

# Oracle® Solaris Cluster システム管理

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、オラクル社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.

このソフトウェアもしく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。このソフトウェアもしくはハードウェアは、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。このソフトウェアもしく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際、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。このソフトウェアもしく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、オラクル社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Oracle Corporationおよびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

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。オラクル社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。オラクル社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。

# 目次

---

はじめに .....	11
<b>1 Oracle Solaris Cluster の管理の概要 .....</b>	<b>15</b>
Oracle Solaris Cluster の管理の概要 .....	16
ゾーンクラスタに関する作業 .....	16
Oracle Solaris OS の機能制限 .....	17
管理ツール .....	18
コマンド行インタフェース .....	18
クラスタ管理の準備 .....	20
Oracle Solaris Cluster ハードウェア構成の記録 .....	20
管理コンソールの使用 .....	20
クラスタのバックアップ .....	20
クラスタ管理の開始 .....	21
リモートからクラスタにログインする .....	23
クラスタコンソールに安全に接続する方法 .....	23
▼ クラスタ構成ユーティリティにアクセスする方法 .....	23
▼ 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 .....	24
▼ 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 .....	26
▼ クラスタコンポーネントのステータスを確認する方法 .....	27
▼ パブリックネットワークのステータスを確認する方法 .....	30
▼ クラスタ構成を表示する方法 .....	30
▼ 基本的なクラスタ構成を検証する方法 .....	39
▼ グローバルマウントポイントを確認する方法 .....	44
▼ Oracle Solaris Cluster のコマンドログの内容を表示する方法 .....	46
<b>2 Oracle Solaris Cluster と RBAC .....</b>	<b>49</b>
RBAC の設定と Oracle Solaris Cluster での使用 .....	49

Oracle Solaris Cluster RBAC の権限プロファイル .....	50
Oracle Solaris Cluster 管理権限プロファイルによる RBAC 役割の作成と割り当て .....	51
▼ コマンド行から役割を作成する方法 .....	51
ユーザーの RBAC プロパティの変更 .....	53
▼ 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 .....	53
<b>3 クラスタの停止とブート .....</b>	<b>55</b>
クラスタの停止とブートの概要 .....	55
▼ クラスタを停止する方法 .....	57
▼ クラスタをブートする方法 .....	59
▼ クラスタをリブートする方法 .....	61
クラスタ内の1つのノードの停止とブート .....	65
▼ ノードを停止する方法 .....	66
▼ ノードをブートする方法 .....	69
▼ ノードをリブートする方法 .....	71
▼ 非クラスタモードでノードをブートする方法 .....	74
満杯の /var ファイルシステムを修復する .....	77
▼ 満杯の /var ファイルシステムを修復する方法 .....	77
<b>4 データ複製のアプローチ .....</b>	<b>79</b>
データ複製についての理解 .....	79
サポートされるデータ複製方式 .....	80
<b>5 グローバルデバイス、ディスクパス監視、およびクラスタファイルシステムの管理 .....</b>	<b>81</b>
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 .....	81
Solaris ボリュームマネージャーのグローバルデバイスのアクセス権 .....	82
グローバルデバイスでの動的再構成 .....	82
クラスタファイルシステムの管理の概要 .....	83
クラスタファイルシステムの制限事項 .....	83
デバイスグループの管理 .....	84
▼ グローバルデバイス名前空間を更新する方法 .....	86
▼ 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 .....	87
グローバルデバイス名前空間を移行する .....	88

▼ 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 .....	88
▼ 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 .....	89
デバイスグループを追加および登録する .....	91
▼ デバイスグループを追加および登録する方法 (Solaris ボリュームマネージャー) .....	91
▼ 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 .....	93
▼ 複製デバイスグループ (ZFS) の追加と登録方法 .....	94
デバイスグループ名を保守する .....	95
デバイスグループを削除して登録を解除する方法 (Solaris ボリュームマネージャー) .....	96
▼ すべてのデバイスグループからノードを削除する方法 .....	96
▼ デバイスグループからノードを削除する方法 (Solaris ボリュームマネージャー) .....	97
▼ 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 .....	99
▼ デバイスグループのプロパティを変更する方法 .....	101
▼ 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 .....	102
▼ デバイスグループ構成の一覧を表示する方法 .....	105
▼ デバイスグループのプライマリノードを切り替える .....	106
▼ デバイスグループを保守状態にする方法 .....	107
ストレージデバイス用の SCSI プロトコル設定の管理 .....	109
▼ 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 .....	110
▼ 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 .....	111
▼ 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 .....	111
▼ 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 .....	113
クラスタファイルシステムの管理 .....	115
▼ クラスタファイルシステムを追加する方法 .....	115
▼ クラスタファイルシステムを削除する方法 .....	118
▼ クラスタ内のグローバルマウントを確認する方法 .....	120
ディスクパス監視の管理 .....	121
▼ ディスクパスを監視する方法 .....	122
▼ ディスクパスの監視を解除する方法 .....	123
▼ 障害のあるディスクパスを表示する方法 .....	124
▼ ディスクパスのステータスエラーを解決する方法 .....	124

▼ファイルからディスクパスを監視する方法 .....	125
▼監視しているすべての共有ディスクパスが失敗したときのノードの自動リブートを有効にする方法 .....	127
▼すべての監視共有ディスクパスが失敗した場合にノードの自動リブートを無効にする方法 .....	128
<b>6 定足数の管理 .....</b>	<b>129</b>
定足数デバイスの管理 .....	129
定足数デバイスへの動的再構成 .....	131
定足数デバイスの追加 .....	132
定足数デバイスの削除または交換 .....	140
定足数デバイスの保守 .....	144
定足数のデフォルトのタイムアウトの変更 .....	151
Oracle Solaris Cluster 定足数サーバーの管理 .....	151
Quorum Server Software の起動および停止 .....	152
▼定足数サーバーを起動する方法 .....	152
▼定足数サーバーを停止する方法 .....	153
定足数サーバーに関する情報の表示 .....	154
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 .....	155
<b>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 .....</b>	<b>159</b>
クラスタインターコネクトの管理 .....	160
クラスタインターコネクトでの動的再構成 .....	161
▼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 .....	162
▼クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを追加する方法 .....	163
▼クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを削除する方法 .....	165
▼クラスタトランスポートケーブルを有効にする方法 .....	168
▼クラスタトランスポートケーブルを無効にする方法 .....	169
▼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 .....	171
▼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 .....	172
パブリックネットワークの管理 .....	175
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 .....	175
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 .....	176

<b>8</b>	<b>ノードの追加と削除</b> .....	179
	クラスタへのノードの追加 .....	179
	▼既存のクラスタにノードを追加する方法 .....	180
	クラスタからのノードの削除 .....	182
	▼ゾーンクラスタからノードを削除する方法 .....	184
	▼クラスタソフトウェア構成からノードを削除する方法 .....	185
	▼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 .....	187
	▼エラーメッセージを修正する方法 .....	190
<b>9</b>	<b>クラスタの管理</b> .....	191
	クラスタの管理の概要 .....	191
	▼クラスタ名を変更する方法 .....	192
	▼ノードIDをノード名にマップする方法 .....	194
	▼新しいクラスタノード認証で作業する方法 .....	194
	▼クラスタの時刻をリセットする方法 .....	196
	▼SPARC: ノードでOpenBoot PROM (OBP) を表示する方法 .....	198
	▼ノードのプライベートホスト名を変更する .....	199
	▼ノード名を変更する .....	202
	▼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 .....	203
	▼ノードを保守状態にする .....	204
	▼ノードを保守状態から戻す .....	206
	▼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 .....	208
	ノードのアンインストールのトラブルシューティング .....	211
	Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理 .....	212
	負荷制限の設定 .....	218
	サービスまたは管理エージェントのポート番号の変更 .....	220
	ゾーンクラスタ管理タスクの実行 .....	222
	▼ゾーンクラスタを削除する .....	223
	▼ゾーンクラスタからファイルシステムを削除する .....	224
	▼ゾーンクラスタからストレージデバイスを削除する .....	226
	トラブルシューティング .....	228
	グローバルクラスタ外でのアプリケーションの実行 .....	228
	破損したディスクセットの復元 .....	230

<b>10</b>	<b>CPU 使用率の制御の構成</b> .....	233
	CPU 制御の概要 .....	233
	シナリオの選択 .....	233
	公平配分スケジューラ .....	234
	CPU 制御の構成 .....	234
	▼ グローバルクラスタの投票ノードで CPU 使用率を制御する方法 .....	234
<b>11</b>	<b>ソフトウェアの更新</b> .....	237
	Oracle Solaris Cluster ソフトウェアの更新の概要 .....	237
	Oracle Solaris Cluster ソフトウェアの更新 .....	238
	新しいリリースへのクラスタのアップグレード .....	238
	特定のパッケージの更新 .....	239
	定足数サーバーまたは AI インストールサーバーの更新 .....	240
	パッケージのアンインストール .....	241
	▼ パッケージのアンインストール .....	241
	▼ 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 .....	241
	更新に関する注意事項 .....	242
<b>12</b>	<b>クラスタのバックアップと復元</b> .....	243
	クラスタのバックアップ .....	243
	▼ ミラーのオンラインバックアップを実行する方法 (Solaris ボリュームマネージャー) .....	243
	▼ クラスタ構成をバックアップする方法 .....	245
	クラスタファイルの復元 .....	246
	▼ ZFS ルート (/) ファイルシステムを復元する方法 (Solaris ボリュームマネージャー) .....	246
<b>A</b>	<b>例</b> .....	251
	StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成 .....	251
	クラスタにおける StorageTek Availability Suite ソフトウェアの理解 .....	252
	クラスタ間でホストベースのデータ複製を構成するためのガイドライン .....	255
	タスクマップ: データ複製の構成例 .....	261
	クラスタの接続とインストール .....	262



---

デバイスグループとリソースグループの構成例 .....	264
データ複製の有効化例 .....	278
データ複製の実行例 .....	281
テイクオーバーの管理の例 .....	286
索引 .....	289



# はじめに

---

『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。

---

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

---

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

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

Bash は、Oracle Solaris 11 のデフォルトのシェルです。Bash シェルのプロンプトに示されているマシン名は、意味を明確にするために表示されています。

## UNIX コマンド

このドキュメントでは、Oracle Solaris Cluster データサービスのインストールと構成に固有のコマンドについて説明します。このドキュメントでは、UNIX の基本的なコマンドや手順(システムの停止、システムのブート、デバイスの構成など)については説明していません。基本的な UNIX コマンドに関する情報および手順については、以下を参照してください。

- Oracle Solaris オペレーティングシステムのオンラインドキュメント
- Oracle Solaris オペレーティングシステムのマニュアルページ
- システムに付属するその他のソフトウェアドキュメント

## 表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表 P-1 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	.login ファイルを編集します。  ls -a を使用してすべてのファイルを表示します。  system%
<b>AaBbCc123</b>	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	system% <b>su</b>  password:
AaBbCc123	変数を示します。実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、rm <i>filename</i> と入力します。
『 』	参照する書名を示します。	『コードマネージャ・ユーザーズガイド』を参照してください。
「 」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第 5 章「衝突の回避」を参照してください。  この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	sun% grep '^#define \  XV_VERSION_STRING'

Oracle Solaris OS に含まれるシェルで使用する、UNIX のデフォルトのシステムプロンプトとスーパーユーザープロンプトを次に示します。コマンド例に示されるデフォルトのシステムプロンプトは、Oracle Solaris のリリースによって異なります。

- C シェル

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

- C シェルのスーパーユーザー

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

- Bash シェル、Korn シェル、および Bourne シェル

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

- Bash シェル、Korn シェル、および Bourne シェルのスーパーユーザー

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

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

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

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

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

## 関連ドキュメント

関連する Oracle Solaris Cluster トピックについての情報は、以下の表に示すドキュメントを参照してください。Oracle Solaris Cluster のすべてのドキュメントは、<http://www.oracle.com/technetwork/indexes/documentation/index.html> で入手可能です。

項目	ドキュメント
ハードウェアの設計と管理	『Oracle Solaris Cluster Hardware Administration Manual』 各ハードウェア管理ガイド
概念	『Oracle Solaris Cluster Concepts Guide』
ソフトウェアのインストール	『Oracle Solaris Cluster ソフトウェアのインストール』
データサービスのインストールと管理	『Oracle Solaris Cluster Data Services Planning and Administration Guide』 および個々のデータサービスガイド
データサービスの開発	『Oracle Solaris Cluster Data Services Developer's Guide』
システム管理	『Oracle Solaris Cluster システム管理』 『Oracle Solaris Cluster Quick Reference』
ソフトウェアアップグレード	『Oracle Solaris Cluster Upgrade Guide』
エラーメッセージ	『Oracle Solaris Cluster Error Messages Guide』
コマンドと関数のリファレンス	『Oracle Solaris Cluster Reference Manual』 『Oracle Solaris Cluster Data Services Reference Manual』 『Oracle Solaris Cluster Geographic Edition Reference Manual』 『Oracle Solaris Cluster Quorum Server Reference Manual』

## Oracle Support へのアクセス

Oracle のお客様は、My Oracle Support を通じて電子的なサポートにアクセスできます。詳細は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> (聴覚に障害をお持ちの場合は <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>) を参照してください。

## 問い合わせについて

Oracle Solaris Cluster をインストールまたは使用しているときに問題が発生した場合は、ご購入先に連絡し、次の情報をお伝えください。

- 名前と電子メールアドレス (利用している場合)
- 会社名、住所、および電話番号
- システムのモデル番号とシリアル番号
- オペレーティング環境のリリース番号 (例: Oracle Solaris 11)
- Oracle Solaris Cluster のバージョン番号 (例: Oracle Solaris Cluster 4.0)

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

コマンド	機能
<code>prtconf -v</code>	システムメモリのサイズと周辺デバイス情報を表示します
<code>psrinfo -v</code>	プロセッサの情報を表示する
<code>pkg list</code>	インストールされているパッケージを報告する
<code>prtdiag -v</code>	システム診断情報を表示する
<code>/usr/cluster/bin/clnode show-rev</code>	Oracle Solaris Cluster のリリースやパッケージのバージョンの情報を、ノードごとに表示します

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

# 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。

- 16 ページの「Oracle Solaris Cluster の管理の概要」
- 17 ページの「Oracle Solaris OS の機能制限」
- 18 ページの「管理ツール」
- 20 ページの「クラスタ管理の準備」
- 21 ページの「クラスタ管理の開始」

このマニュアル内の手順はすべて、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投票ノードからのみ構成されます。グローバルクラスタには、HA for Zones データサービスで構成された solaris ブランドの非大域ゾーン(ノードではなく高可用性コンテナ)をリソースとして含むこともできます。ゾーンクラスタには、グローバルクラスタが必須です。ゾーンクラスタに関する一般的な情報については、『[Oracle Solaris Cluster Concepts Guide](#)』を参照してください。

ゾーンクラスタは、solaris ブランドの1つ以上の投票ノードで構成されます。ゾーンクラスタのすべてのノードは、cluster 属性が設定されている solaris ブランドの非大域ゾーンとして構成されます。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。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投票ノードが動作している場合のみ動作します。あるマシンのグローバルクラスタ投票ノードで障害が発生すると、同じマシン上のすべてのゾーンクラスタノードも動作しなくなります。

# Oracle Solaris Cluster の管理の概要

Oracle Solaris Cluster の高可用性環境によって、重要なアプリケーションの可用性がエンドユーザーに対して保証されます。システム管理者の業務は、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、『Oracle Solaris Cluster 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」および『Oracle Solaris Cluster Concepts Guide』に記載されている計画情報に精通しておいてください。ゾーンクラスタの作成手順については、『Oracle Solaris Cluster ソフトウェアのインストール』の「ゾーンクラスタの構成」を参照してください。Oracle Solaris Cluster の管理は、次のタスクごとに各マニュアルにまとめられています。

- グローバルクラスタまたはゾーンクラスタを定期的に (または毎日) 管理および維持するための標準的なタスク。これらのタスクは、このマニュアルで説明されています。
- インストール、構成、属性の変更などのデータサービスタスク。これらのタスクは、『Oracle Solaris Cluster Data Services Planning and Administration Guide』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。

通常、Oracle Solaris Cluster の管理タスクはクラスタが動作しているときに行うことができます。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の投票ノードで行うべきです。クラスタ全体を停止する必要がある手順については、ダウンタイムのスケジュールを通常の業務時間外に設定してシステムへの影響を最小限に抑えてください。クラスタまたはクラスタノードを停止する予定があるときは、あらかじめユーザーに通知しておいてください。

## ゾーンクラスタに関する作業

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の投票ノードで `cluster` コマンドを使用すると、投票グローバルクラスタおよびすべてのゾーンクラスタに関するすべての情報が得られます。`cluster` コマンドをゾーンクラスタで使用すると、そのゾーンクラスタのみの情報が得られます。

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



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

次の表に、ゾーンクラスタで実行する主な管理タスクを示します。

- ゾーンクラスタの作成 - `clzonecluster configure` コマンドを使用して、新しいゾーンクラスタを作成します。手順については、『[Oracle Solaris Cluster ソフトウェアのインストール](#)』の「[ゾーンクラスタの構成](#)」を参照してください。
- ゾーンクラスタの起動と再起動 - [第3章「クラスタの停止とブート」](#)を参照してください。
- ゾーンクラスタへのノードの追加 - [第8章「ノードの追加と削除」](#)を参照してください。
- ゾーンクラスタからのノードの削除 - [184 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。
- ゾーンクラスタの構成の表示 - [30 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。
- ゾーンクラスタの構成の検証 - [39 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 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 1-1 Oracle Solaris Cluster サービス

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

表 1-1 Oracle Solaris Cluster サービス (続き)

Oracle Solaris Cluster サービス	FMRI
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

Oracle Solaris Cluster 構成で管理タスクを行うときは、コマンド行を使用できません。次のセクションでは、コマンド行ツールの概要を示します。

### 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、対話形式で実行できます。可能なかぎり、本書の管理手順は `clsetup` ユーティリティを使用します。

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

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

Oracle Solaris Cluster の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一覧に示します。詳細は、マニュアルページを参照してください。

<code>if_mpadm(1M)</code>	IP ネットワークマルチパスグループ内のあるアダプタから別のアダプタに IP アドレスを切り換えます。
<code>claccess(1CL)</code>	ノードを追加するために Oracle Solaris Cluster アクセスポリシーを管理します。
<code>cldevice(1CL)</code>	Oracle Solaris Cluster デバイスを管理します。

<code>cldevicegroup(1CL)</code>	Oracle Solaris Cluster デバイスグループを管理します。
<code>clinterconnect(1CL)</code>	Oracle Solaris Cluster インターコネクトを管理します。
<code>clnasdevice(1CL)</code>	Oracle Solaris Cluster 構成の NAS デバイスへのアクセスを管理します。
<code>clnode(1CL)</code>	Oracle Solaris Cluster ノードを管理します。
<code>clquorum(1CL)</code>	Oracle Solaris Cluster 定足数を管理します。
<code>clreslogicalhostname(1CL)</code>	論理ホスト名のために Oracle Solaris Cluster リソースを管理します。
<code>clresource(1CL)</code>	Oracle Solaris Cluster データサービスのリソースを管理します。
<code>clresourcegroup(1CL)</code>	Oracle Solaris Cluster データサービスのリソースを管理します。
<code>clresourcetype(1CL)</code>	Oracle Solaris Cluster データサービスのリソースを管理します。
<code>clressharedaddress(1CL)</code>	共有アドレスのために Oracle Solaris Cluster リソースを管理します。
<code>clsetup(1CL)</code>	Oracle Solaris Cluster 構成を対話形式で構成します。
<code>clsnmphost(1CL)</code>	Oracle Solaris Cluster SNMP ホストを管理します。
<code>clsnmpmib(1CL)</code>	Oracle Solaris Cluster SNMP MIB を管理します。
<code>clsnmpuser(1CL)</code>	Oracle Solaris Cluster SNMP ユーザーを管理します。
<code>cltelemetryattribute(1CL)</code>	システムリソース監視を構成します。
<code>cluster(1CL)</code>	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。
<code>clzonecluster(1CL)</code>	ゾーンクラスタの作成と変更を行います。

さらに、コマンドを使用して Oracle Solaris Cluster 構成のボリューム管理ソフトウェアを管理することもできます。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、サイトに固有なハードウェアの特徴は記録しておきます。クラスタを変更または更新するときには、このハードウェアの記録を参照することで管理作業を少なくすることができます。また、さまざまなクラスタ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、管理が簡単になります。

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

### 管理コンソールの使用

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。

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、専用コンソールを使用すると、次の利点が得られます。

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

### クラスタのバックアップ

ご使用のクラスタを定期的にバックアップしてください。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を記憶装置に保存していますが、これが定期的なバックアップの代わりになるとは考えないでください。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保護する機能を備えていません。したがって、データ損失に対する保護のために、バックアップ手順を用意しておいてください。

次の情報もバックアップしてください。

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

## クラスタ管理の開始

表 1-2 クラスタ管理の開始点を示します。

表 1-2 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 pconsole ユーティリティを使用して、リモートからクラスタにログインします。	23 ページの「リモートからクラスタにログインする」 23 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	clzonecluster コマンドまたは clsetup ユーティリティを使用します。	23 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	clnode コマンドに show-rev -v -node サブコマンドとオプションを付けて使用します。	24 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"> <li>■ clresource</li> <li>■ clresourcegroup</li> <li>■ clresourcetype</li> </ul>	26 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ステータスを確認します。	cluster コマンドに status サブコマンドを付けて使用します。	27 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」

表 1-2 Oracle Solaris Cluster 管理ツール (続き)

タスク	ツール	手順
パブリックネットワーク上の IP ネットワークマルチパスグループのステータス確認	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。  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。	30 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。	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。  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。	30 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。	39 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。	46 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 Administration: Common Tasks』の「Viewing System Messages」
Solaris Volume Manager のステータスの監視	<code>metastat</code> コマンドを使用します。	『Solaris ボリュームマネージャの管理』

## リモートからクラスタにログインする

コマンド行から Parallel Console Access (pconsole) ユーティリティーを使用して、リモートからクラスタにログインします。pconsole ユーティリティーは、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 ユーティリティーは、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ーは、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 ユーティリティーは、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す。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` のように指定できます。

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 ユーティリティー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 ユーティリティーは、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 ユーティリティーは、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ーは、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ーにアクセスする方法

`clsetup` ユーティリティーを使用すると、グローバルクラスタの定足数、リソースグループ、クラスタトランスポート、プライベートホスト名、デバイスグループ、新しいノードのオプションを対話形式で構成できます。`clzonecluster` ユーティリティーは、同様な構成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。

`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なメンバーノードで、スーパーユーザーになります。  
グローバルクラスタのノードから、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
```

メインメニューが表示されます。

- ゾーンクラスタの場合は、**clzonecluster** コマンドでユーティリティを起動します。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> ?
```

- 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 ソフトウェアのインストール](#)』の「[ゾーンクラスタの構成](#)」に記述されている手順を参照してください。

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

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

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

この手順では、長形式の 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 のリリース情報およびバージョン情報の表示:

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



このコマンドは、すべての 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 1-1 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 4.0 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
4.0

phys-schost#% clnode show-rev -v

Oracle Solaris Cluster 4.0 for Solaris 11 sparc
ha-cluster/data-service/apache                :4.0.0-0.21
ha-cluster/data-service/dhcp                  :4.0.0-0.21
ha-cluster/data-service/dns                   :4.0.0-0.21
ha-cluster/data-service/ha-l1dom              :4.0.0-0.21
ha-cluster/data-service/ha-zones              :4.0.0-0.21
ha-cluster/data-service/nfs                   :4.0.0-0.21
ha-cluster/data-service/oracle-database       :4.0.0-0.21
ha-cluster/data-service/tomcat                :4.0.0-0.21
ha-cluster/data-service/weblogic              :4.0.0-0.21
ha-cluster/developer/agent-builder            :4.0.0-0.21
ha-cluster/developer/api                     :4.0.0-0.21
ha-cluster/geo/geo-framework                 :4.0.0-0.21
ha-cluster/geo/manual                        :4.0.0-0.21
ha-cluster/geo/replication/availability-suite :4.0.0-0.21
ha-cluster/geo/replication/data-guard         :4.0.0-0.21
ha-cluster/geo/replication/sbp                :4.0.0-0.21
ha-cluster/geo/replication/srdf              :4.0.0-0.21
ha-cluster/group-package/ha-cluster-data-services-full :4.0.0-0.21
ha-cluster/group-package/ha-cluster-framework-full :4.0.0-0.21
ha-cluster/group-package/ha-cluster-framework-l10n :4.0.0-0.21
ha-cluster/group-package/ha-cluster-framework-minimal :4.0.0-0.21
ha-cluster/group-package/ha-cluster-framework-scm :4.0.0-0.21
ha-cluster/group-package/ha-cluster-framework-slm :4.0.0-0.21
ha-cluster/group-package/ha-cluster-full     :4.0.0-0.21
ha-cluster/group-package/ha-cluster-geo-full :4.0.0-0.21
ha-cluster/group-package/ha-cluster-geo-incorporation :4.0.0-0.21
ha-cluster/group-package/ha-cluster-incorporation :4.0.0-0.21
ha-cluster/group-package/ha-cluster-minimal  :4.0.0-0.21
ha-cluster/group-package/ha-cluster-quorum-server-full :4.0.0-0.21
ha-cluster/group-package/ha-cluster-quorum-server-l10n :4.0.0-0.21
ha-cluster/ha-service/derby                  :4.0.0-0.21
ha-cluster/ha-service/gds                    :4.0.0-0.21
ha-cluster/ha-service/logical-hostname       :4.0.0-0.21
ha-cluster/ha-service/smf-proxy              :4.0.0-0.21
ha-cluster/ha-service/telemetry              :4.0.0-0.21
ha-cluster/library/cacao                     :4.0.0-0.21
ha-cluster/library/ucmm                      :4.0.0-0.21
ha-cluster/locale                            :4.0.0-0.21
ha-cluster/release/name                      :4.0.0-0.21
ha-cluster/service/management                :4.0.0-0.21
ha-cluster/service/management/slm           :4.0.0-0.21
ha-cluster/service/quorum-server             :4.0.0-0.21
```

```

ha-cluster/service/quorum-server/locale           :4.0.0-0.21
ha-cluster/service/quorum-server/manual/locale    :4.0.0-0.21
ha-cluster/storage/svm-mediator                   :4.0.0-0.21
ha-cluster/system/cfgchk                           :4.0.0-0.21
ha-cluster/system/core                             :4.0.0-0.21
ha-cluster/system/dsconfig-wizard                 :4.0.0-0.21
ha-cluster/system/install                         :4.0.0-0.21
ha-cluster/system/manual                           :4.0.0-0.21
ha-cluster/system/manual/data-services            :4.0.0-0.21
ha-cluster/system/manual/locale                   :4.0.0-0.21

```

## ▼ 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。

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

- クラスタで構成されているリソースタイプ、リソースグループ、リソースを表示します。

```
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 1-2 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

```

## ▼ クラスタコンポーネントのステータスを確認する方法

`cluster status` コマンドはゾーンクラスタのステータスを表示します。

`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。

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

- クラスタコンポーネントの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 1-3 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:ngel                      phys-schost-4:ngel                      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-2   sczone-2   Online     Running
    
```

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

- クラスタコンポーネントの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 1-4 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` コマンドで戻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        Online  nge3     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。

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

- グローバルクラスタまたはゾーンクラスタの構成を表示します。

```
% 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の投票ノードから実行すると、そのクラスタに関する詳細な構成情報が表示されます。また、ゾーンクラスタが構成されていれば、ゾーンクラスタの情報も表示されます。

ゾーンクラスタのみに関する構成情報を表示するには、`clzonecluster show` コマンドも使用できます。ゾーンクラスタのプロパティには、ゾーンクラスタ名、IP タイプ、自動ブート、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 1-5 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
globalzonestores: 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.HAStoragePlus:4
Type_version:             4
Group:                    HA_RG
R_description:
Resource_project_name:    SCSLM_HA_RG
Enabled{phys-schost-1}:   True
Enabled{phys-schost-2}:   True
Monitored{phys-schost-1}: True
Monitored{phys-schost-2}: True

```

```

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

```

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

```

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

```

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

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

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

=== DID Device Instances ===

```
DID Device Name:              /dev/did/rdisk/d1
Full Device Path:             phys-schost-1:/dev/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 1-6 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
  zonename:                 /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]

--- 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促されます。機能検査を指定するには、-k functional 検査 id キーワードを使用します。機能検査は一度に1つだけ実行できます。

---

注-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 - 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(ICL) コマンドをグローバルクラスタの投票ノードで実行すると、ゾーンクラスタが正しく機能するために必要な構成を検証する一連のチェックが実行されます。すべてのチェックでエラーがなかった場合、clzonecluster verify はシェルプロンプトに戻ります(その場合は、ゾーンクラスタを安全にインストールできます)。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なメンバーノードで、スーパーユーザーになります。

```
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。

```
# 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。

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

`-k interactive` 実行するインタラクティブな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。

```
# 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。

```
# 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。

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

-k *functional* 実行する機能の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. ゾーンクラスタの構成を検証して、ゾーンクラスタがインストール可能かどうかを確認します。

```
phys-schost# clzonecluster verify zoneclustername
```

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 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 1-7 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` を詳細モードで `phys-schost-1` および `phys-schost-2` ノードに対して実行し、エラーが発見されなかった場合を示しています。

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

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

```
cluster check: phys-schost-2: Starting single-node checks.
cluster check: phys-schost-2: Single-node checks finished.
cluster check: Starting multi-node checks.
cluster check: Multi-node checks finished
#
```

### 例 1-8 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 1-9 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
'/usr/cluster/bin/clresourcegroup switch' on specified resource group
either to specified node or to all nodes in succession.
Version: 1.2
Revision Date: 12/10/10
```

*Take the cluster out of production*

#### # cluster check -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 <<<
```

```
...
```

```
Follow onscreen directions
```

### 例 1-10 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 -h phys-schost-1,
phys-schost-2 -o /var/cluster/logs/cluster_check/Dec5/
```

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

```

cluster check: Multi-node checks finished.
cluster check: One or more checks failed.
cluster check: The greatest severity of all check failures was 3 (HIGH).
cluster check: Reports are in /var/cluster/logs/cluster_check/<Dec5>.
#
# cat /var/cluster/logs/cluster_check/Dec5/cluster_check-results.suncluster.txt
...
=====
= ANALYSIS DETAILS =
=====
-----
CHECK ID : 3065
SEVERITY : HIGH
FAILURE : Global filesystem /etc/vfstab entries are not consistent across
all 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.
...
#

