタデバイスを作成します。

```
# /usr/sbin/metainit -s set-name -a metadvice
```

7. メタデバイスにファイルシステムが存在する場合は、`fsck` コマンドを実行します。

```
# /usr/sbin/fsck -n /dev/md/set-name/rdsk/metadvice
```

`fsck` コマンドが、スーパーブロック数など少数のエラーのみを表示した場合、デバイスは正しく再構築されている可能性が高くなります。その後、`fsck` コマンドを `-n` オプションを指定せずに実行できます。多数のエラーが表示された場合は、メタデバイスが正しく再構築されているかどうかを確認します。そうなっている場合は、`fsck` エラーを確認して、ファイルシステムが回復可能かどうかを判断します。回復できない場合は、バックアップからデータを復元するようにしてください。

8. すべてのクラスタノード上のほかのすべてのメタセットを `/etc/lvm/md.tab` ファイルに連結してから、ローカルディスクセットに連結します。

```
# /usr/sbin/metastat -p >> /etc/lvm/md.tab
```


◆◆◆ 第 10 章

CPU 使用率の制御の構成

CPU の使用率を制御したい場合は、CPU 制御機能を構成します。CPU 制御機能の構成についての詳細は、[rg_properties\(5\)](#) のマニュアルページを参照してください。この章では、次のトピックについて説明します。

- [271 ページの「CPU 制御の概要」](#)
- [272 ページの「CPU 制御の構成」](#)

CPU 制御の概要

Oracle Solaris Cluster ソフトウェアを使用すると、CPU の使用率を制御できます。

CPU 制御機能は、Oracle Solaris OS で利用可能な機能に基づいて構築されています。ゾーン、プロジェクト、リソースプール、プロセッサセット、およびスケジューリングクラスについては、『[Oracle Solaris ゾーンを紹介](#)』を参照してください。

Oracle Solaris 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

注記 - Oracle Solaris Cluster Manager ブラウザインタフェースを使用して、ゾーンクラスタの構成を表示することもできます。Oracle Solaris Cluster Manager のログイン手順については、[290 ページの「Oracle Solaris Cluster Manager にアクセスする方法」](#)を参照してください。

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。

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、Oracle 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す。

CPU 制御の構成

このセクションでは、グローバルクラスタノードで CPU 使用率を制御する方法について説明します。

▼ グローバルクラスタノードで CPU 使用率を制御する方法

グローバルクラスタノードで実行されるリソースグループに CPU シェアを割り当てるには、この手順を実行します。

リソースグループに CPU シェアが割り当てられている場合、Oracle Solaris Cluster ソフトウェアは、グローバルクラスタノードでリソースグループの 1 つのリソースを起動する際に、次のタスクを実行します。

- ノードに割り当てられている CPU シェア (`zone.cpu-shares`) の数を、指定された CPU シェアの数だけ増やします (まだ行われていない場合)。
- ノードに `SCSLM_resource-group` 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_resource-group` プロジェクトのリソースを起動します。

CPU 制御機能の構成についての詳細は、[rg_properties\(5\)](#) のマニュアルページを参照してください。

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
```

次のリブート時に、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、リポート後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [prionctl\(1\)](#) のマニュアルページを参照してください。

注記 - 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に、グローバルクラスタノードに対するシェア数と、デフォルトのプロセッサセットで使用可能な CPU の最小数を構成します。

`globalzoneshares` および `defaultpsetmin` プロパティに値を割り当てない場合、これらのプロパティはデフォルト値をとります。

```
# clnode set [-p globalzoneshares=integer] [-p defaultpsetmin=integer] node
```

```
-p defaultpsetmin=integer
```

デフォルトプロセッサセットで使用可能な CPU の最小数を設定します。デフォルト値は 1 です。

```
-p globalzoneshares=integer
```

ノードに割り当てられるシェアの数を設定します。デフォルト値は 1 です。

```
node
```

プロパティを設定するノードを指定します。

これらのプロパティを設定する際には、ノードのプロパティを設定します。

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
```

```
-p RG_SLM_TYPE=automated
```

CPU 使用率を管理できるようにし、システムリソース管理用に Oracle Solaris OS を構成する手順の一部を自動化します。

`-p RG_SLM_CPU_SHARES=value`

リソースグループ固有のプロジェクトに割り当てられる CPU シェアの数 (`project.cpu-shares`) を指定し、ノードに割り当てられる CPU シェアの数 (`zone.cpu-shares`) を決定します。

リソースグループ

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。ノードでは、このプロパティは値 `default` をとります。

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

5. 構成の変更を有効にします。

```
# clresourcegroup online -eM resource-group
```

リソースグループ

リソースグループの名前を指定します。

注記 - `SCSLM_resource-group` 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。

◆◆◆ 第 11 章

ソフトウェアの更新

この章の以降のセクションでは、Oracle Solaris Cluster ソフトウェアを更新するための情報と手順を提供します。

- [275 ページの「Oracle Solaris Cluster ソフトウェアの更新の概要」](#)
- [276 ページの「Oracle Solaris Cluster ソフトウェアの更新」](#)
- [282 ページの「パッケージのアンインストール」](#)

Oracle Solaris Cluster ソフトウェアの更新の概要

クラスタが正常に動作するには、すべてのクラスタメンバーノードに同じ更新が適用されている必要があります。ノードを更新するときは、更新を行う前に、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、クラスタ全体を停止しておく必要がある場合があります。

Oracle Solaris Cluster ソフトウェアを更新する方法は 2 つあります。

- **アップグレード** - クラスタを最新のメジャーまたはマイナー Oracle Solaris Cluster リリースにアップグレードし、すべてのパッケージを更新することにより Oracle Solaris OS を更新します。メジャーリリースの例は、Oracle Solaris Cluster 4.0 から 5.0 へのアップグレードです。マイナーリリースの例は、Oracle Solaris Cluster 4.1 から 4.2 へのアップグレードです。

`scinstall` コーティリティーまたは `scinstall -u update` コマンドを実行して、新しいブート環境 (イメージのブートインスタンス) を作成し、使用されていないマウントポイントにブート環境をマウントし、ビットを更新し、新しいブート環境を有効化します。クローン環境の作成では、最初に追加の領域が消費されず、ただちに作成されます。この更新を行ったあとに、クラスタをリポートする必要があります。また、アップグレードでは、Oracle Solaris OS が最新の互換性のあるバージョンにアップグレードされます。詳細な手順については、『[Oracle Solaris Cluster 4.3 Upgrade Guide](#)』を参照してください。

ブランドタイプ `solaris` のフェイルオーバーゾーンがある場合は、『[Oracle Solaris Cluster 4.3 Upgrade Guide](#)』の「[How to Upgrade a solaris Branded Failover Zone](#)」の手順に従います。

ゾーンクラスタに `solaris10` ブランドゾーンがある場合は、『[Oracle Solaris Cluster 4.3 Upgrade Guide](#)』の「[Upgrading a solaris10 Branded Zone in a Zone Cluster](#)」のアップグレードの手順に従います。

ブランドではないゾーンクラスタを更新するには、[277 ページの「特定のパッケージの更新」](#)の手順に従って、ベースとなるグローバルクラスタを更新します。グローバルクラスタが更新されると、そのゾーンクラスタも自動的に更新されます。

注記 - Oracle Solaris Cluster Core SRU を適用しても、ソフトウェアを別の Oracle Solaris Cluster リリースにアップグレードした場合と同じ結果にはなりません。

- **更新** - 特定の Oracle Solaris Cluster パッケージを別の SRU レベルに更新します。pkg コマンドのいずれかを使用して、Support Repository Update (SRU) 内の Image Packaging System (IPS) パッケージを更新できます。

SRU は、通常、定期的にリリースされ、更新されたパッケージと不具合の修正が含まれています。このリポジトリには、すべての IPS パッケージと更新されたパッケージが含まれています。pkg update コマンドを実行すると、Oracle Solaris オペレーティングシステムと Oracle Solaris Cluster ソフトウェアの両方が互換バージョンに更新されます。この更新を行ったあとに、クラスタをリブートする必要がある場合があります。手順については、[278 ページの「特定のパッケージの更新」](#)を参照してください。

Oracle Solaris Cluster 製品向けの必要なソフトウェアアップデートを確認してダウンロードするには、My Oracle Support のユーザー登録が必要です。My Oracle Support アカウントを持っていない場合は、Oracle のサービス担当またはセールスエンジニアに連絡するか、<http://support.oracle.com> でオンライン登録してください。ファームウェアのアップデートについては、ハードウェアのドキュメントを参照してください。

注記 - アップデートを適用または削除する前に、ソフトウェアアップデートの README を参照してください。

Oracle Solaris パッケージ管理ユーティリティ `pkg` の使用方法については、『[Oracle Solaris 11.3 ソフトウェアの追加と更新](#)』の第 3 章、「ソフトウェアパッケージのインストールおよび更新」で説明しています。

Oracle Solaris Cluster ソフトウェアの更新

Oracle Solaris Cluster ソフトウェアの Oracle Solaris Cluster リリースまたはパッケージをアップグレードまたは更新する方法を判別するには、次の表を参照してください。

表 20 Oracle Solaris Cluster ソフトウェアの更新

タスク	手順
新しいメジャーリリースまたはマイナーリリースへのクラスタ全体のアップグレード	『 Oracle Solaris Cluster 4.3 Upgrade Guide 』の「 How to Upgrade the Software (Standard Upgrade) 」
更新の成功のヒントの確認	277 ページの「更新のヒント」

タスク	手順
特定のパッケージの更新	278 ページの「特定のパッケージの更新」
定足数サーバーまたは AI インストールサーバーの更新	281 ページの「定足数サーバーまたは AI インストールサーバーの更新」
solaris ブランドゾーンクラスタの更新	279 ページの「solaris ブランドゾーンクラスタを更新する方法」
solaris10 ブランドゾーンクラスタのパッチ	279 ページの「solaris10 ブランドゾーンクラスタ (clzonecluster) にパッチを適用する方法」 280 ページの「ゾーンクラスタの solaris10 ブランドゾーンにパッチを適用する方法 (patchadd)」 281 ページの「solaris10 ブランドゾーンクラスタのゾーンに代替ブート環境のパッチを適用する方法」
Oracle Solaris Cluster パッケージの削除	282 ページの「パッケージのアンインストール」 282 ページの「定足数サーバーまたは AI インストールサーバーパッケージのアンインストール」

更新のヒント

Oracle Solaris Cluster 更新をより効率的に管理するために、次のヒントを使用してください。

- 更新を行う前に、SRU の README ファイルを参照してください。
- ストレージデバイスの更新要件を確認します。
- クラスタを本稼働環境で実行する前に、すべての更新を適用します。
- 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。ファームウェアの更新に関する情報については、ハードウェアのドキュメントを参照してください。
- クラスタメンバーとして機能するノードには、すべて同じ更新を適用する必要があります。
- クラスタサブシステムの更新を最新の状態に保ちます。これらの更新には、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。
- メジャー更新を行ったあとは、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、更新を取り消す準備をしておきます。

特定のパッケージの更新

IPS パッケージは、Oracle Solaris 11 オペレーティングシステムで導入されました。各 IPS パッケージは、Fault Managed Resource Indicator (FMRI) によって記述されており、pkg コマンドを使用して、SRU 更新を実行します。また、scinstall -u コマンドを使用して SRU 更新を実行することもできます。

特定のパッケージを更新して、更新された Oracle Solaris Cluster データサービスエージェントを使用する場合があります。

▼ 特定のパッケージの更新

更新する各グローバルクラスタードでこの手順を実行します。クラスタード上のブランドのないゾーンクラスタも自動的にこの更新を受け取ります。

1. RBAC の承認 `solaris.cluster.admin` を提供する役割になります。

2. パッケージを更新します。

たとえば、特定のパブリッシャーからのパッケージを更新するには、`pkg-fmri` にパブリッシャー名を指定します。

```
# pkg update pkg-fmri
```



注意 - `pkg update` コマンドに `pkg-fmri` を指定しないで使用すると、更新を入手可能なすべてのインストール済みパッケージが更新されます。

インストール済みのパッケージの新しいバージョンが利用可能で、残りのイメージと互換性がある場合、パッケージはそのバージョンに更新されます。`reboot-needed` フラグが `true` に設定されているバイナリがパッケージに含まれている場合、`pkg update pkg-fmri` を実行すると、新しいブート環境が自動的に作成され、更新後に新しいブート環境でブートされます。更新しているパッケージにリブートを強制するバイナリが含まれていない場合、`pkg update` コマンドはライブイメージを更新するため、リブートは必要ありません。

3. データサービスエージェント (`ha-cluster/data-service/*` または `ha-cluster/ha-service/gds` の汎用データサービスエージェント) を更新する場合は、次のコマンドを実行します。

```
# pkg change-facet facet.version-lock.pkg-name=false
```

```
# pkg update pkg-name
```

例を示します。

```
# pkg change-facet facet.version-lock.ha-cluster/data-service/weblogic=false
```

```
# pkg update ha-cluster/data-service/weblogic
```

注記 - エージェントをフリーズして、更新されないようにするには、次のコマンドを実行します。

```
# pkg change-facet facet.version-lock.pkg-name=false
```

```
# pkg freeze pkg-name
```

特定のエージェントのフリーズに関する詳細は、『[Oracle Solaris 11.3 ソフトウェアの追加と更新](#)』の「オプションのコンポーネントのインストールの制御」を参照してください。

4. パッケージが更新されたことを確認します。

```
# pkg verify -v pkg-fmri
```

ゾーンクラスタの更新またはパッチ

solaris ブランドゾーンクラスタを更新するには、`scinstall-u update` コマンドを使用して SRU を適用します。

solaris10 ブランドゾーンクラスタにパッチを適用するには、`clzonecluster install-cluster -p` コマンドまたは `patchadd` コマンドを使用して、または代替ブート環境にパッチを適用することによって、パッチを適用します。

このセクションでは、次の手順について説明します。

- [279 ページの「solaris ブランドゾーンクラスタを更新する方法」](#)
- [279 ページの「solaris10 ブランドゾーンクラスタ \(clzonecluster\) にパッチを適用する方法」](#)
- [280 ページの「ゾーンクラスタの solaris10 ブランドゾーンにパッチを適用する方法 \(patchadd\)」](#)
- [281 ページの「solaris10 ブランドゾーンクラスタのゾーンに代替ブート環境のパッチを適用する方法」](#)

