

# 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 RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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に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 と 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 X/Open Company, Ltd. からライセンスされている登録商標です。

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>17</b> |
| Oracle Solaris Cluster の管理の概要 .....                 | 18        |
| ゾーンクラスタに関する作業 .....                                 | 18        |
| Oracle Solaris OS の機能制限 .....                       | 19        |
| 管理ツール .....   | 20        |
| グラフィカルユーザーインターフェース .....                            | 20        |
| コマンド行インターフェース .....                                 | 20        |
| クラスタ管理の準備 .....                                     | 22        |
| Oracle Solaris Cluster ハードウェア構成の記録 .....            | 22        |
| 管理コンソールの使用 .....                                    | 22        |
| クラスタのバックアップ .....                                   | 23        |
| クラスタ管理の開始 .....                                     | 24        |
| ▼ クラスタに遠隔ログインする .....                               | 26        |
| ▼ クラスタコンソールに安全に接続する .....                           | 27        |
| ▼ クラスタ構成ユーティリティにアクセスする .....                        | 28        |
| ▼ Oracle Solaris Cluster のパッチ情報を表示する .....          | 29        |
| ▼ Oracle Solaris Cluster のリリース情報とバージョン情報を表示する ..... | 29        |
| ▼ 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 .....           | 31        |
| ▼ クラスタコンポーネントの状態を確認する .....                         | 33        |
| ▼ パブリックネットワークの状態を確認する .....                         | 35        |
| ▼ クラスタ構成を表示する .....                                 | 36        |
| ▼ 基本的なクラスタ構成を検証する .....                             | 45        |
| ▼ グローバルマウントポイントを確認する .....                          | 50        |
| ▼ Oracle Solaris Cluster のコマンドログの内容を表示する .....      | 52        |

|  |    |
|--|----|
| <b>2 Oracle Solaris Cluster と RBAC</b> .....               | 55 |
| RBAC の設定と Oracle Solaris Cluster での使用 .....                | 55 |
| Oracle Solaris Cluster RBAC の権限プロファイル .....                | 56 |
| 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 ..... | 57 |
| ▼管理役割ツールを使用して役割を作成する方法 .....                               | 57 |
| ▼コマンド行から役割を作成する方法 .....                                    | 59 |
| ユーザーの RBAC プロパティの変更 .....                                  | 61 |
| ▼ユーザーアカウントツールを使用してユーザーの RBAC プロパティを変更する<br>方法 .....        | 61 |
| ▼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 .....                      | 62 |
| <b>3 クラスタの停止と起動</b> .....                                  | 63 |
| クラスタの停止と起動の概要 .....  | 63 |
| ▼クラスタを停止する .....   | 65 |
| ▼クラスタを起動する .....   | 67 |
| ▼クラスタを再起動する .....  | 69 |
| クラスタ内の1つのノードの停止と起動 .....                                   | 73 |
| ▼ノードを停止する .....  | 74 |
| ▼ノードを起動する .....  | 78 |
| ▼ノードを再起動する .....   | 80 |
| ▼非クラスタモードでノードを起動する .....                                   | 84 |
| 満杯の /var ファイルシステムを修復する .....                               | 87 |
| ▼満杯の /var ファイルシステムを修復する .....                              | 87 |
| <b>4 データ複製のアプローチ</b> .....                                 | 89 |
| データ複製についての理解 .....   | 90 |
| サポートされるデータ複製方式 .....                                       | 91 |
| クラスタ内でのストレージベースのデータ複製の使用 .....                             | 92 |
| クラスタ内でストレージベースのデータ複製を使用する際の要件と制限 .....                     | 94 |
| クラスタ内でストレージベースのデータ複製を使用する際の手動回復に関する懸<br>念事項 .....          | 95 |
| ストレージベースのデータ複製を使用する際のベストプラクティス .....                       | 95 |