```

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

---

- 1 グローバルクラスタのアクティブなメンバーノードで、スーパーユーザーになります。

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
% su
```

- 2 グローバルクラスタ構成を検証します。

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

### 例 1-11 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 -h phys-schost-1,phys-schost-2 -o
/var/cluster/logs/cluster_check/Dec5/

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

...
=====
= ANALYSIS DETAILS =
=====
-----
CHECK ID : 3065
SEVERITY : HIGH
FAILURE : Global filesystem /etc/vfstab entries are not consistent across
all 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 been 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す。

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

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

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

---

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

---

- 実行されたコマンドの終了ステータス

---

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

---

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

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

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

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

## 例 1-12 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
```





## 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。次のトピックについて述べます。

- 49 ページの「RBAC の設定と Oracle Solaris Cluster での使用」
- 50 ページの「Oracle Solaris Cluster RBAC の権限プロファイル」
- 51 ページの「Oracle Solaris Cluster 管理権限プロファイルによる RBAC 役割の作成と割り当て」
- 53 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 を作成して、Oracle Solaris Cluster ソフトウェアで使用するための手順については、この章で後述します。

タスク	手順
RBAC の詳細を調べる	『Oracle Solaris Administration: Security Services』の第 8 章「Using Roles and Privileges (Overview)」
RBAC の設定、要素の管理、RBAC の使用など	『Oracle Solaris Administration: Security Services』の第 9 章「Using Role-Based Access Control (Tasks)」
RBAC の要素とツールの詳細を調べる	『Oracle Solaris Administration: Security Services』の第 10 章「Security Attributes in Oracle Solaris (Reference)」

## Oracle Solaris Cluster RBAC の権限プロファイル

コマンド行で実行する一部の 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 の権限プロファイルは、グローバルクラスタ内の両方の投票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変更する承認。

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 の権限プロファイルには 1 つ以上の RBAC の承認が含まれます。これらの権限プロファイルをユーザーまたは役割に割り当てることで、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限プロファイルを示します。

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

権限プロファイル	含まれる承認	役割に許可されたアクセス権
Oracle Solaris Cluster コマンド	なし。ただし、 <code>eid=0</code> を指定して実行される Oracle Solaris Cluster コマンドのリストが含まれます。	<p>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。</p> <ul style="list-style-type: none"> <li>■ <code>list</code></li> <li>■ <code>show</code></li> <li>■ <code>status</code></li> </ul> <p><code>scha_control</code></p> <p><code>scha_resource_get</code></p> <p><code>scha_resource_setstatus</code></p> <p><code>scha_resourcegroup_get</code></p> <p><code>scha_resourcetype_get</code></p>

権限プロファイル	含まれる承認	役割に許可されたアクセス権
基本 Oracle Solaris ユーザー	この既存の Oracle Solaris 権限プロファイルには、Oracle Solaris の承認のほか、次のものが含まれます。	
	<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の実行。
Cluster Operation	この権限プロファイルは 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の実行。
	<code>solaris.cluster.admin</code>	クラスタオブジェクトの状態の変更。
システム管理者	この既存の Oracle Solaris 権限プロファイルには、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	この権限プロファイルには、Cluster Operation プロファイルに含まれるものと同じ承認のほか、以下の承認が含まれます。	Cluster Operation 役割 ID が実行できるのと同じオペレーションおよびクラスタオブジェクトのプロパティの変更を実行します。
	<code>solaris.cluster.modify</code>	

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

このタスクでは、Oracle Solaris Cluster 管理権限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 スーパーユーザーになるか、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
- 2 次のいずれかの役割の作成方法を選択します。
  - ローカルスコープの役割の場合は、`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、スーパーユーザー、またはその他の役割を作成できる役割による認証を必要とします。`roleadd` コマンドは、すべてのネームサービスに適用できます。

### 3 ネームサービスキャッシュデーモンを起動して停止します。

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

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

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

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

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

Authenticating as user: primaryadmin

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

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

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

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

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

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

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

```

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

```

## ユーザーの RBAC プロパティの変更

ユーザーアカウントツールかコマンド行のいずれかを使用すると、ユーザーの RBAC プロパティを変更できます。ユーザーの RBAC プロパティを変更する場合は、[53 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」](#)を参照してください。

### ▼ コマンド行からユーザーの RBAC プロパティを変更する方法

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 次のように適切なコマンドを選択します。
  - ローカルスコープまたは 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、スーパーユーザー、またはユーザーファイルを変更できる役割による認証を必要とします。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 Administration: Common Tasks](#)』の「[Command-Line Tools for User and Group Account Management](#)」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。



## 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。

- 55 ページの「クラスタの停止とブートの概要」
- 65 ページの「クラスタ内の 1 つのノードの停止とブート」
- 77 ページの「満杯の /var ファイルシステムを修復する」

この章の関連手順の詳細な説明については、74 ページの「非クラスタモードでノードをブートする方法」と表 3-2 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せん。また、アプリケーションエラーによってデータが破損した場合に、グローバルクラスタを停止するときにも使用できます。`clzonecluster halt` コマンドは、特定のノード上のゾーンクラスタ、または構成済みのすべてのノード上のゾーンクラスタ全体を停止します (ゾーンクラスタ内で `cluster shutdown` コマンドを使用することもできます)。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-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、57 ページの「[クラスタを停止する方法](#)」、65 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。

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

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

注-必要であれば、ノードを非クラスタモードで (つまり、ノードがクラスタメンバーシップを取得しないように) 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、74 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 3-1 タスクリスト:クラスタの停止とブート

タスク	手順
クラスタの停止。	57 ページの「 <a href="#">クラスタを停止する方法</a> 」
すべてのノードを起動してクラスタを起動クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクトとの動作中の接続が必要です。	59 ページの「 <a href="#">クラスタをブートする方法</a> 」
クラスタのリポート	61 ページの「 <a href="#">クラスタをリポートする方法</a> 」



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

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

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

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

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

- 特定のゾーンクラスタを停止します。

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

- すべてのゾーンクラスタを停止します。

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

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

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

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

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

### 例 3-1 ゾーンクラスタの停止

次の例では、*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 3-2 SPARC: グローバルクラスタの停止

次に、正常なグローバルクラスタの動作を停止して、すべてのノードを停止し、**ok** プロンプトが表示されたときのコンソールの出力例を示します。ここでは、**-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.
System services are now being stopped.
/etc/rc0.d/K05initrgm: Calling clnode evacuate
The system is down.
syncing file systems... done
Program terminated
ok
```

### 例 3-3 x86: グローバルクラスタの停止

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

```
phys-schost# cluster shutdown -g 0 -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、[59 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-ゾーンクラスタを作成するために、『[Oracle Solaris Cluster ソフトウェアのインストール](#)』の「[ゾーンクラスタの構成](#)」の手順を実行します。

---

- 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 ベースのブートについては、『[Booting and Shutting Down Oracle Solaris on x86 Platforms](#)』の「[Booting an x86 Based System Interactively](#)」を参照してください。

---

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

---

- ゾーンクラスタが1つの場合は、ゾーンクラスタ全体をブートできます。  

```
phys-schost# clzonecluster boot zoneclustername
```
- ゾーンクラスタが複数ある場合は、すべてのゾーンクラスタをブートできます。  

```
zoneclustername
```

の代わりに+を使用してください。

- 2 ノードが問題なくブートし、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ステータスが報告されます。

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

---

注-ノードの `/var` ファイルシステムが満杯になると、そのノード上では 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、[77 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 3-4 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.
...
```

## ▼ 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\(ICL\)](#)、[boot\(1M\)](#)、および [clzonecluster\(ICL\)](#) 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、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になります。  
グローバルクラスタのノードから、次の手順のステップをすべて実行します。

### 3 クラスタを停止します。

- グローバルクラスタを停止します。

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

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

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

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

---

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

---

### 4 各ノードをブートします。

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。

---

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

---

GRUB ベースのブートについては、『[Booting and Shutting Down Oracle Solaris on x86 Platforms](#)』の「[Booting an x86 Based System Interactively](#)」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 zoneclustername
```

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- 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、77 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3-5 ゾーンクラスタのリブート

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

phys-schost#
phys-schost# clzonecluster status

=== Zone Clusters ===

--- Zone Cluster Status ---

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

```

phys-schost#          schost-4   sczone-4             Online   Running

```

### 例 3-6 SPARC: グローバルクラスタのリブート

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:

```



## 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せん。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 3-2 タスクマップ: ノードの停止とブート

タスク	ツール	手順
ノードの停止。	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。 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。	<a href="#">66 ページの「ノードを停止する方法」</a>
ノードの起動。 クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクトとの動作中の接続が必要です。	グローバルクラスタノードの場合は、 <code>boot</code> または <code>b</code> コマンドを使用。 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。	<a href="#">69 ページの「ノードをブートする方法」</a>

表 3-2 タスクマップ: ノードの停止とブート (続き)

タスク	ツール	手順
クラスタ上のノードをいったん停止してから再起動。	グローバルクラスタノードの場合は、 <code>clnode evacuate</code> および <code>shutdown</code> コマンドを使用してから、 <code>boot</code> または <code>b</code> を使用。	71 ページの「ノードをリブートする方法」
クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネクタとの動作中の接続が必要です。	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。	
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 (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。	74 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。

- 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、スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。  
グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 特定のゾーンクラスタメンバーを停止する場合は、手順4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 zoneclustername`  
特定のゾーンクラスタノードを指定すると、そのノードのみが停止します。halt コマンドは、デフォルトではすべてのノード上のゾーンクラスタを停止します。
- 4 すべてのリソースグループ、リソース、およびデバイスグループを、停止するノードから別のグローバルクラスタノードに切り替えます。  
停止するグローバルクラスタノードで、次のようにコマンドを入力します。clnode evacuate コマンドは、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を指定します。  
`phys-schost# shutdown -g0 -y -i0`  
SPARC ベースのシステムではグローバルクラスタノードが ok プロンプトを表示し、x86 ベースのシステムでは GRUB メニューで「Press any key to continue」というメッセージが表示されていることを確認します。
- 6 必要であればノードの電源を切ります。

### 例3-7 SPARC: グローバルクラスタノードの停止

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

```
phys-schost# clnode evacuate nodename
phys-schost# shutdown -g0 -y
Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
```

```

Notice: rgmd is being stopped.
Notice: rpc.pmfd is being stopped.
Notice: rpc.fed is being stopped.
umount: /global/.devices/node@1 busy
umount: /global/phys-schost-1 busy
The system is down.
syncing file systems... done
Program terminated
ok

```

### 例 3-8 x86: グローバルクラスタノードの停止

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

```

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

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

phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
/etc/rc0.d/K05initrgm: Calling 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 3-9 ゾーンクラスタノードの停止

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

```

phys-schost# clzonecluster status

=== Zone Clusters ===

```

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

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

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

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

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

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

```
phys-schost#
```

参照 停止したグローバルクラスタノードを再起動するには、[69 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

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ポートする場合は、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、停止またはリポ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。

---

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

---

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

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

---

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

グローバルクラスタのノードから、次の手順のステップをすべて実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

GRUB メニューが表示された時点で、適切な 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 zoneclustername
```

- 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-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、77 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 3-10 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:
```

**▼ ノードをリブートする方法**

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げません。

`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。



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す。 `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。

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

- 3 **clnode evacuate** および **shutdown** コマンドを使用して、グローバルクラスタノードを停止します。

グローバルクラスタのノード上で実行する **clzonecluster halt** コマンドで、ゾーンクラスタを停止します。(clnode evacuate コマンドと shutdown コマンドもゾーンクラスタ内で動作します)。

グローバルクラスタの場合は、停止するノードで次のコマンドを入力します。clnode evacuate コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定のノード上の大域ゾーンから、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 - 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、ノードを個別にリブートするには、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、ノードを個別にリブートするには、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、リブ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 zoneclustername
```



---

注-クラスタメンバーシップを取得できるように、ノードにはクラスタインターコネク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 3-11 SPARC: グローバルクラスタノードのリブート

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

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

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

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

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

### 例 3-12 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ーシップに参加しません)。非クラスタモードは、クラスタソフトウェアをインストールしたり、ノード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上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。  
グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 2 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、指定の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 zoneclustername
```

  
ゾーンクラスタ内で `clnode evacuate` コマンドと `shutdown` コマンドを使用することもできます。
- 3 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 ベースのブートについては、『[Booting and Shutting Down Oracle Solaris on x86 Platforms](#)』の「[Booting an x86 Based System Interactively](#)」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-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 3-13 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。ここでは、**-g0** オプションで猶予期間をゼロに設定し、**-y** オプションで、確認プロンプトに対して自動的に `yes` と応答するよう指定し、**-i0** で実行レベル 0 (ゼロ) で起動します。このノードの停止メッセージは、グローバルクラスタ内のほかのノードのコンソールにも表示されます。

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

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

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

## 満杯の /var ファイルシステムを修復する

Oracle Solaris ソフトウェアおよび Oracle Solaris Cluster ソフトウェアは、両方ともエラーメッセージを /var/adm/messages ファイルに書き込むため、時間の経過に従って /var ファイルシステムが満杯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、次の手順で、満杯になったファイルシステムを整理してください。詳細は、『[Oracle Solaris Administration: Common Tasks](#)』の「[Viewing System Messages](#)」を参照してください。

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



# データ複製のアプローチ

---

この章では、Oracle Solaris Cluster ソフトウェアで利用できるデータ複製技術について説明します。データ複製は、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。データ複製を使用すると、クラスタの高可用性と耐障害性を確保できます。

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

- クラスタ間 - 障害回復には、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

データ複製を実行するには、複製するオブジェクトと同じ名前のデバイスグループが必要です。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、第5章の84ページの「[デバイスグループの管理](#)」を参照してください。

この章の内容は次のとおりです。

- [79 ページの「データ複製についての理解](#)」

## データ複製についての理解

Oracle Solaris Cluster 4.0 はホストベースのデータ複製をサポートします。

ホストベースのデータ複製は、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複製します。リモートミラー複製を使用すると、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複製できます。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(およびクラスタとクラスタの外にあるホストとの間)の複製に使用されるホストベースの複製ソフトウェアには、StorageTek Availability Suite などがあります。

ホストベースのデータ複製は、特別なストレージレイではなくホストリソースを使用するため、安価な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複製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。Oracle Solaris Cluster Geographic Edition を使用しないホストベースの複製の例については、付録 A の251 ページの「StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成」を参照してください。

## サポートされるデータ複製方式

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複製方式として以下をサポートしています。

1. クラスタ間の複製 - 障害回復の場合、クラスタ間のデータ複製を行うためにホストベースの複製を使用できます。Oracle Solaris Cluster Geographic Edition ソフトウェアで複製を管理できます。
  - ホストベースの複製
    - StorageTek Availability Suite

Oracle Solaris Cluster Geographic Edition ソフトウェアを使用しないでホストベースの複製を使用する場合は、付録 A 「例」の251 ページの「StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成」を参照してください。
2. クラスタ内の複製 - この方式は、ホストベースのミラー化の代替として使用されます。
3. アプリケーションベースの複製 - Oracle Data Guard はアプリケーションベースの複製ソフトウェアの例です。このタイプのソフトウェアは、障害回復で単一インスタンスまたは RAC データベースを複製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。



# グローバルデバイス、ディスクパス監視、およびクラスタファイルシステムの管理

---

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

- 81 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」
- 83 ページの「クラスタファイルシステムの管理の概要」
- 84 ページの「デバイスグループの管理」
- 109 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」
- 115 ページの「クラスタファイルシステムの管理」
- 121 ページの「ディスクパス監視の管理」

この章の関連手順の詳細は、表 5-2 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、ディスクパス監視、およびクラスタファイルシステムに関連する概念については、『[Oracle Solaris Cluster 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 ボリュームマネージャーは「クラスタ対応」であるため、Solaris ボリュームマネージャーの `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、今後クラスタに結合する可能性があるノードでもグローバルな名前空間を更新できます。

## Solaris ボリュームマネージャーのグローバルデバイスのアクセス権

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

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

## グローバルデバイスでの動的再構成

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

- Oracle Solaris Cluster の動的再構成 (DR) のサポートには、Oracle Solaris の DR 機能に述べられている必要条件、手順、および制限がすべて適用されます。ただし、オペレーティングシステムの休止操作は除きます。したがって、Oracle Solaris Cluster ソフトウェアで DR 機能を使用する前に、Oracle Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- Oracle Solaris Cluster は、プライマリノードのアクティブなデバイス上で DR 削除操作を実行できません。DR 操作を実行できるのは、プライマリノードのアクティブでないデバイスか、セカンダリノードの任意のデバイス上でだけです。
- DR 操作が終了すると、クラスタのデータアクセスが前と同じように続けられます。
- Oracle Solaris Cluster は、定足数デバイスの使用に影響を与える DR 操作を拒否します。詳細については、131 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-セカンダリノードに対してDR操作を行なっているときに現在のプライマリノードに障害が発生すると、クラスタの可用性が損なわれます。新しい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DR操作を実行するには、次の手順をその順番どおりに行います。

表 5-1 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. アクティブなデバイスグループに影響するようなDR操作を現在のプライマリノードに実行する必要がある場合、DR削除操作をデバイス上で実行する前に、プライマリノードとセカンダリノードの切替えを実行	106 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. 削除するデバイス上でDR削除操作を実行	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。

## 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投票ノードからのみ認識できません。

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

## クラスタファイルシステムの制限事項

次に、クラスタファイルシステム管理に適用される制限事項を示します。

- `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をマウントし直すとき、`directio` マウントオプションは指定できません。

## デバイスグループの管理

クラスタの要件の変化により、クラスタ上のデバイスグループの追加、削除、または変更が必要となる場合があります。Oracle Solaris Cluster には、このような変更を行うために使用できる、`clsetup` と呼ばれる対話型インタフェースがあります。`clsetup` は `cluster` コマンドを生成します。生成されるコマンドについては、各説明の後にある例を参照してください。次の表に、デバイスグループを管理するためのタスクを示し、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。



注意-ほかのノードが有効なクラスタメンバーであり、それらのノードの少なくとも1つがディスクセットを持つ場合は、クラスタの外側で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。

注-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。

表 5-2 タスクマップ: デバイスグループの管理

タスク	手順
<code>cldevice populate</code> 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	86 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 <code>lofi</code> デバイスのサイズを変更する	87 ページの「グローバルデバイス名前空間で使用する <code>lofi</code> 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	88 ページの「専用パーティションから <code>lofi</code> デバイスにグローバルデバイス名前空間を移行する方法」 89 ページの「 <code>lofi</code> デバイスから専用パーティションにグローバルデバイス名前空間を移行する方法」

表 5-2 タスクマップ: デバイスグループの管理 (続き)

タスク	手順
metaset コマンドを使用することにより、Solaris ポリウムマネージャー ディスクセットを追加し、それらをデバイスグループとして登録する	91 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 ポリウムマネージャー)」
cldevicegroup コマンドを使用することにより、raw ディスクデバイスグループを追加および登録する	93 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 コマンドを使用することにより、ZFS に名前付きデバイスグループを追加する	94 ページの「複製デバイスグループ (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 ポリウムマネージャー デバイスグループを削除する	96 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 ポリウムマネージャー)」
cldevicegroup、metaset、および clsetup コマンドを使用することにより、すべてのデバイスグループからノードを削除する	96 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 コマンドを使用することにより、Solaris ポリウムマネージャー デバイスグループからノードを削除する	97 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 ポリウムマネージャー)」
cldevicegroup コマンドを使用することにより、raw ディスクデバイスグループからノードを削除する	99 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	101 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	105 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	102 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	106 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」

表 5-2 タスクマップ: デバイスグループの管理 (続き)

タスク	手順
metaset コマンドまたは vxdg 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	107 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、`cldevice populate` コマンドを実行して手動でグローバルデバイス名前空間を更新します。

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

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 クラスタの各ノードで、`devfsadm` 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 各ノードで、ディスクセットを作成する前に、「`cldevice populate`」コマンドが完了していることを確認してください。

ノードの1つで `cldevice` コマンド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 5-1 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、スーパーユーザーになるか、RBAC の承認 **solaris.cluster.modify** を提供する役割になります。
- 2 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、[74 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\@'clinfo -n' > /dev/null 2>&1
```

*Ensure that the lofi device is detached*

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

*The command returns no output if the device is detached*

---

注 --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することができます。

---

注-ルートファイルシステムに ZFS がサポートされていますが、重要な例外が1つあります。グローバルデバイスファイルシステムにブートディスクの専用パーティションを使用する場合、ファイルシステムとして UFS のみを使用してください。グローバルデバイス名前空間には、UFS ファイルシステムで動作しているプロキシファイルシステム (PxFS) が必要です。ただし、`/var` または `/home` などの、グローバルデバイス名前空間の UFS ファイルシステムは、ルート (`/`) ファイルシステムやほかのルートファイルシステム用の ZFS ファイルシステムと共存できません。また、lofi デバイスを使用してグローバルデバイス名前空間をホストする場合、ルートファイルシステムに対する ZFS の使用に関する制限はありません。

---

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

- [88 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」](#)
- [89 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したいグローバルクラスタ投票ノードのスーパーユーザーになります。
- 2 ノード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、[74 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せん。

- 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。

```
# /usr/cluster/bin/cldevice populate
```

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

```
# ps -ef | grep 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したいグローバルクラスタ投票ノードのスーパーユーザーになります。

- 2 ノード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、74 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、使用するパーティションが `/dev/did/rdisk/d5s3` の場合、`/etc/vfstab` ファイルに追加する新しいエントリは、`/dev/did/dsk/d5s3 /dev/did/rdisk/d5s3 /global/.devices/node@3 ufs 2 no global` となります。
- 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。
 

```
# /usr/cluster/bin/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 ボリュームマネージャー)

metaset コマンドを使用して Solaris ボリュームマネージャー ディスクセットを作成し、そのディスクセットを 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、複製されたデバイスグループと同じ名前にしてください。

- 1 ディスクセットを作成するディスクに接続されたノードのいずれかで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

- 2 **Solaris** ボリュームマネージャーディスクセットを追加し、このディスクセットをデバイスグループとして **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 **-metaset** コマンドを実行して設定した、クラスタ上の Solaris Volume Manager デバイスグループは、そのデバイスグループに含まれるノード数に関わらず、デフォルトでセカンダリノードになります。デバイスグループが作成されたあと、**clsetup** ユーティリティを使用することで、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、[102 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。

---

- 3 複製されたデバイスグループを構成している場合は、そのデバイスグループの複製プロパティを設定します。

```
# cldevicegroup sync devicegroup
```

- 4 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
```

- 5 **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** の形式の完全な **DID** デバイス名を使用してください。

次の例では、**DID** デバイス **/dev/did/rdisk/d3** のエントリは、ドライブが **phys-schost-1** および **phys-schost-2** によって共有されていることを示しています。

```
=== DID Device Instances ===
DID Device Name:                    /dev/did/rdisk/d1
Full Device Path:                   phys-schost-1:/dev/rdisk/c0t0d0
DID Device Name:                    /dev/did/rdisk/d2
Full Device Path:                   phys-schost-1:/dev/rdisk/c0t6d0
DID Device Name:                    /dev/did/rdisk/d3
Full Device Path:                   phys-schost-1:/dev/rdisk/c1t1d0
Full Device Path:                   phys-schost-2:/dev/rdisk/c1t1d0
...
```

## 6 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 - ディスクセットにドライブを追加するときは、下位デバイス名 (cNtX dY) は使用しないでください。下位レベルデバイス名はローカル名であり、クラスタ全体で一意ではないため、この名前を使用するとディスクセットがスイッチオーバーできなくなる可能性があります。

---

## 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 5-2 Solaris ボリュームマネージャー デバイスグループの追加

次の例は、ディスクドライブ /dev/did/rdisk/d1 および /dev/did/rdisk/d2 を持つディスクセットおよびデバイスグループの作成を示し、デバイスグループが作成されたことを確認しています。

```
# metaset -s dg-schost-1 -a -h phys-schost-1
```

```
# cldevicegroup list
dg-schost-1
```

```
# metaset -s dg-schost-1 -a /dev/did/rdisk/d1 /dev/did/rdisk/d2
```

## ▼ デバイスグループ (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 ソフトウェアで使用できるように再構成します。

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

- 複数の DID をデバイスグループに追加したい
- デバイスグループの名前を変更する必要がある

- `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-複製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名 (Solaris Volume Manager、または raw ディスク) は複製したデバイスグループの名前と同じにする必要があります。

- 1 使用する各デバイスを特定し、事前に規定されたデバイスグループの構成を解除します。

次のコマンドは、d7 および d8 に対する定義済みのデバイスグループを除去します。

```
paris-1# cldevicegroup disable dsk/d7 dsk/d8
paris-1# cldevicegroup offline dsk/d7 dsk/d8
paris-1# cldevicegroup delete dsk/d7 dsk/d8
```

- 2 必要なデバイスを含む、新しい raw ディスクデバイスグループを作成します。

次のコマンドは、グローバルデバイスグループ `rawdg` を作成します。このデバイスグループに d7 および d8 が収められます。

```
paris-1# cldevicegroup create -n phys-paris-1,phys-paris-2 -t rawdisk
-d d7,d8 rawdg
paris-1# /usr/cluster/lib/dcs/cldg show rawdg -d d7 rawdg
paris-1# /usr/cluster/lib/dcs/cldg show rawdg -d d8 rawdg
```

## ▼ 複製デバイスグループ(ZFS)の追加と登録方法

ZFS を複製するには、名前付きデバイスグループを作成し、`zpool` に属するディスクをリストする必要があります。デバイスは、一度に1つのデバイスグループのみに属することができるため、デバイスを含む 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、複製されたデバイスグループと同じ名前にしてください。

- 1 `zpool` のデバイスに対応するデフォルトデバイスグループを削除してください。

たとえば、2つのデバイス `/dev/did/dsk/d2` と `/dev/did/dsk/d13` を含む `mypool` と呼ばれる `zpool` を持っている場合、`d2` と `d13` と呼ばれる2つのデフォルトデバイスグループを削除する必要があります。

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

- 2 **手順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複製したデバイス(デバイスグループ内)の移行を管理します。
 

```
# clrg create -n pnode1,pnode2 migrate_truecopydg-rg
```
- 5 **手順4**で作成したリソースグループに **hasp-rs** リソースを作成し、**globaldevicepaths** プロパティを raw ディスクのデバイスグループに設定します。このデバイスは、**手順2**で作成しました。
 

```
# clrs create -t HASStoragePlus -x globaldevicepaths=mypool -g \
migrate_truecopydg-rg hasp2migrate_mypool
```
- 6 このリソースグループから**手順4**で作成したリソースグループで、**rg\_affinities** プロパティの **+++** 値を設定します。
 

```
# clrg create -n pnode1:zone-1,pnode2:zone-2 -p \
RG_affinities=+++migrate_truecopydg-rg sybase-rg
```
- 7 **手順3**で作成した **zpool** の **HASStoragePlus** リソース (**hasp-rs**) を、**手順4**または**手順6**で作成したリソースグループに作成します。  
**resource\_dependencies** プロパティを**手順5**で作成した **hasp-rs** リソースに設定します。
 

```
# clrs create -g sybase-rg -t HASStoragePlus -p zpools=mypool \
-p resource_dependencies=hasp2migrate_mypool \
-p ZpoolsSearchDir=/dev/did/dsk hasp2import_mypool
```
- 8 デバイスグループ名が必要な場合には、この新しいリソースグループ名を使用します。

## デバイスグループ名を保守する

デバイスグループに対して様々な管理タスクを実行することができます。

## デバイスグループを削除して登録を解除する方法 (Solaris ボリュームマネージャー)

デバイスグループは Oracle Solaris Cluster に登録されている Solaris ボリュームマネージャー ディスクセットです。Solaris ボリュームマネージャー デバイスグループを削除するには、`metaclear` と `metaset` コマンドを使用します。これらのコマンドは、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方法については、Solaris ボリュームマネージャー のドキュメント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

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 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として削除するノード上で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 削除するノードがメンバーになっているデバイスグループ (複数可) を確認します。各デバイスグループの `Device group node list` からこのノード名を検索します。  

```
# cldevicegroup list -v
```
- 3 手順2 で特定したデバイスグループの中に、デバイスグループタイプが `SVM` のものがある場合、そのタイプの各デバイスグループに対して97 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 ボリュームマネージャー)」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 手順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、99 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 ボリュームマネージャー)

Solaris ボリュームマネージャー デバイスグループの潜在的な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、クラスタの外側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** ボリュームマネージャー デバイスグループであることを確認します。  
Solaris ボリュームマネージャー のデバイスグループは、デバイスグループタイプが **SDS/SVM** のものです。  
`phys-schost-1% cldevicegroup show devicegroup`
- 2 どのノードがデバイスグループの現在のプライマリノードであるかを特定します。  
`# cldevicegroup status devicegroup`
- 3 変更したいデバイスグループを所有しているノードでスーパーユーザーになります。

- 4 デバイスグループからこのノードのホスト名を削除します。

```
# metaset -s setname -d -h nodelist
```

-s setname            デバイスグループの名前を指定します。

-d                    -hで指定されたノードをデバイスグループから削除します。

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

---

注-更新が完了するまでに数分間かかることがあります。

---

コマンドが正常に動作しない場合は、コマンドに -f (force) オプションを追加します。

```
# metaset -s setname -d -f -h nodelist
```

- 5 潜在的なプライマリノードとしてノードを削除するデバイスグループごとに手順4を繰り返します。

- 6 デバイスグループからノードが削除されたことを確認します。

デバイスグループ名はmetasetに指定したディスクセット名と一致します。

```
phys-schost-1% cldevicegroup list -v devicegroup
```

### 例 5-3 デバイスグループからノードを削除する (Solaris ボリュームマネージャー)

次に、デバイスグループ構成からホスト名 phys-schost-2 を削除する例を示します。この例では、指定したデバイスグループから phys-schost-2 を潜在的なプライマリノードとして削除します。cldevicegroup show コマンドを実行することにより、ノードが削除されていることを確認します。削除したノードが画面に表示されていないことを確認します。

```
[Determine the Solaris ボリュームマネージャー
device group for the node:]
```

```
# cldevicegroup show dg-schost-1
```

```
=== Device Groups ===
```

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

```
[Determine which node is the current primary for the device group:]
```

```
# cldevicegroup status dg-schost-1
```

```
=== Cluster Device Groups ===
```

```

--- Device Group Status ---

Device Group Name   Primary           Secondary         Status
-----
dg-schost-1         phys-schost-1    phys-schost-2    Online
[Become superuser on the node that currently owns the device group.]
[Remove the host name from the device group:]
# metaset -s dg-schost-1 -d -h phys-schost-2
[Verify removal of the node:]
phys-schost-1% cldevicegroup list -v dg-schost-1
=== Cluster Device Groups ===

--- Device Group Status ---

Device Group Name   Primary           Secondary         Status
-----
dg-schost-1         phys-schost-1    -                 Online

```

## ▼ 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 クラスタ内のノード、ただし削除するノード以外のノード上で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。
- 2 削除されるノードに接続されたデバイスグループを特定し、どれが raw ディスクデバイスグループであるかを判別します。  

```
# cldevicegroup show -n nodename -t rawdisk +
```
- 3 すべての `Local_Diskraw` ディスクデバイスグループの `localonly` プロパティを無効にします。  

```
# cldevicegroup set -p localonly=false devicegroup
```