▼ solaris ブランドゾーンクラスタを更新する方法

この手順を実行し、`scinstall-u update` コマンドを使用して SRU を適用することによって、solaris ブランドゾーンクラスタを更新します。

1. グローバルクラスタのノード上で、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
2. グローバルクラスタのノードから、ノード全体を更新します。

```
phys-schost# scinstall -u update [-b be-name]
```

この段階を各クラスタノード上で繰り返します。

3. クラスタをリブートします。

```
phys-schost# clzonecluster reboot
```

▼ solaris10 ブランドゾーンクラスタ (clzonecluster) にパッチを適用する方法

この手順は、グローバルクラスタのいずれかのノードで実行します。

1. グローバルクラスタのノード上で、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
2. ゾーンクラスタをオフライン/実行モードにブートします。

- ゾーンクラスタがブートされていない場合は、次のコマンドを実行します。

```
phys-schost# clzonecluster boot -o zone-cluster
```

- ゾーンクラスタがブートされている場合は、次のコマンドを実行します。

```
phys-schost# clzonecluster reboot -o zone-cluster
```

3. グローバルクラスタの 1 つのノードから、solaris10 ブランドゾーンクラスタ全体にパッチを適用します。

```
phys-schost# clzonecluster install-cluster -p patch-spec [options] zone-cluster
```

`install-cluster` サブコマンドの詳細は、[clzonecluster\(1CL\)](#) のマニュアルページを参照してください。

注記 - `scinstall -u update` コマンドが、`pkg` コマンドが古いというメッセージを表示して失敗した場合は、メッセージの手順に従います。

4. ゾーンクラスタをリブートします。

```
phys-schost# clzonecluster reboot zone-cluster
```

▼ ゾーンクラスタの solaris10 ブランドゾーンにパッチを適用する方法 (patchadd)

構成された各ゾーンクラスタノードで次の手順を実行します。

1. ゾーンクラスタをオフライン/実行状態にします。

```
phys-schost# clzonecluster reboot -o zone-cluster
```

2. ゾーンクラスタノードにログインします。

```
phys-schost# zlogin zone-cluster
```

3. コマンド行から `patchadd` と入力します。

```
zchost# patchadd
```

4. 新しいリリースに対応するパッチをインストールします。

▼ solaris10 ブランドゾーンクラスタのゾーンに代替ブート環境のパッチを適用する方法

この手順を使用して、ゾーンクラスタノードのゾーンに代替ブート環境 (BE) のパッチを適用します。この方法は、必要に応じてパッチされた BE を取り消すオプションを提供します。

1. パッチを適用するゾーンクラスタノードのゾーンの管理者になります。
2. 新しいブート環境を作成し、パッチを適用して、アクティブ化し、`shutdown` コマンドを使用してゾーンをリブートします。
『Oracle Solaris 10 ゾーンの作成と使用』の「solaris10 ブランドゾーンのブートについて」セクションの「solaris10 ブランドゾーンで複数のブート環境を作成およびアクティブ化する方法」に記載されているタスクの手順に従います。
3. パッチを適用する必要があるほかのすべてのゾーンクラスタノードのゾーンについて、前の手順を繰り返します。

定足数サーバーまたは AI インストールサーバーの更新

定足数サーバーまたは Oracle Solaris 11 Automated Installer (AI) インストールサーバーのパッケージを更新するには、次の手順を使用します。定足数サーバーの詳細は、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の「Oracle Solaris Cluster Quorum Server ソフトウェアをインストールおよび構成する方法」を参照してください。AI の使用の詳細は、『Oracle Solaris Cluster 4.3 ソフトウェアのインストール』の「Oracle Solaris および Oracle Solaris Cluster ソフトウェアをインストールおよび構成する方法 (IPS リポジトリ)」を参照してください。

▼ 定足数サーバーまたは AI インストールサーバーの更新

1. RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
2. 定足数サーバーまたは AI インストールサーバーパッケージを更新します。

```
# pkg update ha-cluster/*
```

インストール済み `ha-cluster` パッケージの新しいバージョンが入手可能で、残りのイメージと互換性がある場合、パッケージはそのバージョンに更新されます。



注意 - `pkg update` コマンドを実行すると、システムにインストールされているすべての `ha-cluster` パッケージが更新されます。

パッケージのアンインストール

単一のパッケージまたは複数のパッケージを削除できます。

▼ パッケージのアンインストール

1. RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
2. 1 つ以上の既存のパッケージをアンインストールします。

```
# pkg uninstall pkg-fmri [pkg-fmri ...]
```

アンインストールしている `pkg-fmri` に依存する別のパッケージがインストールされている場合、`pkg uninstall` コマンドは失敗します。`pkg-fmri` をアンインストールするには、`pkg-fmri` に従属するすべてのものを `pkg uninstall` コマンドに指定する必要があります。パッケージのアンインストールの詳細は、『[Oracle Solaris 11.3 ソフトウェアの追加と更新](#)』および [pkg\(1\)](#) のマニュアルページを参照してください。

▼ 定足数サーバーまたは AI インストールサーバーパッケージのアンインストール

1. RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
2. 定足数サーバーまたは AI インストールサーバーのパッケージをアンインストールします。

```
# pkg uninstall ha-cluster/*
```



注意 - このコマンドは、システムにインストールされているすべての `ha-cluster` パッケージをアンインストールします。

◆◆◆ 第 12 章

クラスタのバックアップと復元

この章は次のセクションから構成されています。

- [283 ページの「クラスタのバックアップ」](#)
- [286 ページの「クラスタファイルの復元」](#)
- [212 ページの「クラスタノードの復元」](#)

クラスタのバックアップ

クラスタをバックアップする前に、バックアップするファイルシステムの名前を確認し、フルバックアップに必要なテープの数を算出し、ZFS ルートファイルシステムをバックアップします。

表 21 タスクリスト：クラスタファイルのバックアップ

タスク	手順
ミラーまたはブ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	283 ページの「ミラーのオンラインバックアップを実行する方法 (Solaris Volume Manager)」
クラスタ構成のバックアップ	285 ページの「クラスタ構成をバックアップする方法」
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ドキュメントを参照

▼ ミラーのオンラインバックアップを実行する方法 (Solaris Volume Manager)

Solaris Volume Manager のミラー化ボリュームは、アンマウントしたりミラー全体をオフラインにしたりしなくても、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。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. バックアップするクラスタノードで、同等の役割になります。
2. **metaset** コマンドを使用して、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。

詳細は、[metaset\(1M\)](#) のマニュアルページを参照してください。

3. **lockfs** コマンドを **-w** オプションとともに使用して、ファイルシステムを書き込みからロックします。

```
# lockfs -w mountpoint
```

詳細は、[lockfs\(1M\)](#) のマニュアルページを参照してください。

4. **metastat** コマンドを使用して、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ステータスを表示します。

詳細は、[metastat\(1M\)](#) のマニュアルページを参照してください。

5. **metadetach** コマンドを使用して、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
```

詳細は、[metadetach\(1M\)](#) のマニュアルページを参照してください。

注記 - 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、ミラーに最初書き込んだ直後から同期がとれなくなります。この不一致は、オフラインのサブミラーをオンラインに戻したときに修正されます。fsck を実行する必要はありません。

6. `-u` オプションを指定して `lockfs` コマンドを使用し、ファイルシステムのロックを解除して書き込みを続行できるようにします。

```
# lockfs -u mountpoint
```

7. ファイルシステムを検査します。

```
# fsck /dev/md/diskset/rdisk/submirror
```

8. オフラインのサブミラーをテープなどのメディアにバックアップします。

注記 - ブロックデバイス (`/disk`) 名ではなく、サブミラーの raw デバイス (`/rdisk`) 名を使用してください。

9. `metattach` コマンドを使用して、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。詳細は、[metattach\(1M\)](#) のマニュアルページを参照してください。

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
```

詳細は、『[Oracle Solaris 11.3 での ZFS ファイルシステムの管理](#)』を参照してください。

▼ クラスタ構成をバックアップする方法

クラスタ構成を確実にアーカイブし、クラスタ構成を簡単に回復できるようにするため、定期的にクラスタ構成をバックアップしてください。Oracle Solaris Cluster 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、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。

```
# cluster export -o configfile
```

`configfile`

クラスタコマンドのクラスタ構成情報のエクスポート先である XML 構成ファイルの名前。XML 構成ファイルについては、[clconfiguration\(5CL\)](#) のマニュアルページ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。

```
# pfedit configfile
```

クラスタファイルの復元

ZFS ルートファイルシステムを新しいディスクに復元できます。

統合アーカイブからクラスタまたはノードを復元したり、特定のファイルまたはファイルシステムを復元したりできます。

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 22 タスクリスト：クラスタファイルの復元

タスク	手順
Solaris Volume Manager の場合、ZFS ルート (/) ファイルシステムを復元	286 ページの「ZFS ルート (/) ファイルシステム (Solaris Volume Manager) を復元する方法」

▼ ZFS ルート (/) ファイルシステム (Solaris Volume Manager) を復元する方法

障害の発生したルートディスクを交換したあとなどに、この手順を使用して ZFS ルート (/) ファイルシステムを新しいディスクに復元します。復元中のノードはブートしないおささないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。UFS はサポートされます (ルートファイルシステムとして使用する場合を除く)。UFS は共有ディスクの Solaris Volume Manager メタセットのメタデバイスで使用できます。

注記 - 新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

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

この手順では、長形式の Oracle Solaris Cluster コマンドを使用します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

1. **復元するノードが接続先でもあるディスクセットへのアクセス権を持つクラスタノードで、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。回復し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. **ZFS ルートファイルシステム (/) を復元します。**

詳細は、『Oracle Solaris 11.3 での ZFS ファイルシステムの管理』の「ZFS ルートプール内のディスクの交換」を参照してください。

ZFS ルートプールまたはルートプールスナップショットを回復する場合は、『Oracle Solaris 11.3 での ZFS ファイルシステムの管理』の「ZFS ルートプール内のディスクの交換」に記載されている手順に従います。

注記 - /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。

バックアップディレクトリに /globaldevices バックアップファイルが存在する場合は、ZFS ルートの復元とともに復元されます。globaldevices SMF サービスは、このファイルを自動的に作成しません。

4. **ノードをマルチユーザーモードでリブートします。**

```
# reboot
```

5. **デバイス ID を交換します。**

```
# cldevice repair root-disk
```

6. **metadb コマンドを使用し、状態データベースのレプリカを再作成します。**

```
# metadb -c copies -af raw-disk-device
```

-c copies 作成するレプリカの数指定します。

-f raw-disk-device レプリカの作成先の raw ディスクデバイス名を指定します。

-a レプリカを追加します。

詳細は、[metadb\(1M\)](#) のマニュアルページを参照してください。

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リポートします。これでクラスタを使用できるようになります。

例 89 ZFS ルート (/) ファイルシステム (Solaris Volume Manager) の復元

次に、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。その他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レプリカが `/dev/rdisk/c0t0d0s4` に再作成されます。データの復元の詳細は、『[Oracle Solaris 11.3](#)での ZFS ファイルシステムの管理』の「ZFS ストレージプール内のデータの問題を解決する」を参照してください。

メタセットからノードを削除します

```
phys-schost-2# metaset -s schost-1 -f -d -h phys-schost-1
```

障害の発生したディスクを交換し、ノードをブートします

Oracle Solaris ドキュメントの手順を使用して、ルート (/) および /usr ファイルシステムを復元します

ノードをリポートします

```
# reboot
```

ディスク ID を交換します

```
# cldevice repair /dev/dsk/c0t0d0
```

状態データベースのレプリカを再作成します

```
# metadb -c 3 -af /dev/rdisk/c0t0d0s4
```

メタセットに再度ノードを追加します

```
phys-schost-2# metaset -s schost-1 -a -h phys-schost-1
```

◆◆◆ 第 13 章

13

Oracle Solaris Cluster Manager ブラウザインタフェースの使用

この章では、クラスタのさまざまな側面を管理するために使用できる Oracle Solaris Cluster Manager ブラウザインタフェースについて説明します。この章には、Oracle Solaris Cluster Manager にアクセスして使用するための手順も含まれています。

注記 - Oracle Solaris Cluster Manager は、Oracle Solaris Cluster 製品に付属のプライベートバージョンの Oracle GlassFish Server ソフトウェアを使用します。パブリックバージョンの Oracle GlassFish Server ソフトウェアをインストールしたり、パッチセットに更新したりしないでください。そうすると、Oracle Solaris Cluster ソフトウェアの更新時または Oracle Solaris Cluster SRU のインストール時にパッケージの問題が発生する可能性があります。Oracle Solaris Cluster Manager で必要なプライベートバージョンの Oracle GlassFish Server に対するバグ修正は、Oracle Solaris Cluster SRU で提供されます。

この章の内容は、次のとおりです。

- [289 ページの「Oracle Solaris Cluster Manager の概要」](#)
- [290 ページの「Oracle Solaris Cluster Manager ソフトウェアへのアクセス」](#)
- [293 ページの「Oracle Solaris Cluster Manager のアクセシビリティのサポートの構成」](#)
- [294 ページの「トポロジを使用したクラスタのモニタリング」](#)

Oracle Solaris Cluster Manager の概要

Oracle Solaris Cluster Manager ブラウザインタフェースを使用すると、クラスタ情報のグラフィカルな表示、クラスタコンポーネントのステータスのチェック、および構成変更のモニタリングを行うことができます。また、Oracle Solaris Cluster Manager では、次の Oracle Solaris Cluster コンポーネントの管理を含む多くの管理タスクを実行することもできます。

- データサービス
- ゾーンクラスタ

- ノード
- プライベートアダプタ、ケーブル、およびスイッチ
- 定足数デバイス
- デバイスグループ
- ディスク
- NAS デバイス
- ノードの負荷制限
- リソースグループとリソース
- Geographic Edition パートナーシップ

Oracle Solaris Cluster Manager は現在、Oracle Solaris Cluster の一部の管理タスクを実行できません。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。

Oracle Solaris Cluster Manager ソフトウェアへのアクセス

Oracle Solaris Cluster Manager ブラウザインタフェースには、Oracle Solaris Cluster ソフトウェアの多くのタスクを簡単に管理する方法が用意されています。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

クラスタをブートすると、共通エージェントコンテナが自動的に起動します。共通エージェントコンテナが実行していることを確認する必要がある場合は、[291 ページの「Oracle Solaris Cluster Manager のトラブルシューティング」](#)を参照してください。