|   |   |     |
|---|---|-----|
| 5 | グローバルデバイス、ディスクパス監視、およびクラスタファイルシステムの管理 .....                     | 97  |
|   | グローバルデバイスとグローバルな名前空間の管理の概要 .....                                | 97  |
|   | Solaris Volume Manager のグローバルデバイスのアクセス権 .....                   | 98  |
|   | グローバルデバイスでの動的再構成 .....  | 98  |
|   | Veritas Volume Manager による管理に関する注意事項 .....                      | 99  |
|   | ストレージベースの複製されたデバイスの管理 .....                                     | 101 |
|   | Hitachi TrueCopy で複製されたデバイスの管理 .....                            | 101 |
|   | EMC Symmetrix Remote Data Facility で複製したデバイスの管理 .....           | 112 |
|   | クラスタファイルシステムの管理の概要 .....  | 124 |
|   | クラスタファイルシステムの制限事項 .....   | 125 |
|   | VxFS サポートについてのガイドライン .....                                      | 125 |
|   | デバイスグループの管理 .....   | 127 |
|   | ▼ グローバルデバイス名前空間を更新する .....                                      | 129 |
|   | ▼ 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 ....                    | 130 |
|   | グローバルデバイス名前空間を移行する .....  | 131 |
|   | ▼ 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 .....                | 132 |
|   | ▼ 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 .....                 | 133 |
|   | デバイスグループを追加および登録する .....  | 135 |
|   | ▼ デバイスグループを追加および登録する (Solaris Volume Manager) .....             | 135 |
|   | ▼ デバイスグループ (raw ディスク) を追加および登録する .....                          | 137 |
|   | ▼ 複製デバイスグループ (ZFS) の追加と登録方法 .....                               | 138 |
|   | ▼ ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager) .....       | 139 |
|   | デバイスグループ名を維持する .....  | 140 |
|   | デバイスグループを削除して登録を解除する (Solaris Volume Manager) .....             | 140 |
|   | ▼ すべてのデバイスグループからノードを削除する .....                                  | 141 |
|   | ▼ デバイスグループからノードを削除する (Solaris Volume Manager) .....             | 142 |
|   | ▼ ディスクをカプセル化する際に新しいディスクグループを作成する (Veritas Volume Manager) ..... | 144 |
|   | ▼ 新しいボリュームを既存のデバイスグループに追加する (Veritas Volume Manager) .....      | 146 |
|   | ▼ 既存のディスクグループをデバイスグループに変換する (Veritas Volume Manager) .....      | 147 |
|   | ▼ デバイスグループに新しいマイナー番号を割り当てる (Veritas Volume                      |     |

|   |     |
|---|-----|
| Manager) .....  | 147 |
| ▼ ディスクグループをデバイスグループとして登録する (Veritas Volume Manager) ..... | 148 |
| ▼ ディスクグループの構成変更を登録する (Veritas Volume Manager) .....       | 151 |
| ▼ ローカルディスクグループをデバイスグループに変換する (VxVM) .....                 | 152 |
| ▼ デバイスグループをローカルディスクグループに変換する (VxVM) .....                 | 153 |
| ▼ デバイスグループからボリュームを削除する (Veritas Volume Manager) .....     | 154 |
| ▼ デバイスグループを削除して登録を解除する (Veritas Volume Manager) .....     | 155 |
| ▼ デバイスグループにノードを追加する (Veritas Volume Manager) .....        | 156 |
| ▼ デバイスグループからノードを削除する (Veritas Volume Manager) .....       | 157 |
| ▼ raw ディスクデバイスグループからノードを削除する .....                        | 159 |
| ▼ デバイスグループのプロパティを変更する .....                               | 161 |
| ▼ デバイスグループの二次ノードの希望数を設定する .....                           | 163 |
| ▼ デバイスグループ構成の一覧を表示する .....                                | 166 |
| ▼ デバイスグループの主ノードを切り替える .....                               | 167 |
| ▼ デバイスグループを保守状態にする .....                                  | 168 |
| ストレージデバイス用の SCSI プロトコル設定の管理 .....                         | 170 |
| ▼ 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<br>を表示する .....  | 171 |
| ▼ 単一ストレージデバイスの SCSI プロトコルを表示する .....                      | 171 |
| ▼ 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<br>設定を変更する .....  | 172 |
| ▼ 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 .....                      | 174 |
| クラスタファイルシステムの管理 .....                                     | 175 |
| ▼ クラスタファイルシステムを追加する .....                                 | 176 |
| ▼ クラスタファイルシステムを削除する .....                                 | 180 |
| ▼ クラスタ内のグローバルマウントを確認する .....                              | 182 |
| ディスクパス監視の管理 .....   | 182 |
| ▼ ディスクパスを監視する .....                                       | 183 |
| ▼ ディスクパスの監視を解除する方法 .....                                  | 185 |
| ▼ 障害のあるディスクパスを表示する .....                                  | 186 |
| ▼ ディスクパスの状態エラーを解決する .....                                 | 186 |
| ▼ ファイルからディスクパスを監視する .....                                 | 187 |
| ▼ 監視しているすべての共有ディスクパスが失敗したときのノードの自動再起動を<br>有効にする .....     | 189 |
| ▼ すべての監視共有ディスクパスが失敗した場合にノードの自動再起動を無効にす<br>る .....         | 190 |