`localonly` プロパティについての詳細は、`cldevicegroup(1CL)` のマニュアルページを参照してください。

- 4 削除するノードに接続されているすべての raw ディスクデバイスグループの `localonly` プロパティが無効になっていることを確認します。  
 デバイスグループタイプ `Disk` は、この raw ディスクデバイスグループの `localonly` プロパティが無効になっていることを表します。

```
# cldevicegroup show -n nodename -t rawdisk -v +
```

- 5 手順2で特定されたすべての raw ディスクデバイスグループからノードを削除します。

この手順は、削除するノードに接続されている raw ディスクデバイスグループごとに行う必要があります。

```
# cldevicegroup remove-node -n nodename devicegroup
```

#### 例 5-4 raw デバイスグループからノードを削除する

この例では、raw ディスクデバイスグループからノード (`phys-schost-2`) を削除します。すべてのコマンドは、クラスタの別のノード (`phys-schost-1`) から実行します。

```
[Identify the device groups connected to the node being removed, and determine which are raw-disk device groups:]
```

```
phys-schost-1# cldevicegroup show -n phys-schost-2 -t rawdisk -v +
```

```
Device Group Name:      dsk/d4
Type:                   Disk
failback:               false
Node List:              phys-schost-2
preferenced:            false
localonly:              false
autogen:                true
numsecondaries:        1
device names:          phys-schost-2
```

```
Device Group Name:      dsk/d1
Type:                   SVM
failback:               false
Node List:              pbrave1, pbrave2
preferenced:            true
localonly:              false
autogen:                true
numsecondaries:        1
diskset name:          ms1
```

```
(dsk/d4) Device group node list: phys-schost-2
(dsk/d2) Device group node list: phys-schost-1, phys-schost-2
(dsk/d1) Device group node list: phys-schost-1, phys-schost-2
```

```
[Disable the localonly flag for each local disk on the node:]
```

```
phys-schost-1# cldevicegroup set -p localonly=false dsk/d4
```

```
[Verify that the localonly flag is disabled:]
```

```
phys-schost-1# cldevicegroup show -n phys-schost-2 -t rawdisk +
```

```
(dsk/d4) Device group type:      Disk
(dsk/d8) Device group type:      Local_Disk
```

```
[Remove the node from all raw-disk device groups:]
```

```
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d4
```

```
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d2
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d1
```

## ▼ デバイスグループのプロパティを変更する方法

デバイスグループの主所有権を確立する方法は、`preferenced` という所有権設定属性の設定に基づきます。この属性を設定していない場合は、ほかで所有されていないデバイスグループの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、5 を使用して、Solaris ボリュームマネージャー 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、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。
- 2 `clsetup` ユーティリティを起動します。  
`# clsetup`  
メインメニューが表示されます。
- 3 デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプション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** ボリュームマネージャー デバイスグループの重要なプロパティを変更するためのオプション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**、**failback**などのプロパティを変更する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 5-5 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様に下がります。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 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際のデバイスグループのオプションの詳細については、`cldevicegroup(1CL)` を参照してください。

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

この手順では、長形式の 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、スーパーユーザーになるか、RBACの承認 `solaris.cluster.read` および `solaris.cluster.modify` を提供する役割になります。
- 2 `clsetup` ユーティリティを起動します。  
`# clsetup`  
メインメニューが表示されます。
- 3 デバイスグループを使用して作業するには、「デバイスグループとボリューム (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。すると対応する `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-このような構成変更には、ボリュームの追加や削除、既存ボリュームのグループ、所有者、アクセス権の変更などがあります。構成変更後に登録を行うと、グローバルな名前空間が正しい状態になります。[86 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 5-6 セカンダリノードの希望数の変更 (Solaris ボリュームマネージャー)

次に、デバイスグループ (`dg-schost-1`) のセカンダリノードの希望数を構成するとき、`clsetup` によって生成される `cldevicegroup` コマンドの例を示します。この例では、ディスクグループとボリュームは以前に作成されているものと想定していません。

```
# cldevicegroup set -p numsecondaries=1 dg-schost-1
# cldevicegroup show -v dg-schost-1
```

```
=== Device Groups ===
```

```
Device Group Name:      dg-schost-1
Type:                   SVM
failback:               yes
Node List:              phys-schost-1, phys-schost-2
preferenced:            yes
numsecondaries:         1
diskset names:         dg-schost-1
```



## 例 5-7 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、スーパーユーザーになる必要はありません。ただし、`solaris.cluster.read` の権限は必要です。

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

この手順では、長形式の 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。

<code>cldevicegroup show</code>	<code>cldevicegroup show</code> を使用して、クラスタ内のすべてのデバイスグループの構成を一覧表示します。
<code>cldevicegroup show <i>devicegroup</i></code>	<code>cldevicegroup show <i>devicegroup</i></code> を使用して、1つのデバイスグループの構成を一覧表示します。
<code>cldevicegroup status <i>devicegroup</i></code>	<code>cldevicegroup status <i>devicegroup</i></code> を使用して、1つのデバイスグループのステータスを判別します。
<code>cldevicegroup status +</code>	<code>cldevicegroup status +</code> を使用して、クラスタ内のすべてのデバイスグループのステータスを判別します。

詳細情報を表示するには、上記のコマンドと `-v` オプションを使用します。

### 例 5-8 すべてのデバイスグループのステータスの一覧表示

```
# cldevicegroup status +

=== Cluster Device Groups ===

--- Device Group Status ---

Device Group Name      Primary      Secondary      Status
-----
dg-schost-1            phys-schost-2 phys-schost-1  Online
dg-schost-2            phys-schost-1 --              Offline
dg-schost-3            phys-schost-3 phy-shost-2    Online
```

### 例 5-9 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。

`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、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供するプロファイルを使用します。

- 2 `cldevicegroup switch` を使用して、デバイスグループの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 5-10 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ことによって、デバイスのいずれかにアクセスされたときに、デバイスグループが自動的にオンラインになることを防ぎます。デバイスグループを保守状態にするべきなのは、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またはディスクグループを修復していても、当該ノード上のデバイスグループはオンラインにならないため、データの損失を防ぎます。

破損したディスクセットを復元する方法については、[230 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-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。

- 1 デバイスグループを保守状態にします。
  - a. デバイスグループが有効である場合は、デバイスグループを無効にします。  

```
# cldevicegroup disable devicegroup
```
  - b. デバイスグループをオフラインにします。  

```
# cldevicegroup offline devicegroup
```
- 2 修理手順を実行するときに、ディスクセットまたはディスクグループの所有権が必要な場合は、ディスクセットまたはディスクグループを手動でインポートします。  
 Solaris ボリュームマネージャー の場合:  

```
# metaset -C take -f -s diskset
```



注意-Solaris ボリュームマネージャー ディスクセットの所有権を取得する場合、デバイスグループが保守状態にあるときは、metaset -C take コマンドを使用する必要があります。metaset -t を使用すると、所有権の取得作業の一部として、デバイスグループがオンラインになります。

---

- 3 必要な修理手順を実行します。
- 4 ディスクセットまたはディスクグループの所有権を解放します。



注意-デバイスグループを保守状態から戻す前に、ディスクセットまたはディスクグループの所有権を解放する必要があります。所有権を解放しないと、データが失われる可能性があります。

---

- Solaris ボリュームマネージャー の場合:  

```
# metaset -C release -s diskset
```

- 5 デバイスグループをオンラインにします。

```
# cldevicegroup online devicegroup
# cldevicegroup enable devicegroup
```

#### 例 5-11 デバイスグループを保守状態にする

次に、デバイスグループ `dg-schost-1` を保守状態にし、保守状態からデバイスグループを削除する方法の例を示します。

```
[Place the device group in maintenance state.]
# cldevicegroup disable dg-schost-1
# cldevicegroup offline dg-schost-1
[If needed, manually import the disk set or disk group.]
For Solaris ボリュームマネージャー:
# metaset -C take -f -s dg-schost-1

[Complete all necessary repair procedures.]
[Release ownership.]
For Solaris ボリュームマネージャー:
# metaset -C release -s dg-schost-1

[Bring the device group online.]
# cldevicegroup online dg-schost-1
# cldevicegroup enable dg-schost-1
```

## ストレージデバイス用の SCSI プロトコル設定の管理

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。

- 110 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」
- 111 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」
- 111 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」
- 113 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 スーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` を提供する役割になります。
- 2 任意のノードから、現在のグローバルなデフォルト SCSI プロトコル設定を表示します。

```
# cluster show -t global
```

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

### 例 5-12 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 スーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` を提供する役割になります。
- 2 任意のノードから、ストレージデバイスの SCSI プロトコル設定を表示します。

```
# cldevice show device
```

*device* デバイスパスの名前またはデバイス名。

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

### 例 5-13 単一デバイスの SCSI プロトコルの表示

次の例に、デバイス `/dev/rdsk/c4t8d0` の SCSI プロトコルを示します。

```
# cldevice show /dev/rdsk/c4t8d0
```

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

```
DID Device Name:                /dev/did/rdsk/d3
Full Device Path:                phappy1:/dev/rdsk/c4t8d0
Full Device Path:                phappy2:/dev/rdsk/c4t8d0
Replication:                     none
default_fencing:                 global
```

## ▼ すべてのストレージデバイスのデフォルトのグローバルな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、そのデバイスの単一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、113 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 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 定足数デバイスではないすべてのストレージデバイスのフェンシングプロトコルを設定します。

<code>cluster set -p global_fencing={pathcount   prefer3   nofencing   nofencing-noscrub}</code>	
<code>-p global_fencing</code>	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。
<code>prefer3</code>	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。
<code>pathcount</code>	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せん)。
nofencing-noscrub	ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスが消去され、クラスタの外側にあるシステムからストレージへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大な問題があるストレージデバイスに対してのみ使用してください。

#### 例 5-14 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- 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

---

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す。

- 1 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 ストレージデバイスのフェンシングプロトコルを設定します。

```
cldevice set -p default_fencing ={pathcount | \
scsi3 | global | nofencing | nofencing-noscrub} device
```

-p default\_fencing デバイスの default\_fencing プロパティを変更します。

pathcount 共有デバイスに接続されている DID パスの数でフェンシングプロトコルを決定します。

scsi3 SCSI-3 プロトコルを使用します。

大域 (global) グローバルなデフォルトのフェンシング設定を使用します。global 設定は、定足数デバイス以外のデバイスで使用されます。

指定された DID インスタンスのフェンシングステータスを設定することで、フェンシングをオフに設定します。

nofencing-noscrub ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスが消去され、クラスタの外側にあるシステムからストレージデバイスへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大な問題があるストレージデバイスに対してのみ使用してください。

device デバイスパスの名前またはデバイス名を指定します。

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

#### 例 5-15 単一デバイスのフェンシングプロトコルの設定

次の例では、(デバイス番号で指定される) デバイス d5 を SCSI-3 プロトコルに設定します。

```
# cldevice set -p default_fencing=prefer3 d5
```

次の例では、d11 デバイスのデフォルトフェンシングをオフに設定します。

```
#cldevice set -p default_fencing=nofencing d11
```

## クラスタファイルシステムの管理

クラスタファイルシステムは、クラスタのどのノードからでも読み取りやアクセスが可能なグローバルなファイルシステムです。

表 5-3 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
クラスタファイルシステムを Oracle Solaris Cluster の最初のインストール後に追加する	115 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	118 ページの「クラスタファイルシステムを削除する方法」
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	120 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上でスーパーユーザー特権が確立されていること。
- ボリュームマネージャーソフトウェアがクラスタ上にインストールおよび構成されていること。
- クラスタファイルシステムの作成先のデバイスグループ (Solaris ボリュームマネージャー デバイスグループなど) またはブロックディスクスライスが存在すること。

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は、クラスタファイルシステムがすでに自動的に作成されています (十分な共有ディスクが存在する場合)。

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

この手順では、長形式の 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 クラスタ内にある任意のノード上でスーパーユーザーになります。

---

ヒント-ファイルシステムを迅速に作成するには、ファイルシステムを作成するグローバルデバイスの現在のプライマリノードでスーパーユーザーになります。

---

- 2 **newfs** コマンドを使用して 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。

ボリューム管理ソフトウェア	ディスクデバイス名の例	説明
Solaris ボリュームマネージャー	/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

- 3 クラスタ内の各ノードで、クラスタファイルシステムのマウントポイントのディレクトリを作成します。

そのノードからはクラスタファイルシステムにアクセスしない場合でも、マウントポイントはノードごとに必要です。

---

ヒント-管理を行いやすくするには、マウントポイントを `/global/device-group/` ディレクトリに作成します。この場所を使用すると、グローバルに利用できるクラスタファイルシステムとローカルファイルシステムを区別しやすくなります。

---

phys-schost# **mkdir -p /global/device-group/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。

- 4 クラスタ内の各ノードで、マウントポイント用の `/etc/vfstab` ファイルにエントリを追加します。

詳細については、`vfstab(4)` のマニュアルページを参照してください。

- a. 各エントリで、使用する種類のファイルシステムに必要なマウントオプションを指定します。
- b. クラスタファイルシステムを自動的にマウントするには、`mount at boot` フィールドを `yes` に設定します。
- c. 各クラスタファイルシステムで、`/etc/vfstab` エントリの情報が各ノードで同じになるようにします。
- d. 各ノードの `/etc/vfstab` ファイルのエントリに、デバイスが同じ順序で表示されることを確認します。
- e. ファイルシステムの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上で、構成確認ユーティリティを実行します。

```
phys-schost# cluster check -k vfstab
```

構成確認ユーティリティは、マウントポイントが存在することを確認します。また、`/etc/vfstab` ファイルのエントリが、クラスタのすべてのノードで正しいことを確認します。エラーが発生していない場合は、何も出力されません。

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)` マニュアルページまたは `mount(1M)` マニュアルページを参照してください。

## 例 5-16 UFS クラスタファイルシステムの作成

次に、Solaris ボリュームマネージャー ボリューム /dev/md/oracle/rdsk/d1 上に、UFS クラスタファイルシステムを作成する例を示します。各ノードの `vfstab` ファイルにクラスタファイルシステムのエントリが追加されます。次に、1つのノードから `cluster check` コマンドを実行します。構成確認プロセスが正しく終了すると、1つのノードからクラスタファイルシステムがマウントされ、全ノードで確認されます。

```
phys-schost# newfs /dev/md/oracle/rdsk/d1
...
phys-schost# mkdir -p /global/oracle/d1
phys-schost# vi /etc/vfstab
#device          device          mount   FS      fsck    mount  mount
#to mount        to fsck         point  type   pass   at boot options
#
/dev/md/oracle/dsk/d1 /dev/md/oracle/rdsk/d1 /global/oracle/d1 ufs 2 yes global,logging
...
phys-schost# cluster check -k vfstab
phys-schost# mount /global/oracle/d1
phys-schost# mount
...
/global/oracle/d1 on /dev/md/oracle/dsk/d1 read/write/setuid/global/logging/largefiles
on Sun Oct 3 08:56:16 2005
```

## ▼ クラスタファイルシステムを削除する方法

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 - クラスタファイルシステムは、`cluster shutdown` を実行してクラスタ全体を停止したときに、システム停止処理の一環として自動的にマウント解除されません。`shutdown` を実行して単独でノードを停止したときはマウント解除されません。なお、停止するノードが、ディスクに接続されている唯一のノードの場合は、そのディスク上のクラスタファイルシステムにアクセスしようとするエラーが発生します。

---

クラスタファイルシステムをマウント解除する前に、次の必要条件が満たされていることを確認します。

- クラスタ内のノード上でスーパーユーザー特権が確立されていること。

- ファイルシステムが使用中でないこと。ファイルシステムが使用中と見なされるのは、ユーザーがファイルシステム内のディレクトリにアクセスしている場合や、プログラムがファイルシステム内のファイルを開いているからです。ユーザーやプログラムは、クラスタ内のどのノードでもアクセスできます。

1 クラスタ内にある任意のノード上でスーパーユーザーになります。

2 マウントされているクラスタファイルシステムを確認します。

```
# mount -v
```

3 各ノードで、クラスタファイルシステムを使用中の全プロセスの一覧を表示し、停止するプロセスを判断します。

```
# fuser -c [ -u ] mountpoint
```

-c                   ファイルシステムのマウントポイントとなっているファイルと、マウントされているファイルシステム内のファイルがすべて表示されます。

-u                   (任意) 各プロセスIDのユーザーログイン名を表示します。

mountpoint         プロセスを停止するクラスタファイルシステムの名前を指定します。

4 各ノードで、クラスタファイルシステムのプロセスをすべて停止します。

プロセスは任意の方法で停止できます。必要であれば、次のコマンドを使用して、クラスタファイルシステムに関するプロセスを強制終了してください。

```
# fuser -c -k mountpoint
```

クラスタファイルシステムを使用している各ノードに SIGKILL が送信されます。

5 各ノードで、ファイルシステムを使用しているプロセスがないことを確認します。

```
# fuser -c mountpoint
```

6 1つのノードからファイルシステムをマウント解除します。

```
# umount mountpoint
```

mountpoint         マウント解除するクラスタファイルシステムの名前を指定します。クラスタファイルシステムがマウントされているディレクトリの名前や、ファイルシステムのデバイス名パスを指定できません。

7 (任意) `/etc/vfstab` ファイルを編集して、削除するクラスタファイルシステムのエントリを削除します。

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 (任意) ディスクデバイス `group/metadevice/volume/plex` を削除します。  
詳細については、ボリューム管理ソフトウェアのドキュメントを参照してください。

### 例 5-17 クラスタファイルシステムの削除

次に、Solaris ボリュームマネージャー メタデバイスまたはボリューム `/dev/md/oracle/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

(On each node, remove the highlighted entry:)
# vi /etc/vfstab
#device          device          mount  FS      fsck    mount  mount
#to mount        to fsck         point  type    pass   at boot options
#
/global/oracle/dsk/d1 /dev/md/oracle/rdsk/d1 /global/oracle/d1 ufs 2 yes global,logging
```

[Save and exit.]

クラスタファイルシステム上のデータを削除するには、配下のデバイスを削除します。詳細については、ボリューム管理ソフトウェアの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。エラーが発生していない場合は、何も戻されません。

---

注-クラスタファイルシステムの削除など、デバイスやボリューム管理コンポーネントに影響を及ぼすような変更をクラスタ構成に加えたあとに `cluster check` コマンドを実行します。

---

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。



- 2 クラスタのグローバルマウントを確認します。

```
# cluster check -k vfstab
```

## ディスクパス監視の管理

ディスクパス監視 (DPM) の管理コマンドを使用すれば、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、ディスクパスの監視に必要な管理タスクを行うための手順を説明します。ディスクパス監視デーモンに関する概念については、『Oracle Solaris Cluster Concepts Guide』の第3章「Key Concepts for System Administrators and Application Developers」を参照してください。コマンドオプションと関連するコマンドについては、`cldevice(1CL)`のマニュアルページを参照してください。`scdpmd`デーモンの調整に関する詳細は、`scdpmd.conf(4)`のマニュアルページを参照してください。デーモンが報告するログエラーに関しては、`syslogd(1M)`のマニュアルページも参照してください。

注 - `cldevice` コマンドを使ってノードに入出力デバイスを追加すると、監視を行っていた監視リストにディスク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監視から除外されます。

表 5-4 タスクマップ: ディスクパス監視の管理

タスク	手順
ディスクパスを監視します。	122 ページの「ディスクパスを監視する方法」
ディスクパスの監視を解除します。	123 ページの「ディスクパスの監視を解除する方法」
あるノードに対する障害のあるディスクパスのステータスを表示します。	124 ページの「障害のあるディスクパスを表示する方法」
ファイルからディスクパスを監視します。	125 ページの「ファイルからディスクパスを監視する方法」
監視しているすべての共有ディスクパスが失敗したときのノードの自動リブートを有効化または無効化します。	127 ページの「監視しているすべての共有ディスクパスが失敗したときのノードの自動リブートを有効にする方法」 128 ページの「すべての監視共有ディスクパスが失敗した場合にノードの自動リブートを無効にする方法」

表 5-4 タスクマップ:ディスクパス監視の管理 (続き)

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。ブート時に監視対象の DID デバイスを利用できず、DID インスタンスが DID ドライバにアップロードされない場合、不正なディスクパスステータスが報告されることがあります。	124 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。

## ▼ ディスクパスを監視する方法

このタスクは、クラスタのディスクパスを監視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。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 ディスクパスを監視します。  
`# cldevice monitor -n node disk`
- 3 ディスクパスが監視されているか確認します。  
`# cldevice status device`

### 例 5-18 単一ノードのディスクパスを監視

次の例では、単一ノードから `schost-1:/dev/did/rdisk/d1` ディスクパスを監視します。ディスク `/dev/did/dsk/d1` へのパスを監視するのは、ノード `schost-1` 上の DPM デーモンだけです。

```
# cldevice monitor -n schost-1 /dev/did/dsk/d1
# cldevice status d1
```

```
Device Instance   Node           Status
-----
/dev/did/rdisk/d1 phys-schost-1 Ok
```

**例 5-19** すべてのノードのディスクパスを監視

次の例では、すべてのノードから `schost-1:/dev/did/dsk/d1` ディスクパスを監視します。DPM は、`/dev/did/dsk/d1` が有効なパスであるすべてのノードで起動されます。

```
# cldevice monitor /dev/did/dsk/d1
# cldevice status /dev/did/dsk/d1
```

Device Instance	Node	Status
/dev/did/rdsk/d1	phys-schost-1	Ok

**例 5-20** CCR からディスク構成を読み直す

次の例では、デーモンが CCR からディスク構成を読み直し、監視されているディスクパスをそのステータスとともに出力します。

```
# cldevice monitor +
# cldevice status
```

Device Instance	Node	Status
/dev/did/rdsk/d1	schost-1	Ok
/dev/did/rdsk/d2	schost-1	Ok
/dev/did/rdsk/d3	schost-1	Ok
	schost-2	Ok
/dev/did/rdsk/d4	schost-1	Ok
	schost-2	Ok
/dev/did/rdsk/d5	schost-1	Ok
	schost-2	Ok
/dev/did/rdsk/d6	schost-1	Ok
	schost-2	Ok
/dev/did/rdsk/d7	schost-2	Ok
/dev/did/rdsk/d8	schost-2	Ok

**▼ ディスクパスの監視を解除する方法**

ディスクパスの監視を解除する場合は、この手順を使用します。

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

- 2 監視を解除するディスクパスの状態を調べます。

```
# cldevice status device
```

- 3 各ノードで、適切なディスクパスの監視を解除します。

```
# cldevice unmonitor -n node disk
```

### 例 5-21 ディスクパスの監視解除

次の例では、schost-2:/dev/did/rdsk/d1 ディスクパスの監視を解除し、クラスタ全体のディスクパスの一覧とそのステータスを出力します。

```
# cldevice unmonitor -n schost2 /dev/did/rdsk/d1
# cldevice status -n schost2 /dev/did/rdsk/d1
```

Device Instance	Node	Status
/dev/did/rdsk/d1	schost-2	Unmonitored

## ▼ 障害のあるディスクパスを表示する方法

クラスタに障害のあるディスクパスを表示する場合は、次の手順を使用します。

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 全クラスタ内の障害のあるディスクパスを表示します。

```
# cldevice status -s fail
```

### 例 5-22 障害のあるディスクパスを表示する

次の例では、全クラスタ内の障害のあるディスクパスを表示します。

```
# cldevice status -s fail
```

Device Instance	Node	Status
dev/did/dsk/d4	phys-schost-1	fail

## ▼ ディスクパスの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。

- 監視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、監視対象の DID パスの下のデバイスがオンラインに戻らない。

ブート時に監視対象の DID デバイスを利用できず、このため DID インスタンスが DID ドライバにアップロードされないため、不正なディスクパス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 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
```

- 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
```

- 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監視する方法

ファイルを使ってディスクパスを監視したり、その監視を解除する場合は、次の手順を使用します。

ファイルを使用してクラスタ構成を変更するには、まず現在の構成をエクスポートします。このエクスポート操作により XML ファイルが作成されます。このファイルは、変更する構成項目を設定するために修正できます。この手順では、このプロセス全体を説明します。

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。

- クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- デバイス構成を XML ファイルにエクスポートします。

```
# cldevice export -o configurationfile
```

-o configurationfile XML ファイルのファイル名を指定します。

- 3 デバイスパスが監視されるよう、構成ファイルを変更します。  
監視するデバイスパスを検索し、`monitored` 属性を `true` に設定します。
- 4 デバイスパスを監視します。  
`# cldevice monitor -i configurationfile`  
`-i configurationfile` 変更された XML ファイルのファイル名を指定します。
- 5 この時点でデバイスパスが監視されていることを確認します。  
`# cldevice status`

### 例 5-23 ファイルからディスクパスを監視する

次の例では、ノード `phys-schost-2` とデバイス `d3` の間のデバイスパスが、XML ファイルを使用することによって監視されています。

最初に、現在のクラスタ構成をエクスポートします。

```
# cldevice export -o deviceconfig
```

`deviceconfig` XML ファイルは、`phys-schost-2` と `d3` の間のパスが現在は監視されていないことを示しています。

```
<?xml version="1.0"?>
<!DOCTYPE cluster SYSTEM "/usr/cluster/lib/xml/cluster.dtd">
<cluster name="brave_clus">
.
.
.
  <deviceList readonly="true">
    <device name="d3" ctd="c1t8d0">
      <devicePath nodeRef="phys-schost-1" monitored="true"/>
      <devicePath nodeRef="phys-schost-2" monitored="false"/>
    </device>
  </deviceList>
</cluster>
```

そのパスを監視するには、次のように、監視される `attribute` を `true` に設定します。

```
<?xml version="1.0"?>
<!DOCTYPE cluster SYSTEM "/usr/cluster/lib/xml/cluster.dtd">
<cluster name="brave_clus">
.
.
.
  <deviceList readonly="true">
    <device name="d3" ctd="c1t8d0">
      <devicePath nodeRef="phys-schost-1" monitored="true"/>
      <devicePath nodeRef="phys-schost-2" monitored="true"/>
    </device>
  </deviceList>
</cluster>
```

```

    </device>
  </deviceList>
</cluster>

```

cldevice コマンドを使用して、ファイルを読み込み、監視を有効にします。

```
# cldevice monitor -i deviceconfig
```

cldevice コマンドを使用して、この時点でデバイスが監視されていることを確認します。

```
# cldevice status
```

参照 クラスタ構成のエクスポート、および結果の XML ファイルを使用したクラスタ構成の設定の詳細は、[cluster\(1CL\)](#) および [clconfiguration\(5CL\)](#) のマニュアルページを参照してください。

## ▼ 監視しているすべての共有ディスクパスが失敗したときのノードの自動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監視対象の共有ディスクパスが失敗した。
- 少なくとも1つのディスクがクラスタ内の異なるノードからアクセス可能である。

ノードが再起動すると、そのノード上でマスターされているすべてのリソースグループとデバイスグループが別のノード上で再起動します。

ノードが自動リブートしたあと、ノード上のすべての監視対象共有ディスクパスがアクセス不能のままである場合、そのノードは再び自動リブートしません。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。監視された共有ディスクのみが影響を受けます。

- 1 クラスタ内の任意のノードで、スーパーユーザーになるか、`solaris.cluster.modify RBAC` の承認を提供する役割になります。

- 2 クラスタのすべてのノードに対して、監視共有ディスクパスがすべて失敗したときの、ノードの自動リブートを有効にします。

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

## ▼ すべての監視共有ディスクパスが失敗した場合にノードの自動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監視共有ディスクパスに障害が発生しても、ノードは自動的にはリブートしません。

- 1 クラスタ内の任意のノードで、スーパーユーザーになるか、**solaris.cluster.modify RBAC**の承認を提供する役割になります。
- 2 クラスタ内のすべてのノードに対して、ノードへのすべての監視共有ディスクパスに障害が発生した場合の、ノードの自動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 Concepts Guide](#)』の「[Quorum and Quorum Devices](#)」を参照してください。

- 129 ページの「定足数デバイスの管理」
- 151 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を使用すると、いくつかの手順を行うことができます。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。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投票ノードで実行する必要があります。
- `clquorum` コマンドが中断または失敗すると、定足数の構成情報は、クラスタ構成データベースで矛盾することになります。このような矛盾が発生した場合は、このコマンドを再度実行するか、`clquorum reset` コマンドを実行して定足数構成をリセットします。

- クラスタの可用性を最高にするには、定足数デバイスによる合計の投票数が、ノードによる合計の投票数よりも少なくなるようにします。少なくなければ、すべてのノードが機能していても、すべての定足数デバイスを使用できない場合、そのノードはクラスタを形成できません。
- 現在定足数デバイスとして構成されているディスクは、Solaris ZFS ストレージプールには追加しないでください。構成済みの定足数デバイスを Solaris ZFS ストレージプールに追加すると、ディスクはEFI ディスクとしてラベルが変更され、また定足数構成情報が失われ、ディスクはクラスタへの定足数投票を提供しなくなります。ディスクがストレージプール内に入ると、そのディスクは定足数デバイスとして構成できます。または、ディスクの定足数デバイス構成を解除し、ディスクをストレージプールに追加した後に、そのディスクを定足数デバイスとして再構成することができます。

---

注 - `clsetup` コマンドは、ほかの 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 6-1 タスクリスト:定足数の管理

タスク	説明
<code>clsetup</code> ユーティリティを使用し、クラスタに定足数デバイスを追加する	132 ページの「定足数デバイスの追加」
<code>clsetup</code> ユーティリティを使用する ( <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	140 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	142 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	143 ページの「定足数デバイスを交換する方法」

---

表 6-1 タスクリスト:定足数の管理 (続き)

タスク	説明
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	144 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」
clsetup ユーティリティを使用する (clquorum を生成する) ことにより、定足数デバイスを保守状態にする  (保守状態にある場合、定足数デバイスは定足数確立の投票に参加しません。)	146 ページの「定足数デバイスを保守状態にする方法」
clsetup ユーティリティを使用する (clquorum を生成する) ことにより、定足数構成をデフォルト状態にリセットする	148 ページの「定足数デバイスを保守状態から戻す」
clquorum 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	149 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 (DR) を実行するときには、いくつかの問題を考慮する必要があります。

- Oracle Solaris の DR 機能の説明で述べられているすべての必要条件、手順、制限は、Oracle Solaris Cluster の DR サポートにも適用されます (オペレーティングシステムでの休止操作は除く)。したがって、Oracle Solaris Cluster ソフトウェアで DR 機能を使用する前に、Oracle Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- 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場合 DR 削除操作を実行できません。
- DR 操作がアクティブなデバイスに影響する場合、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デバイスを識別します。

定足数デバイスを削除するには、次の手順をその順番どおりに行います。

表 6-2 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	132 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	140 ページの「定足数デバイスを削除する方法」
3. 削除する定足数デバイス上で DR 削除操作を実行	

## 定足数デバイスの追加

ここでは、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 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
  - Oracle 社製 Sun ZFS Storage Appliance
- 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。

- [133 ページの「共有ディスク定足数デバイスを追加する方法」](#)
- [136 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-複製されたディスクを定足数デバイスとして構成することはできません。複製されたディスクを定足数デバイスとして追加しようとすると、次のエラーメッセージが表示され、コマンドはエラーコードとともに終了します。

*Disk-name* is a replicated device. Replicated devices cannot be configured as quorum devices.

---

共有ディスク定足数デバイスは、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せん。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。

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。

この手順を実行するには、ノードが共有するデバイス ID (Device Identifier、DID) によりディスクドライブを確認します。cldevice show コマンドを使用して、DID 名の一覧を参照します。詳細は、[cldevice\(ICL\)](#) 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、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 `1` ユーティリティーを起動します。  
`# clsetup`  
`clsetup` のメインメニューが表示されます。
- 3 定足数のオプション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` ユーティリティーのプロンプトが表示されたら `yes` と入力します。  
`clsetup` ユーティリティーにより、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
```