ヒント - Oracle Solaris Cluster Manager を終了するためにブラウザで「戻る」をクリックしないでください。

▼ Oracle Solaris Cluster Manager にアクセスする方法

この手順では、クラスタの Oracle Solaris Cluster Manager にアクセスする方法を説明します。

1. クラスタメンバーで、root 役割になります。
2. 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。
3. 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。
4. 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。

5. ブラウザから、クラスタの 1 つのノード上にある Oracle Solaris Cluster Manager のポートに接続します。

デフォルトのポート番号は 8998 です。

```
https://node:8998/scm
```

6. 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。
Oracle Solaris Cluster Manager ログインページが表示されます。
7. 管理するクラスタ内のノードの名前を入力するか、現在のクラスタを管理する場合はデフォルトの localhost をそのまま使用します。
8. ノードのユーザー名とパスワードを入力します。
9. 「サインイン」をクリックします。
Oracle Solaris Cluster Manager アプリケーション起動ページが表示されます。

注記 - 複数のクラスタが構成されている場合は、ドロップダウンリストから「その他」を選択し、別のクラスタにログインしてそのクラスタに関する情報を表示できます。クラスタが 1 つ以上のパートナーシップのメンバーである場合は、「パートナーシップ」フォルダにアクセスすると、すべてのパートナー名がドロップダウンリストに自動的に追加されます。認証が完了すると、「クラスタの切り替え」を選択できます。

Oracle Solaris Cluster Manager に接続できない場合は、[291 ページの「Oracle Solaris Cluster Manager のトラブルシューティング」](#)を参照してください。Oracle Solaris のインストール中に制限されたネットワークプロファイルを選択すると、Oracle Solaris Cluster Manager の外部アクセスが制限されます。このネットワークは Oracle Solaris Cluster Manager ブラウザインタフェースを使用する必要があります。

Oracle Solaris Cluster Manager のトラブルシューティング

- 2 つのマネージャーサービスが実行されていることを確認します。

```
# svcs system/cluster/manager\*
```

```
STATE      STIME      FMRI
online    Oct_30    svc:/system/cluster/manager-glassfish3:default
online    Oct_30    svc:/system/cluster/manager:default
```

svcadm コマンドを使用して、system/cluster/manager-glassfish3 を無効または有効にします。このアクションにより、アプリケーションサーバーがそれぞれ停止して再起動しま

す。system/cluster/manager はオンラインのままにしてください。無効または有効にする必要はありません。

- Oracle Solaris Cluster Manager に接続できない場合は、usr/sbin/cacaoadm status を入力して、共通エージェントコンテナが実行されているかどうかを調べます。共通エージェントコンテナが実行されていない場合、ログインページは表示されますが、認証はできません。

共通エージェントコンテナを手動で起動するには、/usr/sbin/cacaoadm start を入力します。

▼ 共通エージェントコンテナのセキュリティ鍵を構成する方法

Oracle Solaris Cluster Manager は、強力な暗号化手法を使用して、Oracle Solaris Cluster Manager Web サーバーと各クラスタノードの間のセキュアな通信を確保します。

Oracle Solaris Cluster Manager でデータサービス構成ウィザードを使用したり、またはその他の Oracle Solaris Cluster Manager タスクを実行したりしているときに Cacao 接続エラーが発生する場合があります。この手順では、共通エージェントコンテナのセキュリティファイルがすべてのクラスタノードにコピーされます。これにより、共通エージェントコンテナのセキュリティファイルがすべてのクラスタノード上で同じであり、コピーされたファイルが正しいファイルアクセス権を保持することが保証されます。この手順を実行すると、セキュリティ鍵が同期されます。

1. 各ノードで、セキュリティファイルエージェントを停止します。

```
phys-schost# /usr/sbin/cacaoadm stop
```

2. 1 つのノードで、/etc/cacao/instances/default/ ディレクトリに変更します。

```
phys-schost-1# cd /etc/cacao/instances/default/
```

3. /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# tar cf /tmp/SECURITY.tar security
```

4. /tmp/Security.tar ファイルを各クラスタノードにコピーします。

5. /tmp/SECURITY.tar ファイルをコピーした各ノード上で、セキュリティファイルを抽出します。/etc/cacao/instances/default/ ディレクトリにすでにセキュリティ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
```

6. セキュリティリスクを回避するために、tar ファイルの各コピーを削除します。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
```

7. 各ノード上で、セキュリティーファイルエージェントを起動します。

```
phys-schost# /usr/sbin/cacaoadm start
```

▼ ネットワークバインドアドレスを確認する方法

Oracle Solaris Cluster Manager を実行しているノード以外のノードに関する情報を表示しようとしたときにシステムエラーメッセージが表示された場合は、共通エージェントコンテナのネットワークバインドアドレスパラメータが `0.0.0.0` の正しい値に設定されているかどうかを確認します。

クラスタの各ノード上で次の手順を実行します。

1. ネットワークバインドアドレスを確認します。

```
phys-schost# cacaoadm list-params | grep network
network-bind-address=0.0.0.0
```

ネットワークバインドアドレスが `0.0.0.0` 以外の値に設定されている場合は、それを目的のアドレスに変更する必要があります。

2. 変更の前後で cacao を停止および起動します。