|          |  |     |
|----------|--|-----|
| <b>6</b> | <b>定足数の管理</b> .....  | 191 |
|          | 定足数デバイスの管理 .....   | 191 |
|          | 定足数デバイスへの動的再構成 .....   | 193 |
|          | 定足数デバイスの追加 .....   | 194 |
|          | 定足数デバイスの削除または交換 .....  | 206 |
|          | 定足数デバイスの保守 .....   | 209 |
|          | 定足数のデフォルトのタイムアウトの変更 .....                                    | 217 |
|          | Oracle Solaris Cluster 定足数サーバーの管理 .....                      | 217 |
|          | Quorum Server Software の起動および停止 .....                        | 218 |
|          | ▼ 定足数サーバーを起動する .....   | 218 |
|          | ▼ 定足数サーバーを停止する .....   | 219 |
|          | 定足数サーバーに関する情報の表示 .....                                       | 220 |
|          | 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 .....                             | 221 |
| <b>7</b> | <b>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> .....                     | 225 |
|          | クラスタインターコネクトの管理 .....  | 225 |
|          | クラスタインターコネクトでの動的再構成 .....                                    | 227 |
|          | ▼ クラスタインターコネクトの状態を確認する .....                                 | 227 |
|          | ▼ Cluster Transport Cable 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 ..... | 228 |
|          | ▼ Cluster Transport Cable 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 ..... | 231 |
|          | ▼ Cluster Transport Cable を有効にする .....                       | 234 |
|          | ▼ Cluster Transport Cable を無効にする .....                       | 235 |
|          | ▼ 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 .....                            | 237 |
|          | ▼ 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 .....               | 238 |
|          | パブリックネットワークの管理 .....   | 241 |
|          | 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 .....                          | 241 |
|          | 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 .....                              | 242 |
| <b>8</b> | <b>ノードの追加と削除</b> .....                                       | 245 |
|          | クラスタへのノードの追加 .....   | 245 |
|          | ▼ ノードを認証ノードリストに追加する .....                                    | 246 |
|          | グローバルクラスタ内の非投票ノード(ゾーン)の作成 .....                              | 249 |