### 例 6-1 共有ディスク定足数デバイスの追加

次の例は、共有ディスク定足数デバイスを追加する際に `clsetup` によって生成される `clquorum` コマンドと、検証ステップを示しています。

```
Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.
```

```
[Start the clsetup utility:]
# clsetup
[Select Quorum>Add a quorum device]
[Answer the questions when prompted.]
[You will need the following information.]
  [Information:                Example:]
  [Directly attached shared disk  shared disk]
  [Global device                  d20]

[Verify that the clquorum command was completed successfully:]
clquorum add d20
```

Command completed successfully.

```
[Quit the clsetup Quorum Menu and Main Menu.]
```

```
[Verify that the quorum device is added:]
```

```
# clquorum list -v
```

```
Quorum      Type
-----
d20          shared_disk
scphyshost-1 node
scphyshost-2 node
```

## ▼ Sun ZFS Storage Appliance の NAS 定足数デバイスを追加する方法

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。

`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 iSCSI デバイスのセットアップ手順については、**Sun ZFS 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.LUNName,IPAddress_of_NASDevice
# devfsadm -i iscsi
# cldevice refresh
```

- 3 1つのクラスタノードから DID を iSCSI LUN 用に構成します。

```
# /usr/cluster/bin/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、スーパーユーザーになるか、RBAC の承認 solaris.cluster.modify を提供する役割になります。

- 6 clquorum コマンドを使用して、手順 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 Quorum Server を定足数デバイスとして追加する前に、ホストマシンに Oracle Solaris Cluster Quorum Server ソフトウェアをインストールし、定足数サーバーを起動して動作させておく必要があります。定足数サーバーのインストールについては、『Oracle Solaris Cluster 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。



- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。
- 2 すべての Oracle Solaris Cluster ノードがオンライン状態であり、Oracle Solaris Cluster Quorum Server と通信が行えることを確認します。
  - a. クラスタノードに直接接続されているネットワークスイッチが次の基準のいずれかを満たすことを確認します。
    - スイッチは RSTP (Rapid Spanning Tree Protocol) をサポートしています。
    - スイッチ上で高速ポートモードが有効になっています。

クラスタノードと定足数サーバー間ですぐに通信できるようにするには、これらの機能の1つが必要です。この通信がスイッチによって大幅に遅延すると、クラスタはこの通信の中断を定足数デバイスが失われたものと解釈します。
  - b. パブリックネットワークで可変長サブネット化 (CIDR (Classless Inter-Domain Routing) とも呼ばれる) 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 定足数サーバー定足数デバイスのオプションの番号を入力します。**quorum\_server** 定足数デバイスを追加することを確認するには、「**yes**」と入力します。  
新しい定足数デバイスの名前を入力するように、clsetup ユーティリティ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- 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` ユーティリティーではその旨のメッセージが表示されます。

- 11 定足数デバイスが追加されていることを確認します。

```
# clquorum list -v
```

## 例6-2 定足数サーバー定足数デバイスの追加

次の例は、定足数サーバー定足数デバイスを追加する際に `clsetup` によって生成される `clquorum` コマンドを示しています。またこの例では検証ステップも示します。

```
Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.
```

```
[Start the clsetup utility:]
```

```
# clsetup
```

```
[Select Quorum > Add a quorum device]
```

```
[Answer the questions when prompted.]
```

```
[You will need the following information.]
```

```
[Information:          Example:]
[Quorum Device        quorum_server quorum device]
[Name:                qd1]
[Host Machine Name:   10.11.124.84]
[Port Number:        9001]
```

```
[Verify that the clquorum command was completed successfully:]
```

```
clquorum add -t quorum_server -p qshost=10.11.124.84 -p port=9001 qd1
```

```
Command completed successfully.
```

```
[Quit the clsetup Quorum Menu and Main Menu.]
```

```
[Verify that the quorum device is added:]
```

```
# clquorum list -v
```

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

```
# clquorum status
```

```
=== Cluster Quorum ===
```

```
-- Quorum Votes Summary --
```

Needed	Present	Possible
-----	-----	-----
3	5	5

```
-- Quorum Votes by Node --
```

Node Name	Present	Possible	Status
-----	-----	-----	-----

```
phys-schost-1    1      1      Online
phys-schost-2    1      1      Online
```

```
-- Quorum Votes by Device --
```

Device Name	Present	Possible	Status
-----	-----	-----	-----
qd1	1	1	Online
d3s2	1	1	Online
d4s2	1	1	Online

## 定足数デバイスの削除または交換

このセクションでは、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。

- 140 ページの「定足数デバイスを削除する方法」
- 142 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」
- 143 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。構成されているデバイスが、クラスタの最後の定足数デバイスの場合は、`clquorum(1CL)` はデバイスを構成から削除できません。ノードを削除する場合は、そのノードに接続されている定足数デバイスをすべて削除してください。

---

注-削除するデバイスがクラスタの最後の定足数デバイスの場合は、[142 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。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。

- 2 削除する定足数デバイスを判別します。  
# **clquorum list -v**
- 3 **clsetup**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**

### 例6-3 定足数デバイスの削除

次に、2つ以上の定足数デバイスが構成されているクラスタから定足数デバイスを削除する例を示します。

Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.

```
[Determine the quorum device to be removed:]
# clquorum list -v
[Start the clsetup utility:]
# clsetup
[Select Quorum>Remove a quorum device]
[Answer the questions when prompted.]
[Quit the clsetup Quorum Menu and Main Menu.]
[Verify that the quorum device is removed:]
# clquorum list -v
```

Quorum	Type
-----	----
scphyshost-1	node
scphyshost-2	node
scphyshost-3	node

**注意事項** 定足数サーバー定足数デバイスの削除中に、クラスタと定足数サーバーホストの間の通信が失われた場合、定足数サーバーホストに関する無効な構成情報をクリーンアップする必要があります。このクリーンアップの実行に関する説明は、[155 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` 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、140 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。132 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 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 `clquorum` コマンドを使用して定足数デバイスを削除します。  
定足数デバイスに障害が発生した場合は、`-F` (強制) オプションを使用して、障害が発生したデバイスを削除します。

```
# clquorum remove -F qd1
```

---

注-また、削除するノードを保持状態とし、定足数デバイスを `clquorum removequorum` コマンドを使用して削除することができます。`clsetup` 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、204 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
```

## 例 6-4 最後の定足数デバイスの削除

この例では、クラスタを保持モードにし、クラスタ構成で最後の定足数デバイスを削除する方法を示しています。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any
cluster node.]
[Place the cluster in install mode:]
# cluster set -p installmode=enabled
[Remove the quorum device:]
# clquorum remove d3
[Verify that the quorum device has been removed:]
# clquorum list -v
  Quorum      Type
  -----
scphyshost-1  node
scphyshost-2  node
scphyshost-3  node
```

### ▼ 定足数デバイスを交換する方法

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、[132 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、[140 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。

- 144 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」
- 146 ページの「定足数デバイスを保守状態にする方法」
- 148 ページの「定足数デバイスを保守状態から戻す」
- 149 ページの「クラスタ構成を一覧表示する方法」
- 150 ページの「定足数デバイスを修復する方法」
- 151 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` ユーティリティを使用すると、既存の定足数デバイスのノードリストにノードを追加したり、ノードリストからノードを削除したりできます。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、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 変更したい定足数デバイスの名前を判別します。  

```
# clquorum list -v
```
- 3 `clsetup` ユーティリティを起動します。  

```
# clsetup
```

  
メインメニューが表示されます。
- 4 定足数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`

### 例 6-5 定足数デバイスノードリストの変更

次の例に、`clsetup` ユーティリティを使用して、定足数デバイスのノードリストにノードを追加したり、ノードリストからノードを削除する方法を示します。この例では、定足数デバイスの名前は `d2` であり、この手順の最終目的は別のノードを定足数デバイスのノードリストに追加することです。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC
authorization on any node in the cluster.]
```

```
[Determine the quorum device name:]
```

```
# clquorum list -v
Quorum          Type
-----
d2              shared_disk
sc-phys-schost-1 node
sc-phys-schost-2 node
sc-phys-schost-3 node
```

```
[Start the clsetup utility:]
```

```
# clsetup
```

```
[Type the number that corresponds with the quorum option.]
```

```
.
```

```
[Type the number that corresponds with the option to remove a quorum device.]
```

```
.
```

```
[Answer the questions when prompted.]
```

```
[You will need the following information:]
```

```
Information:      Example:
Quorum Device Name:  d2
```

```
[Verify that the clquorum command completed successfully:]
```

```
clquorum remove d2
Command completed successfully.
```

```
[Verify that the quorum device was removed.]
```

```
# clquorum list -v
Quorum          Type
-----
sc-phys-schost-1 node
sc-phys-schost-2 node
sc-phys-schost-3 node
```

```

[Type the number that corresponds with the Quorum option.]
.
[Type the number that corresponds with the option to add a quorum device.]
.
[Answer the questions when prompted.]
[You will need the following information:]

    Information          Example:
    quorum device name   d2

[Verify that the clquorum command was completed successfully:]
clquorum add d2
    Command completed successfully.

Quit the clsetup utility.

[Verify that the correct nodes have paths to the quorum device.
In this example, note that phys-schost-3 has been added to the
enabled hosts list.]
# clquorum show d2 | grep Hosts
=== Quorum Devices ===

Quorum Device Name:      d2
  Hosts (enabled):      phys-schost-1, phys-schost-2, phys-schost-3

[Verify that the modified quorum device is online.]

# clquorum status d2
=== Cluster Quorum ===

--- Quorum Votes by Device ---

Device Name      Present      Possible      Status
-----
d2                1            1            Online

```

## ▼ 定足数デバイスを保守状態にする方法

clquorum コマンドを使用して定足数デバイスを保守状態にします。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。

サービスから定足数デバイスを長時間はずす場合は、その定足数デバイスを保守状態にします。定足数デバイスの定足数投票数 (quorum vote count) はゼロに設定されるため、そのデバイスが稼働中でも定足数確立の投票には参加しません。保守状態でも定足数デバイスの構成情報は保持されます。

---

注-2 ノードクラスタでは、定足数デバイスが少なくとも1つは構成されている必要があります。構成されているデバイスが2 ノードクラスタの最後の定足数デバイスの場合、clquorum は失敗してデバイスは保守状態になりません。

---

クラスタノードを保守状態にする方法については、204 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。

- 1 クラスタの任意のノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 定足数デバイスを保守状態にします。

```
# clquorum disable device
```

`device` 変更するディスクデバイスの DID 名 (d4 など) を指定します。

- 3 定足数デバイスが保守状態にあることを確認します。

保守状態にしたデバイスの出力は、定足数デバイスの投票数 (以下の例の Quorum device votes) がゼロになっていなければなりません。

```
# clquorum status device
```

#### 例 6-6 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、148 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、204 ページの「ノードを保守状態にする」を参照してください。

## ▼ 定足数デバイスを保守状態から戻す

この作業は、定足数デバイスが保守状態にある場合にその状態から定足数デバイスを戻して定足数投票数をデフォルトにリセットするときに実行します。



注意 - globaldev または node オプションのどちらも指定しない場合、定足数投票数はクラスタ全体でリセットされます。

定足数デバイスを構成する場合、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、206 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 Concepts Guide』の「About Quorum Vote Counts」を参照してください。

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

この手順では、長形式の 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、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 定足数投票数をリセットします。  

```
# clquorum enable device
```

*device* リセットする定足数デバイスの DID 名 (d4 など) を指定します。
- 3 ノードが保守状態にあったために定足数投票数をリセットする場合は、このノードをリブートします。
- 4 定足数投票数を確認します。  

```
# clquorum show +
```

### 例 6-7 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

定足数構成を一覧表示するには、スーパーユーザーになる必要はありません。RBACの承認 `solaris.cluster.read` を提供する任意の役割にすることができます。

---

注- 定足数デバイスに対するノード接続の数を増減させる場合、定足数が自動的に再計算されることはありません。すべての定足数デバイスをいったん削除し、その後それらを構成に追加し直すと、正しい定足数が再構成されます。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。

- **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 6-8 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/rdsdsk/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

この作業は、動作が不正な定足数デバイスを交換する場合に行なってください。

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

この手順では、長形式の 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- 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、[132 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、[140 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、132 ページの「定足数デバイスの追加」を参照してください。

---

注-手順1で定足数デバイスを別途追加した場合は、デバイスを削除しても安全です。定足数デバイスを削除する方法については、140 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の値は、『Oracle Solaris Cluster 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」の指示に従って増分することができます。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 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-Oracle RAC (Oracle Real Application Clusters) 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。

- 152 ページの「Quorum Server Software の起動および停止」
- 152 ページの「定足数サーバーを起動する方法」

- 153 ページの「定足数サーバーを停止する方法」
- 154 ページの「定足数サーバーに関する情報の表示」
- 155 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 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 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を変更する方法については、151 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スーパーユーザーになります。
- 2 ソフトウェアを起動するには、`clquorumserver start` コマンドを使用します。

```
# /usr/cluster/bin/clquorumserver start quorumserver
```

`quorumserver` 定足数サーバーを識別します。定足数サーバーが待機するポート番号を使用できます。構成ファイルでインスタンス名を指定した場合は、代わりにその名前を使用できます。

1台の定足数サーバーを起動するには、インスタンス名とポート番号のいずれかを指定します。複数の定足数サーバーを構成している場合、すべての定足数サーバーを起動するには、+オペランドを使用します。



**例 6-9** すべての構成済み定足数サーバーの起動

次の例では、構成されているすべての定足数サーバーを起動します。

```
# /usr/cluster/bin/clquorumserver start +
```

**例 6-10** 特定の定足数サーバーの起動

次の例では、ポート番号 2000 で待機している定足数サーバーを起動します。

```
# /usr/cluster/bin/clquorumserver start 2000
```

## ▼ 定足数サーバーを停止する方法

- 1 **Oracle Solaris Cluster** ソフトウェアを起動するホスト上でスーパーユーザーになります。
- 2 ソフトウェアを停止するには、**clquorumserver stop** コマンドを使用します。

```
# /usr/cluster/bin/clquorumserver stop [-d] quorumserver
```

**-d**                    マシンを次回起動したときに、定足数サーバーを起動するかどうかを制御します。**-d** オプションを指定すると、次のマシン起動時に定足数サーバーは起動しません。

**quorumserver**        定足数サーバーを識別します。定足数サーバーが待機するポート番号を使用できます。構成ファイルでインスタンス名を指定した場合は、代わりにその名前を使用できます。

1 台の定足数サーバーを停止するには、インスタンス名とポート番号のいずれかを指定します。複数の定足数サーバーを構成している場合、すべての定足数サーバーを停止するには、**+** オペランドを使用します。

**例 6-11** すべての構成済み定足数サーバーの停止

次の例では、構成されているすべての定足数サーバーを停止します。

```
# /usr/cluster/bin/clquorumserver stop +
```

**例 6-12** 特定の定足数サーバーの停止

次の例では、ポート番号 2000 で待機している定足数サーバーを停止します。

```
# /usr/cluster/bin/clquorumserver stop 2000
```

## 定足数サーバーに関する情報の表示

定足数サーバーについての構成情報を表示することができます。このコマンドは、定足数サーバーを定足数デバイスとして構成しているすべてのクラスタごとに、対応するクラスタ名、クラスタ ID、予約鍵のリスト、および登録鍵のリストを表示します。

### ▼ 定足数サーバーに関する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スーパーユーザーになります。スーパーユーザー以外のユーザーには、`solaris.cluster.read` RBAC (Role-Based Access Control) の承認が必要です。RBAC 権限プロファイルの詳細については、[rbac\(5\)](#) のマニュアルページを参照してください。
- 2 `clquorumserver` コマンドを使用することで、定足数サーバーの構成情報を表示します。

```
# /usr/cluster/bin/clquorumserver show quorumserver
```

`quorumserver` 1つまたは複数の定足数サーバーを識別します。インスタンス名またはポート番号で定足数サーバーを指定できます。すべての定足数サーバーの構成情報を表示するには、+オペランドを使用します。

#### 例 6-13 1つの定足数サーバーの構成の表示

次の例では、ポート 9000 を使用する定足数サーバーの構成情報を表示します。次のコマンドは、定足数サーバーが定足数デバイスとして構成されているすべてのクラスタの情報を表示します。この情報にはクラスタの名前と ID、およびデバイスの予約鍵と登録鍵のリストが含まれます。

次の例では、クラスタ `bastille` の ID が 1、2、3、および 4 であるノードが、定足数サーバー上に鍵を登録しています。また、ノード 4 は定足数デバイスの予約を所有しているため、その鍵は予約リストに表示されます。

```
# /usr/cluster/bin/clquorumserver show 9000
```

```
=== Quorum Server on port 9000 ===
```

```
--- Cluster bastille (id 0x439A2EFB) Reservation ---
```

```
Node ID:                4
  Reservation key:      0x439a2efb00000004
```

```
--- Cluster bastille (id 0x439A2EFB) Registrations ---
```

```
Node ID:                1
  Registration key:     0x439a2efb00000001
```

```

Node ID:                2
Registration key:       0x439a2efb00000002

Node ID:                3
Registration key:       0x439a2efb00000003

Node ID:                4
Registration key:       0x439a2efb00000004

```

#### 例 6-14 複数の定足数サーバーの構成の表示

次の例では、3つの定足数サーバー qs1、qs2、および qs3 の構成情報を表示します。

```
# /usr/cluster/bin/clquorumserver show qs1 qs2 qs3
```

#### 例 6-15 動作しているすべての定足数サーバーの構成の表示

次の例では、動作しているすべての定足数サーバーの構成情報を表示します。

```
# /usr/cluster/bin/clquorumserver show +
```

## 期限切れの定足数サーバークラスタ情報のク リーンアップ

quorumserver のタイプの定足数デバイスを削除するには、How to Remove a Quorum Deviceで説明されているように、140 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 140 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でスーパーユーザーになります。
- 2 **clquorumserver clear** コマンドを使用して、構成ファイルをクリーンアップします。  

```
# clquorumserver clear -c clustername -I clusterID quorumserver [-y]
```

**-c clustername** 以前に定足数サーバーを定足数デバイスとして使用していたクラスタの名前です。

クラスタ名を取得するには、クラスタノード上で `cluster show` を実行します。

**-I clusterID** クラスタ ID です。

クラスタ ID は 8 桁の 16 進数です。クラスタ ID を取得するには、クラスタノード上で `cluster show` を実行します。

**quorumserver** 1 つまたは複数の定足数サーバーの識別子です。

定足数サーバーは、ポート番号かインスタンス名で識別できません。ポート番号は、クラスタノードが定足数サーバーと通信するために使用されます。インスタンス名は、定足数サーバーの構成ファイル `/etc/scqsd/scqsd.conf` で指定されます。

**-y** 実行前に確認のプロンプトを表示することなく、`clquorumserver clear` コマンドに、構成ファイルからクラスタ情報をクリーンアップさせます。

期限切れのクラスタ情報を定足数サーバーから削除したいことが確かである場合のみ、このオプションを使用します。
- 3 (省略可能) このサーバーインスタンスでほかに定足数デバイスが構成されていない場合は、定足数サーバーを停止します。

### 例 6-16 定足数サーバー構成からの期限切れのクラスタ情報のクリーンアップ

次の例は、`sc-cluster` という名前のクラスタについての情報を、ポート 9000 を使用する定足数サーバーから削除します。

```
# clquorumserver clear -c sc-cluster -I 0x4308D2CF 9000
The quorum server to be unconfigured must have been removed from the cluster.
Unconfiguring a valid quorum server could compromise the cluster quorum. Do you
want to continue? (yes or no) y
```



# クラスタインターコネク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のソフトウェア上の作業手順について説明します。

クラスタインターコネクとパブリックネットワークの管理には、ハードウェア上の作業とソフトウェア上の作業が含まれます。通常、初めてクラスタをインストールおよび構成するときには、IP ネットワークマルチパス (IP Network Multipathing) グループを含むクラスタインターコネクとパブリックネットワークを構成します。マルチパスは Oracle Solaris 11 OS に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、175 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。

この章では、次のトピックの手順について説明します。

- 160 ページの「クラスタインターコネクの管理」
- 175 ページの「パブリックネットワークの管理」

この章の関連手順の詳細な説明については、表 7-1 および表 7-3 を参照してください。

クラスタインターコネク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 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

ここでは、クラスタインターコネクット(クラスタトランスポートアダプタ、クラスタトランスポートケーブルなど)を再構成する手順を説明します。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` ユーティリティを使用すると、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投票ノードで実行する必要があります。

クラスタソフトウェアをインストールする手順については、『Oracle Solaris Cluster 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。

注-クラスタインターコネクット手順中、通常は、(適切であれば)デフォルトのポート名を選択してもかまいません。デフォルトのポート名は、ケーブルのアダプタ側が接続されているノードの内部ノード ID 番号と同じです。

表 7-1 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	23 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」
<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	162 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	163 ページの「クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを追加する方法」
<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	165 ページの「クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチを削除する方法」
<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	168 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	169 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	171 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」



表 7-1 タスクリスト:クラスタインターコネクトの管理 (続き)

タスク	手順
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	172 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トでの動的再構成

クラスタインターコネクト上で動的再構成 (DR) を実行するときには、いくつかの問題を考慮する必要があります。

- Oracle Solaris の DR 機能の説明で述べられているすべての必要条件、手順、制限は、Oracle Solaris Cluster の DR サポートにも適用されます (オペレーティングシステムでの休止操作は除く)。したがって、Oracle Solaris Cluster ソフトウェアで DR 機能を使用する前に、Oracle Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。
- 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トインタフェース上で実行された DR ボード削除操作を拒否します。
- アクティブなクラスタインターコネクトで DR を実行するには、クラスタからアクティブなアダプタを完全に削除する必要があります。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には、他のすべてのクラスタノードに対する有効なパスが、少なくとも 1 つは存在していなければなりません。したがって、個々のクラスタノードへの最後のパスをサポートするプライベートインターコネクトインタフェースを無効にしないでください。

パブリックネットワークインタフェース上で DR 操作を実行するときは、次の手順をその順番どおりに行います。

表 7-2 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トからインタフェースを無効にして削除	176 ページの「パブリックネットワークインタフェースでの動的再構成」
2. パブリックネットワークインタフェース上で DR 操作を実行	

## ▼ 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スーパーユーザーとしてログインする必要はありません。

- 1 クラスタインターコネクットの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 clinterconnect status を実行してもメッセージが繰り返される場合は、適切な処置を行なってください。

### 例 7-1 クラスタインターコネクッ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ッ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、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。

- 1 クラスタトランスポートケーブルが物理的に取り付けられていることを確認します。

クラスタトランスポートケーブルのインストール手順については、『[Oracle Solaris Cluster Hardware Administration Manual](#)』を参照してください。

- 2 クラスタ内にある任意のノード上でスーパーユーザーになります。

- 3 `clsetup` ユーティリティを起動します。

```
# clsetup
```

メインメニューが表示されます。

- 4 クラスタインターコネクットメニューを表示するためのオプション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ットで次のアダプタのいずれかを使用する予定の場合、関連するエントリを各クラスタノードの `/etc/system` ファイルに追加します。このエントリは、次のシステム再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 7-2 クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチの追加

次の例に、clsetup ユーティリティを使用し、トランスポートケーブル、トランスポートアダプタ、トランスポートスイッチをノードに追加する方法を示します。

```
[Ensure that the physical cable is installed.]
[Start the clsetup utility:]
# clsetup
[Select Cluster interconnect]

[Select either Add a transport cable,
Add a transport adapter to a node,
or Add a transport switch.]
[Answer the questions when prompted.]
  [You Will Need: ]
[Information:      Example:]
  node names      phys-schost-1
  adapter names   net5
  switch names    hub2
  transport type  dlpi
[Verify that the clinterconnect
command completed successfully:]Command completed successfully.
Quit the clsetup Cluster Interconnect Menu and Main Menu.
[Verify that the cable, adapter, and switch are added:]
# clinterconnect show phys-schost-1: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 SState (0):                  Enabled

```

```
# clinterconnect show phys-schost-1:hub2
```

```

=== Transport Switches ===
Transport Switch:                hub2
Switch State:                    Enabled
Switch Type:                     switch
Switch Port Names:               1 2
Switch Port State(1):            Enabled
Switch Port State(2):            Enabled

```

次の手順 クラスタトランスポートケーブルのインターコネクトのステータスを確認するには、[162 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、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-各クラスタノードには、他のすべてのクラスタノードに対する(機能している)トランスポートバスが少なくとも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。

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 残りのクラスタトランスポートパスの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ッ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- 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-ポートがトランスポートケーブルの終端として使用されている場合、スイッチは削除できません。

---

- 9 ケーブル、アダプタ、またはスイッチが削除されたことを確認します。

```
# clinterconnect show node:adapter,adapternode
# clinterconnect show node:adapter
# clinterconnect show node:switch
```

ノードからトランスポートケーブルやトランスポートアダプタが削除された場合は、このコマンドの出力には表示されません。

### 例 7-3 トランスポートケーブル、トランスポートアダプタ、トランスポートスイッチの削除

次の例に、`clsetup` コマンドを使用して、トランスポートケーブル、トランスポートアダプタ、またはトランスポートスイッチを削除する方法を示します。

```
[Become superuser on any node in the cluster.]
[Start the utility:]
# clsetup
[Select Cluster interconnect.]
[Select either Remove a transport cable,
Remove a transport adapter to a node,
or Remove a transport switch.]
[Answer the questions when prompted.]
  You Will Need:
  Information           Example:
  node names            phys-schost-1
  adapter names         net0
  switch names         hub1
[Verify that the clinterconnect
command was completed successfully:]
Command completed successfully.
[Quit the clsetup utility Cluster Interconnect Menu and Main Menu.]
[Verify that the cable, adapter, or switch is removed:]
# clinterconnect show phys-schost-1: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
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 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、すでに存在するクラスタのトランスポートケーブルを有効にするために使用します。

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 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 **clsetup** ユーティリティを起動します。  

```
# clsetup
```

 メインメニューが表示されます。
- 3 クラスタインターコネク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、**Return** キーを押します。
- 4 トランスポートケーブルを有効にするためのオプションの番号を入力し、**Return** キーを押します。

プロンプトが表示されたなら、指示に従います。ケーブルのいずれかの終端のノード名およびアダプタ名の両方を入力する必要があります。



- 5 ケーブルが有効になっていることを確認します。

```
# clinterconnect show node:adapter,adapternode
```

#### 例 7-4 クラスタトランスポートケーブルを有効にする

次の例に、ノード `phys-schost-2` にあるアダプタ `net0` のクラスタトランスポートケーブルを有効にする方法を示します。

```
[Become superuser on any node.]
[ Start the clsetup utility:]
# clsetup
[Select Cluster interconnect>Enable a transport cable.]

[Answer the questions when prompted.]
[You will need the following information.]
  You Will Need:
Information:                               Example:
node names                                 phys-schost-2
adapter names                              net0
switch names                              hub1
[Verify that the scinterconnect
  command was completed successfully:]

clinterconnect enable phys-schost-2:net0

Command completed successfully.
[Quit the clsetup Cluster Interconnect Menu and Main Menu.]
[ Verify that the cable is enabled:]
# 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tpas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tpで発生する問題の解決や、クラスタインターコネクtpのハードウェアの交換に便利です。

ケーブルを無効にした場合、このケーブルの2つのエンドポイントは構成されたままになります。トランスポートケーブルの終端として使用されているアダプタは削除できません。



注意-各クラスタノードには、他のすべてのクラスタノードに対する(機能している)トランスポートパスが少なくとも1つずつ必要です。2つのノードは必ず接続されており、お互いに分離されているノードは存在しません。ケーブルを無効にする前には、必ず、ノードのクラスタインターコネクットの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。

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 ケーブルを無効にする前に、クラスタインターコネクットの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ッ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、**Return** キーを押します。
- 5 トランスポートケーブルを無効にするためのオプションの番号を入力し、**Return** キーを押します。

指示に従い、必要な情報を入力します。このクラスタインターコネクットのすべてのコンポーネントは無効になります。ケーブルのいずれかの終端のノード名およびアダプタ名の両方を入力する必要があります。

- 6 ケーブルが無効になっていることを確認します。

```
# clinterconnect show node:adapter,adapternode
```

### 例 7-5 クラスタトランスポートケーブルを無効にする

次の例に、ノード phys-schost-2 にあるアダプタ net0 のクラスタトランスポートケーブルを無効にする方法を示します。

```
[Become superuser on any node.]
[Start the clsetup utility:]
# clsetup
[Select Cluster interconnect>Disable a transport cable.]