```
phys-schost# cacaoadm stop
phys-schost# cacaoadm set-param network-bind-address=0.0.0.0
phys-schost# cacaoadm start
```

Oracle Solaris Cluster Manager のアクセシビリティのサポートの構成

Oracle Solaris Cluster Manager は次のアクセシビリティ設定を提供します。

- スクリーンリーダー
- 高コントラスト
- 大きいフォント

これらの 1 つまたは複数の制御を設定するには、Oracle Solaris Cluster Manager の上部メニューバーの「アクセシビリティ」をクリックします。次に、有効または無効にするアクセシビリティ制御のチェックボックスをクリックします。

アクセシビリティ制御設定の変更は、ログインセッションをまたいで保持されません。設定は、同じログインセッション中に別のクラスタノードに対して認証する場合は保持されます。

トポロジを使用したクラスタのモニタリング

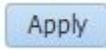
「トポロジ」ビューは、クラスタをモニターして問題を識別するのに役立ちます。オブジェクト間の関係をすばやく表示し、どのリソースグループおよびリソースが各ノードに属するかを確認できます。

▼ トポロジを使用してクラスタのモニタリングと更新を行う方法

「トポロジ」ページに移動するには、Oracle Solaris Cluster Manager にログインし、「リソースグループ」をクリックして、「トポロジ」タブをクリックします。依存関係と共用関係が線によって表されます。

オンラインヘルプには、ビューの各要素、ビューをフィルタ処理するためのオブジェクトの選択方法、およびオブジェクトを右クリックして関連するアクションのコンテキストメニューを表示する方法が詳しく説明されています。オンラインヘルプを折りたたんだり、元に戻したりするには、オンラインヘルプの横にある矢印をクリックします。フィルタを折りたたんだり、元に戻したりすることもできます。

次の表は、「リソーストポロジ」ページ上のコントロールの一覧です。

コントロール	機能	説明
	ズーム	ページの一部分を拡大または縮小します。
	概要	図の上でビューポートをドラッグして、ビューをパンします。
	分離	リソースグループまたはリソースをシングルクリックして、ほかのすべてのオブジェクトを表示から取り除きます。
	ドリル	リソースグループをシングルクリックして、そのグループのリソースにドリルします。
	リセット	分離やドリルのあとで、全体表示に戻ります。
	フィルタ	タイプ、インスタンス、またはステータスでオブジェクトを選択し、表示内容を絞り込みます。

次の手順は、クラスタノードにクリティカルエラーがないかどうかをモニターする方法を示しています。

1. 「トポロジ」タブで、「潜在的マスター」領域を見つけます。
2. ズームインして、クラスタ内の各ノードのステータスを確認します。
3. 「クリティカル」ステータスを示す赤いアイコンのノードを見つけ、ノードを右クリックして「詳細の表示」を選択します。
4. ノードの「ステータス」ページで、「システムログ」をクリックしてログメッセージを表示し、フィルタ処理します。

◆◆◆ 付録 A

デプロイメントの例: Availability Suite ソフトウェアによるクラスタ間のホストベースのデータレプリケーションの構成

この付録では、Oracle Solaris Cluster Geographic Edition を使用せずにホストベースのレプリケーションを使用する方法について説明します。クラスタ間のホストベースのレプリケーションの構成と操作を簡素化するには、Oracle Solaris Cluster Geographic Edition をホストベースのレプリケーションに使用します。[98 ページの「データレプリケーションについての理解」](#)を参照してください。

この付録の例は、Oracle Solaris の Availability Suite 機能 ソフトウェアを使用してクラスタ間のホストベースのデータレプリケーションを構成する方法を示しています。この例では、NFS アプリケーション用の完全なクラスタ構成を示し、個別のタスクの実行方法に関する詳細情報を提供します。すべてのタスクをグローバルクラスタで実行してください。例には、ほかのアプリケーションやクラスタ構成で必要な手順がすべて含まれているわけではありません。

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 Oracle Solaris 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。一連のデータレプリケーション手順には、次の Oracle Solaris Cluster RBAC の承認が必要です。

- `solaris.cluster.modify`
- `solaris.cluster.admin`
- `solaris.cluster.read`

RBAC の役割の使用に関する詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』を参照してください。各 Oracle Solaris Cluster サブコマンドで必要となる RBAC の承認については、Oracle Solaris 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 Availability Suite ソフトウェアの理解

このセクションでは、耐障害性について紹介し、Availability Suite ソフトウェアが使用するデータレプリケーション方式について説明します。

耐障害性は、プライマリクラスタに障害が発生した場合に、アプリケーションを代替クラスタに復元する機能です。災害耐性のベースは、データレプリケーションとテイクオーバーです。テイクオーバーは、1 つ以上のリソースグループおよびデバイスグループをオンラインにすることにより、アプリケーションサービスをセカンダリクラスタに再配置します。

プライマリクラスタおよびセカンダリクラスタ間でデータが同期してレプリケートされている場合、プライマリサイトで障害が発生してもコミットされたデータは失われません。ただし、データが非同期でレプリケートされていた場合、プライマリサイトで障害が発生する前にセカンダリクラスタにレプリケートされていなかったデータがある可能性があり、それらのデータは失われます。

Availability Suite ソフトウェアが使用するデータレプリケーション方式

このセクションでは、Availability Suite が使用するリモートミラーレプリケーション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` と `iiadm` コマンドを使用してデータをレプリケートします。詳細は、[sndradm\(1M\)](#) および [iiadm\(1M\)](#) のマニュアルページを参照してください。

リモートミラーレプリケーション

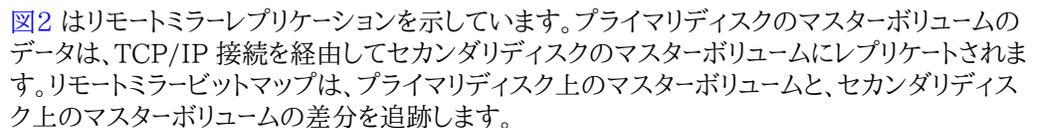
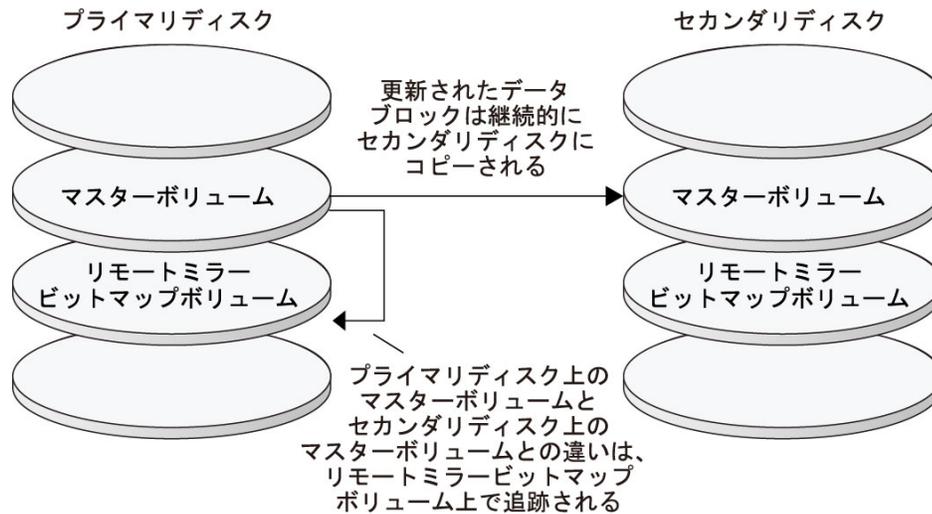
 **図2** はリモートミラーレプリケーションを示しています。プライマリディスクのマスターボリュームのデータは、TCP/IP 接続を経由してセカンダリディスクのマスターボリュームにレプリケートされます。リモートミラービットマップは、プライマリディスク上のマスターボリュームと、セカンダリディスク上のマスターボリュームの差分を追跡します。

図 2 リモートミラーレプリケーション



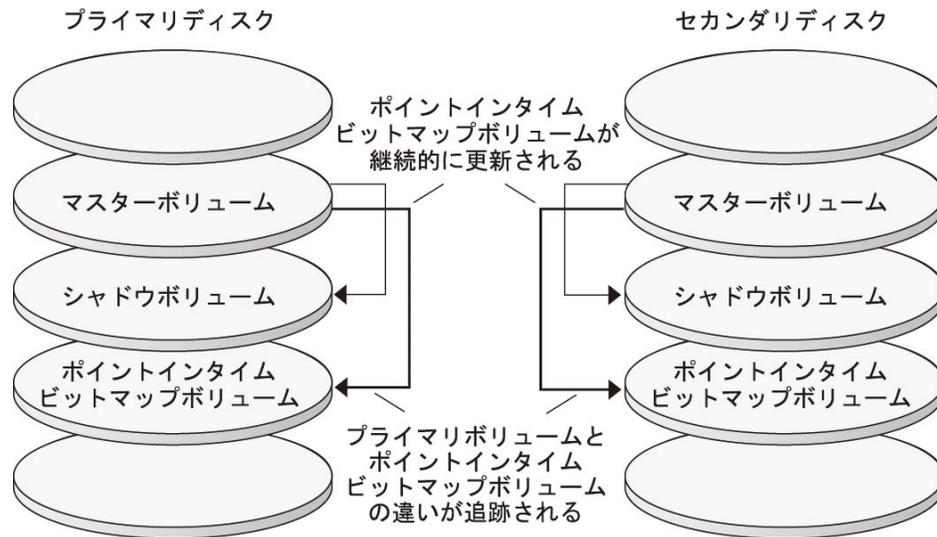
リモートミラーレプリケーションは、リアルタイムに同期で実行することも非同期で実行することもできます。各クラスターの各ボリュームセットはそれぞれ、同期レプリケーションまたは非同期レプリケーションに構成できます。

- 同期データレプリケーションでは、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。
- 非同期データレプリケーションでは、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。非同期データレプリケーションは、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3 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。

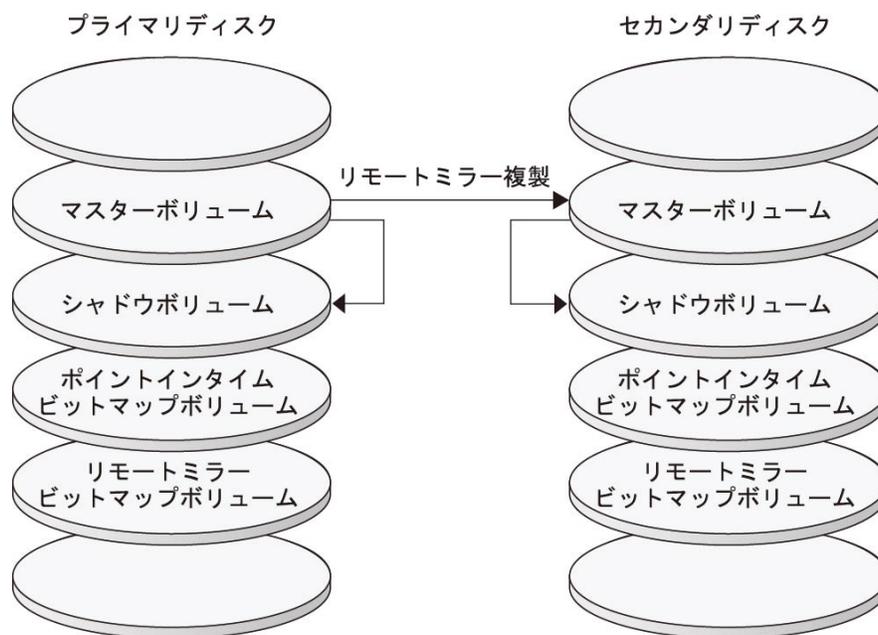
図 3 ポイントインタイムスナップショット



構成例でのレプリケーション

図4に、この構成例でリモートミラーレプリケーション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 構成例でのレプリケーション



クラスタ間でホストベースのデータレプリケーションを構成するためのガイドライン

このセクションでは、クラスタ間のデータレプリケーションの構成ガイドラインを提供します。また、レプリケーション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、クラスタのデータレプリケーションを構成する際に使用してください。

このセクションでは、次の項目について説明します。

- [302 ページの「レプリケーションリソースグループの構成」](#)
- [302 ページの「アプリケーションリソースグループの構成」](#)
 - [303 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」](#)
 - [304 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」](#)
- [306 ページの「テイクオーバーの管理のガイドライン」](#)

レプリケーションリソースグループの構成

レプリケーションリソースグループは、Availability Suite ソフトウェアの制御下にあるデバイスグループを論理ホスト名リソースと結び付けます。論理ホスト名は、データレプリケーションストリームの各終端に存在し、デバイスへのプライマリ入出力パスとして動作しているのと同じクラスタノードにある必要があります。レプリケーション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、各クラスタ (プライマリおよびセカンダリ) 内のいずれかのノードでホストされ、Availability Suite ソフトウェアのデータレプリケーションストリームのソースおよびターゲットアドレスを提供するために使用されます。

- HAStoragePlus リソースを持つ

HAStoragePlus リソースは、レ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たときに、デバイスグループをフェイルオーバーします。Oracle Solaris 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、レプリケーションリソースグループをフェイルオーバーします。このようにレプリケーションリソースグループとデバイスグループは常に共用関係を持ち、同じノードから制御されます。

HAStoragePlus リソースで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、レ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 についての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 共用関係を持つデバイスグループ名に `-stor-rg` を付けた名前を持つ

たとえば、`devgrp-stor-rg` などです。

- プライマリクラスタとセカンダリ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

高可用性にするためには、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。

HAStoragePlus リソース内で `ZPoolsSearchDir` 拡張プロパティを定義する必要があります。この拡張プロパティは、ZFS ファイルシステムを使用するために必要です。

プライマリ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、セカンダリ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、セカンダリクラスタにレプリケートする必要があります。

このセクションでは、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。

- [303 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」](#)
- [304 ページの「スケラブルアプリケーション向けのリソースグループの構成」](#)

フェイルオーバーアプリケーション向けのリソースグループの構成

フェイルオーバーアプリケーションでは、1つのアプリケーションが1度に1ノード上で動作します。ノードで障害が発生すると、アプリケーションは同じクラスタ内の別のノードにフェイルオーバーします。フェイルオーバーアプリケーション向けリソースグループは、以下の特徴を持っていないければなりません。

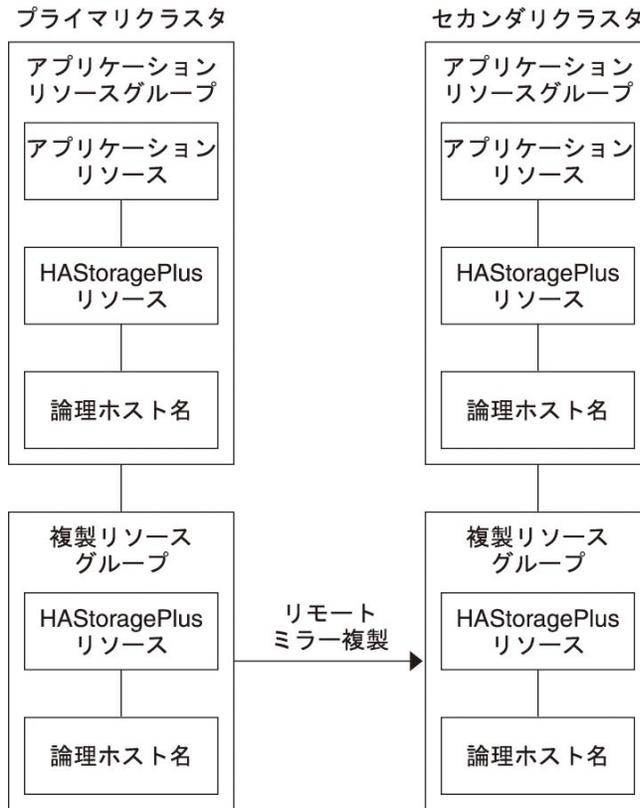
- アプリケーションリソースグループがスイッチオーバーまたはフェイルオーバーされた場合、HAStoragePlus リソースにファイルシステムまたは zpool をフェイルオーバーさせるデバイスグループは、レプリケーションリソースグループおよびアプリケーションリソースグループと共有関係を持ちます。したがって、アプリケーションリソースグループがフェイルオーバーすると、デバイスグループとレプリケーションリソースグループもフェイルオーバーします。アプリケーションリソースグループ、レプリケーションリソースグループおよびデバイスグループは、同じノードによって制御されます。
ただし、デバイスグループやレプリケーションリソースグループがフェイルオーバーしても、アプリケーションリソースグループはフェイルオーバーを行いません。
- アプリケーションデータがグローバルマウントされている場合は、アプリケーションリソースグループに HAStoragePlus リソースを必ず入れなければならないわけではありませんが、入れることをお勧めします。
- アプリケーションデータがローカルマウントされている場合は、アプリケーションリソースグループに HAStoragePlus リソースを必ず入れなければなりません。

HAStoragePlus についての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- プライマリクラスタでオンライン、セカンダリクラスタでオフラインとなる
セカンダリクラスタがプライマリクラスタをテイクオーバーした場合は、セカンダリ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5](#) に、フェイルオーバーアプリケーションでのアプリケーションリソースグループとレプリケーションリソースグループの構成を示します。

図 5 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。

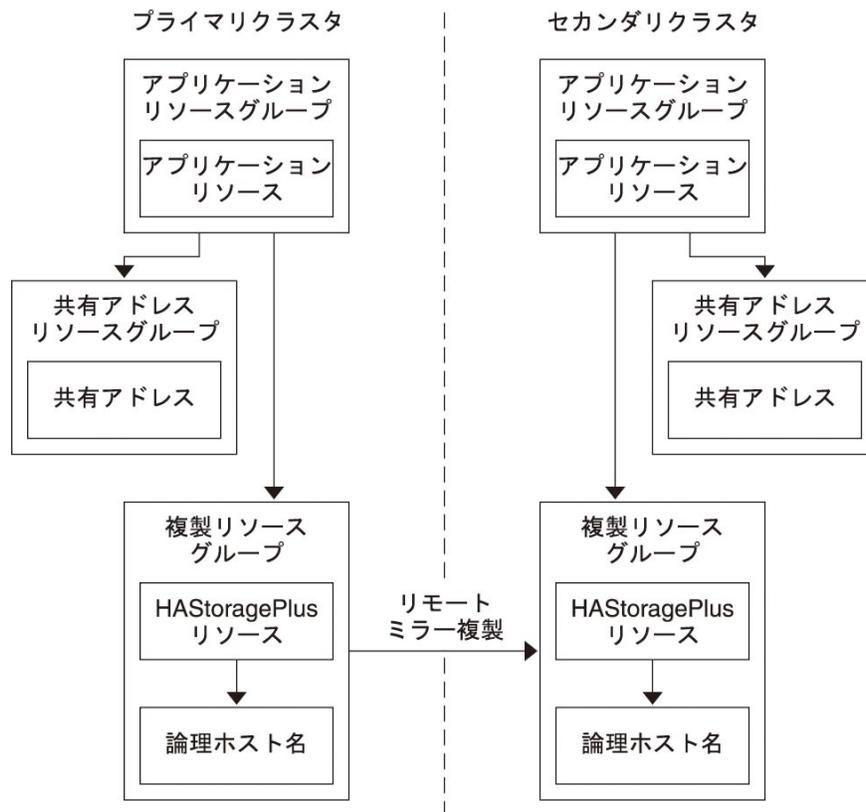
スケーラブルアプリケーションをアプリケーションリソースグループのリソースとして管理している場合は、アプリケーションリソースグループをデバイスグループと結び付ける必要はありません。したがって、アプリケーションリソースグループ向けに HAStoragePlus リソースを作成する必要はありません。

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となりません。

- 共有アドレスのリソースグループと依存関係を持っている
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。
- プライマリクラスタでオンライン、セカンダリクラスタでオフラインとなる

図6 に、スケーラブルアプリケーションでのリソースグループの構成を示します。

図 6 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

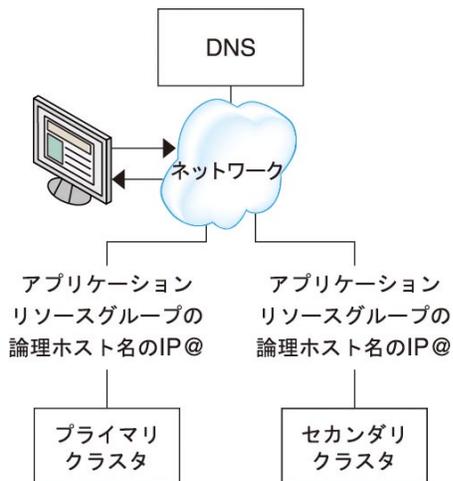


テイクオーバーの管理のガイドライン

プライマリクラスタに障害が発生した場合、アプリケーションをできるだけ早くセカンダリクラスタにスイッチオーバーする必要があります。セカンダリ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。

クライアントは DNS を使用して、アプリケーションの論理ホスト名を IP アドレスにマップします。アプリケーションをセカンダリクラスタに移動することによりテイクオーバーを行ったあとに、アプリケーションの論理ホスト名と新しい IP アドレス間のマッピングが反映されるように DNS 情報を更新する必要があります。

図 7 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。テイクオーバーの管理の例については、[333 ページの「テイクオーバーの管理の例」](#)を参照してください。

修復後、プライマリクラスタをオンラインに戻すことができます。元のプライマリ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. プライマリクラスタとセカンダリクラスタを同期させ、プライマリボリュームが最新であることを確認します。これを行うには、レプリケーションデータストリームがなくなるように、セカンダリノードのリソースグループを停止します。
2. データレプリケーションの方向を逆に、元のプライマリクラスタが元のセカンダリクラスタにふたたびデータをレプリケーションするようにします。
3. プライマリクラスタでリソースグループを起動します。

4. クライアントがプライマリ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。

タスクマップ: データレプリケーションの構成例

表23「タスクマップ: データレプリケーションの構成例」に、Availability Suite ソフトウェアを使用して NFS アプリケーション向けにどのようにデータレプリケーションが構成されたかを示すこの例でのタスクを示します。

表 23 タスクマップ: データレプリケーションの構成例

タスク	手順
1. クラスタを接続およびインストールする	307 ページの「クラスタの接続とインストール」
2. プライマリクラスタとセカンダリクラスタで、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	309 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. プライマリクラスタとセカンダリクラスタでデータレプリケーションを有効にする	324 ページの「プライマリクラスタでレプリケーションを有効にする方法」 326 ページの「セカンダリクラスタでレプリケーションを有効にする方法」
4. データレプリケーションを実行する	327 ページの「リモートミラーレプリケーションを実行する方法」 329 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方法」
5. データレプリケーションの構成を確認する	330 ページの「レプリケーションが正しく構成されていることを確認する方法」

クラスタの接続とインストール

[図8](#) に、構成例で使用するクラスタ構成を示します。構成例のセカンダリ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 8 クラスタ構成例

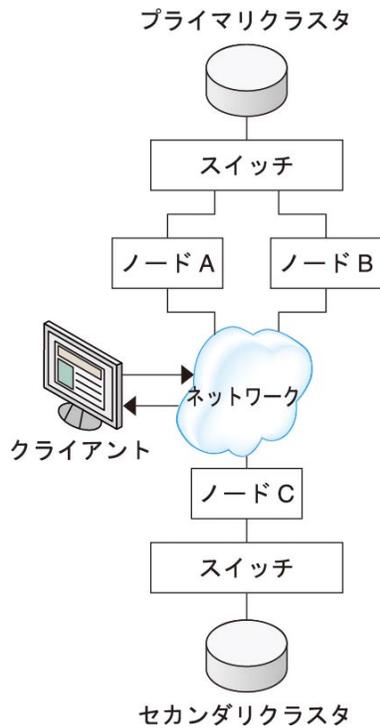


表24「必要なハードウェアとソフトウェア」では、構成例に必要なハードウェアとソフトウェアをまとめています。Oracle Solaris OS、Oracle Solaris Cluster ソフトウェア、およびボリューム管理ソフトウェアは、Availability Suite ソフトウェアとソフトウェア更新をインストールする前にクラスタノードにインストールしてください。

表 24 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	Availability Suite ソフトウェアは、Oracle 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、『 Oracle Solaris Cluster Hardware Administration Manual 』を参照してください。
ディスク容量	約 15M バイト。
Oracle Solaris OS	Oracle Solaris Cluster ソフトウェアがサポートする Oracle Solaris OS のリリース。すべてのノードが同じバージョンの Oracle Solaris OS を使用する必要があります。

ハードウェアまたはソフトウェア	要件
	インストールについては、『 Oracle Solaris Cluster 4.3 ソフトウェアのインストール 』を参照してください
Oracle Solaris Cluster ソフトウェア	少なくとも Oracle Solaris Cluster 4.1 ソフトウェア。 インストールについては、『 Oracle Solaris Cluster 4.3 ソフトウェアのインストール 』を参照してください。
ボリュームマネージャーソフトウェア	Solaris Volume Manager ソフトウェア。すべてのノードで、同じバージョンのボリューム管理ソフトウェアを使用する。 インストールについては、『 Oracle Solaris Cluster 4.3 ソフトウェアのインストール 』の第 4 章、「 Solaris Volume Manager ソフトウェアの構成 」を参照してください。
Availability Suite ソフトウェア	別個のクラスタでは、異なるバージョンの Oracle Solaris OS および Oracle Solaris Cluster ソフトウェアを使用できますが、クラスタ間で同じバージョンの Availability Suite ソフトウェアを使用する必要があります。 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 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。
Availability Suite ソフトウェアの更新	最新のソフトウェア更新については、 My Oracle Support にログインしてください。

デバイスグループとリソースグループの構成例

このセクションでは、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、[302 ページの「レプリケーションリソースグループの構成」](#)および[302 ページの「アプリケーションリソースグループの構成」](#)を参照してください。

ここでは、次の手順について説明します。

- [310 ページの「プライマリクラスタでデバイスグループを構成する方法」](#)
- [311 ページの「セカンダリクラスタでデバイスグループを構成する方法」](#)
- [313 ページの「プライマリ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方法」](#)
- [314 ページの「セカンダリ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方法」](#)
- [315 ページの「プライマリクラスタでレプリケーションリソースグループを作成する方法」](#)
- [317 ページの「セカンダリクラスタでレプリケーションリソースグループを作成する方法」](#)
- [319 ページの「プライマリクラスタで NFS アプリケーションリソースグループを作成する方法」](#)
- [322 ページの「セカンダリクラスタで NFS アプリケーションリソースグループを作成する方法」](#)
- [330 ページの「レプリケーションが正しく構成されていることを確認する方法」](#)

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 25 構成例内のグループとリソースのサマリー

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
レプリケーションリソースグループとリソース	devgrp-stor-rg	レプリケーションリソースグループ
	lhost-reprg-prim, lhost-reprg-sec	プライマリクラスタとセカンダリクラスタのレプリケーションリソースグループの論理ホスト名
	devgrp-stor	レプリケーション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	プライマリクラスタとセカンダリ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。レプリケーションリソースグループの名前は、*devicegroupname-stor-rg* というフォーマットでなければいけません。

Solaris Volume Manager ソフトウェアについては、『[Oracle Solaris Cluster 4.3 ソフトウェアのインストール](#)』の第 4 章、「[Solaris Volume Manager ソフトウェアの構成](#)」を参照してください。

▼ プライマリクラスタでデバイスグループを構成する方法

始める前に 次のタスクを完成していることを確認してください。

- 次のセクションのガイドラインと要件を確認します。
 - [297 ページの「クラスタにおける Availability Suite ソフトウェアの理解」](#)
 - [301 ページの「クラスタ間でホストベースのデータレプリケーションを構成するためのガイドライン」](#)
- [307 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、プライマリクラスタおよびセカンダリクラスタを設定します。

1. RBAC の承認 `solaris.cluster.modify` を提供する役割になって `nodeA` にアクセスします。
`nodeA` は、プライマリ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8](#)を参照してください。
2. NFS データおよび関連するレプリケーションが含まれるようにメタセットを作成します。