|  |            |
|--|------------|
| クラスタからのノードの削除 .....  | 251        |
| ▼ゾーンクラスタからノードを削除する .....   | 252        |
| ▼クラスタソフトウェア構成からノードを削除する .....                                    | 253        |
| ▼グローバルクラスタから非投票ノード(ゾーン)を削除する .....                               | 256        |
| ▼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 ..                           | 256        |
| ▼エラーメッセージを修正する .....   | 259        |
| <br>   |            |
| <b>9 クラスタの管理 .....</b>   | <b>261</b> |
| クラスタの管理の概要 .....   | 261        |
| ▼クラスタ名を変更する .....  | 262        |
| ▼クラスタ名の変更 (Veritas CVM を使用する場合) .....                            | 264        |
| ▼ノードIDをノード名にマップする .....  | 265        |
| ▼新しいクラスタノード認証で作業する .....   | 265        |
| ▼クラスタの時刻をリセットする .....  | 267        |
| ▼SPARC: ノードで OpenBoot PROM (OBP) を表示する .....                     | 269        |
| ▼ノードのプライベートホスト名を変更する .....                                       | 270        |
| ▼グローバルクラスタ上の非投票ノードのプライベートホスト名を追加する .....                         | 273        |
| ▼グローバルクラスタ上の非投票ノードのプライベートホスト名を変更する .....                         | 273        |
| ▼グローバルクラスタ上の非投票ノードのプライベートホスト名を削除する .....                         | 275        |
| ▼ノード名を変更する .....   | 275        |
| ▼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 ..... | 277        |
| ▼ノードを保守状態にする .....   | 278        |
| ▼ノードを保守状態から戻す .....  | 279        |
| ▼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 .....         | 282        |
| ノードのアンインストールに伴う問題の解決 .....                                       | 284        |
| Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理 .....          | 286        |
| 負荷制限の設定 .....  | 292        |
| ゾーンクラスタ管理タスクの実行 .....  | 294        |
| ▼ゾーンクラスタを削除する .....  | 295        |
| ▼ゾーンクラスタからファイルシステムを削除する .....                                    | 296        |
| ▼ゾーンクラスタからストレージデバイスを削除する .....                                   | 298        |
| 障害追跡 .....   | 300        |
| グローバルクラスタ外でのアプリケーションの実行 .....                                    | 300        |
| 破損したディスクセットの復元 .....   | 302        |



|           |   |     |
|-----------|---|-----|
| <b>10</b> | <b>CPU 使用率の制御の構成</b> .....  | 307 |
|           | CPU 制御の概要 .....   | 307 |
|           | シナリオの選択 .....   | 307 |
|           | 公平配分スケジューラ .....  | 308 |
|           | CPU 制御の構成 .....   | 309 |
|           | ▼ 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 .....                             | 309 |
|           | ▼ デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する .....         | 311 |
|           | ▼ 専用のプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する .....            | 314 |
| <b>11</b> | <b>Oracle Solaris Cluster ソフトウェアとファームウェアのパッチ</b> .....            | 319 |
|           | Oracle Solaris Cluster へのパッチの適用の概要 .....                          | 319 |
|           | Oracle Solaris Cluster パッチの適用に関する注意事項 .....                       | 320 |
|           | Oracle Solaris Cluster ソフトウェアへのパッチの適用 .....                       | 321 |
|           | ▼ 再起動パッチを適用する (ノード) .....   | 321 |
|           | ▼ 再起動パッチを適用する (クラスタ) .....  | 326 |
|           | ▼ 非再起動 Oracle Solaris Cluster パッチを適用する .....                      | 329 |
|           | ▼ フェイルオーバーゾーン構成をしている場合に、シングルユーザーモードでノードにパッチを適用する .....            | 331 |
|           | Oracle Solaris Cluster パッチの変更 .....                               | 335 |
| <b>12</b> | <b>クラスタのバックアップと復元</b> .....                                       | 337 |
|           | クラスタのバックアップ .....   | 337 |
|           | ▼ バックアップするファイルシステム名を確認する .....                                    | 338 |
|           | ▼ 完全なバックアップに必要なテープ数を決定する .....                                    | 338 |
|           | ▼ ルート(/) ファイルシステムをバックアップする .....                                  | 339 |
|           | ▼ ミラーのオンラインバックアップを実行する (Solaris Volume Manager) .....             | 342 |
|           | ▼ ボリュームのオンラインバックアップを実行する (Veritas Volume Manager) .....           | 345 |
|           | ▼ クラスタ構成をバックアップする .....   | 349 |
|           | クラスタファイルの復元の作業マップ .....   | 349 |
|           | ▼ 個々のファイルを対話形式で復元する (Solaris Volume Manager) .....                | 350 |
|           | ▼ ルート(/)ファイルシステムを復元する (Solaris Volume Manager) .....              | 350 |
|           | ▼ Solaris Volume Manager ボリューム上に存在していたルート (/) ファイルシステムを復元する ..... | 353 |
|           | ▼ カプセル化されていないルート (/) ファイルシステムを復元する (Veritas Volume                |     |