[Answer the questions when prompted.]
[You will need the following information.]
 [ You Will Need:]
Information:          Example:
node names            phys-schost-2
adapter names         net0
switch names         hub1
[Verify that the clinterconnect
command was completed successfully:]
Command completed successfully.
[Quit the clsetup Cluster Interconnect Menu and Main Menu.]
[Verify that the cable is disabled:]
# 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の追加と削除を行うには、トランスポートアダプタのインスタンス番号を確認する必要があります。アダプタ名は、アダプタの種類とアダプタのインスタンス番号を組み合わせたものです。

- 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,400/pci11c8,0@2" 0 "ttt"
"/xyz@1f,4000.pci11c8,0@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。

始める前に スーパーユーザーのリモートシェル (**rsh(1M)**) または Secure Shell (**ssh(1)**) アクセスが、すべてのクラスタノードで有効になっていることを確認します。

- 1 各クラスタノード上で次のサブステップを実行することで、すべてのクラスタノードをリブートし、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上で、スーパーユーザーになるか、RBACの承認 **solaris.cluster.admin** を提供する役割になります。

- b. **clnode evacuate** および **cluster shutdown** コマンドを使用してノードを停止します。

**clnode evacuate** コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定ノード上の投票ノードから、次に優先される投票ノードへ、すべてのリソースグループを切り替えます。

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

- 2 1つのノードから、**clsetup** ユーティリティを起動します。

非クラスタモードで動作している場合、**clsetup** ユーティリティは非クラスタモード動作のメインメニューを表示します。

- 3 「クラスタトランスポート用のネットワークアドレス指定と範囲の変更」メニュー項目を選択します。  
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**ユーティリティーは、ユーザーが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**ユーティリティーは2つの推奨ネットマスクを計算します。
    - 第一のネットマスクは、ユーザーが指定したノード、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**ユーティリティーの質問に対しては、「yes」と入力します。
- 8 完了後**clsetup**ユーティリティーを終了します。
- 9 各クラスタノードに対して次のサブステップを実行することで、各クラスタノードをリポートし、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

# パブリックネットワークの管理

Oracle Solaris Cluster ソフトウェアは、Oracle Solaris ソフトウェアに実装されている、パブリックネットワーク用の Internet Protocol network Multipathing (IPMP) をサポートします。IPMP の基本的な管理は、クラスタ環境でも非クラスタ環境でも同じです。マルチパスは 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 を管理する前にガイドラインを確認してください。

## 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。

- 各パブリックネットワークアダプタは、IPMP グループに属している必要があります。
- `local-mac-address?` 変数には、Ethernet アダプタの値として `true` が指定されていなければなりません。
- 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、優れた保護を提供しますが、可用性が損なわれる場合もあります。
- 次に示すタイプのマルチパスグループ内に存在するアダプタごとにテスト IP アドレスを構成する必要があります。
  - すべての多重アダプタマルチパスグループで、テスト IP アドレスが必要です。単一アダプタマルチパスグループでは、テスト IP アドレスは必要ありません。
- 同一マルチパスグループ内のすべてのアダプタ用のテスト IP アドレスは、単一の IP サブネットに属する必要があります。
- テスト IP アドレスは高可用性でないため、通常のアプリケーションが使用しないようにします。
- マルチパスグループの命名に制限はありません。しかし、リソースグループを構成するとき、`netiflist` には、任意のマルチパス名にノード ID 番号またはノード名が続くものを指定します。たとえば、マルチパスグループの名前が `sc_ipmp0` であるとき、ノード ID が 1 である `phys-schost-1` というノード上にアダプタが存在する場合、`netiflist` には `sc_ipmp0@1` または `sc_ipmp0@phys-schost-1` のどちらかを指定してもかまいません。

- `if_mpadm` コマンドを使用して、削除するアダプタからグループ内の代替アダプタに IP アドレスを最初に切り替えてから、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`) または停止してください。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監視中の場合でもアダプタに対して行うことができます。
- クラスタ内の各ノードについて、最低 1 つのパブリックネットワーク接続を維持しなければなりません。クラスタは、パブリックネットワーク接続がないとアクセスできません。
- クラスタ上の IP ネットワークマルチパスグループのステータスを表示するには、`clinterconnect status` コマンドを使用します。

IP ネットワークマルチパスの詳細は、Oracle Solaris OS システム管理ドキュメントセットの該当するドキュメントを参照してください。

表 7-3 タスクリスト:パブリックネットワークの管理

Oracle Solaris OS のリリース	手順
Oracle Solaris 11 OS	『Oracle Solaris Administration: Network Interfaces and Network Virtualization』の第 15 章「Administering IPMP」

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 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。

## パブリックネットワークインタフェースでの動的再構成

クラスタ内のパブリックネットワークインタフェース上で動的再構成 (DR) を実行するときには、いくつかの問題を考える必要があります。

- Oracle Solaris の DR 機能の説明で述べられているすべての必要条件、手順、制限は、Oracle Solaris Cluster の DR サポートにも適用されます (オペレーティングシステムでの休止操作は除く)。したがって、Oracle Solaris Cluster ソフトウェアで DR 機能を使用する前に、Oracle Solaris の DR 機能についての説明を参照してください。特に、DR の切り離し操作中に、ネットワークに接続されていない入出力デバイスに影響する問題について確認してください。



- DR ボード削除操作は、パブリックネットワークインタフェースがアクティブでないときだけ成功します。アクティブなパブリックネットワークインタフェースを削除する前に、`if_mpadm` コマンドを使用して、マルチバ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。
- アクティブなネットワークインタフェースを適切に無効にせずにパブリックネットワークインタフェースカードを削除しようとした場合、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つのアダプタを持つマルチバスグループの場合、無効にしたネットワークアダプタ上で DR 削除操作を実行している間に残りのネットワークアダプタに障害が発生すると、可用性に影響が生じます。これは、DR 操作の間は、残りのネットワークアダプタのフェイルオーバー先が存在しないためです。

パブリックネットワークインタフェース上で DR 操作を実行するときは、次の手順をその順番どおりに行います。

表 7-4 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、マルチバ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	<a href="#">if_mpadm(1M)</a> のマニュアルページ 『Oracle Solaris Administration: Network Interfaces and Network Virtualization』の「How to Move an Interface From One IPMP Group to Another Group」
2. <code>ipadm</code> コマンドを使用して、マルチバスグループからアダプタを削除	<a href="#">ipadm(1M)</a> のマニュアルページ 『Oracle Solaris Administration: Network Interfaces and Network Virtualization』の「How to Add or Remove IP Addresses」
3. パブリックネットワークインタフェース上で DR 操作を実行	



## ノードの追加と削除

---

この章では、クラスタへのノードの追加とノードの削除方法を説明します。

- 179 ページの「クラスタへのノードの追加」
- 182 ページの「クラスタからのノードの削除」

クラスタのメンテナンスに関する情報は、[第9章「クラスタの管理」](#)を参照してください。

### クラスタへのノードの追加

このセクションでは、グローバルクラスタまたはゾーンクラスタにノードを追加する方法を説明します。新しいゾーンクラスタノードは、そのゾーンクラスタをホストするグローバルクラスタのノード上に作成できますが、それは、グローバルクラスタノードが、そのゾーンクラスタのノードをまだホストしていない場合に限られます。

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 - 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せん。
- 

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 8-1 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』
認証ノードリストへのノードの追加	<code>/usr/cluster/bin/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 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	179 ページの「クラスタへのノードの追加」
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 Geographic Edition System Administration Guide』の「How to Add a New Node to a Cluster in a Partnership」

## ▼ 既存のクラスタにノードを追加する方法

Oracle Solaris ホストまたは仮想マシンを既存のグローバルクラスタまたはゾーンクラスタに追加する前に、プライベートクラスタインターコネクトへの運用面の物理接続を含む、必要なハードウェアすべてがノードに正しく取り付けられ、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スーパーユーザーになります。次の手順は、グローバルクラスタのノードから実行します。
- 2 表 8-1 のタスクマップに記載されている必要なハードウェアのインストールと構成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。  
scinstall ユーティリティを使用して、新しいノードのインストールと構成を完了します。詳細は、『Oracle Solaris Cluster ソフトウェアのインストール』を参照してください。
- 4 新しいノードで **scinstall** ユーティリティを使用して、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 ソフトウェアのインストール』の「ゾーンクラスタの構成」を参照してください。

- 6 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。  

```
# clzc install zone-cluster
```
- 7 新しいマシンがクラスタに追加されないようにするために、新しいマシンを追加する要求を無視するようクラスタに指示するオプションの番号を **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。

**8 clsetup ユーティリティを終了します。**

**例 8-1 認証ノードリストへのグローバルクラスタノードの追加**

次に、ノード phys-schost-3 を既存のクラスタの認証ノードリストに追加する方法を示します。

```
[Become superuser and execute the clsetup utility.]
phys-schost# clsetup
[Select New nodes>Specify the name of a machine which may add itself.]
[Answer the questions when prompted.]
[Verify that the command completed successfully.]

claccess allow -h phys-schost-3

      Command completed successfully.
[Select Prevent any new machines from being added to the cluster.]
[Quit the clsetup New Nodes Menu and Main Menu.]
[Install the cluster software.]
```

**参照** [clsetup\(1CL\)](#)

クラスタノードを追加するタスクの一連の手順については、[表 8-1](#)、「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方法については、『[Oracle Solaris Cluster Data Services Planning and Administration Guide](#)』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 8-2 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。	<pre>clnode evacuate node</pre> <p>184 ページの「ゾーンクラスタからノードを削除する方法」</p>
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。	<pre>claccess show</pre> <pre>claccess allow -h node-to-remove</pre>
ノードを削除できない場合に、クラスタ構成へのアクセス権をノードに付与する。	
すべてのデバイスグループからノードを削除する。	97 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 ボリュームマネージャー)」
削除するノードに接続されているすべての定足数デバイスを削除する。	<p>2 ノードクラスタのノードを削除する場合、この手順は省略可能です。</p> <p>140 ページの「定足数デバイスを削除する方法」</p> <p>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。</p> <p>142 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」</p>
削除するノードを非クラスタモードにする。	204 ページの「ノードを保守状態にする」
ゾーンクラスタからノードを削除する。	184 ページの「ゾーンクラスタからノードを削除する方法」

表 8-2 タスクマップ: ノードの削除 (続き)	
タスク	手順
クラスタソフトウェア構成からノードを削除する。	185 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」
(省略可能) 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。	208 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」

## ▼ ゾーンクラスタからノードを削除する方法

ゾーンクラスタからノードを削除するには、ノードを停止してアンインストールし、構成からそのノードを削除します。あとでノードをゾーンクラスタに戻す場合は、表 8-1 の手順に従います。ここからの手順のほとんどは、グローバルクラスタノードから実行します。

- 1 グローバルクラスタのノードのスーパーユーザーになります。
- 2 ノードとそのゾーンクラスタを指定して、削除するゾーンクラスタノードを停止します。

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

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

- 3 ゾーンクラスタ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# clrg remove-node -n zonehostname -Z zoneclustername rg-name
```

- 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 zoneclustername
```

- 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 zoneclustername
```

```
clzc:sczone> remove node physical-host=zoneclusternodename
```

```
clzc:sczone> exit
```

- 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、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。  
グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 削除するグローバルクラスタノードを非クラスタモードでブートします。  
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、[184 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 ベースのブートについては、『[Booting and Shutting Down Oracle Solaris on x86 Platforms](#)』の「[Booting an x86 Based System Interactively](#)」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** 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。

---

注-削除するノードが使用できない場合や、ブートできなくなっている場合は、アクティブな任意のクラスタノードで `clnode clear -F <node-to-be-removed>` コマンドを実行します。`clnode status <nodename>` を実行して、ノードが削除されていることを確認します。

---

#### 4 クラスタからノードを削除します。

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。

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-クラスタ内の最後のノードを削除する場合は、そのノードがクラスタモードモードでないこと、およびクラスタ内にアクティブなノードがないことが必要です。

---

#### 5 別のクラスタノードから、ノードの削除を確認します。

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

#### 6 ノードの削除を完了します。

- 削除するノードから **Oracle Solaris Cluster** ソフトウェアをアンインストールする場合は、[208 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」](#)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しない場合は、『Oracle Solaris Cluster Hardware Administration Manual』で説明されているように、ハードウェア接続を削除することにより、クラスタからノードを物理的に削除できます。

## 例 8-2 クラスタソフトウェア構成からのノードの削除

次に、ノード `phys-schost-2` をクラスタから削除する方法を示します。`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。

```
[Remove the node from the cluster:]
phys-schost-2# clnode remove
phys-schost-1# clnode clear -F phys-schost-2
[Verify node removal:]
phys-schost-1# clnode status
-- Cluster Nodes --

```

	Node name	Status
	-----	-----
Cluster node:	phys-schost-1	Online

参照 削除するノードから 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、208 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」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概要については、表 8-2 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、180 ページの「既存の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、動作している Oracle RAC データベースのインスタンスを停止します。手順については、『Oracle Database Administration Guide』を参照してください。

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

`clnode evacuate` コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、指定ノード上の投票ノードから、次に優先される投票ノードへ、すべてのリソースグループを切り替えます。

- 4 デバイスグループを保守状態にします。  
デバイスグループを保守状態にする手順については、204 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 **HASStoragePlus** 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
node    ノードの名前。
```

リソースグループのノードリストを変更する方法については、『Oracle Solaris Cluster Data Services Planning and Administration Guide』を参照してください。

---

注 - `clresourcegroup` を実行するときには、リソースタイプ、リソースグループ、およびリソースのプロパティ名には大文字と小文字の区別があります。

---

- 7 削除する予定のストレージレイがノードに接続されている最後のストレージレイである場合、当該ストレージレイに接続されているハブまたはスイッチとノードの間にある光ケーブルを取り外します (そうでない場合、この手順は省略します)。
- 8 切断するノードからホストアダプタを削除する場合、ノードの電源を切ります。切断するノードからホストアダプタを削除する場合は、[手順 11](#)に進みます。
- 9 ノードからホストアダプタを削除します。  
ホストアダプタの削除手順については、ノード用ドキュメントを参照してください。
- 10 ブートが行われないようにして、ノードに電源を入れます。
- 11 **Oracle RAC** ソフトウェアがインストールされている場合、切断する予定のノードからそのパッケージを削除します。

```
phys-schost# 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
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、[206 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結合を試みます。  
この手順は、グローバルクラスタ上のみで実行します。  
`phys-schost# boot`
- 2 ノードがクラスタに正常に再結合されているかどうかを確認します。
  - 再結合されていない場合は、[手順b](#)に進みます。
  - 再結合されている場合は、次の各手順を行なってノードをデバイスグループから削除します。
    - a. ノードが正常にクラスタに再結合された場合は、残っているデバイスグループからノードを削除します。  
[96](#)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、[208](#)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」に戻り、その手順を繰り返します。
- 3 ノードがクラスタに再結合されなかった場合は、ノードの `/etc/cluster/ccr` ファイルを他の名前に変更します (たとえば、`ccr.old`)。  
`# mv /etc/cluster/ccr /etc/cluster/ccr.old`
- 4 [208](#)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」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。

- 191 ページの「クラスタの管理の概要」
- 222 ページの「ゾーンクラスタ管理タスクの実行」
- 228 ページの「トラブルシューティング」

クラスタへのノードの追加または削除に関する詳細は、第8章「ノードの追加と削除」を参照してください。

### クラスタの管理の概要

このセクションでは、グローバルクラスタやゾーンクラスタ全体の管理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結合すると、構成の変更点が Oracle Solaris Cluster によって再現されます。

---

注-デフォルトでは、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ませんが、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 9-1 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「ノードの追加と削除」
クラスタ名を変更	192 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	194 ページの「ノード ID をノード名にマップする方法」
クラスタへの新しいノードの追加を許可または拒否	194 ページの「新しいクラスタノード認証で作業する方法」
NTP を使用して、クラスタの時刻を変更する	196 ページの「クラスタの時刻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	198 ページの「SPARC: 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	199 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	204 ページの「ノードを保守状態にする」
ノード名の変更	202 ページの「ノード名を変更する」
クラスタノードを保守状態から復帰	206 ページの「ノードを保守状態から戻す」
クラスタノードからソフトウェアをアンインストール	208 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」
SNMP Event MIB の追加および管理	213 ページの「SNMP イベント MIB を有効にする」 216 ページの「SNMP ユーザーをノードに追加する」
各ノードの負荷制限の構成	219 ページの「ノードに負荷制限を設定する」
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	222 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。

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にある任意のノード上でスーパーユーザーになります。
- 2 **8**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 9-1 クラスタ名の変更

次の例に、新しいクラスタ名 `dromedary` へ変更するために、`clsetup` 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、最初にクラスタに加わったときの順番でノードに割り当てられます。ノード ID 番号が割り当てられたあとでは、番号は変更できません。ノード ID 番号は、通常、エラーメッセージが発生したクラスタノードを識別するために、エラーメッセージで使用されます。この手順を使用し、ノード ID とノード名間のマッピングを判別します。

グローバルクラスタまたはゾーンクラスタ用の構成情報を表示するために、スーパーユーザーになる必要はありません。グローバルクラスタのノードから、このプロシージャの 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\(ICL\)](#) のマニュアルページを参照してください。

- 2 1つのゾーンクラスタに対して、複数のノード ID を一覧表示することも可能です。ゾーンクラスタノードは、実行中のグローバルクラスタノードと同じノード ID を持っています。

```
phys-schost# zlogin sczone clnode -v | grep Node
```

### 例 9-2 ノード名のノード ID へのマップ

次の例は、グローバルクラスタに対するノード ID の割り当てを示しています。

```
phys-schost# clnode show | grep Node
=== Cluster Nodes ===
Node Name:          phys-schost1
Node ID:            1
Node Name:          phys-schost2
Node ID:            2
Node Name:          phys-schost3
Node ID:            3
```

## ▼ 新しいクラスタノード認証で作業する方法

Oracle Solaris Cluster では、新しいノードをグローバルクラスタに追加できるようにするかどうかと、使用する認証の種類を指定できます。パブリックネットワーク上のクラスタに加わる新しいノードを許可したり、新しいノードがクラスタに加わることを拒否したり、クラスタに加わるノードを特定できます。新しいノードは、標準 UNIX または Diffie-Hellman (DES) 認証を使用し、認証することができます。DES

認証を使用して認証する場合、ノードが加わるには、すべての必要な暗号化鍵を構成する必要があります。詳細は、[keyserv\(1M\)](#)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にある任意のノード上でスーパーユーザーになります。
- 2 `clsetup` ユーティリティーを起動します。  
`phys-schost# clsetup`  
メインメニューが表示されます。
- 3 クラスタ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 9-3 新しいマシンがグローバルクラスタに追加されないようにする

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、新しいマシンがクラスタに追加されないようにする `claccess` コマンドを示しています。

```
phys-schost# claccess deny -h hostname
```

#### 例 9-4 すべての新しいマシンがグローバルクラスタに追加されることを許可する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、すべての新しいマシンをクラスタに追加できるようにする `claccess` コマンドを示しています。

```
phys-schost# claccess allow-all
```

#### 例 9-5 グローバルクラスタに追加される新しいマシンを指定する

`clsetup` ユーティリティーにより、`claccess` コマンドを生成します。次の例は、1台の新しいマシンをクラスタに追加できるようにする `claccess` コマンドを示しています。

```
phys-schost# claccess allow -h hostname
```

### 例 9-6 認証を標準 UNIX に設定する

clsetup ユーティリティーにより、claccess コマンドを生成します。次の例は、クラスタに参加している新規ノードの標準 UNIX 認証に対し、リセットを行う claccess コマンドを示しています。

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

### 例 9-7 認証を DES に設定する

clsetup ユーティリティーにより、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加わるには、すべての必要な暗号化鍵を構成します。詳細は、[keyserv\(1M\)](#) と [publickey\(4\)](#) のマニュアルページを参照してください。

## ▼ クラスタの時刻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、クラスタノード間で時刻を同期させています。グローバルクラスタの時刻の調整は、ノードが時刻を同期するときに、必要に応じて自動的に行われます。詳細は、『[Oracle Solaris Cluster Concepts Guide](#)』および <http://download.oracle.com/docs/cd/E19065-01/servers.10k/> の『[Network Time Protocol's User's Guide](#)』を参照してください。



注意 - NTP を使用する場合、クラスタの稼動中はクラスタの時刻を調整しないでください。date、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、時刻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にある任意のノード上でスーパーユーザーになります。

- 2 グローバルクラスタを停止します。  

```
phys-schost# cluster shutdown -g0 -y -i 0
```
- 3 SPARC ベースのシステムではノードが **ok** プロンプトを表示し、x86 ベースのシステムでは **GRUB** メニューで「**Press any key to continue**」というメッセージが表示されていることを確認します。
- 4 非クラスタモードでノードをブートします。
  - 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 ベースのブートについては、『[Booting and Shutting Down Oracle Solaris on x86 Platforms](#)』の「[Booting an x86 Based System Interactively](#)」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** と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-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刻を設定します。

```
phys-schost# date HHMM.SS
```

- 6 ほかのマシンで、**rdate(1M)** コマンドを実行し、時刻をこのノードに同期化します。

```
phys-schost# rdate hostname
```

- 7 各ノードを起動し、クラスタを再起動します。

```
phys-schost# reboot
```

- 8 すべてのクラスタノードで変更が行われたことを確認します。

各ノードで、`date` コマンドを実行します。

```
phys-schost# date
```

## ▼ SPARC: ノードで OpenBoot PROM (OBP) を表示する方法

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** コマンドを使用してから、**shutdown** コマンドを使用することで、クラスタノードを正常に停止します。

**clnode evacuate** コマンドは、すべてのデバイスグループを、指定ノードから次に優先されるノードに切り替えます。またこのコマンドは、グローバルクラスタの指定した投票ノードから、次に優先される投票ノードへ、すべてのリソースグループを切り替えます。

```
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、次の手順に従います。

デフォルトのプライベートホスト名は、クラスタの初期インストール時に割り当てられます。デフォルトのプライベートホスト名の形式は、**clusternode<nodeid>-priv** です (**clusternode3-priv** など)。名前がすでにドメイン内で使用されている場合にかぎり、プライベートホスト名を変更します。



注意 - 新しいプライベートホスト名には IP アドレスを割り当てないでください。クラスタソフトウェアが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 Data Services Planning and Administration Guide](#)』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- 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 pred1 です。次の clnode コマンドを実行したら、[手順 6](#)に進みます。

```
phys-schost# /usr/cluster/bin/clnode set -p privatehostname=New-private-nodename pred1
```

- 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 (clusternode<nodeid> -priv) および新しいプライベートホスト名を入力してください。

- 6 ネームサービスキャッシュをフラッシュします。

クラスタの各ノードで次の手順を実行します。フラッシュすることによって、クラスタアプリケーションとデータサービスが古いプライベートホスト名にアクセスしないようにします。

```
phys-schost# nscd -i hosts
```



- 7 **NTP** 構成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トがある場合は、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 **手順 1** で無効にしたデータサービスリソースとそのほかのアプリケーションをすべて有効にします。  
phys-schost# `clresource 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 Data Services Planning and Administration Guide](#)』を参照してください。

### 例 9-8 プライベートホスト名を変更する

次に、ノード `phys-schost-2` 上のプライベートホスト名 `clusternode2-priv` を `clusternode4-priv` に変更する例を示します。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# vi /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、スーパーユーザーになるか、`solaris.cluster.modify RBAC`の承認を提供する役割になります。

- 2 Oracle Solaris 構成のパートナーシップにある Oracle Solaris Cluster Geographic Edition クラスタのノードの名前を変更する場合は、追加の手順を行う必要があります。

Geographic Edition クラスタおよびノードの詳細は、『Oracle Solaris Cluster Geographic Edition System Administration Guide』の第5章「Administering Cluster Partnerships」を参照してください。

名前変更手順を行なっているクラスタが保護グループのプライマリクラスタで、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、名前変更手順を行なっている間、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。

- 3 『Oracle Solaris Administration: Common Tasks』の「How to Change a System's Identity (nodename)」の手順を完了して(ただし、手順の最後にあるリポートは実行しない)、Oracle Solaris のホスト名を変更します。

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。

一度に1つのノード名を変更します。

```
# clnode rename -n newnodename oldnodename
```

- 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再起動します。  
# `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 **Geographic Edition** クラスタノード上で名前を変更していて、名前が変更されたノードを含むクラスタのパートナークラスタが引き続き以前のノード名を参照している場合、保護グループの同期状態はエラーとして表示されます。  
geopg update <pg> を使用して、名前が変更されたノードが含まれているパートナークラスタのいずれかのノードから保護グループを更新する必要があります。その手順の完了後、geopg start -e global <pg> コマンドを実行します。あとで、名前が変更されたノードのクラスタに保護グループを再び切り替えることができます。
- 11 論理ホスト名リソースの `hostnameList` プロパティーを変更することを選択できます。  
この省略可能な手順については、203 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

202 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変更したあとに、論理ホスト名リソースの `hostnameList` プロパティーを変更することもできます。この手順は省略可能です。

- 1 グローバルクラスタでは、スーパーユーザーになるか、`solaris.cluster.modify RBAC` の承認を提供する役割になります。
- 2 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。  
# `clrg offline apache-rg`

- b. Apache 論理ホスト名リソースを無効にします。  

```
# clrs disable apache-lh-res
```
- c. 新しいホスト名リストを指定します。  

```
# clrs 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。  

```
# clrs enable apache-lh-res
```
- f. Apache リソースグループをオンラインにします。  

```
# clrg online -emM 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。  

```
# clrs status apache-rs
```

## ▼ ノードを保守状態にする

サービスからグローバルクラスタノードを長時間外す場合は、そのノードを保守状態にします。保守状態のノードは、サービス対象中に定足数確立の投票に参加しません。ノードを保守状態にするには、`clnode evacuate` および `cluster shutdown` コマンドを使用して、ノードをシャットダウンしておく必要があります。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- ノードを1つだけ停止する場合は、Oracle Solaris の `shutdown` コマンドを使用します。クラスタ全体を停止する場合にだけ、`cluster shutdown` コマンドを使用します。

---

クラスタノードが停止されて保守状態になると、そのノードのポートで構成されるすべての定足数デバイスの、定足数投票数 (quorum vote count) が1つ減ります。このノードが保守状態から移動してオンラインに戻されると、ノードおよび定足数デバイスの投票数は1つ増えます。

`clquorum disable` コマンドを使用して、クラスタノードを保守状態にします。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、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。
- 2 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とデバイスグループを切り替えます。  
`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 -i 0`
- 4 クラスタ内の別のノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になり、手順3で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`  
`node` 保守モードにするノードの名前を指定します。
- 5 グローバルクラスタノードが保守状態にあることを確認します。  
`phys-schost# clquorum status node`  
 保守状態にしたノードの Status は `offline` で、その Present と Possible の定足数投票数は 0 (ゼロ) である必要があります。

### 例 9-9 グローバルクラスタノードを保守状態にする

次に、クラスタノードを保守状態にして、その結果を確認する例を示します。`clnode status` の出力では、`phys-schost-1` のノードの Node votes は 0 (ゼロ) で、その状態は `Offline` です。Quorum Summary では、投票数も減っているはずですが、構成によって異なりますが、Quorum Votes by Device の出力では、いくつかの定足数ディスクデバイスもオフラインである可能性があります。

```
[On the node to be put into maintenance state:]
phys-schost-1# clnode evacuate phys-schost-1
phys-schost-1# shutdown -g0 -y -i0
```

```
[On another node in the cluster:]
phys-schost-2# clquorum disable phys-schost-1
phys-schost-2# clquorum status phys-schost-1
```

```
-- Quorum Votes by Node --
```

Node Name	Present	Possible	Status
-----------	---------	----------	--------

-----	-----	-----	-----
phys-schost-1	0	0	Offline
phys-schost-2	1	1	Online
phys-schost-3	1	1	Online

参照 ノードをオンライン状態に戻す方法については、206 ページの「ノードを保守状態から戻す」を参照してください。

## ▼ ノードを保守状態から戻す

次の手順を使用して、グローバルクラスタノードをオンラインに戻し、定足数投票数をリセットしてデフォルト設定に戻します。クラスタノードのデフォルトの投票数は1です。定足数デバイスのデフォルトの投票数は $N-1$ です。 $N$ は、投票数が0以外で、定足数デバイスが構成されているポートを持つノードの数を示します。

ノードが保守状態になると、そのノードの投票数は1つ減ります。また、このノードのポートに構成されているすべての定足数デバイスの投票数も(1つ)減ります。投票数がリセットされ、ノードが保守状態から戻されると、ノードの投票数と定足数デバイスの投票数の両方が1つ増えます。

保守状態にしたグローバルクラスタノードを保守状態から戻した場合は、必ずこの手順を実行してください。



注意 -globaldev または node オプションのどちらも指定しない場合、定足数投票数はクラスタ全体でリセットされます。

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

この手順では、長形式の 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、保守状態のノード以外の任意のノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。
- 2 グローバルクラスタ構成内にあるノードの数に応じて、次の手順のいずれかを実行します。
  - クラスタ構成内に2つのノードがある場合は、手順4に進みます。
  - クラスタ構成内に3つ以上のノードがある場合は、手順3に進みます。

- 3 保守状態から解除するノードに定足数デバイスがある場合は、保守状態にあるノード以外のノードからクラスタ定足数のカウントをリセットします。  
保守状態ではないノードの定足数投票数をリセットするのは、そのノードを再起動する前である必要があります。そうしないと、定足数の確立を待機中にハングアップすることがあります。

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

```
reset          定足数をリセットする変更フラグです。
```

- 4 保守状態を解除するノードを起動します。

- 5 定足数投票数を確認します。

```
phys-schost# clquorum status
```

保守状態を解除したノードの状態は `online` であり、`Present` と `Possible` の定足数投票数は適切な値である必要があります。

#### 例 9-10 クラスタノードの保守状態を解除して、定足数投票数をリセットする

次に、クラスタノードの定足数投票数をリセットして、その定足数デバイスをデフォルトに戻し、その結果を確認する例を示します。`cluster status` の出力では、`phys-schost-1` の `Node votes` は 1 で、その状態は `online` です。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/rdsk/d3s2	1	1	Online
/dev/did/rdsk/d17s2	0	1	Online
/dev/did/rdsk/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

完全に確立されているクラスタ構成からグローバルクラスタノードを切断する前に、グローバルクラスタノードから Oracle Solaris Cluster ソフトウェアを構成解除するには、この手順を行います。この手順では、クラスタに存在する最後のノードからソフトウェアをアンインストールできます。

---

注-クラスタにまだ結合されていない、あるいはまだインストールモードであるノードから Oracle Solaris Cluster ソフトウェアをアンインストールする場合、この手順を使用してはいけません。その代わりに、『Oracle Solaris Cluster 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 作業マップにあるクラスタノードを削除するための前提作業がすべて完了していることを確認します。

表 8-2 を参照してください。

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- 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スーパーユーザーになります。
- 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。ただし、`/global/.devices` グローバルマウントを除きます。
- 5 ノードを再起動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 ベースのブートについては、『[Booting and Shutting Down Oracle Solaris on x86 Platforms](#)』の「[Booting an x86 Based System to a Specified State \(Task Map\)](#)」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起動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起動します。

---

注-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起動するには、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]
```

- |  |   |
|--|---|
| <code>-r</code>                            | クラスタノードから、クラスタの構成情報を削除し、 <b>Oracle Solaris Cluster</b> 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。 |
| <code>-b <i>bootenvironmentname</i></code> | 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。名前の指定は省略可能です。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。   |

詳細については、[scinstall\(IM\)](#) 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むりの場合は、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しない場合は、ほかのクラスタデバイスからトランスポートケーブルとトランスポートスイッチを切断します (存在する場合)。
  - a. アンインストールしたノードが、並列 **SCSI** インタフェースを使用する記憶装置デバイスに接続されている場合は、トランスポートケーブルを切り離れた後で、この記憶装置デバイスのオープン **SCSI** コネクタに **SCSI** ターミネータを取り付ける必要があります。

アンインストールしたノードが、**Fibre Channel** インタフェースを使用する記憶装置デバイスに接続されている場合は、終端処理は必要ありません。
  - b. 切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` への移行については、[88 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トリ

次のエラーメッセージは、削除したグローバルクラスタノードに、`vfstab` ファイルから参照されているクラスタファイルシステムがまだあることを示しています。