```
nodeA# metaset -s nfsset a -h nodeA nodeB
```
3. メタセットにディスクを追加します。

```
nodeA# metaset -s nfsset -a /dev/did/dsk/d6 /dev/did/dsk/d7
```

4. メタセットにメタデータを追加します。

```
nodeA# metaset -s nfsset -a -m nodeA nodeB
```

5. 必要なボリューム (またはメタデバイス) を作成します。

ミラーのコンポーネントを 2 つ作成します。

```
nodeA# metainit -s nfsset d101 1 1 /dev/did/dsk/d6s2
nodeA# metainit -s nfsset d102 1 1 /dev/did/dsk/d7s2
```

いずれかのコンポーネントを使用してミラーを作成します。

```
nodeA# metainit -s nfsset d100 -m d101
```

もう 1 つのコンポーネントをミラーに接続して、同期できるようにします。

```
nodeA# metattach -s nfsset d100 d102
```

次の例のようにミラーからソフトパーティションを作成します。

■ d200 - NFS データ (マスターボリューム)

```
nodeA# metainit -s nfsset d200 -p d100 50G
```

■ d201 - NFS データのポイントインタイムコピーボリューム

```
nodeA# metainit -s nfsset d201 -p d100 50G
```

■ d202 - ポイントインタイムビットマップボリューム

```
nodeA# metainit -s nfsset d202 -p d100 10M
```

■ d203 - リモートシャドウビットマップボリューム

```
nodeA# metainit -s nfsset d203 -p d100 10M
```

■ d204 - Oracle Solaris Cluster SUNW.NFS 構成情報用のボリューム

```
nodeA# metainit -s nfsset d204 -p d100 100M
```

6. NFS データおよび構成ボリュームのファイルシステムを作成します。

```
nodeA# yes | newfs /dev/md/nfsset/rdisk/d200
nodeA# yes | newfs /dev/md/nfsset/rdisk/d204
```

次の手順 [311 ページの「セカンダリクラスタでデバイスグループを構成する方法」](#)に進みます。

▼ セカンダリクラスタでデバイスグループを構成する方法

始める前に 手順[310 ページの「プライマリクラスタでデバイスグループを構成する方法」](#)を完了します。

1. RBAC の承認 `solaris.cluster.modify` を提供する役割になって `nodec` にアクセスします。
2. NFS データおよび関連するレプリケーションが含まれるようにメタセットを作成します。

```
nodeC# metaset -s nfsset a -h nodeC
```

3. メタセットにディスクを追加します。

次の例では、ディスク DID 番号が異なると仮定しています。

```
nodeC# metaset -s nfsset -a /dev/did/dsk/d3 /dev/did/dsk/d4
```

注記 - 単一のノードクラスタではメディエータは必要ありません。

4. 必要なボリューム (またはメタデバイス) を作成します。

ミラーのコンポーネントを 2 つ作成します。

```
nodeC# metainit -s nfsset d101 1 1 /dev/did/dsk/d3s2
nodeC# metainit -s nfsset d102 1 1 /dev/did/dsk/d4s2
```

いずれかのコンポーネントを使用してミラーを作成します。

```
nodeC# metainit -s nfsset d100 -m d101
```

もう 1 つのコンポーネントをミラーに接続して、同期できるようにします。

```
metattach -s nfsset d100 d102
```

次の例のようにミラーからソフトパーティションを作成します。

■ *d200* - NFS データのマスターボリューム

```
nodeC# metainit -s nfsset d200 -p d100 50G
```

■ *d201* - NFS データのポイントインタイムコピーボリューム

```
nodeC# metainit -s nfsset d201 -p d100 50G
```

■ *d202* - ポイントインタイムビットマップボリューム

```
nodeC# metainit -s nfsset d202 -p d100 10M
```

■ *d203* - リモートシャドウビットマップボリューム

```
nodeC# metainit -s nfsset d203 -p d100 10M
```

■ *d204* - Oracle Solaris Cluster SUNW.NFS 構成情報用のボリューム

```
nodeC# metainit -s nfsset d204 -p d100 100M
```

5. NFS データおよび構成ボリュームのファイルシステムを作成します。

```
nodeC# yes | newfs /dev/md/nfsset/rdisk/d200
nodeC# yes | newfs /dev/md/nfsset/rdisk/d204
```

次の手順 313 ページの「プライマリ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方法」に進みます。

▼ プライマリ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方法

始める前に 手順311 ページの「セカンダリクラスタでデバイスグループを構成する方法」を完了します。

1. **nodeA** および **nodeB** で、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。
例を示します。

```
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ない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/md/nfsset/dsk/d200 /dev/md/nfsset/rdisk/d200 \  
/global/mountpoint ufs 3 no global,logging
```

4. **nodeA** と **nodeB** で、メタデバイス `d204` 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
```

5. **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メタデバイス `d204` 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/md/nfsset/dsk/d204 /dev/md/nfsset/rdisk/d204 \  
/global/etc ufs 3 yes global,logging
```

6. **nodeA** にメタデバイス `d204` をマウントします。

nodeA# mount /global/etc
7. Oracle Solaris Cluster HA for NFS データサービスの構成ファイルおよび情報を作成します。
 - 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 314 ページの「セカンダリ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方法」に進みます。

▼ セカンダリ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方法

始める前に 手順313 ページの「プライマリ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方法」を完了します。

1. nodeC で、RBAC の承認 solaris.cluster.admin を提供する役割になります。
2. nodeC で、NFS ファイルシステム向けのマウントポイントディレクトリを作成します。
例を示します。

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

3. nodeC で、マウントポイントに自動でマウントされるようにマスターボリュームを構成します。
nodeC の /etc/vfstab ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは 1 行で記述してください。

```
/dev/md/nfsset/dsk/d200 /dev/md/nfsset/rdisk/d200 \  
/global/mountpoint ufs 3 yes global,logging
```

4. nodeA にメタデバイス d204 をマウントします。

```
nodeC# mount /global/etc
```
5. Oracle Solaris Cluster HA for NFS データサービスの構成ファイルおよび情報を作成します。

- a. nodeA に /global/etc/SUNW.nfs というディレクトリを作成します。

```
nodeC# 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。

```
nodeC# 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 315 ページの「プライマリクラスタでレプリケーションリソースグループを作成する方法」に進みます。

▼ プライマリクラスタでレプリケーションリソースグループを作成する方法

- 始める前に
- 手順314 ページの「セカンダリ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方法」を完了します。
 - すべての論理ホスト名の IP アドレスのサブネットとネットマスクのエントリが /etc/netmasks ファイルにあることを確認してください。必要に応じて、/etc/netmasks ファイルを編集して、不足しているエントリがある場合は追加します。

1. RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割として、nodeA にアクセスします。

2. SUNW.HASStoragePlus リソースタイプを登録します。

```
nodeA# clresourcetype register SUNW.HASStoragePlus
```

3. デバイスグループのレプリケーション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 がレプリケーション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
```

レプリケーション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. SUNW.HASStoragePlus リソースをレプリケーションリソースグループに追加します。

```
nodeA# clresource create -g devgrp-stor-rg -t SUNW.HASStoragePlus \
-p GlobalDevicePaths=nfsset \
-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=

Availability Suite ソフトウェアが依存するデバイスグループを指定します。

-p AffinityOn=True

SUNW.HAStoragePlus リソースが、-p 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必要があることを指定します。したがって、レ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。

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

5. 論理ホスト名リソースをレ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 devgrp-stor-rg lhost-reprg-prim
```

プライマリクラスタ上のレプリケーションリソースグループの論理ホスト名は lhost-reprg-prim です。

6. リソースを有効にし、リソースグループを管理し、リソースグループをオンラインにします。

```
nodeA# clresourcegroup onLine -emM -n nodeA devgrp-stor-rg
```

-e

関連付けられたリソースを有効にします。

-M

リソースグループを管理状態にします。

-n

リソースグループをオンラインにするノードを指定します。

7. リソースグループがオンラインであることを確認します。

```
nodeA# clresourcegroup status devgrp-stor-rg
```

リソースグループの状態フィールドを調べ、レプリケーション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 [317 ページの「セカンダリクラスタでレプリケーションリソースグループを作成する方法」](#)に進みます。

▼ セカンダリクラスタでレプリケーションリソースグループを作成する方法

- 始める前に
- 手順315 ページの「プライマリクラスタでレプリケーションリソースグループを作成する方法」を完了します。
 - すべての論理ホスト名の IP アドレスのサブネットとネットマスクのエントリが `/etc/netmasks` ファイルにあることを確認してください。必要に応じて、`/etc/netmasks` ファイルを編集して、不足しているエントリがある場合は追加します。

1. RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割として、`nodeC` にアクセスします。

2. リソースタイプとして `SUNW.HAStoragePlus` を登録します。

```
nodeC# clresourcetype register SUNW.HAStoragePlus
```

3. デバイスグループのレプリケーション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
```

レプリケーションリソースグループの名前。

4. レ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 \  
-t SUNW.HAStoragePlus \  
-p GlobalDevicePaths=nfsset \  
-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=

Availability Suite ソフトウェアが依存するデバイスグループを指定します。

-p AffinityOn=True

SUNW.HAStoragePlus リソースが、**-p 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必要があることを指定します。したがって、レ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。

devgrp-stor

レプリケーションリソースグループの HAStoragePlus リソース

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

5. 論理ホスト名リソースをレ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 devgrp-stor-rg lhost-reprg-sec
```

セカンダリクラスタ上のレプリケーション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 -eM -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、レプリケーション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 [319 ページの「プライマリクラスタで NFS アプリケーションリソースグループを作成する方法」](#)に進みます。

▼ プライマリクラスタで NFS アプリケーションリソースグループを作成する方法

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317 ページの「セカンダリクラスタでレプリケーションリソースグループを作成する方法」](#)を完了します。
 - すべての論理ホスト名の IP アドレスのサブネットとネットマスクのエントリが `/etc/netmasks` ファイルにあることを確認してください。必要に応じて、`/etc/netmasks` ファイルを編集して、不足しているエントリがある場合は追加します。

1. RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割として、`nodeA` にアクセスします。

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. NFS サービス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_affinities=+++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_affinities=+++devgrp-stor-rg
```

アプリケーションリソースグループを結び付ける必要があ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レプリケーションリソースグループ `devgrp-stor-rg` に結び付いている必要があります。

レプリケーションリソースグループが新しいプライマリノードにスイッチオーバーすると、アプリケーション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、この操作により結び付きの要件が破られるため、アプリケーションリソースグループを新しいプライマリノードにスイッチオーバーしようとするするとブロックされます。

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

リソースが SUNW.HAStoragePlus タイプであることを指定します。

-p FileSystemMountPoints=/global/mountpoint

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 FileSystemMountPoints 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必要があることを指定します。したがって、アプリケーションリソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

nfs-dg-rs

NFS アプリケーション向けの HAStoragePlus リソースの名前。

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

6. 論理ホスト名リソースをアプリケーションリソースグループに追加します。

```
nodeA# clreslogicalhostname create -g nfs-rg \  
lhost-nfsrg-prim
```

プライマリ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. `dfstab.resource-name` 構成ファイルを作成し、そのリソースを含むリソースグループの `Pathprefix` ディレクトリの下に `SUNW.nfs` サブディレクトリ内に配置します。

- a. `nodeA` に `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 `dfstab.resource-name` 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
```

8. 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 -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。

9. 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 [322 ページの「セカンダリクラスタで NFS アプリケーションリソースグループを作成する方法」](#)に進みます。

▼ セカンダリクラスタで NFS アプリケーションリソースグループを作成する方法

- 始める前に
- 手順319 ページの「プライマリクラスタで NFS アプリケーションリソースグループを作成する方法」を完了します。
 - すべての論理ホスト名の IP アドレスのサブネットとネットマスクのエントリが `/etc/netmasks` ファイルにあることを確認してください。必要に応じて、`/etc/netmasks` ファイルを編集して、不足しているエントリがある場合は追加します。

1. RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割として、`nodec` にアクセスします。

2. `SUNW.nfs` をリソースタイプとして登録します。

```
nodeC# clresourcetype register SUNW.nfs
```

3. `SUNW.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。

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

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_affinities=+++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_affinities=+++devgrp-stor-rg
```

アプリケーションリソースグループを結び付ける必要があ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レプリケーションリソースグループ `devgrp-stor-rg` に結び付いている必要があります。

レプリケーションリソースグループが新しいプライマリノードにスイッチオーバーすると、アプリケーション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、これにより結び付き

の要件が破られるため、アプリケーションリソースグループを新しいプライマリノードにスイッチオーバーしようとするするとブロックされます。

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

リソースが SUNW.HAStoragePlus タイプであることを指定します。

-p

リソースのプロパティを指定します。

FileSystemMountPoints=/global/mountpoint

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 -p FileSystemMountPoints= 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必要があることを指定します。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。

6. 論理ホスト名リソースをアプリケーションリソースグループに追加します。

```
nodeC# clreslogicalhostname create -g nfs-rg \  
lhost-nfsrg-sec
```

セカンダリクラスタ上のアプリケーションリソースグループの論理ホスト名は lhost-nfsrg-sec です。

7. NFS リソースをアプリケーションリソースグループに追加します。

```
nodeC# clresource create -g nfs-rg \  
-t SUNW.nfs -p Resource_dependencies=nfs-dg-rs nfs-rg
```

8. グローバルボリュームがプライミクラスタにマウントされている場合は、セカンダリ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
```

ボリュームがセカンダリ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 [324 ページの「データレプリケーションの有効化例」](#)に進みます。

データレプリケーションの有効化例

このセクションでは、構成例のデータレプリケーションをどのように有効にするかを説明します。このセクションでは、Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、Availability Suite のドキュメントを参照してください。

ここでは、次の手順について説明します。

- [324 ページの「プライミクラスタでレプリケーションを有効にする方法」](#)
- [326 ページの「セカンダリクラスタでレプリケーションを有効にする方法」](#)

▼ プライミクラスタでレプリケーションを有効にする方法

1. RBAC の承認 `solaris.cluster.read` を提供する役割として、`nodeA` 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。

```
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. プライミクラスタからセカンダリクラスタへのリモートミラーレプリケーションを有効にします。この手順では、プライミクラスタからセカンダリクラスタへのレプリケーションを有効にします。この手順によって、プライミクラスタのマスターボリューム (`d200`) からセカンダリクラスタのマスターボリューム (`d200`) へのレプリケーションが有効になります。さらに、この手順で `d203` のリモートミラービットマップへのレプリケーションも有効になります。

- プライマリクラスタとセカンダリ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。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -e lhost-reprg-prim \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 ip sync
```

- プライマリクラスタとセカンダリ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。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -E lhost-reprg-prim \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 ip sync
```

5. 自動同期機能を有効にします。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -a on lhost-reprg-prim \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 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、システムがリブート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。

次のコマンドを実行します。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/md/nfsset/rdisk/d200 ->  
lhost-reprg-sec:/dev/md/nfsset/rdisk/d200  
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せん。

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。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/iiadm -e ind \  

```

```

/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d201 \
/dev/md/nfsset/rdisk/d202
nodeA# /usr/sbin/iiadm -w \
/dev/md/nfsset/rdisk/d201

```

この手順によって、プライマリ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 d200、シャドウボリュームは d201、ポイントインタイムビットマップボリュームは d203 です。

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。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```

nodeA# /usr/sbin/sndradm -I a \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d201 \
/dev/md/nfsset/rdisk/d202

```

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。Availability Suite ソフトウェアは、リモートミラーレプリケーションの前にポイントインタイムスナップショットを必ず取ります。

次の手順 [326 ページの「セカンダリクラスタでレプリケーションを有効にする方法」](#)に進みます。

▼ セカンダリクラスタでレプリケーションを有効にする方法

始める前に 手順[324 ページの「プライマリクラスタでレプリケーションを有効にする方法」](#)を完了します。

1. root 役割として、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. プライマリクラスタからセカンダリクラスタへのリモートミラーレプリケーションを有効にします。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```

nodeC# /usr/sbin/sndradm -n -e lhost-reprg-prim \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 ip sync

```

プライマリクラスタがセカンダリ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、Availability Suite のシステムログファイル /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。
次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/iiadm -e ind \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d201 \  
/dev/md/nfsset/rdisk/d202  
nodeC# /usr/sbin/iiadm -w \  
/dev/md/nfsset/rdisk/d201
```

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。
次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -I a \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d201 \  
/dev/md/nfsset/rdisk/d202
```

次の手順 [327 ページの「データレプリケーションの実行例」](#)に進みます。

データレプリケーションの実行例

このセクションでは、構成例のデータレプリケーションをどのように実行するかを説明します。このセクションでは、Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、Availability Suite のドキュメントを参照してください。

ここでは、次の手順について説明します。

- [327 ページの「リモートミラーレプリケーションを実行する方法」](#)
- [329 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方法」](#)
- [330 ページの「レプリケーションが正しく構成されていることを確認する方法」](#)

▼ リモートミラーレプリケーションを実行する方法

この手順では、プライマリディスクのマスターボリュームがセカンダリディスクのマスターボリュームにレプリケートされます。マスターボリュームは `d200` で、リモートミラービットマップボリュームは `d203` です。

1. `root` 役割として、`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。
次のコマンドを実行します。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/md/nfsset/rdisk/d200 ->
lhost-reprg-sec:/dev/md/nfsset/rdisk/d200
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せん。

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 でステップ 1 からステップ 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。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -m lhost-reprg-prim \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 ip sync
```

6. レプリケーションが完了し、ボリュームが同期化されるのを待ちます。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -w lhost-reprg-prim \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 ip sync
```

7. クラスタがレプリケートモードであることを確認します。

次のコマンドを実行します。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/md/nfsset/rdisk/d200 ->
lhost-reprg-sec:/dev/md/nfsset/rdisk/d200
autosync: on, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: replicating
```

レプリケートモードでは、状態は replicating で、自動同期のアクティブ状態は on です。プライマリボリュームに書き込みが行われると、Availability Suite ソフトウェアがセカンダリボリュームを更新します。

次の手順 [329 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方法」](#)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方法

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、プライマリクラスタのシャドウボリュームをプライマリ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 d200、ビットマップボリュームは d203、シャドウボリュームは d201 です。

始める前に 手順[327 ページの「リモートミラーレプリケーションを実行する方法」](#)を完了します。

1. RBAC の承認 `solaris.cluster.modify` および `solaris.cluster.admin` を提供する役割として、`nodeA` にアクセスします。

2. `nodeA` で実行されているリソースを無効にします。

```
nodeA# clresource disable nfs-rs
```

3. プライマリクラスタをロギングモードに変更します。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 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。レプリケーションは行われません。

4. プライマリクラスタのシャドウボリュームをプライマリクラスタのマスターボリュームに同期化させます。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/iiadm -u s /dev/md/nfsset/rdisk/d201  
nodeA# /usr/sbin/iiadm -w /dev/md/nfsset/rdisk/d201
```

5. セカンダリクラスタのシャドウボリュームをセカンダリクラスタのマスターボリュームに同期化させます。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/iiadm -u s /dev/md/nfsset/rdisk/d201  
nodeC# /usr/sbin/iiadm -w /dev/md/nfsset/rdisk/d201
```

6. `nodeA` でアプリケーションを再起動します。

```
nodeA# clresource enable nfs-rs
```

7. セカンダリボリュームをプライマリボリュームと再同期化させます。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 ip sync
```

次の手順 [330 ページの「レプリケーションが正しく構成されていることを確認する方法」](#)に進みます。

▼ レプリケーションが正しく構成されていることを確認する方法

始める前に 手順[329 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方法」](#)を完了します。

1. RBAC の承認 `solaris.cluster.admin` を提供する役割として、`nodeA` および `nodeC` にアクセスします。
2. プライマリクラスタがレプリケートモードで、自動同期機能がオンになっていることを確認します。

次のコマンドを実行します。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/md/nfsset/rdisk/d200 ->  
lhost-reprg-sec:/dev/md/nfsset/rdisk/d200  
autosync: on, max q writes:4194304, max q fbas:16384, mode:sync,ctag:  
devgrp, state: replicating
```

レプリケートモードでは、状態は `replicating` で、自動同期のアクティブ状態は `on` です。プライマリボリュームに書き込みが行われると、Availability Suite ソフトウェアがセカンダリボリュームを更新します。

3. プライマリクラスタがレプリケートモードでない場合は、レプリケートモードにします。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 ip sync
```

```
/dev/md/nfsset/rdisk/d203 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. root 役割として、クライアントマシンにログインします。

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. プライマリボリュームをアプリケーションディレクトリにマウントし、マウントしたディレクトリを表示します。

- a. プライマリボリュームをアプリケーションディレクトリにマウントします。

```
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. アプリケーションディレクトリからプライマリボリュームのマウントを解除します。

- a. アプリケーションディレクトリからプライマリボリュームのマウントを解除します。

```
client-machine# umount /dir
```

- b. プライマリ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 -g nfs-rg +
nodeA# clresourcegroup offline nfs-rg
```

- c. プライマリクラスタをロギングモードに変更します。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \
/dev/md/nfsset/rdisk/d200 \
/dev/md/nfsset/rdisk/d203 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。レプリケーションは行われません。

- 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. ファイルシステムがセカンダリクラスタへのマウントに適していることを確認します。

```
nodeC# fsck -y /dev/md/nfsset/rdisk/d200
```

- f. アプリケーションを管理状態にして、セカンダリクラスタでオンラインにします。

```
nodeC# clresourcegroup online -eM nodeC nfs-rg
```

- g. root 役割として、クライアントマシンにアクセスします。

次のようなプロンプトが表示されます。

```
client-machine#
```

- h. [ステップ 4](#) で作成したアプリケーションディレクトリをセカンダリボリュームのアプリケーションディレクトリにマウントします。

```
client-machine# mount -o rw lhost-nfsrg-sec:/global/mountpoint /dir
```

- i. マウントしたディレクトリを表示します。

```
client-machine# ls /dir
```

7. [ステップ 6](#) で表示されたディレクトリが[ステップ 5](#) 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. プライマリボリュームのアプリケーションをマウントされたアプリケーションディレクトリに戻します。

- a. セカンダリボリューム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 -g nfs-rg +
nodeC# clresourcegroup offline nfs-rg
```

- b. 必ずグローバルボリュームをセカンダリボリュームからマウント解除します。

```
nodeC# umount /global/mountpoint
```

- c. アプリケーションリソースグループを管理状態にして、プライマリクラスタでオンラインにします。

```
nodeA# clresourcegroup online -eM nodeA nfs-rg
```