|   |     |
|---|-----|
| Manager) .....  | 358 |
| ▼ カプセル化されたルート (/) ファイルシステムを復元する (Veritas Volume Manager) .....          | 360 |
| <b>13</b> グラフィカルユーザーインターフェースによる <b>Oracle Solaris Cluster</b> の管理 ..... | 365 |
| Oracle Solaris Cluster Manager の概要 .....                                | 365 |
| SPARC: Sun Management Center の概要 .....                                  | 366 |
| Oracle Solaris Cluster Manager の構成 .....                                | 367 |
| RBAC の役割の設定 .....   | 367 |
| ▼ 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 .....                    | 367 |
| ▼ Oracle Solaris Cluster Manager のサーバーアドレスを変更する .....                   | 368 |
| ▼ 共通エージェントコンテナのセキュリティー鍵を再生成する .....                                     | 369 |
| Oracle Solaris Cluster Manager ソフトウェアの起動 .....                          | 370 |
| ▼ Oracle Solaris Cluster Manager を起動する .....                            | 370 |
| <b>A</b> 例 .....  | 375 |
| Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成 .....      | 375 |
| クラスタにおける Sun StorageTek Availability Suite ソフトウェアの理解 .....              | 376 |
| 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 .....                                  | 379 |
| 作業マップ: データ複製の構成例 .....  | 385 |
| クラスタの接続とインストール .....  | 386 |
| デバイスグループとリソースグループの構成例 .....   | 388 |
| データ複製の有効化例 .....  | 402 |
| データ複製の実行例 .....   | 405 |
| フェイルオーバーの管理方法の例 .....   | 410 |
| 索引 .....  | 413 |

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

---

このマニュアルは、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。

## UNIX コマンド

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

これらの情報については、次を参照してください。

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

## 表記上の規則

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

表 P-1 表記上の規則

| 字体または記号          | 意味  | 例   |
|------------------|---|---|
| AaBbCc123        | コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。 | .login ファイルを編集します。<br><br>ls -a を使用してすべてのファイルを表示します。<br><br>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><br><br>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。<br><br>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 \<br><br>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。

## 関連マニュアル

関連する Oracle Solaris Cluster トピックについての情報は、以下の表に示すマニュアルを参照してください。すべての Oracle Solaris Cluster ドキュメントは、<http://www.oracle.com/technetwork/indexes/documentation/index.html> で閲覧できます。

| 項目                | ドキュメント  |
|-------------------|---|
| 概念                | 『Oracle Solaris Cluster Concepts Guide』   |
| ハードウェアの設計と管理      | 『Oracle Solaris Cluster 3.3 Hardware Administration Manual』<br>各ハードウェア管理ガイド   |
| ソフトウェアのインストール     | 『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』<br>各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』<br>『Oracle Solaris Cluster Quick Reference』 ( <a href="http://docs.oracle.com/cd/E19680-01/pdf/821-1573.pdf">http://docs.oracle.com/cd/E19680-01/pdf/821-1573.pdf</a> ) |
| 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』<br>『Oracle Solaris Cluster Data Services Reference Manual』  |

Oracle Solaris Cluster ドキュメントの完全なリストについては、<http://wikis.sun.com/display/SunCluster/Home/> で Oracle Solaris Cluster ソフトウェアの使用しているリリースのリリースノートを参照してください。