```
Verifying that no unexpected global mounts remain in /etc/vfstab ... failed
clnode: global-mount1 is still configured as a global mount.
clnode: global-mount1 is still configured as a global mount.
clnode: /global/dgl 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、[208 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」](#)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、この[手順 4](#) が正しく行われているか確認してください。

## デバイスグループに削除されていないリストがある場合

次のエラーメッセージは、削除したノードが依然としてデバイスグループにリストされていることを示しています。

```
Verifying that no device services still reference this node ... failed
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 `clsnmphot` コマンドによって定義されているすべてのホストに自動的に送信します。MIB には、最新の 50 イベントの読み取り専用のテーブルが保持されます。クラスタは多数の通知を生成するので、重要度が `warning` 以上のイベントだけがトラップ通知として送信されます。この情報は、レポートが実行されると消失します。

SNMP イベント MIB は、`sun-cluster-event-mib.mib` ファイルで定義されており、`/usr/cluster/lib/mib` ディレクトリにあります。この定義を使用して、SNMP トラップ情報を解釈できます。

イベント SNMP モジュールのデフォルトのポート番号は 11161 で、SNMP トラップのデフォルトのポートは 11162 です。これらのポート番号は、共通エージェントコンテナのプロパティファイル (`/etc/cacao/instances/default/private/cacao.properties`) を変更することによって変更できます。

Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理には、次の作業が含まれます。

表 9-2 作業マップ: 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	213 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	213 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	214 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	215 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	216 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	216 ページの「SNMP ユーザーをノードに追加する」
SNMP ユーザーの削除	217 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 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 SNMP イベント MIB を有効にします。

```
phys-schost-1# clsnmpmib enable [-n node] MIB
```

`[-n node]`      有効にするイベント MIB がある `node` を指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`MIB`            有効にする MIB の名前を指定します。この場合、MIB 名は `event` にしてください。

## ▼ SNMP イベント MIB を無効にする

この手順では、SNMP イベント MIB を無効化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 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 **SNMP イベント MIB** を無効にします。

```
phys-schost-1# clsnmpmib disable -n node MIB
```

`-n node` 無効にするイベント MIB がある `node` を指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`MIB` 無効にする MIB の種類を指定します。この場合、`event` を指定してください。

## ▼ **SNMP イベント MIB** を変更する

この手順では、SNMP イベント MIB のプロトコルを変更する方法を説明します。

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

この手順では、長形式の 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 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 **SNMP イベント MIB** のプロトコルを変更します。

```
phys-schost-1# clsnmpmib set -n node -p version=value MIB
```

`-n node`

変更するイベント MIB がある `node` を指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`-p version=value`

MIB で使用する SNMP プロトコルのバージョンを指定します。 `value` は次のように指定します。

- `version=SNMPv2`
- `version=snmpv2`
- `version=2`
- `version=SNMPv3`

- version=snmpv3
- version=3

#### MIB

サブコマンドが適用される単数または複数の MIB の名前を指定します。この場合、`event` を指定してください。このオペランドを指定しない場合は、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。

### ▼ SNMP ホストがノード上の SNMP トラップを受信できるようにする

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

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

この手順では、長形式の 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 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 ホストを、別のノード上のコミュニティの SNMP ホストリストに追加します。

```
phys-schost-1# clnsmphost add -c SNMPcommunity [-n node] host
```

`-c SNMPcommunity`

ホスト名とともに使用される SNMP コミュニティ名を指定します。

ホストを `public` 以外のコミュニティに追加する場合は、コミュニティ名 `SNMPcommunity` を指定してください。add サブコマンドを `-c` オプションなしで使用すると、このサブコマンドは `public` をデフォルトのコミュニティ名として使用します。

指定されたコミュニティ名が存在しない場合、このコマンドはそのコミュニティを作成します。

`-n node`

クラスタ内の SNMP MIB に対するアクセス権を付与されている SNMP ホストの `node` の名前を指定します。ノード名またはノード ID を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`host`

クラスタ内の SNMP MIB に対するアクセス権が付与されたホストの名前、IP アドレス、または IPv6 アドレスを指定します。

## ▼ SNMP ホストがノード上の SNMP トラップを受信できないようにする

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

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

この手順では、長形式の 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 スーパーユーザーになるか、RBAC 承認 `solaris.cluster.modify` を提供する役割になります。
- 2 指定のノード上のコミュニティの SNMP ホストリストからホストを削除します。

```
phys-schost-1# clsnmphost remove -c SNMPcommunity -n node host
```

`remove`

指定のノードから指定の SNMP ホストを削除します。

`-c SNMPcommunity`

SNMP ホストを削除する SNMP コミュニティの名前を指定します。

`-n node`

構成から削除される SNMP ホストの `node` の名前を指定します。ノード名またはノード ID を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`host`

構成から削除されるホストの名前、IP アドレス、または IPv6 アドレスを指定します。

指定の SNMP コミュニティ内のすべてのホストを削除するには、`-c` オプション付きの `host` に正符号 (+) を使用します。すべてのホストを削除するには、`host` に正符号 + を使用します。

## ▼ SNMP ユーザーをノードに追加する

この手順では、ノード上の SNMP ユーザー構成に SNMP ユーザーを追加する方法を説明します。

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

この手順では、長形式の 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 スーパーユーザーになるか、RBAC承認 `solaris.cluster.modify` を提供する役割になります。
- 2 **SNMP ユーザー**を追加します。

```
phys-schost-1# 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す。このオプションを指定しないと、デフォルトで現在のノードが使用されます。
- a *authentication* ユーザーの承認に使用する認証プロトコルを指定します。認証プロトコルの値は SHA または MD5 です。
- f *password* SNMP ユーザーパスワードを含むファイルを指定します。新しいユーザーを作成する際にこのオプションを指定しないと、コマンドはパスワードを求めるプロンプトを表示します。このオプションは、add サブコマンドとだけ有効です。

ユーザーパスワードは、次の形式で、独立した行の上に指定します。

```
user:password
```

パスワードには次に示す文字または空白文字を含めることはできません。

- ;(セミコロン)
- :(コロン)
- \ (バックスラッシュ)
- \n (復帰改行)

*user* 追加する SNMP ユーザーの名前を指定します。

## ▼ **SNMP ユーザーをノードから削除する**

この手順では、ノード上の SNMP ユーザー構成から SNMP ユーザーを削除する方法を説明します。

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

この手順では、長形式の 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 スーパーユーザーになるか、RBAC承認 `solaris.cluster.modify` を提供する役割になります。

## 2 SNMP ユーザーを削除します。

```
phys-schost-1# clsnmpuser delete -n node user
```

`-n node` SNMP ユーザーが削除されるノードを指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`user` 削除する SNMP ユーザーの名前を指定します。

## 負荷制限の設定

負荷制限を設定することによって、ノードまたはゾーンのリソースグループの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。各ノードの強い負荷制限値と弱い負荷制限値の作成と変更には、`clnode create-loadlimit`、`clnode set-loadlimit`、および `clnode delete-loadlimit` コマンドを使用します。詳細については、[clnode\(ICL\)](#) 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設定すると、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リソースグループが優先度の高いリソース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` プロパティのデフォルト値は、`FALSE` です。

---

注- 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設定できません。負荷制限を設定するには、コマンド行、`clsetup` ユーティリティ、または Oracle Solaris Cluster Manager インタフェースを使用します。次の手順は、コマンド行を使用して負荷制限を設定する方法を示したものです。

---

## ▼ ノードに負荷制限を設定する

- 1 グローバルクラスタの任意のノードで、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。

- 2 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。

```
# clnode create-loadlimit -p limitname=mem_load -Z zc1 -p  
softlimit=11 -p hardlimit=20 node1 node2 node3
```

この例では、ゾーンクラスタ名は `zc1` です。サンプルプロパティは `mem_load` で、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省略可能な引数で、特に定義しなかった場合、デフォルトは無制限です。詳細については、`clnode(1CL)` のマニュアルページを参照してください。

- 3 負荷係数値を各リソースグループに割り当てます。

```
# clresourcegroup set -p load_factors=mem_load@50,factor2@1 rg1 rg2
```

この例では、2つのリソースグループ (`rg1` と `rg2`) で負荷係数が設定されています。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。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、既存の負荷を再分散できます (`clrg remaster`)。

```
# 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。

- 5 必要に応じて、一部のリソースグループに、ほかのリソースグループより高い優先度を与えることができます。

```
# 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。

- 6 必要に応じて、`Preemption_mode` プロパティを設定できます。

```
# 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。

- 7 必要に応じて、**Concentrate\_load** プロパティを設定できます。

```
# cluster set -p Concentrate_load=TRUE
```

- 8 必要に応じて、リソースグループ間のアフィニティを指定できます。

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
```

- 9 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状態を確認します。

```
# 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。

## サービスまたは管理エージェントのポート番号の変更

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。

---

注- 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正しい値である `0.0.0.0` に設定されていることを確認してください。

クラスタの各ノードで、次の手順を実行します。

1. `network-bind-address` パラメータの値を表示します。

```
# cacaoadm get-param network-bind-address
network-bind-address=0.0.0.0
```

2. パラメータ値が `0.0.0.0` に設定されていない場合は、この値に変更します。

```
# cacaoadm stop
# cacaoadm set-param network-bind-address=0.0.0.0
# cacaoadm start
```

---

## ▼ 共通エージェントコンテナを使用して、サービスまたは管理エージェントのポート番号を変更する

共通エージェントコンテナサービスのデフォルトのポート番号が実行中の別のプロセスと競合する場合、`cacaoadm` コマンドを使用し、クラスタの各ノード上で、競合しているサービスまたは管理エージェントのポート番号を変更できます。

- 1 すべてのクラスタ上で共通エージェントコンテナ管理デーモンを停止します。

```
# /opt/bin/cacaoadm stop
```

- 2 `get-param` サブコマンドを使用して、共通エージェントコンテナサービスによって現在使用されているポート番号を取得します。

```
# /opt/bin/cacaoadm get-param parameterName
```

`cacaoadm` コマンドを使用して、次の共通エージェントコンテナサービスのポート番号を変更できます。次のリストは、共通エージェントコンテナで管理できるサービスとエージェント、および対応するパラメータ名の例を示しています。

JMX コネクタポート	<code>jmxmp-connector-port</code>
SNMP ポート	<code>snmp-adapter-port</code>
SNMP トラップポート	<code>snmp-adapter-trap-port</code>
コマンドストリームポート	<code>commandstream-adapter-port</code>

---

注- 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正しい値である `0.0.0.0` に設定されていることを確認してください。

クラスタの各ノードで、次の手順を実行します。

1. `network-bind-address` パラメータの値を表示します。

```
# cacaoadm get-param network-bind-address
network-bind-address=0.0.0.0
```

2. パラメータ値が `0.0.0.0` に設定されていない場合は、この値に変更します。

```
# cacaoadm stop
# cacaoadm set-param network-bind-address=0.0.0.0
# cacaoadm start
```

- 3 ポート番号を変更します。

```
# /opt/bin/cacaoadm set-param parameterName=parameterValue
```

- 4 クラスタの各ノードで、[手順3](#)を繰り返します。

- 5 すべてのクラスタノードで共通エージェントコンテナ管理デーモンを再起動します。

```
# /opt/bin/cacaoadm start
```

## ゾーンクラスタ管理タスクの実行

ゾーンクラスタに関するほかの管理作業(ゾーンパスの移動、アプリケーションを実行するためのゾーンクラスタの準備、ゾーンクラスタの複製など)を実行できます。これらのコマンドは、必ずグローバルクラスタの投票ノードから実行してください。

---

注 - `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。 `-c config_profile` オプションの使用法については、『[Oracle Solaris Cluster ソフトウェアのインストール](#)』の「[ゾーンクラスタの構成](#)」を参照してください。

---



---

注 - グローバルクラスタ内の投票ノードからのみ実行する Oracle Solaris Cluster コマンドは、ゾーンクラスタには使用できません。各種ゾーンでのコマンドの有効な使用方法については、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 9-3 その他のゾーンクラスタの作業

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 zoneclustername</code>
アプリケーション実行用のゾーンクラスタの準備	<code>clzonecluster ready -n nodename zoneclustername</code>
ゾーンクラスタの複製	<code>clzonecluster clone -Z source- zoneclustername [-m copymethod] zoneclustername</code>  clone サブコマンドを使用する前に、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、構成済みであることが必要です。
ゾーンクラスタの削除	<a href="#">223 ページの「ゾーンクラスタを削除する」</a>
ゾーンクラスタからファイルシステムを削除	<a href="#">224 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」</a>

表 9-3 その他のゾーンクラスタの作業 (続き)

タスク	手順
ゾーンクラスタからストレージデバイスを削除	226 ページの「ゾーンクラスタからストレージデバイスを削除する」
ノードのアンインストールに関するトラブルシューティング	211 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	212 ページの「Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理」

## ▼ ゾーンクラスタを削除する

グローバルクラスタ上に構成されているゾーンクラスタは、特定の1つのゾーンクラスタを削除することも、ワイルドカードを使用してすべてのゾーンクラスタを削除することもできます。構成されていないゾーンクラスタは、削除できません。

- 1 グローバルクラスタのノードで、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。  
グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 2 ゾーンクラスタからすべてのリソースグループとそのリソースを削除します。

```
phys-schost# clresourcegroup delete -F -Z zoneclustername +
```

注- この手順は、グローバルクラスタノードから実行されます。この手順をゾーンクラスタのノードから実行するには、ゾーンクラスタノードにログインし、コマンドの「-Z zonecluster」を省略します。

- 3 ゾーンクラスタを停止します。  
phys-schost# `clzonecluster halt zoneclustername`
- 4 ゾーンクラスタをアンインストールします。  
phys-schost# `clzonecluster uninstall zoneclustername`
- 5 ゾーンクラスタを構成解除します。  
phys-schost# `clzonecluster delete zoneclustername`

### 例 9-11 グローバルクラスタからのゾーンクラスタの削除

```
phys-schost# clresourcegroup delete -F -Z sczone +
```

```
phys-schost# clzonecluster halt sczone
```

```
phys-schost# clzonecluster uninstall sczone
```

```
phys-schost# clzonecluster delete sczone
```

## ▼ ゾーンクラスタからファイルシステムを削除する

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
- 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
- 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 `HASStoragePlus` 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は、『[Oracle Solaris Cluster 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。

`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。

- 1 目的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、スーパーユーザーになります。  
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** のリソースタイプ (`HASStoragePlus`、`SUNW.ScalMountPoint` など) を特定し、削除します。

```
phys-schost# clresource delete -F -Z zoneclustername fs_zone_resources
```

- b. 削除するファイルシステム用のグローバルクラスタ内に構成されている **Oracle Solaris Cluster** リソースがあれば、そのリソースを特定して削除します。

```
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:

```
phys-schost# clzonecluster configure zoneclustername
```

- 4 ファイルシステムをゾーンクラスタの構成から削除します。

```
phys-schost# clzonecluster configure zoneclustername
```

```
clzc:zoneclustername> remove fs dir=filesystemdirectory
```

```
clzc:zoneclustername> commit
```

ファイルシステムのマウントポイントは、**dir=**で指定します。

- 5 ファイルシステムが削除されたことを確認します。

```
phys-schost# clzonecluster show -v zoneclustername
```

## 例 9-12 ゾーンクラスタ内の高可用性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 HAStoragePlus です。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name:                fs
  dir:                        /local/ufs-1
  special:                    /dev/md/dsl/dsk/d0
  raw:                        /dev/md/dsl/rdisk/d0
  type:                        ufs
  options:                    [logging]
...
phys-schost# clresource delete -F -Z sczone hasp-rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove fs dir=/local/ufs-1
clzc:sczone> commit
phys-schost# clzonecluster show -v sczone
```

### 例 9-13 ゾーンクラスタ内の高可用性 ZFS ファイルシステムの削除

この例は、リソース `hasp-rs`、タイプ `SUNW.HAStoragePlus` 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ストレージデバイス (Solaris Volume Manager ディスクセット、DID デバイスなど) 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。

- 1 目的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、スーパーユーザーになります。  
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 のリソースタイプ (`SUNW.HAStoragePlus`、`SUNW.ScalDeviceGroup` など) を特定し、削除します。

```
phys-schost# clresource delete -F -Z zoneclustername dev_zone_resources
```

- 3 削除するデバイスに対して一致するエントリを調べます。

```
phys-schost# clzonecluster show -v zoneclustername
...
Resource Name:      device
match:              <device_match>
...
```

- 4 デバイスをゾーンクラスタの構成から削除します。

```
phys-schost# clzonecluster configure zoneclustername
clzc:zoneclustername> remove device match=<devices_match>
clzc:zoneclustername> commit
clzc:zoneclustername> end
```

- 5 ゾーンクラスタを再起動します。  
phys-schost# **clzonecluster reboot zoneclustername**
- 6 デバイスの削除を確認します。  
phys-schost# **clzonecluster show -v zoneclustername**

#### 例 9-14 SVM ディスクセットをゾーンクラスタから削除する

この例は、sczone というゾーンクラスタに構成された apachedg という 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 9-15 DID デバイスをゾーンクラスタから削除する

この例は、DID デバイス d10 および d11 を削除する方法を示しています。このデバイスは、sczone というゾーンクラスタに構成されています。このデバイスは、クラスタに構成された zc\_rs のリソースにより使用されます。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name:      device
match:              /dev/did/*dsk/d10*
Resource Name:      device
match:              /dev/did/*dsk/d11*
...
phys-schost# clresource delete -F -Z sczone zc_rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove device match=/dev/did/*dsk/d10*
clzc:sczone> remove device match=/dev/did/*dsk/d11*
clzc:sczone> commit
clzc:sczone> end
phys-schost# clzonecluster reboot sczone
phys-schost# clzonecluster show -v sczone
```

# トラブルシューティング

この節では、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起動したノードから **Solaris Volume Manager** メタセットを取得する

この手順を使用して、テスト用にグローバルクラスタ外でアプリケーションを実行します。

- 1 **Solaris Volume Manager** メタセットで定足数デバイスが使用されているかどうかを確認し、定足数デバイスが **SCSI2** または **SCSI3** 予約を使用するかどうかを確認します。

```
phys-schost# clquorum show
```

- a. 定足数デバイスが **Solaris Volume Manager** メタセットにある場合は、あとで非クラスタモードにするメタセットには含まれない、新しい定足数デバイスを追加します。

```
phys-schost# clquorum add did
```

- b. 古い定足数デバイスを削除します。

```
phys-schost# clquorum remove did
```

- c. 定足数デバイスが **SCSI2** 予約を使用する場合は、古い定足数からの **SCSI2** 予約をスクラブして、**SCSI2** 予約が残らないことを確認します。

pgre コマンドの実行方法を確認するには、Oracle Support から提供されている診断ツールキットパッケージ (ha-cluster/diagnostic/tool-kit) をインストールして使用する必要があります。

- 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 targetnode
```

- 3 **HAStorage** または **HAStoragePlus** リソースを含み、あとで非クラスタモードにするメタセットの影響を受けるデバイスまたはファイルシステムを含む、1つまたは複数のリソースグループをオフラインにします。

```
phys-schost# clresourcegroup offline resourcegroupname
```

- 4 オフラインにしたリソースグループ内のすべてのリソースを無効にします。

```
phys-schost# clresource disable resourcename
```

- 5 リソースグループを非管理状態に切り替えます。  

```
phys-schost# clresourcegroup unmanage resourcegroupname
```
- 6 対応する1つまたは複数のデバイスグループをオフラインにします。  

```
phys-schost# cldevicegroup offline devicegroupname
```
- 7 1つまたは複数のデバイスグループを無効にします。  

```
phys-schost# cldevicegroup disable devicegroupname
```
- 8 パッシブノードを非クラスタモードでブートします。  

```
phys-schost# reboot -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 devicegroupname
```
- 17 1つまたは複数のリソースグループを起動します。  

```
phys-schost# clresourcegroup online -emM resourcegroupname
```

## 破損したディスクセットの復元

この手順は、ディスクセットが破損している場合、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になっている場合に使用します。状態を明らかにしようとしたができなかった場合は、ディスクセットを修正するための最後の試みとして次の手順に従います。

次の手順は、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します。

### ▼ Solaris ボリュームマネージャー ソフトウェア構成を保存する

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。代替の方法として適切なのは、`metastat` コマンドを使用して定期的に複製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方法です。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します。(prtvtoc 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ます。231 ページの「Solaris ボリュームマネージャーソフトウェア構成を再作成する」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/prtvtoc /dev/global/rdisk/diskname > /etc/lvm/diskname.vtoc
```

- 2 Solaris ボリュームマネージャー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 setname >> /etc/lvm/md.tab
```

---

注 - /etc/vfstab ファイルなどのほかの構成ファイルが、Solaris ボリュームマネージャー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 ボリュームマネージャー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。セットの所有権を持つノード上で Oracle Explorer (SUNWexplo) を実行すると、prtvtoc 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

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、ノードにディスクセットの所有権があってははいけません。

- 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 setname -P
```

このコマンドを実行すると、データベースの複製から、ディスクセット情報のほか、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 ボリュームマネージャー 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- ノードをクラスタモードからブートしたときに複数所有者ディスクセットを削除する場合は、Oracle Support から提供されている診断ツールキットパッケージ (ha-cluster/diagnostic/tool-kit) をインストールして使用する必要があることがあります。ツールキットにより、dcs 構成ファイルから情報が削除されます。手順2を参照してください。

- 2 データベースの複製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 setname -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 ソフトウェアは引き続き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時に問題が発生する場合があります。

- 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、診断ツールキットパッケージ (ha-cluster/diagnostic/tool-kit) を使用して、dcs 構成ファイルから情報を削除します。

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、Oracle Solaris Cluster にアクセスします。

## ▼ Solaris ボリュームマネージャーソフトウェア構成を再作成する

この手順に従うのは、Solaris ボリュームマネージャー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 ボリュームマネージャー構成とそのコンポーネントが保存され、破損したディスクセットが削除されていることを想定しています。

注- メディエータは、2 ノードクラスタでのみ使用するようにしてください。

- 1 新しいディスクセットを作成します。

```
# /usr/sbin/metaset -s setname -a -h nodename1 nodename2
```

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 setname -aM -h nodename1 nodename2
```

- 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 setname -a -m nodename1 nodename2
```

- 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ふたたび追加します。

```
/usr/sbin/metaset -s setname -a /dev/did/rdisk/diskname /dev/did/rdisk/diskname
```

- 4 削除したディスクセットを再作成する場合は、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、復元するセットを再作成する場合は、`/etc/lvm/diskname.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:

```
# /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 setname -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 setname -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/setname/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。正しく再構築されている場合は、`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
```



# 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。

- 233 ページの「CPU 制御の概要」
- 234 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。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 のシステム管理 \(Oracle Solaris ゾーン、Oracle Solaris 10 ゾーン、およびリソース管理\)](#)』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

## シナリオの選択

構成の選択肢と、選択するオペレーティングシステムのバージョンに応じて、さまざまなレベルの CPU 制御を行うことができます。この章で説明する CPU 制御のすべての局面は、リソースグループプロパティ `RG_SLM_TYPE` が `automated` に設定されていることに依存します。

表 10-1 で、使用可能なさまざまな構成シナリオを説明します。

表 10-1 CPU 制御のシナリオ

説明	手順
<p>リソースグループは、グローバルクラスタ投票ノードで動作します。</p> <p>CPU シェアをリソースグループに割り当て、<code>project.cpu-shares</code> および <code>zone.cpu-shares</code> の値を指定します。</p>	<p>234 ページの「グローバルクラスタの投票ノードで CPU 使用率を制御する方法」</p>

## 公平配分スケジューラ

CPU シェアをリソースグループに割り当てる手順の最初のステップは、システムのスケジューラを公平配分スケジューラ (FSS) に設定することです。デフォルトでは、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せん。

## CPU 制御の構成

このセクションでは次の作業について説明します。

- 234 ページの「グローバルクラスタの投票ノードで CPU 使用率を制御する方法」

### ▼ グローバルクラスタの投票ノードで CPU 使用率を制御する方法

グローバルクラスタの投票ノードで実行されるリソースグループに CPU シェアを割り当てるには、この手順を実行します。

リソースグループに CPU シェアが割り当てられている場合、Oracle Solaris Cluster ソフトウェアは、グローバルクラスタの投票ノードでリソースグループの 1 つのリソースを起動する際に、次のタスクを実行します。

- 投票ノードに割り当てられている CPU シェア (`zone.cpu-shares`) の数を、指定された CPU シェアの数だけ増やします (まだ行われていない場合)。
- 投票ノードに `SCSLM_resourcegroup_name` という名前のプロジェクトを作成します (まだ行われていない場合)。このプロジェクトはリソースグループに固有で、指定された数の CPU シェア (`project.cpu-shares`) が割り当てられています。
- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

CPU制御機能の構成についての詳細は、[rg\\_properties\(5\)](#)のマニュアルページを参照してください。

- 1 システムのデフォルトのスケジューラを、公平配分スケジューラ (**FSS**) に設定します。

```
# dispadmin -d FSS
```

次のリブート時に、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
```

`priocntl` コマンドと `dispadmin` コマンドを組み合わせることで、FSSがすぐにデフォルトのスケジューラになり、リブ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 [priocntl\(1\)](#) のマニュアルページを参照してください。

---

注 - FSSがデフォルトのスケジューラでない場合、CPUシェアの割り当ては有効になりません。

---

- 2 各ノードでCPU制御を使用するため、グローバルクラスタの投票ノードに対するシェア数と、デフォルトのプロセッサセットで使用可能なCPUの最小数を構成します。

`globalzonestshares` 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 globalzonestshares=integer] \
[-p defaultpsetmin=integer] \
node
```

`-p defaultpsetmin= defaultpsetmininteger` デフォルトのプロセッサセットで利用可能なCPUシェアの最小数を設定します。デフォルト値は1です。

`-p globalzonestshares= integer` 投票ノードに割り当てられるシェアの数を設定します。デフォルト値は1です。

`node` プロパティを設定するノードを指定します。

これらのプロパティを設定する際には、投票ノードのプロパティを設定しています。

- 3 これらのプロパティを正しく設定したことを確認します。

```
# clnode show node
```

指定するノードに対して、`clnode` コマンドは、設定されているプロパティ、およびこれらのプロパティに設定されている値を出力します。`clnode` を使用して CPU 制御プロパティを設定しないと、これらはデフォルト値をとります。

#### 4 CPU 制御機能を構成します。

```
# clresourcegroup create -p RG_SLM_TYPE=automated \
  [-p RG_SLM_CPU_SHARES=value] resource_group_name
```

`-p RG_SLM_TYPE=automated` CPU 使用率を管理できるようにし、システム資源管理用に 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、投票ノードに割り当てられる CPU シェアの数 (`zone.cpu-shares`) を決定します。

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

この手順では、`RG_SLM_PSET_TYPE` プロパティは設定しません。投票ノードでは、このプロパティは値 `default` をとります。

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

#### 5 構成の変更を有効にします。

```
# clresourcegroup online -emM resource_group_name
```

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

---

注 - `SCSLM_resource_group_name` プロジェクトは削除または変更しないでください。手動で、たとえば `project.max-lwps` プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

# 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。

- 237 ページの「Oracle Solaris Cluster ソフトウェアの更新の概要」
- 238 ページの「Oracle Solaris Cluster ソフトウェアの更新」
- 241 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ためには、すべてのクラスタの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 4.0 から 4.1 へのアップグレードです。scinstall ユーティリティまたは 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) の作成、使用されていないマウントポイントへのブート環境のマウント、ビットの更新、および新しいブート環境の有効化を行います。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、クラスタをリブ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 Upgrade Guide』を参照してください。

ブランドタイプが solaris であるフェイルオーバーゾーンがある場合は、『Oracle Solaris Cluster Upgrade Guide』の「How to Upgrade Failover Zones」に記載されている手順に従います。

- 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、Service 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、定期的に取り替えられ、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性のある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、239 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 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 11-1 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 Upgrade Guide』の「How to Perform a Standard Upgrade」
特定のパッケージの更新	239 ページの「特定のパッケージの更新」
定数数サーバーまたは AI インストールサーバーの更新	240 ページの「定数数サーバーまたは AI インストールサーバーの更新」
Oracle Solaris Cluster パッケージの削除	241 ページの「パッケージのアンインストール」
	241 ページの「定数数サーバーまたは 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」

## 新しいリリースへのクラスタのアップグレード

アップグレードは常に新しいブート環境で行われ、既存のブート環境は変更されないため、このアップグレードを行う前に、クラスタを非クラスタモードにする必要はありません。新しいブート環境には、名前を指定することも、自動的に生成される名前を使用することもできます。手順については、『Oracle Solaris Cluster Upgrade Guide』の「How to Perform a Standard Upgrade」を参照してください。

Oracle Solaris Cluster ソフトウェアをアップグレードするときは、データサービスおよび Geographic Edition ソフトウェアもアップグレードするようにしてください。ただし、データサービスを個別にアップグレードする場合は、『[Oracle Solaris Cluster Data Services Planning and Administration Guide](#)』の「[Overview of the Installation and Configuration Process](#)」を参照してください。Oracle Solaris Cluster Geographic Edition を個別にアップグレードする場合は、『[Oracle Solaris Cluster Geographic Edition Installation Guide](#)』を参照してください。

Oracle Solaris Cluster ソフトウェアをアップグレードすると、Oracle Solaris OS も最新のリリースにアップグレードされます。

## 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とともに導入されます。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(1)` コマンドを使用して、SRU 更新を行います。また、`scinstall - u` コマンドを使用して SRU 更新を行うこともできます。

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

- 1 スーパーユーザーになるか、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
- 2 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、利用可能な更新があるすべてのインストール済みの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、次の手順を行います。

a. `# pkg change-facet facet.version-lock.pkg name=false`

b. `# pkg update pkg name`

例:

```
# pkg change-facet facet.version-lock.ha-cluster/data-service/weblogic =false
```

```
# pkg update ha-cluster/data-service/weblogic
```

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、次の手順を行います。

```
# 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 ソフトウェア パッケージの追加および更新』の「省略可能な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
```

## 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 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。定足数サーバーについては、『Oracle Solaris Cluster 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 の使用方法については、『Oracle Solaris Cluster 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 (Automated Installer)」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 スーパーユーザーになるか、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
- 2 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/*
```



インストール済みの `ha-cluster` 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。



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 スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。
- 2 既存のパッケージをアンインストールします。

```
# pkg uninstall pkg-fmri
```

複数のパッケージをアンインストールする場合は、次の構文を使用します。

```
# 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については、『[Oracle Solaris 11 ソフトウェアパッケージの追加および更新](#)』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 スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。
- 2 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。

---

## 更新に関する注意事項

Oracle Solaris Cluster の更新をより効率的に管理するために、次の点に注意してください。

- 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。
- 新しい Oracle Solaris Cluster バージョンにアップグレードする場合は、『[Oracle Solaris Cluster Upgrade Guide](#)』に記載されている手順に従います。