- d. プライマリボリュームをレプリケートモードに変更します。

次のコマンドを実行します。Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 lhost-reprg-sec \  
/dev/md/nfsset/rdisk/d200 \  
/dev/md/nfsset/rdisk/d203 ip sync
```

プライマリボリュームに書き込みが行われると、Availability Suite ソフトウェアがセカンダリボリュームを更新します。

参照 [333 ページの「テイクオーバーの管理の例」](#)

テイクオーバーの管理の例

ここでは、DNS エントリを更新する方法を説明します。詳細は、[306 ページの「テイクオーバーの管理のガイドライン」](#)を参照してください。

▼ DNS エントリを更新する方法

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7](#)を参照してください。

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

プライマリクラスタの IP アドレス (逆順) です。

ipaddress2rev

セカンダリクラスタの 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。

プライマリ論理ホスト名をセカンダリクラスタの IP アドレスにマッピングし、セカンダリ論理ホスト名をプライマリクラスタの 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

セカンダリクラスタの IP アドレス (正順) です。

ipaddress1fwd

プライマリクラスタの IP アドレス (正順) です。

索引

あ

- アクセシビリティ
 - Oracle Solaris Cluster Manager, 293
- アクセス
 - Oracle Solaris Cluster Manager, 290
- アクセス権、グローバルデバイス, 106
- アダプタ、トランスポート, 194
- アップグレード
 - 概要, 275
 - ブランドタイプ solaris10 のフェイルオーバーゾーン, 275
 - ブランドタイプ solaris のフェイルオーバーゾーン, 275
- アフィニティスイッチオーバー
 - データレプリケーションの構成, 316
- アプリケーションリソースグループ
 - ガイドライン, 302
 - データレプリケーションの構成, 319
- アンインストール
 - lofi デバイスファイル, 244
 - Oracle Solaris Cluster ソフトウェア, 241
 - パッケージ, 282
- 移行
 - グローバルデバイス名前空間, 124
- 一覧表示
 - 定足数構成, 181
 - デバイスグループ構成, 142
- イベント MIB
 - log_number の変更, 247
 - min_severity の変更, 247
 - SNMP の有効化と無効化, 246, 246
 - SNMP プロトコルの変更, 247
- インターコネクト
 - トラブルシューティング, 202
 - 有効化, 202
- エラーメッセージ

- /var/adm/messages ファイル, 96
- ノードの削除, 223

か

- 回復
 - ストレージベースのデータレプリケーションを使用するクラスタ, 103
- 概要
 - 定足数, 165
- 確認
 - vfstab 構成, 154
 - クラスタノードのステータス, 216
 - グローバルマウントポイント, 56, 156
 - データレプリケーション構成, 330
- 完全な修復 /var/adm/messages ファイル, 96
- 管理
 - EMC SRDF でレプリケートされたデバイス, 108
 - IPMP, 204
 - PNM, 189
 - クラスタインターコネクトとパブリックネットワーク, 189
 - クラスタファイルシステム, 120
 - グローバルクラスタ, 24
 - グローバルクラスタ設定, 225
 - ストレージベースのレプリケートされたデバイス, 107
 - ゾーンクラスタ, 24, 254
 - 定足数, 165
- 管理コンソール, 29
- 管理ツール
 - Oracle Solaris Cluster Manager, 289
- 起動
 - pconsole ユーティリティ, 215
 - グローバルクラスタ, 71
 - グローバルクラスタノード, 82
 - ゾーンクラスタ, 71

- ゾーンクラスタノード, 82
- キャンパスクラスタ
 - ストレージベースのデータレプリケーション, 100
 - ストレージベースのデータレプリケーションを使用した回復, 103
- 共通エージェントコンテナ
 - セキュリティー鍵の構成, 292
- 共有ディスクパス
 - 自動リブートの無効化, 164
 - 自動リブートの有効化, 163
 - モニタリング, 157
- 共有 SCSI ディスク
 - 定足数デバイスとしてサポート, 167
- 切り替え
 - デバイスグループのプライマリノード, 143
- クラスタ
 - 時間の設定, 230
 - スコープ, 33
 - 名前の変更, 226
 - ノード認証, 228
 - バックアップ, 30, 283
 - ファイルの復元, 286
- クラスタインターコネクト
 - 管理, 189
 - 動的再構成, 191
- クラスタコンソールへのセキュア接続, 32
- クラスタの時間の設定, 230
- クラスタのモニタリング
 - Oracle Solaris Cluster Manager トポロジによる, 294
- クラスタファイルシステム, 105
 - 管理, 120
 - 構成の確認, 154
 - 削除, 154
 - 追加, 152
 - マウントオプション, 153
- クラスタファイルシステムのマウントオプション
 - 要件, 153
- グローバル
 - デバイス, 105
 - アクセス権の設定, 106
 - 動的再構成, 106
 - 名前空間, 105, 122
 - マウントポイント, 確認, 56, 156
 - グローバルクラスタ
 - 管理, 225
 - 構成情報の表示, 42
 - 構成の検証, 50
 - コンポーネントのステータス, 38
 - 定義, 24
 - 停止, 67
 - ノードの削除, 219
 - ブート, 67
 - リポート, 75
 - グローバルクラスタノード
 - CPU シェア, 272
 - 確認
 - ステータス, 216
 - 停止, 82
 - ブート, 82
 - リポート, 90
 - グローバルデバイス名前空間
 - 移行, 124
 - グローバル名前空間の更新, 122
 - ケーブル、トランスポート, 194
 - ゲストドメイン, 86
- 検索
 - グローバルクラスタ用ノード ID, 227
 - ゾーンクラスタ用ノード ID, 227
- 検証
 - グローバルクラスタ構成, 50
 - ゾーンクラスタ構成, 50
- 権利プロファイル
 - RBAC, 61
- 更新, 279
 - 参照 パッチ
 - AI サーバー, 281
 - Oracle GlassFish の制限, 289
 - Oracle Solaris Cluster Manager の制限, 289
 - solaris ブランドゾーンクラスタ, 279
 - 概要, 275
 - 定足数サーバー, 281
 - 特定のパッケージ, 278
 - ヒント, 277
- 構成
 - HAStoragePlus を使用しないローカル ZFS ストレージプール, 131
 - ストレージベースのレプリケートされたデバイス, 107
 - セキュリティー鍵, 292
 - データレプリケーション, 297
 - ノード上の負荷制限, 252
 - レプリケートされた ZFS デバイスグループ, 130

- 構成例 (キャンパスクラスタ)
 - 2 ルーム、ストレージベースのデータレプリケーション, 100
 - 2 ルーム、ストレージベースのレプリケーション, 100
- 公平配分スケジューラ
 - CPU シェアの構成, 272
- コマンド, 27
 - boot, 71
 - cconsole, 31
 - claccess, 27
 - cldevice, 27
 - cldevicegroup, 27
 - clinterconnect, 27
 - clnasdevice, 27
 - clquorum, 27
 - clreslogicalhostname, 27
 - clresource, 27
 - clresourcegroup, 27
 - clresourcetype, 27
 - clressharedaddress, 27
 - clsetup, 27
 - clsnmpghost, 27
 - clsnmpmib, 27
 - clsnmpuser, 27
 - cltelemetryattribute, 27
 - cluster check, 27, 30, 50, 56
 - cluster shutdown, 67
 - clzonecluster boot, 71
 - clzonecluster, 27, 67
 - clzonecluster verify, 50
 - metaset, 105
- コマンド行管理ツール, 26
- コンソール
 - 接続, 31

- さ
- 再起動
 - グローバルクラスタノード, 90
 - ゾーンクラスタノード, 90
- 最後の定足数デバイス
 - 削除, 175
- 削除
 - SNMP
 - ホスト, 249
 - ユーザー, 251
 - Solaris Volume Manager デバイスグループ, 133
 - クラスタファイルシステム, 154
 - 最後の定足数デバイス, 175
 - ストレージレイ, 221
 - すべてのデバイスグループからのノード, 133
 - ゾーンクラスタから, 217
 - ゾーンクラスタからのリソースとリソースグループ, 259
 - 定足数デバイス, 167, 174
 - トランスポートケーブル、アダプタ、およびスイッチ, 194
 - ノード, 216, 219
 - サポートされている定足数デバイスタイプ, 167
- 修復
 - 定足数デバイス, 182
- 出力
 - 障害のあるディスクパス, 160
- 使用
 - 役割 (RBAC), 61
 - スイッチ、トランスポート, 194
- スイッチバック
 - データレプリケーションで実行するためのガイドライン, 306
- ステータス
 - グローバルクラスタのコンポーネント, 38
 - ゾーンクラスタのコンポーネント, 38
- ストレージレイ
 - 削除, 221
- ストレージベースのデータレプリケーション, 100
 - 回復, 103
 - 制限, 102
 - 定義, 98
 - 定足数デバイス, 103
 - ベストプラクティス, 103
 - 要件, 102
- ストレージベースのレプリケートされたデバイス
 - 管理, 107
 - 構成, 107
- スナップショット 参照 ストレージベースのレプリケーション
 - ポイントインタイム, 299
- セカンダリ
 - 希望数の設定, 140

- デフォルト数, 138
- セキュリティー鍵
 - 構成, 292
- 設定
 - 役割 (RBAC), 61
- ゾーンクラスタ
 - solaris10 ブランド
 - clzonecluster によるパッチの適用, 279
 - patchadd によるパッチの適用, 280
 - 代替 BE のパッチの適用, 281
 - solaris ブランド
 - 更新, 279
 - アプリケーションのための準備, 254
 - 管理, 225
 - クローニング, 254
 - 構成情報の表示, 49
 - 構成の検証, 50
 - コンポーネントのステータス, 38
 - 作成, 24
 - サポートされている直接マウント, 260
 - ゾーンパスの移動, 254
 - 定義, 24
 - 停止, 67
 - 統合アーカイブからのインストール, 256
 - 統合アーカイブからの構成, 255
 - ネットワークアドレスの追加, 257
 - ファイルシステムの削除, 254
 - ブート, 67
 - ブランド, 23
 - リブート, 75
- ゾーンクラスタノード
 - IP アドレスおよび NIC の指定, 209
 - 停止, 82
 - ブート, 82
 - リブート, 90
- ゾーンパス
 - 移動, 254
- 属性 参照 プロパティ
- ソフトウェアの更新
 - 概要, 275
- た**
- 代替 BE
 - solaris10 ブランドゾーンクラスタへのパッチの適用, 281
- タイムアウト
 - 定足数デバイスのデフォルトの変更, 183
- 直接接続共有ディスク定足数デバイス
 - 追加, 169
- 直接マウント
 - ゾーンクラスタへのファイルシステムのエクスポート, 260
- 追加
 - SNMP
 - ホスト, 248
 - ユーザー, 250
 - Solaris Volume Manager デバイスグループ, 129
 - ZFS
 - HAStoragePlus を使用しないストレージプール, 131
 - レプリケートされたデバイスグループ, 130
 - カスタム役割 (RBAC), 64
 - クラスタファイルシステム, 152
 - ゾーンクラスタへのネットワークアドレス, 257
 - 直接接続共有ディスク定足数デバイス, 169
 - 定足数デバイス, 168
 - Oracle ZFS Storage Appliance NAS, 170
 - 定足数サーバー, 171
 - デバイスグループ, 127, 129
 - トランスポートケーブル、アダプタ、およびスイッチ, 38, 42, 190
 - ノード, 209
 - グローバルクラスタへの, 210
 - ゾーンクラスタへの, 210
 - 役割 (RBAC), 63
- 停止
 - グローバルクラスタ, 67, 75
 - グローバルクラスタノード, 82, 82
 - ゾーンクラスタ, 67, 75
 - ゾーンクラスタノード, 82, 82
 - ノード, 82
- ディスクパス
 - ステータスエラーの解決, 161
 - モニタリング, 105, 157, 158
 - 障害のあるディスクパスの出力, 160
 - モニタリング解除, 159
- 定足数
 - 概要, 165
 - 管理, 165
- 定足数サーバー 参照 定足数デバイス

- 定足数サーバー定足数デバイス
 - インストールの要件, 171
 - 削除のトラブルシューティング, 175
 - 追加, 171
- 定足数デバイス
 - 交換, 176
 - 構成の一覧表示, 181
 - 削除, 167, 174
 - 最後の定足数デバイス, 175
 - サポートされているタイプ, 167
 - 修復, 182
 - ストレージベースのデータレプリケーション, 103
 - 追加, 168
 - Oracle ZFS Storage Appliance NAS, 170
 - 直接接続共有ディスク定足数デバイス, 169
 - 定足数サーバー, 171
 - デバイスの動的再構成, 167
 - デフォルトのタイムアウトの変更, 183
 - ノードリストの変更, 177
 - 保守状態、デバイス, 178
 - 保守状態、デバイスを保守状態から戻す, 180
- 定足数デバイスの交換, 176
- データレプリケーション, 97
 - DNS エントリの更新, 333
 - ガイドライン
 - スイッチオーバーの管理, 306
 - テイクオーバーの管理, 306
 - リソースグループの構成, 301
 - 概要, 297
 - 構成
 - NFS アプリケーション用ファイルシステム, 313
 - NFS アプリケーションリソースグループ, 319
 - アフィニティスイッチオーバー, 302, 316
 - デバイスグループ, 310
 - 構成の確認, 330
 - 構成例, 307
 - ストレージベース, 98, 100
 - 定義, 98
 - テイクオーバーの管理, 333
 - 同期, 299
 - 必要なハードウェアとソフトウェア, 308
 - 非同期, 299
 - ポイントインタイムスナップショット, 299, 329
 - ホストベースの, 98, 297
 - 有効化, 324
 - リソースグループ
 - アプリケーション, 302
 - 共有アドレス, 305
 - 構成, 302
 - 作成, 315
 - スケーラブルアプリケーション, 304
 - フェイルオーバーアプリケーション, 303
 - 命名規則, 302
 - リモートミラー, 298, 327
 - 例, 327
 - データレプリケーションの拡張プロパティ
 - アプリケーションリソース, 320, 323
 - レプリケーションリソース, 317
 - データレプリケーションのスイッチオーバー
 - アフィニティスイッチオーバー, 302
 - 実行, 333
 - データレプリケーションのためのスケーラブルアプリケーション, 304
 - データレプリケーションのフェイルオーバーアプリケーション
 - AffinityOn プロパティ, 302
 - GlobalDevicePaths, 302
 - ZPoolsSearchDir, 302
 - ガイドライン
 - テイクオーバーの管理, 306
 - リソースグループ, 303
 - 管理, 333
 - データレプリケーション用の共有アドレスリソースグループ, 305
 - デバイス
 - グローバル, 105
 - デバイスグループ
 - raw ディスク
 - 追加, 129
 - Solaris Volume Manager
 - 追加, 127
 - 管理の概要, 120
 - 構成の一覧表示, 142
 - 削除
 - と登録解除, 133
 - 追加, 129
 - データレプリケーションの構成, 310
 - プライマリ所有権, 138
 - プロパティの変更, 138
 - 保守状態, 144
 - デバイスグループのプライマリ所有権, 138
 - デバイスグループのプライマリノードの切り替え, 143