# ◆◆◆ 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。

- 243 ページの「クラスタのバックアップ」
- 246 ページの「クラスタファイルの復元」

### 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 12-1 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
ミラーまたはプレックスファイルシステムのオンラインバックアップの実行	243 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 ボリュームマネージャー)」
クラスタ構成のバックアップ	245 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 ボリュームマネージャー)

Solaris Volume Manager のミラー化ボリュームは、マウント解除したりミラー全体をオフラインにしたりしなくても、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のですが、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 バックアップするクラスタノード上で、スーパーユーザーまたは同等の役割になります。
- 2 `metaset` 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。  
詳細については、[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- 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- 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 の管理: ZFS ファイルシステム』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

クラスタ構成をアーカイブし、クラスタ構成の簡単な復元を実現するため、定期的にクラスタ構成をバックアップします。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、スーパーユーザーになるか、RBAC の承認 **solaris.cluster.read** を提供する役割になります。

- 2 クラスタ構成情報をファイルにエクスポートします。

```
# /usr/cluster/bin/cluster export -o configfile
```

*configfile* クラスタコマンドのクラスタ構成情報のエクスポート先である XML 構成ファイルの名前。XML 構成ファイルについては、[clconfiguration\(5CL\)](#) 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。

```
# vi 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。

ファイルまたはファイルシステムの復元を開始する前に、次の点を確認してください。

- 必要なテープ
- ファイルシステムを復元する raw デバイス名
- 使用するテープドライブの種類
- テープドライブのデバイス名(ローカルまたはリモート)
- 障害が発生したディスクのパーティション分割方式。これは、パーティションとファイルシステムを交換用ディスクに正確に複製しなければならないためです。

表 12-2 タスクリスト:クラスタファイルの復元

タスク	手順
Solaris ボリュームマネージャーの場合、ZFS ルート (/) ファイルシステムを復元	246 ページの「ZFS ルート (/) ファイルシステムを復元する方法 (Solaris ボリュームマネージャー)」

### ▼ ZFS ルート (/) ファイルシステムを復元する方法 (Solaris ボリュームマネージャー)

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おさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。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 ボリュームマネージャー メタセットのメタデバイスで使用できます。

---

注-新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

---

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

この手順では、長形式の 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 復元するノードの添付先であるディスクセットへのアクセス権があるクラスタノード上で、スーパーユーザーになるか、**solaris.cluster.modify RBAC** 権限を提供する役割になります。  
復元する以外のノードを使用します。

- 2 すべてのメタセットから、復元するノードのホスト名を削除します。  
このコマンドは、削除するノード以外のメタセットのノードから実行します。復元を行なっているノードはオフラインであるため、システムは「RPC: Rpcbnd failure - RPC: Timed out」というエラーを表示します。このエラーを無視し、次のステップを続けます。

```
# metaset -s setname -f -d -h nodelist
```

```
-s setname      ディスクセット名を指定します。
-f             ディスクセットから最後のホストを削除します。
-d            ディスクセットから削除します。
-h nodelist    ディスクセットから削除するノードの名前を指定します。
```

- 3 **ZFS** ルートファイルシステム (/) を復元します。  
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 Administration: ZFS File Systems](#)』の「[How to Replace a Disk in a ZFS Root Pool](#)」に記載されている手順に従います。

---

注 - /global/.devices/node@nodeid ファイルシステムが作成されていることを確認します。

---

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 rootdisk**
- 6 **metadb** コマンドを使用し、状態データベースの複製を再作成します。  
# **metadb -c copies -af raw-disk-device**  
-c *copies*                    作成する複製の数を指定します。  
-f *raw-disk-device*        複製の作成先の raw ディスクデバイス名を指定します。  
-a                                複製を追加します。  
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ブートします。これでクラスタを使用できるようになります。

#### 例 12-1 ZFS ルート (/) ファイルシステムの復元 (Solaris ボリュームマネージャー)

次に、ノード `phys-schost-1` に復元したルート (/) ファイルシステムの例を示します。 `metaset` コマンドは、クラスタの別のノード `phys-schost-2` から実行し、ノード `phys-schost-1` を削除し、後でディスクセット `schost-1` に追加します。そのコマンドはすべて `phys-schost-1` から実行します。新しいブートブロックが `/dev/rdisk/c0t0d0s0` に作成され、3つの状態データベースの複製が `/dev/rdisk/c0t0d0s4` に再作成されます。データの復元の詳細については、『[Oracle Solaris Administration: ZFS File Systems](#)』の「[Repairing Damaged Data](#)」を参照してください。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on a cluster node
other than the node to be restored.]
[Remove the node from the metaset:]
phys-schost-2# metaset -s schost-1 -f -d -h phys-schost-1
[Replace the failed disk and boot the node:]
Restore the root (/) and /usr file system using the procedure in the Solaris system
administration documentation
[Reboot:]
# reboot
[Replace the disk ID:]
# cldevice repair /dev/dsk/c0t0d0
[Re-create state database replicas:]
```



---

```
# metadb -c 3 -af /dev/rdisk/c0t0d0s4  
[Add the node back to the metaset:]  
phys-schost-2# metaset -s schost-1 -a -h phys-schost-1
```



## StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成

この付録では、Oracle Solaris Cluster Geographic Edition を使用しない、ホストベースの複製の代替方法を説明します。Oracle では、ホストベースの複製に Oracle Solaris Cluster Geographic Edition を使用して、クラスタ間のホストベースの複製の構成と操作を簡素化することをお勧めします。79 ページの「データ複製についての理解」を参照してください。

この付録の例は、StorageTek Availability Suite ソフトウェアを使用してクラスタ間のホストベースのデータ複製を構成する方法を示しています。この例では、NFS アプリケーション用の完全なクラスタ構成を示し、個別のタスクの実行方法に関する詳細情報を提供します。すべてのタスクはグローバルクラスタの投票ノードで行われます。例には、ほかのアプリケーションやクラスタ構成に必要な手順がすべて含まれているわけではありません。

スーパーユーザーの代わりに役割に基づくアクセス制御 (RBAC) を使用してクラスタノードにアクセスする場合は、すべての 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ますようにします。ユーザーがスーパーユーザーでない場合、一連のデータ複製手順には、次の Oracle Solaris Cluster RBAC の承認が必要です。

- `solaris.cluster.modify`
- `solaris.cluster.admin`
- `solaris.cluster.read`

RBAC の役割についての詳細は、『[Oracle Solaris の管理: セキュリティーサービ](#)ス』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 StorageTek Availability Suite ソフトウェアの理解

ここでは、耐障害性について紹介し、StorageTek Availability Suite ソフトウェアが使用するデータ複製方式について説明します。

耐障害性は、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、データ複製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複製されている場合、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、データが非同期で複製されていた場合、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複製されていなかったデータがある可能性があります、それらのデータは失われます。

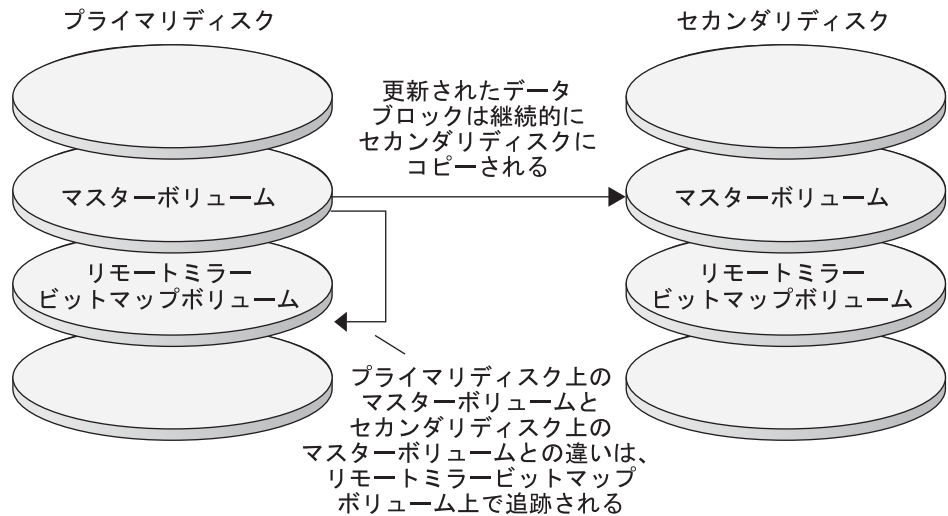
### StorageTek Availability Suite ソフトウェアが使用するデータ複製方式

このセクションでは、StorageTek Availability Suite が使用するリモートミラー複製方式とポイントインタイムスナップショット方式について説明します。このソフトウェアは、`sndradm` と `iiadm` コマンドを使用してデータを複製します。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。

#### リモートミラー複製

図 A-1 はリモートミラー複製を示しています。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複製されます。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 A-1 リモートミラー複製



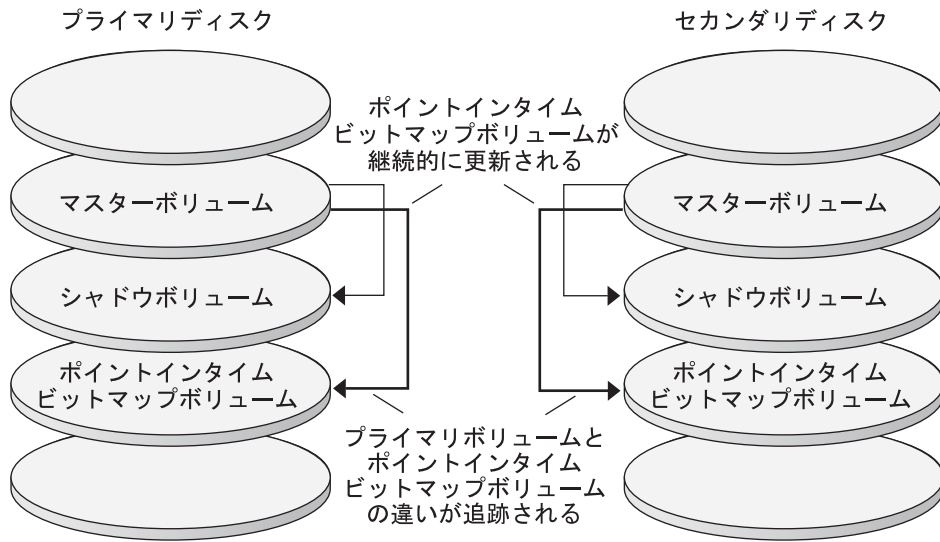
リモートミラー複製は、リアルタイムに同期で実行することも非同期で実行することもできます。各クラスタの各ボリュームセットはそれぞれ、同期複製または非同期複製に構成できます。

- 同期データ複製では、リモートボリュームが更新されるまで、書き込み操作は完了したとは確認されません。
- 非同期データ複製では、リモートボリュームが更新される前に書き込み操作が完了したと確認されます。非同期データ複製は、長い距離や低い帯域幅で大きな柔軟性を発揮します。

## ポイントインタイムスナップショット

図 A-2 は、ポイントインタイムスナップショットを示しています。各ディスクのマスターボリュームのデータは、同じディスクのシャドウボリュームにコピーされます。ポイントインタイムビットマップは、マスターボリュームとシャドウボリューム間の違いを追跡調査します。データがシャドウボリュームにコピーされると、ポイントインタイムビットマップはリセットされます。

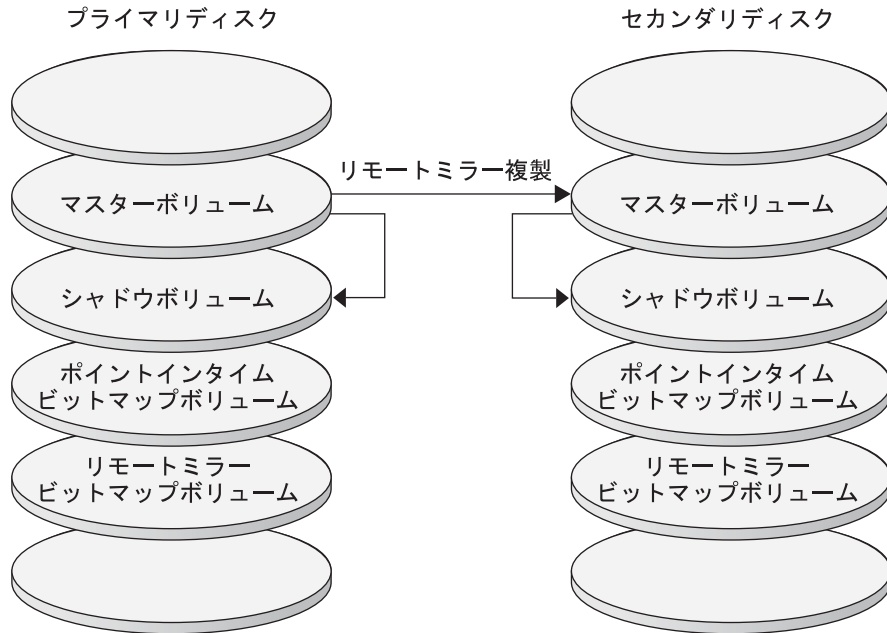
図A-2 ポイントインタイムスナップショット



### 構成例での複製

図A-3に、この構成例でミラー複製とポイントインタイムスナップショットがどのように使用されているかを示します。

図 A-3 構成例での複製



## クラスタ間でホストベースのデータ複製を構成するためのガイドライン

このセクションでは、クラスタ間のデータ複製の構成ガイドラインを提供します。また、複製リソースグループとアプリケーションリソースグループの構成のコツも紹介します。これらのガイドラインは、クラスタのデータ複製を構成する際に使用してください。

このセクションでは、次の項目について説明します。

- [256 ページの「複製リソースグループの構成」](#)
- [257 ページの「アプリケーションリソースグループの構成」](#)
  - [257 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」](#)
  - [258 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」](#)
- [260 ページの「テイクオーバーの管理のガイドライン」](#)

## 複製リソースグループの構成

複製リソースグループは、StorageTek Availability Suite ソフトウェアの制御下にあるデバイスグループと論理ホスト名リソースを結び付けます。論理ホスト名は、データ複製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。複製リソースグループには、次の特徴があります。

- フェイルオーバーリソースグループである  
フェイルオーバーリソースは、常に単一のノード上で実行されます。フェイルオーバーが発生すると、フェイルオーバーリソースがフェイルオーバーに加わります。
- 論理ホスト名リソースを持つ  
論理ホスト名は、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、StorageTek Availability Suite ソフトウェアのデータ複製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、複製リソースグループがスイッチオーバーまたはフェイルオーバーしたときに、デバイスグループをフェイルオーバーします。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、複製リソースグループをフェイルオーバーします。このように複製リソースグループとデバイスグループは常に結び付き、同じノードから制御されます。

HAStoragePlus リソース内に次の拡張プロパティを定義する必要があります。

- *GlobalDevicePaths*。この拡張プロパティは、ボリュームが属するデバイスグループを定義します。
- *AffinityOn property = True*。この拡張プロパティは、複製リソースグループがスイッチオーバーまたはフェイルオーバーしたときに、デバイスグループをスイッチオーバーまたはフェイルオーバーします。この機能はアフィニティスイッチオーバーと呼ばれます。

HAStoragePlus についての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 結び付いているデバイスグループに `-stor-rg` を付けた名前になる  
たとえば、`devgrp-stor-rg` などです。
- 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

高可用性を実現するためには、アプリケーションはアプリケーションリソースグループのリソースとして管理される必要があります。アプリケーションリソースグループは、フェイルオーバーアプリケーションまたはスケー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複製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。

- [257 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」](#)
- [258 ページの「スケーラブルアプリケーション向けのリソースグループの構成」](#)

## フェイルオーバーアプリケーション向けのリソースグループの構成

フェイルオーバーアプリケーションでは、1つのアプリケーションが1度に1ノード上で動作します。ノードで障害が発生すると、アプリケーションは同じクラスタ内の別のノードにフェイルオーバーします。フェイルオーバーアプリケーション向けリソースグループは、以下の特徴を持っていない限りなりません。

- 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、複製リソースグループとアプリケーションリソースグループに結び付けられています。したがって、アプリケーションリソースグループがフェイルオーバーすると、デバイスグループと複製リソースグループもフェイルオーバーします。アプリケーションリソースグループ、複製リソースグループおよびデバイスグループは、同じノードによって制御されます。

ただし、デバイスグループや複製リソースグループがフェイルオーバーしても、アプリケーションリソースグループはフェイルオーバーを行いません。

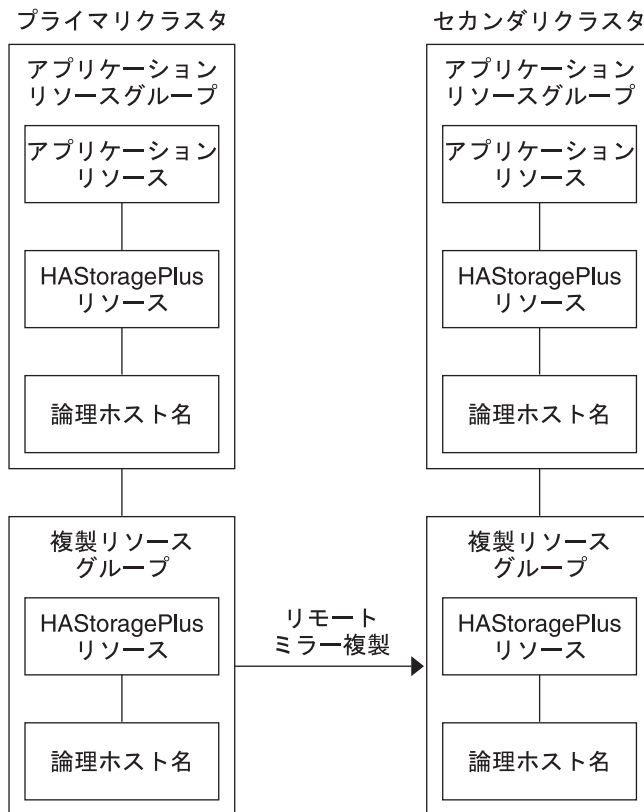
- アプリケーションデータがグローバルマウントされている場合は、アプリケーションリソースグループに HAStoragePlus リソースを必ず入れなければならないわけではありませんが、入れることをお勧めします。
- アプリケーションデータがローカルマウントされている場合は、アプリケーションリソースグループに HAStoragePlus リソースを必ず入れなければならない限りません。

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 A-4 に、フェイルオーバーアプリケーションでのアプリケーションリソースグループと複製リソースグループの構成を示します。

図 A-4 フェイルオーバーアプリケーションでのリソースグループの構成



### スケーラブルアプリケーション向けのリソースグループの構成

スケーラブルアプリケーションでは、アプリケーションは複数のノードで実行されて、1つの論理サービスを作成します。スケーラブルアプリケーションを実行しているノードで障害が発生しても、フェイルオーバーは起こりません。アプリケーションは別のノードで引き続き実行されます。

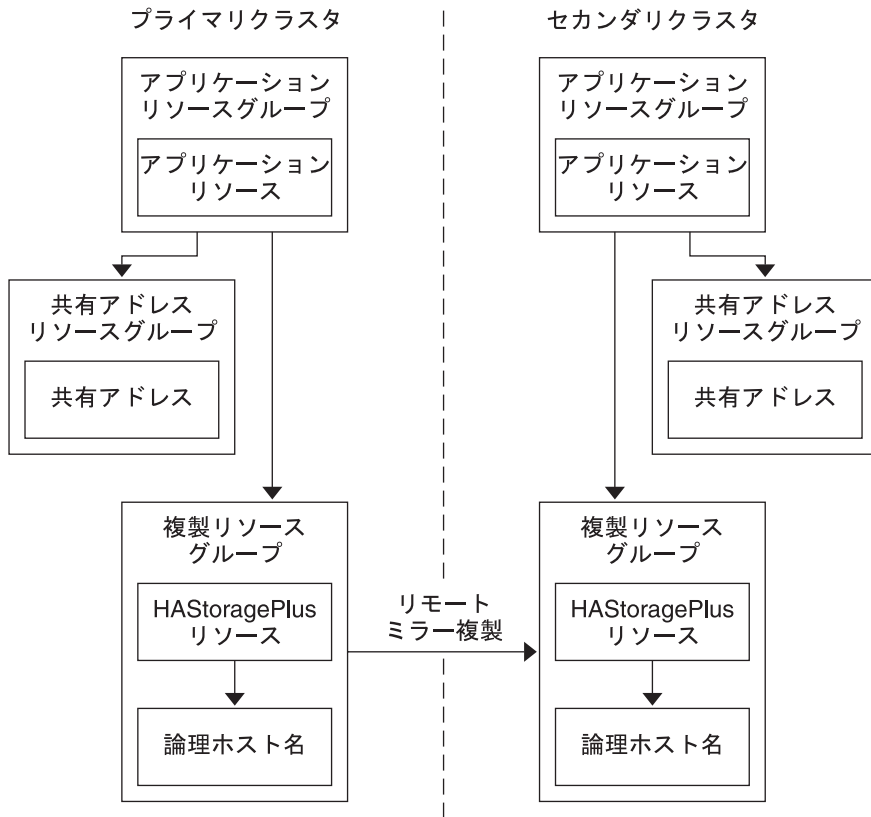
スケーラブルアプリケーションをアプリケーションリソースグループのリソースとして管理している場合は、アプリケーションリソースグループをデバイスグループと結び付ける必要はありません。したがって、アプリケーションリソースグループ向けに HAStoragePlus リソースを作成する必要はありません。

スケーラブルアプリケーション向けリソースグループは、以下の特徴を持っていないければなりません。

- 共有アドレスのリソースグループに依存する  
共有アドレスは、受信データを配信するためにスケーラブルアプリケーションを実行するノードで使用されます。
- 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 A-5 に、スケーラブルアプリケーションでのリソースグループの構成を示します。

図 A-5 スケーラブルアプリケーションでのリソースグループの構成

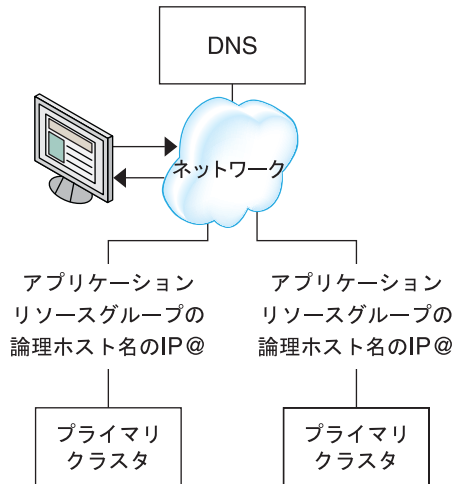


## テイクオーバーの管理のガイドライン

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 A-6 クライアントからクラスタへの DNS マッピング



DNSを更新するには、`nsupdate` コマンドを使用します。詳細は、[nsupdate\(1M\)](#)のマニュアルページを参照してください。テイクオーバーの管理の例については、[286 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。

プライマリクラスタが修復されたら、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のものであることを確認します。これを行うには、複製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. データ複製の方向を逆にして、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複製するようにします。
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。

## タスマップ:データ複製の構成例

表 A-1 に、StorageTek Availability Suite ソフトウェアを使用して NFS アプリケーション向けにどのようにデータ複製を構成するかを示すこの例でのタスクを示します。

表 A-1 タスマップ:データ複製の構成例

タスク	手順
1. クラスタを接続およびインストールする	<a href="#">262 ページ</a> の「 <a href="#">クラスタの接続とインストール</a> 」

表 A-1 タスクマップ:データ複製の構成例 (続き)

タスク	手順
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	264 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複製を有効にする	278 ページの「プライマリクラスタで複製を有効にする方法」 280 ページの「セカンダリクラスタで複製を有効にする方法」
4. データ複製を実行する	281 ページの「リモートミラー複製を実行する方法」 282 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. データ複製の構成を確認する	283 ページの「複製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 A-7 に、構成例で使用するクラスタ構成を示します。構成例の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 A-7 クラスタ構成例

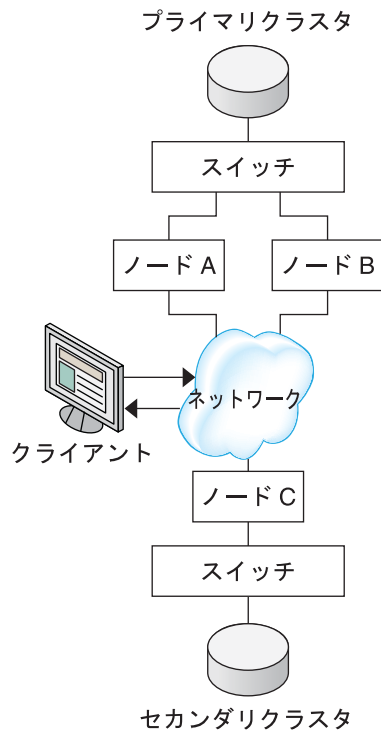


表 A-2 に、構成例で必要となるハードウェアとソフトウェアをまとめました。Oracle Solaris OS、Oracle Solaris Cluster ソフトウェア、およびボリューム管理ソフトウェアは、StorageTek Availability Suite ソフトウェアとソフトウェア更新をインストールする前にクラスタノードにインストールしてください。

表 A-2 必要なハードウェアとソフトウェア

ハードウェアまたはソフトウェア	要件
ノードハードウェア	StorageTek 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、『 <a href="#">Oracle Solaris Cluster Hardware Administration Manual</a> 』を参照してください。
ディスク容量	約 15M バイト

表 A-2 必要なハードウェアとソフトウェア (続き)

ハードウェアまたはソフトウェア	要件
Oracle Solaris OS	<p>Oracle Solaris Cluster ソフトウェアがサポートする Oracle Solaris OS のリリース。</p> <p>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。</p> <p>インストールについては、『Oracle Solaris Cluster ソフトウェアのインストール』を参照してください。</p>
Oracle Solaris Cluster ソフトウェア	<p>Oracle Solaris Cluster 4.0 ソフトウェア</p> <p>インストールについては、『Oracle Solaris Cluster ソフトウェアのインストール』を参照してください。</p>
ボリューム管理ソフトウェア	<p>Solaris ボリュームマネージャーソフトウェア</p> <p>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。</p> <p>インストールについては、『Oracle Solaris Cluster ソフトウェアのインストール』の第4章「Solaris ボリュームマネージャーソフトウェアの構成」を参照してください。</p>
StorageTek Availability Suite ソフトウェア	<p>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 StorageTek Availability Suite ソフトウェアを使用する必要があります。</p> <p>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 StorageTek Availability Suite ソフトウェアのインストールマニュアルを参照してください。</p> <ul style="list-style-type: none"> <li>StorageTek Availability Suite - StorageTek Availability のドキュメント</li> </ul>
StorageTek Availability Suite ソフトウェアの更新	<p>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。</p>

## 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、256 ページの「複製リソースグループの構成」および257 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。

- 266 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」
- 267 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」



- 268 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」
- 269 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」
- 270 ページの「プライマリクラスタで複製リソースグループを作成する方法」
- 272 ページの「セカンダリクラスタで複製リソースグループを作成する方法」
- 273 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」
- 275 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」
- 283 ページの「複製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 A-3 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
複製リソースグループ とリソース	devgrp-stor-rg	複製リソースグループ
	lhost-reprg-prim、 lhost-reprg-sec	プライマリクラスタとセカンダリクラスタの複製リソースグループの論理ホスト名
	devgrp-stor	複製リソースグループの HAStoragePlus リソース
アプリケーションリ ソースグループとリ ソース	nfs-rg	アプリケーションリソースグループ
	lhost-nfsrg-prim、 lhost-nfsrg-sec	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。複製リソースグループは、*devicegroupname-stor-rg* というフォーマットでなければなりません。

Solaris ボリュームマネージャーソフトウェアについては、『Oracle Solaris Cluster ソフトウェアのインストール』の第4章「Solaris ボリュームマネージャーソフトウェアの構成」を参照してください。

## ▼ 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。
  - 252 ページの「クラスタにおける StorageTek Availability Suite ソフトウェアの理解」
  - 255 ページの「クラスタ間でホストベースのデータ複製を構成するためのガイドライン」
- 262 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 **nodeA** にスーパーユーザー、または RBAC の承認 `solaris.cluster.modify` を提供する役割になりますしてアクセスします。

`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 A-7](#) を参照してください。

- 2 NFS データおよび関連する複製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 メタセットにメディアータを追加します。

```
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* - 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 [267 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 [266 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 **nodeC** にスーパーユーザー、または RBAC の承認 `solaris.cluster.modify` を提供する役割になりますしてアクセスします。
- 2 NFS データおよび関連する複製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- 単一のノードクラスタではメディア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* - 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 [268 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 [267 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、スーパーユーザー、または RBAC の承認 `solaris.cluster.admin` を提供する役割になります。
- 2 **nodeA** と **nodeB** で、NFS ファイルシステム向けのマウントポイントディレクトリを作成します。

例:

```
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` ファイルに次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 [269 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 [268 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、スーパーユーザー、または RBAC の承認 `soLaris.cluster.admin` を提供する役割になります。

- 2 **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:

```
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 [270 ページの「プライミクラスタで複製リソースグループを作成する方法」](#)に進みます。

## ▼ プライミクラスタで複製リソースグループを作成する方法

始める前に 手順 [269 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。

- 1 **nodeA** にスーパーユーザー、または RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割になりますしてアクセスします。
- 2 **SUNW.HASStoragePlus** というリソースタイプを登録します。  

```
nodeA# clresource type register SUNW.HASStoragePlus
```
- 3 デバイスグループの複製リソースグループを作成します。  

```
nodeA# clresourcegroup create -n nodeA,nodeB devgrp-stor-rg
```

-n nodeA,nodeB クラスタノード **nodeA** および **nodeB** が複製リソースグループをマスターできることを指定します。

`devgrp-stor-rg` 複製リソースグループの名前。この名前でも、`devgrp` はデバイスグループの名前を指定します。

**4 SUNW.HASStoragePlus リソースを複製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= StorageTek 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.HASStoragePlus 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、複製リソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

これらの拡張プロパティについての詳細は、[SUNW.HASStoragePlus\(5\)](#) のマニュアルページを参照してください。

**5 論理ホスト名リソースを複製リソースグループに追加します。**

```
nodeA# clreslogicalhostname create -g devgrp-stor-rg lhost-reprg-prim
```

プライマリクラスタ上の複製リソースグループの論理ホスト名は lhost-reprg-prim です。

**6 リソースを有効にし、リソースグループを管理し、リソースグループをオンラインにします。**

```
nodeA# clresourcegroup online -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、複製リソースグループが nodeA でオンラインとなっていることを確認します。

次の手順 [272 ページの「セカンダリクラスタで複製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複製リソースグループを作成する方法

始める前に 手順270 ページの「プライマリクラスタで複製リソースグループを作成する方法」を完了します。

- 1 **nodeC** にスーパーユーザー、または RBAC の承認 **solaris.cluster.modify**、**solaris.cluster.admin**、および **solaris.cluster.read** を提供する役割になりますしてアクセスします。

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

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

- 3 デバイスグループの複製リソースグループを作成します。

```
nodeC# clresourcegroup create -n nodeC devgrp-stor-rg
```

**create**                   リソースグループを作成します。

**-n**                       リソースグループのノードリストを指定します。

**devgrp**                 デバイスグループの名前。

**devgrp-stor-rg**       複製リソースグループの名前。

- 4 複製リソースグループに **SUNW.HASStoragePlus** リソースを追加します。

```
nodeC# clresource create \
-t SUNW.HASStoragePlus \
-p GlobalDevicePaths=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=**   StorageTek 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.HASStoragePlus** 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、複製リソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

**devgrp-stor**           複製リソースグループの **HAStoragePlus** リソース

これらの拡張プロパティについての詳細は、[SUNW.HASStoragePlus\(5\)](#) のマニュアルページを参照してください。



- 5 論理ホスト名リソースを複製リソースグループに追加します。

```
nodeC# clreslogicalhostname create -g devgrp-stor-rg lhost-reprg-sec
```

セカンダリクラスタ上の複製リソースグループの論理ホスト名は `lhost-reprg-sec` です。

- 6 リソースを有効にし、リソースグループを管理し、リソースグループをオンラインにします。

```
nodeC# clresourcegroup online -emM -n nodeC devgrp-stor-rg
```

`onLine` オンラインにします。

`-e` 関連付けられたリソースを有効にします。

`-M` リソースグループを管理状態にします。

`-n` リソースグループをオンラインにするノードを指定します。

- 7 リソースグループがオンラインであることを確認します。

```
nodeC# clresourcegroup status devgrp-stor-rg
```

リソースグループの状態フィールドを調べ、複製リソースグループが `nodeC` でオンラインとなっていることを確認します。

次の手順 [273 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 [272 ページの「セカンダリクラスタで複製リソースグループを作成する方法」](#)を完了します。

- 1 `nodeA` にスーパーユーザー、または RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割になりますしてアクセスします。
- 2 `SUNW.nfs` をリソースタイプとして登録します。

```
nodeA# clresourcetype register SUNW.nfs
```

- 3 **SUNW.HASStoragePlus** をリソースタイプとして登録していない場合は、登録します。

```
nodeA# clresource type 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複製リソースグループ devgrp-stor-rg に結び付いている必要があります。

複製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.HASStoragePlus** リソースを追加します。

```
nodeA# clresource create -g nfs-rg \  
-t SUNW.HASStoragePlus \  
-p FileSystemMountPoints=/global/mountpoint \  
-p AffinityOn=True \  
nfs-dg-rs
```

create

リソースを作成します。

-g

リソースを追加するリソースグループを指定します。

-t SUNW.HASStoragePlus

リソースのタイプに SUNW.HASStoragePlus を指定します。

-p 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 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 -emM -n nodeA nfs-rg
```

online リソースグループをオンラインにします。

-e 関連付けられたリソースを有効にします。

-M リソースグループを管理状態にします。

-n リソースグループをオンラインにするノードを指定します。

nfs-rg リソースグループの名前。

- 8 アプリケーションリソースグループがオンラインであることを確認します。

```
nodeA# clresourcegroup status
```

アプリケーションリソースグループの状態フィールドを調べ、複製リソースグループが nodeA と nodeB でオンラインとなっているかどうかを調べます。

次の手順 [275 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 [273 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。

- 1 **nodeC** にスーパーユーザー、または RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割になりますしてアクセスします。

- 2 **SUNW.nfs** をリソースタイプとして登録します。

```
nodeC# clresourcetype register SUNW.nfs
```

- 3 **SUNW.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複製リソースグループ `devgrp-stor-rg` に結び付いている必要があります。

複製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.HASStoragePlus** リソースを追加します。

```
nodeC# clresource create -g nfs-rg \  
-t SUNW.HASStoragePlus \  
-p FileSystemMountPoints=/global/mountpoint \  
-p AffinityOn=True \  
nfs-dg-rs
```

`create`

リソースを作成します。

`-g`

リソースを追加するリソースグループを指定します。

`-t SUNW.HASStoragePlus`

リソースのタイプに `SUNW.HASStoragePlus` を指定します。

`-p`

リソースのプロパティを指定します。

`FileSystemMountPoints=/global/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マリ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 [278 ページの「データ複製の有効化例」](#)に進みます。

## データ複製の有効化例

このセクションでは、構成例のデータ複製をどのように有効にするかを説明します。このセクションでは、StorageTek Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、StorageTek Availability 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。

- 278 ページの「プライマリクラスタで複製を有効にする方法」
- 280 ページの「セカンダリクラスタで複製を有効にする方法」

### ▼ プライマリクラスタで複製を有効にする方法

- 1 **nodeA** にスーパーユーザー、または RBAC の承認 `solaris.cluster.read` を提供する役割になりますしてアクセスします。

- 2 すべてのトランザクションをフラッシュします。

```
nodeA# lockfs -a -f
```

- 3 論理ホスト名 `lhost-reprg-prim` と `lhost-reprg-sec` がオンラインであることを確認します。

```
nodeA# clresourcegroup status
nodeC# clresourcegroup status
```

リソースグループの状態フィールドを調べます。

- 4 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複製を有効にします。

この手順では、プライマリクラスタからセカンダリクラスタへの複製を有効にします。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 (`d200`) からセカンダリクラスタのマスターボリューム (`d200`) への複製が有効になります。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複製も有効になります。

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

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/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す。

## 7 ポイントインタイムスナップショットを有効にします。

次のコマンドを実行します。StorageTek 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 になります。

- 8 ポイントインタイムスナップショットをリモートミラーセットに設定します。  
次のコマンドを実行します。StorageTek 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。StorageTek Availability Suite ソフトウェアは、リモートミラー複製の前にポイントインタイムスナップショットを必ず取ります。

次の手順 280 ページの「セカンダリクラスタで複製を有効にする方法」に進みます。

### ▼ セカンダリクラスタで複製を有効にする方法

始める前に 手順278 ページの「プライマリクラスタで複製を有効にする方法」を完了します。

- 1 スーパーユーザーとして **nodeC** にアクセスします。
- 2 すべてのトランザクションをフラッシュします。
- 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複製を有効にします。

次のコマンドを実行します。StorageTek 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、StorageTek 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。StorageTek 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。StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -I a \
/dev/md/nfsset/rdsk/d200 \
/dev/md/nfsset/rdsk/d201 \
/dev/md/nfsset/rdsk/d202
```

次の手順 [281 ページの「データ複製の実行例」](#)に進みます。

## データ複製の実行例

このセクションでは、構成例のデータ複製をどのように実行するかを説明します。このセクションでは、StorageTek Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、StorageTek 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。

- [281 ページの「リモートミラー複製を実行する方法」](#)
- [282 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」](#)
- [283 ページの「複製が正しく構成されていることを確認する方法」](#)

### ▼ リモートミラー複製を実行する方法

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複製されます。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 スーパーユーザーとして `nodeA` にアクセスします。
- 2 クラスタがロギングモードであることを確認します。

次のコマンドを実行します。StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/md/nfsset/rdsk/d200 ->
lhost-reprg-sec:/dev/md/nfsset/rdsk/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す。

- 3 すべてのトランザクションをフラッシュします。

```
nodeA# lockfs -a -f
```

- 4 **nodeC** で **手順1** から **手順3** を繰り返します。
- 5 **nodeA** のマスターボリュームを **nodeC** のマスターボリュームにコピーします。  
次のコマンドを実行します。StorageTek 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 複製が完了し、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。StorageTek 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 クラスタが複製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。StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/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
```

複製モードでは、状態は `replicating` で、自動同期のアクティブ状態は `on` です。プライマリボリュームに書き込みが行われると、StorageTek 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 [282 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 [281 ページの「リモートミラー複製を実行する方法」](#) を完了します。

- 1 **nodeA** にスーパーユーザー、または RBAC の承認 `solaris.cluster.modify` および `solaris.cluster.admin` を提供する役割になりますしてアクセスします。

- 2 **nodeA** で実行されているリソースを無効にします。

```
nodeA# clresource disable 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。StorageTek 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。複製は行われません。

- 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。StorageTek 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。StorageTek 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。StorageTek 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 [283 ページの「複製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。

## ▼ 複製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 [282 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 **nodeA** および **nodeC** にスーパーユーザー、または RBAC の承認 `solaris.cluster.admin` を提供する役割になりますを使用してアクセスします。

- 2 プライマリクラスタが複製モードで、自動同期機能がオンになっていることを確認します。

次のコマンドを実行します。StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/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
```

複製モードでは、状態は `replicating` で、自動同期のアクティブ状態は `on` です。プライマリボリュームに書き込みが行われると、StorageTek 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複製モードでない場合は、複製モードにします。

次のコマンドを実行します。StorageTek 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
```

- 4 クライアントマシンにディレクトリを作成します。

- a. クライアントマシンにスーパーユーザーとしてログインします。

次のようなプロンプトが表示されます。

```
client-machine#
```

- b. クライアントマシンにディレクトリを作成します。

```
client-machine# mkdir /dir
```

- 5 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。StorageTek 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。複製は行われません。

- d. PathPrefix ディレクトリが使用可能であることを確認します。

```
nodeC# mount | grep /global/etc
```

- e. 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 -emM nodeC nfs-rg
```

- g. クライアントマシンにスーパーユーザーとしてアクセスします。

次のようなプロンプトが表示されます。

```
client-machine#
```

- h. 手順 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 手順5で表示されたディレクトリが手順6で表示されたディレクトリと同じであることを確認します。
- 8 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 -emM nodeA nfs-rg
```

- d. プライマリボリュームを複製モードに変更します。

次のコマンドを実行します。StorageTek 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、StorageTek 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 [286 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、[260 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。

ここでは、次の手順について説明します。

- 287 ページの「DNS エントリを更新する方法」

## ▼ 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 A-6](#) 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

---

## A

*AffinityOn* プロパティ、データ複製の拡張プロパティ、256  
Availability Suite、データ複製に使用、251

## B

boot コマンド、59-61

## C

cconsole コマンド、「pconsole コマンド」を参照  
claccess コマンド、18  
cldevicegroup コマンド、18  
cldevice コマンド、18  
clinterconnect コマンド、18  
clnasdevice コマンド、18  
clnode check コマンド、18  
clnode コマンド、218, 219-220  
clquorum コマンド、18  
clreslogicalhostname コマンド、18  
clresourcegroup コマンド、18, 219-220  
clresourcetype コマンド、18  
clresource コマンド、18  
リソースとリソースグループの削除、223  
clressharedaddress コマンド、18  
clsetup ユーティリティ、18, 23  
clsnmp host コマンド、18  
clsnmp mib コマンド、18  
clsnmp user コマンド、18

cltelemattribute コマンド、18

cluster check  
コマンド  
への変更、39

cluster check コマンド、18  
vfstab ファイル確認、117

cluster shutdown コマンド、55-64

clzonecluster  
boot、59-61  
説明、23  
停止、55-64

clzonecluster コマンド、18

CPU シェア  
グローバルクラスタの投票ノード、234  
構成、233  
制御、233

## D

DID 情報、手動更新、124-125  
DID 情報の手動更新、124-125  
DR、「動的再構成」を参照

## E

/etc/vfstab ファイル、44  
構成の確認、117  
マウントポイントの追加、116

**F**

failback プロパティ, 101

**G**

GlobalDevicePaths, データ複製の拡張プロパティ, 256

**I**

IPMP

管理, 175

ステータスの確認, 30

**K**

/kernel/drv/, md.conf ファイル, 91

**L**

lofi ファイル, アンインストール, 211

**M**

md.tab ファイル, 20

MIB

SNMP イベントの有効化と無効化, 213

SNMP イベントプロトコルの変更, 214

**N**

ntp.conf.sc ファイル, 201

numsecondaries プロパティ, 102

**O**

OpenBoot PROM (OBP), 198

Oracle Solaris OS

CPU 制御, 233

svcadm コマンド, 199

グローバルクラスタの管理タスク, 16

グローバルクラスタの定義, 15

ゾーンクラスタ定義, 15

ノードの再起動に関する特別な指示, 71-74

ノードのブートに関する特別な指示, 69-71

ホストベースの複製, 80

Oracle Solaris Cluster 定足数サーバー, 定足数デバイスとしてサポート, 132

**P**

pconsole, 保護付きの接続, 23

pconsole コマンド, 23

**R**

raw ディスクデバイス, 命名規則, 116

raw ディスクデバイスグループ, 追加, 93-94

RBAC, 49-53

グローバルクラスタの投票ノードの場合, 50

権限プロファイル (説明), 50-51

タスク

カスタム役割を追加, 52

使用, 49

設定, 49

役割の追加, 51

ユーザーの変更, 53

Role-Based Access Control, 「RBAC」を参照

**S**

SATA, 134

SATA ストレージ, 定足数デバイスとしてサポート, 132

showrev -p コマンド, 24

SNMP

イベント MIB の有効化と無効化, 213

プロトコルの変更, 214

ホストの無効化, 216

## SNMP (続き)

ホストの有効化, 215

ユーザーの削除, 217

ユーザーの追加, 216

SNMP イベント MIB の有効化と無効化, 213

Solaris OS, 「Oracle Solaris OS」を参照

Solaris ボリュームマネージャー, raw ディスクデバイス名, 116

ssh, 23

StorageTek Availability Suite, データ複製に使用, 251

Sun NAS, 定足数デバイスとしてサポート, 132

Sun ZFS Storage Appliance, 定足数デバイスとして追加, 135

Sun 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, 132

## U

/usr/cluster/bin/clresource, リソースグループの削除, 223

/usr/cluster/bin/cluster check コマンド, vfstab ファイル確認, 117

## V

/var/adm/messages ファイル, 77

vfstab ファイル

構成の確認, 117

マウントポイントの追加, 116

## Z

ZFS

デバイスグループ追加, 94

ファイルシステムの削除, 224-226

複製, 94

ルートファイルシステムの制限, 83-84

ZFS Storage Appliance, 「Sun ZFS Storage Appliance 定足数デバイス」を参照

ZPoolsSearchDir, データ複製の拡張プロパティ, 256

## あ

アクセス権、グローバルデバイスの, 82

アダプタ、トランスポート, 165

アップグレード、ブランドタイプが solaris である  
フェイルオーバーゾーン, 238アフィニティスイッチオーバー、データ複製の  
構成, 271

アプリケーションリソースグループ

ガイドライン, 257

データ複製の構成, 273-275

アンインストール、Oracle Solaris Cluster ソフト  
ウェア, 208

アンインストール

lofi デバイスファイル, 211

パッケージ, 241

## い

移行、グローバル名前空間, 88

一覧表示

定足数の構成, 149

デバイスグループ構成, 105

イベント MIB

SNMP の有効化と無効化, 213

SNMP プロトコルの変更, 214

## え

エラーメッセージ

/var/adm/messages ファイル, 77

ノードの削除, 190

## か

開始

グローバルクラスタノード, 65-77

ゾーンクラスタノード, 65-77

ノード, 65-77

概要、定足数, 129-151

確認

vfstab 構成, 117

グローバルマウントポイント, 44, 120

## 確認 (続き)

- データ複製構成, 283-286
- 監視, ディスクパス, 122-123
- 監視解除, ディスクパス, 123-124
- 完全に修復する/var/adm/messages ファイル, 77
- 管理
  - IPMP, 159-177
  - クラスタインターコネクとパブリックネットワーク, 159-177
  - クラスタファイルシステム, 83
  - グローバルクラスタ, 16
  - グローバルクラスタ設定, 191-232
  - ゾーンクラスタ, 16, 222
  - 定足数, 129-151
- 管理コンソール, 20

## き

技術サポート, 14

## 起動

- グローバルクラスタ, 59-61
- グローバルクラスタノード, 65-77
- ゾーンクラスタ, 59-61
- ゾーンクラスタノード, 65-77
- ノード, 65-77
- 共通エージェントコンテナ, ポート番号の変更, 221
- 共有 SCSI ディスク, 定足数デバイスとしてサポート, 132
- 共有ディスクパス
  - 監視, 121-128
  - 自動リブートの無効化, 128
  - 自動リブートの有効化, 127-128
- 切り換え, デバイスグループのプライマリノード, 106-107

## く

## クラスタ

- 時刻の設定, 196
- 名前の変更, 192-193
- ノード認証, 194
- バックアップ, 20, 243-246

## クラスタ (続き)

- ファイルの復元, 246
- クラスタインターコネクと
  - 管理, 159-177
  - 動的再構成, 161
- クラスタコンソールへの保護付きの接続, 23
- クラスタの時刻の設定, 196
- クラスタファイルシステム, 81-128
  - 管理, 83
  - グローバルクラスタの投票ノード, 83
  - 構成の確認, 117
  - 削除, 118-120
  - 追加, 115-118
  - マウントオプション, 117
- クラスタファイルシステムのマウントオプション, 要件, 117
- グローバル
  - デバイス, 81-128
    - アクセス権の設定, 82
    - 動的再構成, 82-83
    - 名前空間, 81-83, 86
    - マウントポイント, 確認, 44, 120
- グローバルクラスタ
  - 管理, 191-232
  - 構成情報の表示, 31-38
  - 構成の検証, 39
  - コンポーネントのステータス, 27
  - 定義, 16
  - 停止, 55-77
  - ノードの削除, 185
  - ブート, 55-77
  - リポート, 61
- グローバルクラスタノード
  - 起動, 65-77
  - 再起動, 71-74
  - シャットダウン, 65-77
- グローバルクラスタの投票ノード
  - CPU シェア, 234
  - クラスタファイルシステムの管理, 83
- グローバル名前空間, 移行, 88
- グローバル名前空間の更新, 86
- グローバルファイルシステム, 「クラスタファイルシステム」を参照

## け

ケーブル、トランスポート、165

権限プロファイル, RBAC, 50-51

## 検索

グローバルクラスタ用ノード ID, 194

ゾーンクラスタ用ノード ID, 194

## 検証

グローバルクラスタ構成, 39

ゾーンクラスタ構成, 39

## こ

## 更新

概要, 237

注意事項, 242

構成, データ複製, 251-287

公平配分スケジューラ, CPU シェアの構成, 234

## コマンド

boot, 59-61

cconsole, 23

claccess, 18

cldevice, 18

cldevicegroup, 18

clinterconnect, 18

clnasdevice, 18

clnode check, 18

clquorum, 18

clreslogicalhostname, 18

clresource, 18

clresourcegroup, 18

clresourcetype, 18

clressharedaddress, 18

clsetup, 18

clsnmp host, 18

clsnmp mib, 18

clsnmp user, 18

cltelemetryattribute, 18

cluster check, 18, 21, 39, 44

cluster shutdown, 55-64

clzonecluster, 18, 55-64

clzonecluster boot, 59-61

clzonecluster verify, 39

メタセット, 81-83

コマンド行管理ツール, 18

コンソール, に接続, 23

## さ

## 再起動

グローバルクラスタノード, 71-74

ゾーンクラスタノード, 71-74

最後の定足数デバイス, 削除, 142

## 削除

SNMP ホスト, 216

SNMP ユーザー, 217

Solaris Volume Manager デバイスグループ, 96

クラスタファイルシステム, 118-120

最後の定足数デバイス, 142

ストレージレイ, 187

すべてのデバイスグループからノードを, 96

ゾーンクラスタから, 184

定足数デバイス, 131, 140

トランスポートケーブル、アダプタ、およびス  
イッチ, 165

ノード, 182, 185

リソースとリソースグループをゾーンクラスタ  
から削除, 223

サポートされている定足数デバイスタイプ, 132

## し

## シャットダウン

グローバルクラスタノード, 65-77

ゾーンクラスタノード, 65-77

ノード, 65-77

修復, 定足数デバイス, 150

使用, 役割 (RBAC), 49

## す

スイッチ、トランスポート, 165

スイッチバック, データ複製で実行するためのガ  
イドライン, 261

## ステータス

グローバルクラスタコンポーネント, 27

ゾーンクラスタコンポーネント, 27

ストレージレイ、削除, 187  
スナップショット、ポイントインタイム, 253

## せ

セカンダリノード  
希望数の設定, 102  
デフォルト数, 101  
設定、役割 (RBAC), 49

## そ

ゾーンクラスタ  
アプリケーション用に準備, 222  
管理, 191-232  
構成情報の表示, 38  
構成の検証, 39  
コンポーネントのステータス, 27  
サポートされている直接マウント, 224-226  
ゾーンパスの移動, 222  
定義, 16  
停止, 55-77  
ファイルシステムの削除, 222  
ブート, 55-77  
複製, 222  
リブート, 61  
ゾーンクラスタノード  
IPアドレスおよびNICの指定, 179-182  
起動, 65-77  
再起動, 71-74  
シャットダウン, 65-77  
ゾーンパス、移動, 222  
属性、「プロパティ」を参照  
ソフトウェアの更新、概要, 237

## た

耐障害性、定義, 252  
タイムアウト、定足数デバイスのデフォルト値の  
変更, 151

## ち

直接接続共有ディスク定足数デバイス、追加, 134  
直接マウント、ファイルシステムをゾーンクラ  
スタにエクスポート, 224-226

## つ

追加  
SNMP ホスト, 215  
SNMP ユーザー, 216  
Solaris Volume Manager デバイスグループ, 93  
Sun ZFS Storage Appliance NAS 定足数デバイ  
ス, 135  
ZFS デバイスグループ, 94  
カスタム役割 (RBAC), 52  
クラスタファイルシステム, 115-118  
直接接続共有ディスク定足数デバイス, 134  
定足数サーバー定足数デバイス, 136  
定足数デバイス, 133  
デバイスグループ, 91, 93-94  
トランスポートケーブル、アダプタ、およびス  
イッチ, 28, 31-38, 160-174  
ノード, 179-182  
ノードをグローバルクラスタへ, 180  
ノードをゾーンクラスタへ, 180  
役割 (RBAC), 51

## て

停止  
グローバルクラスタ, 55-77  
グローバルクラスタノード, 65-77  
ゾーンクラスタ, 55-77  
ゾーンクラスタノード, 65-77  
ノード, 65-77  
ディスクパス  
監視, 81-128  
障害のあるディスクパスを表示, 124  
監視解除, 123-124  
ディスクパスの、ステータスエラーの解  
決, 124-125  
定足数  
概要, 129-151

- 定足数 (続き)
  - 管理, 129-151
- 定足数サーバー, 「定足数サーバー定足数デバイス」を参照
- 定足数サーバー定足数デバイス
  - インストールの要件, 136
  - 削除のトラブルシューティング, 141
  - 追加, 136
- 定足数デバイス
  - 交換, 143
  - 構成の一覧表示, 149
  - 最後の定足数デバイスの削除, 142
  - 削除, 131, 140
  - 修復, 150
  - 追加, 133
    - Sun ZFS Storage Appliance NAS 定足数デバイス, 135
      - 直接接続共有ディスク定足数デバイス, 134
      - 定足数サーバー定足数デバイス, 136
    - デバイスの動的再構成, 131
    - デフォルトのタイムアウトの変更, 151
    - ノードリストの変更, 144
    - 保守状態、デバイス, 146
    - 保守状態、デバイスを保守状態から戻す, 148
- 定足数デバイスタイプ, サポートされているタイプのリスト, 132
- 定足数デバイスの交換, 143
- データ複製, 79-80
  - DNS エントリの更新, 287
  - ガイドライン
    - スイッチオーバーの管理, 260
    - テイクオーバーの管理, 260
    - リソースグループの構成, 255
  - 概要, 252
  - 構成
    - NFS アプリケーション用ファイルシステム, 268-269
    - NFS アプリケーションリソースグループ, 273-275
      - アフィニティスイッチオーバー, 256, 271
      - デバイスグループ, 265
  - 構成の確認, 283-286
  - 構成例, 261
  - 定義, 79-80
- データ複製 (続き)
  - テイクオーバーの管理, 286-287
  - 同期, 253
  - 必要なハードウェアとソフトウェア, 263
  - 非同期, 253
  - ポイントインタイムスナップショット, 253, 282-283
  - ホストベースの, 79-80, 251-287
  - 有効化, 278-281
  - リソースグループ
    - アプリケーション, 257
    - 共有アドレス, 259
    - 構成, 256
    - 作成, 270-271
    - スケーラブルアプリケーション, 258-259
    - フェイルオーバーアプリケーション, 257-258
    - 命名規則, 256
    - リモートミラー, 252, 281-282
    - 例, 281-286
- データ複製の拡張プロパティ
  - アプリケーションリソース, 274, 276
  - 複製リソース, 272
- データ複製のスイッチオーバー
  - アフィニティスイッチオーバー, 256
  - 実行, 286-287
- データ複製のためのスケーラブルアプリケーション
  - セッション, 258-259
- データ複製のフェイルオーバーアプリケーション
  - AffinityOn* プロパティ, 256
  - GlobalDevicePaths*, 256
  - ZPoolsSearchDir*, 256
  - ガイドライン
    - テイクオーバーの管理, 260
    - リソースグループ, 257-258
  - 管理, 286-287
- データ複製用の共有アドレスリソースグループ, 259
- デバイス, グローバル, 81-128
- デバイスグループ
  - raw ディスク
    - 追加, 93-94
  - SVM
    - 追加, 91

## デバイスグループ (続き)

- 管理の概要, 84
  - 構成の表示, 105
  - 削除と登録解除, 96
  - 主所有者権, 101
  - 追加, 93
  - データ複製の構成, 265
  - 変更プロパティ, 101
  - 保守状態, 107
- デバイスグループの主所有者権, 101
- デバイスグループのプライマリノードの切り替え, 106-107
- 電源管理, 191

## と

- 問い合わせ, 14
- 同期データ複製, 253
- 動的再構成, 82-83
  - クラスタインターコネクト, 161
  - 定足数デバイス, 131
  - パブリックネットワークインタフェース, 176
- 登録解除, Solaris Volume Manager デバイスグループ, 96
- ドメインネームシステム (DNS)
  - 更新のガイドライン, 260
  - データ複製での更新, 287
- トランスポートアダプタ, 追加, 28, 31-38, 160-174
- トランスポートケーブル
  - 追加, 28, 31-38, 160-174
  - 無効化, 169
  - 有効化, 168
- トランスポートケーブルの無効化, 169
- トランスポートケーブルの有効化, 168
- トランスポートスイッチ, 追加, 28, 31-38, 160-174

## な

- 名前空間
  - 移行, 88
  - グローバル, 81-83

## ね

- ネットワークファイルシステム (Network File System、NFS), データ複製用アプリケーションファイルシステムの構成, 268-269

## の

- ノード
  - ID 検索, 194
  - 起動, 65-77
  - グローバルクラスタからのノードの削除, 185
  - グローバルクラスタでの名前変更, 202
  - 削除
    - エラーメッセージ, 190
    - シャットダウン, 65-77
    - セカンダリ, 101
    - ゾーンクラスタから削除, 184
    - 追加, 179-182
    - デバイスグループから削除, 96
    - に接続, 23
    - 認証, 194
    - 負荷制限の設定, 219-220
    - プライマリ, 82-83, 101
    - 保守状態にする, 204
  - ノード名の変更
    - グローバルクラスタでの, 202

## は

- バックアップ
  - クラスタ, 20, 243-246
  - ミラーのオンライン, 243
- パッケージ, アンインストール, 241
- パブリックネットワーク
  - 管理, 159-177
  - 動的再構成, 176

## ひ

- 非クラスタモードでのブート, 74
- ビットマップ
  - ポイントインタイムスナップショット, 253



## ビットマップ (続き)

- リモートミラー複製, 252
- 非同期データ複製, 253
- 表示, 障害のあるディスクパス, 124

## ふ

## ファイル

- /etc/vfstab, 44
- md.conf, 91
- md.tab, 20
- ntp.conf.sc, 201

## ファイルシステム

- NFS アプリケーション
  - データ複製の構成, 268-269
- ゾーンクラスタ内で削除, 222
- ルートの復元
  - 説明, 246

## ブート

- グローバルクラスタ, 55-77
- ゾーンクラスタ, 55-77
- 非クラスタモード, 74

## 負荷制限

- concentrate\_load プロパティ, 218
- preemption\_mode プロパティ, 218
- ノード上での構成, 218
- ノード上での設定, 219-220

## 負荷制限の設定, ノード上, 219-220

## 復元

- クラスタファイル, 246
- ルートファイルシステム, 246

## 複製, 「データ複製」を参照

## プライベートホスト名, 変更, 199

## プロパティ

- failback, 101
- numsecondaries, 102
- preferenced, 101

## プロファイル, RBAC 権限, 50-51

## へ

## 変更

- numsecondaries プロパティ, 102

## 変更 (続き)

- SNMP イベント MIB プロトコル, 214
- クラスタ名, 192-193
- 定足数デバイスのノードリスト, 144
- プライベートホスト名, 199
- プライマリノード, 106-107
- プロパティ, 101
- ポート番号, 共通エージェントコンテナの使用, 221
- ユーザー (RBAC), 53

## ほ

## ポイントインタイムスナップショット

- 実行, 282-283
- 定義, 253

## ポート番号, 共通エージェントコンテナを使用した変更, 221

## 保守, 定足数デバイス, 146

## 保守状態

- 定足数デバイス, 146
- 定足数デバイスを保守状態から戻す, 148
- ノード, 204

## ホスト

- SNMP の追加と削除, 215, 216

## ホストベースのデータ複製

- 定義, 79-80
- 例, 251-287

## ま

## マウントポイント

- /etc/vfstab ファイルの修正, 116
- グローバル, 44

## み

## ミラー, オンラインバックアップ, 243

- め  
命名規則  
    raw ディスクデバイス, 116  
    複製リソースグループ, 256  
メタセットコマンド, 81-83
- も  
モニタリング, 共有ディスクパス, 127-128
- や  
役割  
    カスタム役割を追加, 52  
    設定, 49  
    役割の追加, 51
- ゆ  
ユーザー  
    SNMP の削除, 217  
    SNMP の追加, 216  
    プロパティの変更, 53  
ユーザーアカウントツール, 説明, 53
- り  
リソース  
    構成情報の表示, 26-27  
    削除, 223  
リソースグループ  
    データ複製  
        構成, 256  
        構成のガイドライン, 255  
        フェイルオーバーでの役割, 256  
リブート  
    グローバルクラスタ, 61  
    ゾーンクラスタ, 61  
リモートミラー複製  
    実行, 281-282  
    定義, 252  
リモートログイン, 23  
リリース情報, 24
- る  
ループバックマウント, ファイルシステムを  
    ゾーンクラスタにエクスポート, 224-226
- れ  
例  
    インタラクティブな妥当性検査のリスト, 42  
    機能の妥当性検査の実行, 42-43  
    クラスタファイルシステムの作成, 118
- ろ  
ログイン, リモート, 23  
論理ホスト名リソース, データ複製テイク  
    オーバーでの役割, 256