- 電源管理, 225
 - 同期データレプリケーション, 101, 299
 - 統合アーカイブ
 - クラスタノードの復元, 212
 - ゾーンクラスタのインストール, 256
 - ゾーンクラスタの構成, 255
 - 動的再構成, 106, 106
 - クラスタインターコネクト, 191
 - 定足数デバイス, 167
 - パブリックネットワークインタフェース, 206
 - 登録解除
 - Solaris Volume Manager デバイスグループ, 133
 - トポロジ
 - Oracle Solaris Cluster Manager での使用, 294
 - ドメインネームシステム (DNS)
 - 更新のガイドライン, 306
 - データレプリケーションでの更新, 333
 - トラブルシューティング
 - Oracle Solaris Cluster Manager, 291
 - トランスポートケーブル、アダプタ、およびスイッチ, 202
 - ネットワークバインドアドレス, 293
 - トランスポートアダプタ, 追加, 38, 42
 - トランスポートアダプタ, 追加, 190, 194
 - トランスポートケーブル
 - 追加, 38, 42, 190, 194
 - 無効化, 198
 - 有効化, 197
 - トランスポートケーブルの無効化, 198
 - トランスポートケーブルの有効化, 197
 - トランスポートスイッチ, 追加, 38, 42
 - トランスポートスイッチ, 追加, 190, 194
- な**
- 名前空間
 - 移行, 124
 - グローバル, 105
 - ネットワークアドレス
 - ゾーンクラスタへの追加, 257
 - ネットワークバインドアドレス
 - 確認, 293
 - ネットワークファイルシステム (Network File System、NFS)
 - データレプリケーション用アプリケーションファイルシステム構成, 313
 - ネットワークファイルシステム (NFS)
 - プライマリクラスタ上のリソース, 321
 - ノード
 - ID 検索, 227
 - グローバルクラスタでの名前変更, 235, 235
 - 削除
 - エラーメッセージ, 223
 - グローバルクラスタからの, 219
 - ゾーンクラスタから, 217
 - デバイスグループから, 133
 - セカンダリ, 138
 - 接続, 31
 - 追加, 209
 - 停止, 82
 - 認証, 228
 - ブート, 82
 - 負荷制限の構成, 252
 - プライマリ, 106, 138
 - 保守状態にする, 237
- ノード名の変更
- グローバルクラスタでの, 235, 235
- は**
- バックアップ
 - クラスタ, 30, 283
 - ミラーのオンライン, 283
 - パッケージ
 - アンインストール, 282
 - 特定のパッケージの更新, 278
 - パッチ, 279
 - 参照 更新
 - solaris10 ブランドゾーンクラスタ
 - clzonecluster による, 279
 - patchadd による, 280
 - 代替 BE, 281
 - パブリックネットワーク
 - 管理, 189, 204
 - 動的再構成, 206
 - 非稼働状態
 - 定足数デバイス, 178
 - 非クラスタモードでのブート, 93
 - ビットマップ
 - ポイントインタイムスナップショット, 299

- リモートミラーレプリケーション, 298
 - 非同期データレプリケーション, 101, 299
 - ファイル
 - /etc/vfstab, 56
 - md.conf, 127
 - md.tab, 30
 - ntp.conf.sc, 234
 - ファイルシステム
 - NFS アプリケーション
 - データレプリケーションの構成, 313
 - ゾーンクラスタでの削除, 254
 - ルートの復元
 - 説明, 286
 - ブート
 - グローバルクラスタ, 67
 - グローバルクラスタノード, 82
 - ゾーンクラスタ, 67
 - ゾーンクラスタノード, 82
 - 非クラスタモード, 93
 - 負荷制限
 - concentrate_load プロパティ, 251
 - preemption_mode プロパティ, 251
 - ノード上での構成, 251, 252
 - 復元
 - クラスタノード
 - scinstall の使用, 212
 - 統合アーカイブから, 212
 - クラスタファイル, 286
 - ルートファイルシステム, 286
 - プライベートホスト名
 - 変更, 232
 - プロパティ
 - failback, 138
 - numsecondaries, 140
 - preferenced, 138
 - プロファイル
 - RBAC の権利, 61
 - ベストプラクティス, 103
 - EMC SRDF, 103
 - ストレージベースのデータレプリケーション, 103
 - 変更
 - numsecondaries プロパティ, 140
 - SNMP イベント MIB プロトコル, 247
 - クラスタ名, 226
 - 定足数デバイスのノードリスト, 177
 - プライベートホスト名, 232
 - プライマリノード, 143
 - プロパティ, 138
 - ユーザー (RBAC), 64
 - ポイントインタイムスナップショット
 - 実行, 329
 - 定義, 299
 - 保守
 - 定足数デバイス, 178
 - 保守状態
 - 定足数デバイス, 178
 - 定足数デバイスを保守状態から戻す, 180
 - ノード, 237
 - ホスト
 - SNMP の追加と削除, 248, 249
 - ホストベースのデータレプリケーション
 - 定義, 98
 - 例, 297
 - ボリューム 参照 ストレージベースのレプリケーション
- ## ま
- マウントポイント
 - /etc/vfstab ファイルの変更, 153
 - グローバル, 56
 - マニフェスト
 - Automated Installer, 214
 - ミラー、オンラインバックアップ, 283
 - 命名規則
 - raw ディスクデバイス, 153
 - レプリケーションリソースグループ, 302
 - モニタリング
 - 共有ディスクパス, 163
 - ディスクパス, 158
 - モニタリング解除
 - ディスクパス, 159
- ## や
- 役割
 - カスタム役割を追加, 64
 - 設定, 61
 - 役割の追加, 63
 - 役割に基づくアクセス制御 参照 RBAC
 - ユーザー
 - SNMP の削除, 251

SNMP の追加, 250
プロパティの変更, 64

ら

リソース
構成情報の表示, 36
削除, 259
リソースグループ
データレプリケーション
構成, 302
構成のガイドライン, 301
フェイルオーバーでの役割, 302
リブート
グローバルクラスタ, 75
グローバルクラスタノード, 90
ゾーンクラスタ, 75
ゾーンクラスタノード, 90
リモートのミラー化 参照 ストレージベースのレプリケーション
リモートミラーレプリケーション
実行, 327
定義, 298
リモートレプリケーション 参照 ストレージベースのレプリケーション
リモートログイン, 31
リリース情報, 33
ループバックマウント
ゾーンクラスタへのファイルシステムのエクスポート, 260
例
インタラクティブ妥当性検査のリスト, 54
機能妥当性検査の実行, 54
レプリケーション 参照 データレプリケーション
レプリケーション、ストレージベース, 100
ローカルのミラー化 参照 ストレージベースのレプリケーション
ログイン
リモート, 31
論理ホスト名リソース
データレプリケーションテイクオーバーでの役割, 302

わ

耐障害性

定義, 298

A

AffinityOn プロパティ
データレプリケーションの拡張プロパティ, 302
Automated Installer
マニフェスト, 214
Availability Suite
データレプリケーションの使用, 297

B

boot コマンド, 71
BUI 参照 Oracle Solaris Cluster Manager

C

cconsole コマンド 参照 pconsole ユーティリティ
claccess コマンド, 27
cldevice コマンド, 27
cldevicegroup コマンド, 27
clinterconnect コマンド, 27
clnasdevice コマンド, 27
clnode コマンド, 251, 252
clquorum コマンド, 27
clreslogicalhostname コマンド, 27
clresource コマンド, 27
リソースグループの削除, 259
リソースとリソースグループの削除, 259
clresourcegroup コマンド, 27, 252
clresourcetype コマンド, 27
clressharedaddress コマンド, 27
clsetup, 27, 27, 32
ゾーンクラスタの管理, 254
ゾーンクラスタの作成, 25
ゾーンクラスタへのネットワークアドレスの追加, 257
定数デバイスの管理, 165
デバイスグループの管理, 120
トランスポートスイッチの管理, 190
clsetup ユーティリティ
ゾーンクラスタの作成, 23
clsnmphot コマンド, 27

clsnmpmib コマンド, 27
 clsnmpuser コマンド, 27
 cltelemattribute コマンド, 27
 cluster check
 コマンド, 50
 cluster check コマンド, 27
 vfstab ファイルのチェック, 154
 cluster shutdown コマンド, 67
 clzonecluster コマンド, 27
 solaris10 ブランドゾーンクラスタへのパッチの適
 用, 279
 説明, 32
 停止, 67
 ブート, 71
 CPU
 構成, 272
 CPU シェア
 グローバルクラスタノード, 272
 構成, 271
 制御, 271

D

DID 情報
 手動更新, 161
 DID 情報の手動更新, 161
 DLPI, 194

E

/etc/vfstab ファイル, 56
 構成の確認, 154
 マウントポイントの追加, 153
 EMC SRDF
 DID デバイスの構成, 110
 管理, 108
 キャンパスクラスタのプライマリルームが完全にフェ
 イルオーバーしたあとの回復, 117
 構成の確認, 112
 構成例, 113
 制限, 102
 ドミノモード, 101
 ベストプラクティス, 103
 要件, 102
 レプリケーショングループの構成, 108

F

failback プロパティ, 138
 fence_level 参照 レプリケーション中

G

GlobalDevicePaths
 データレプリケーションの拡張プロパティ, 302
 GUI 参照 Oracle Solaris Cluster Manager

I

IPMP
 管理, 204
 ステータスの確認, 41
 IPMP グループからのネットワークインタフェースの
 unplumb, 205
 iSCSI ストレージ
 定足数デバイスとして使用される
 プローブベースの IPMP, 204
 リンクベースの IPMP, 204

K

/kernel/drv/
 md.conf ファイル, 127

L

labeled ブランドゾーンクラスタ, 23
 lofi ファイル
 アンインストール, 244

M

md.tab ファイル, 30
 metaset コマンド, 105
 MIB
 SNMP イベントの有効化と無効化, 246, 246
 SNMP イベントプロトコルの変更, 247

N

ntp.conf.sc ファイル, 234

numsecondaries プロパティ, 140

O

OpenBoot PROM (OBP), 232

Oracle GlassFish

更新の制限, 289

Oracle Solaris Cluster Manager, 26, 289

アクセシビリティ制御の設定, 293

アクセス, 290

更新の制限, 289

実行できるタスク

Oracle ZFS Storage Appliance NAS デバイスの追加, 170

共有ストレージの追加, 226

クラスタインターコネクトのステータスのチェック, 192

クラスタインターコネクトの有効化, 203

クラスタ構成の表示, 41

クラスタ構成ユーティリティへのアクセス, 32

クラスタコンポーネントステータスのチェック, 38

グローバルクラスタノードの停止, 83

グローバルクラスタノードの負荷制限の作成, 252

ケーブルの削除, 195

ケーブルの追加, 193

ケーブルを無効にする, 198

ケーブルを有効にする, 197

ゾーンクラスタからのストレージデバイスの削除, 263

ゾーンクラスタからのファイルシステムの削除, 261

ゾーンクラスタノードからソフトウェアをアンインストールする, 25

ゾーンクラスタノード上の負荷制限を作成する, 252

ゾーンクラスタノードの停止, 83, 218

ゾーンクラスタノードのブート, 86

ゾーンクラスタノードのリブート, 90

ゾーンクラスタの削除, 259

ゾーンクラスタの作成, 25, 254

ゾーンクラスタのシャットダウン, 25

ゾーンクラスタの「リソースセキュリティ」プロパティの編集, 254

ゾーンクラスタへの共有ストレージの追加, 210

ゾーンクラスタへのストレージデバイスの追加, 254

ゾーンクラスタへのネットワークアドレスの追加, 257

ゾーンクラスタへのファイルシステムの追加, 254

ゾーン構成の表示, 25

ディスクパスのモニタリングを無効にする, 160

ディスクパスのモニタリングを有効にする, 158

定足数サーバーの作成, 171

定足数情報の表示, 182

定足数デバイスの削除, 174

定足数デバイスの作成, 168

定足数デバイスを無効にする, 179

定足数デバイスを有効にする, 180

定足数デバイスをリセットする, 179

デバイスグループをオフラインにする, 120, 145

デバイスグループをオンラインにする, 120, 143

トランスポートアダプタの削除, 195

トランスポートアダプタの追加, 193

ノードシステムメッセージの表示, 31, 32

ノードのステータスの表示, 41

ノードの退避, 68

ノードプロパティの編集, 164

ノードを退避する, 237

ノードを保守状態にする, 237

ファイルシステムの削除, 155

ファイルシステムの追加, 152

プライベートアダプタの削除, 195

プライベートアダプタの追加, 193

リソースとリソースグループの表示, 36

使用, 289

トポロジ, 289

「トポロジ」ビュー, 294

トラブルシューティング, 289

ログオンする方法, 289

Oracle Solaris Cluster 定足数サーバー

定足数デバイスとしてサポート, 167

Oracle Solaris OS

CPU 制御, 271

svcadm コマンド, 232

グローバルクラスタの管理タスク, 24

グローバルクラスタの定義, 23

ストレージベースのレプリケーション, 98

ゾーンクラスタ定義, 23

特別な指示

ノードのブート, 86
 ノードのリブート, 90
 ホストベースのレプリケーション, 98
 Oracle ZFS Storage Appliance
 定足数デバイスの追加, 170
 Oracle ZFS ストレージアプライアンス
 定足数デバイスとしてサポート, 167

P

patchadd
 solaris10 ブランドゾーンクラスタへのパッチの適用, 280
 pconsole ユーティリティ, 31
 使用, 215
 セキュア接続, 32

R

raw ディスクデバイス
 命名規則, 153
 raw ディスクデバイスグループ
 追加, 129
 RBAC, 61
 権利プロファイル (説明), 61
 タスク
 カスタム役割を追加, 64
 使用, 61
 設定, 61
 役割の追加, 63
 ユーザーの変更, 64

S

SATA ストレージ, 169
 定足数デバイスとしてサポート, 167
 scinstall コマンド
 クラスタノードの復元, 212
 showrev -p コマンド, 33
 SNMP
 イベント MIB の有効化と無効化, 246, 246
 プロトコルの変更, 247
 ホストの無効化, 249
 ホストの有効化, 248

ユーザーの削除, 251
 ユーザーの追加, 250
 SNMP イベント MIB の有効化と無効化, 246, 246
 solaris10 ブランドゾーンクラスタ, 23
 clzonecluster によるパッチの適用, 279
 patchadd によるパッチの適用, 280
 代替 BE のパッチの適用, 281
 Solaris OS 参照 Oracle Solaris OS
 Solaris Volume Manager
 raw ディスクデバイス名, 153
 solaris ブランドゾーンクラスタ, 23
 更新, 279
 SRDF 参照 EMC SRDF
 ssh, 32

V

/var/adm/messages ファイル, 96
 vfstab ファイル
 構成の確認, 154
 マウントポイントの追加, 153

Z

ZFS Storage Appliance 参照 定足数デバイス
 ZFS
 構成
 HAStoragePlus を使用しないローカルストレージプール, 131
 レプリケートされたデバイスグループ, 130
 追加
 HAStoragePlus を使用しないストレージプール, 131
 レプリケートされたデバイスグループ, 130
 ファイルシステムの削除, 260
 ルートファイルシステムの制限, 120
 レプリケーション, 130
 zpool
 HAStoragePlus を使用しないローカルの構成, 130, 131
 ZPoolsSearchDir
 データレプリケーションの拡張プロパティ, 302