## ドキュメントおよびサポート

その他のリソースについては、次の Web サイトを参照してください。

- ドキュメント (<http://www.oracle.com/technetwork/indexes/documentation/index.html>)
- サポート (<http://www.oracle.com/us/support/systems/index.html>)

## Oracle ソフトウェアリソース

Oracle Technology Network (<http://www.oracle.com/technetwork/index.html>) では、Oracle ソフトウェアに関する広範なリソースを提供しています。

- ディスカッションフォーラム (<http://forums.oracle.com>) では、技術的な問題と解決方法についての情報交換を行えます。
- Oracle By Example (<http://www.oracle.com/technetwork/tutorials/index.html>) では、手順を追った実践的なチュートリアルを利用できます。

## 問い合わせについて

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

- 名前と電子メールアドレス
- 会社名、住所、および電話番号
- システムのモデルとシリアル番号
- オペレーティングシステムのバージョン番号 (例: Oracle Solaris 10)
- Oracle Solaris Cluster のリリース番号 (例: Oracle Solaris Cluster 3.3)

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              |

---

| コマンド  | 機能  |
|---|---|
| <code>showrev -p</code>                       | インストールされているパッチを報告する                             |
| <code>SPARC:prtdiag -v</code>                 | システム診断情報を表示する                                   |
| <code>/usr/cluster/bin/clnode show-rev</code> | Oracle Solaris Cluster のリリースとパッケージのバージョン情報を表示する |

---

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





# 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。

- 18 ページの「Oracle Solaris Cluster の管理の概要」
- 19 ページの「Oracle Solaris OS の機能制限」
- 20 ページの「管理ツール」
- 22 ページの「クラスタ管理の準備」
- 24 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 10 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のみで構成されますが、オプションで0個以上のグローバルクラスタ非投票ノードを含むこともできます。グローバルクラスタには、オプションで、LINUX OS、またはネイティブブランドの非大域ゾーン(ノードではなく高可用性コンテナ)をリソースとして含めることもできます。ゾーンクラスタには、グローバルクラスタが必須です。ゾーンクラスタの概要については、『[Oracle Solaris Cluster Concepts Guide](#)』を参照してください。

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

## 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。

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

- グローバルクラスタまたはゾーンクラスタを定期的に (または毎日) 管理および維持するための標準的な作業。これらの作業は、このマニュアルで説明されています。
- インストール、構成、属性の変更などのデータサービス作業。これらの作業は、『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作業。これらの作業は、『Oracle Solaris Cluster 3.3 Hardware Administration Manual』で説明されています。

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

## ゾーンクラスタに関する作業

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

次の表に、ゾーンクラスタで実行する主な管理作業を示します。

- ゾーンクラスタの作成 - `clzonecluster configure` コマンドを使用して、新しいゾーンクラスタを作成します。手順については、『[Oracle Solaris Cluster ソフトウェアのインストール](#)』の「[ゾーンクラスタの設定](#)」を参照してください。
- ゾーンクラスタの起動と再起動 - 第3章「[クラスタの停止と起動](#)」を参照してください。
- ゾーンクラスタへのノードの追加 - 第8章「[ノードの追加と削除](#)」を参照してください。
- ゾーンクラスタからのノードの削除 - 252 ページの「[ゾーンクラスタからノードを削除する](#)」を参照してください。
- ゾーンクラスタの構成の表示 - 36 ページの「[クラスタ構成を表示する](#)」を参照してください。
- ゾーンクラスタの構成の検証 - 45 ページの「[基本的なクラスタ構成を検証する](#)」を参照してください。
- ゾーンクラスタの停止 - 第3章「[クラスタの停止と起動](#)」を参照してください。

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

### グラフィカルユーザーインターフェース

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

### コマンド行インターフェース

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

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

- 定足数 (quorum)
- リソースグループ
- データサービス
- クラスタインターコネクト

- デバイスグループとボリューム
- プライベートホスト名
- 新規ノード
- そのほかのクラスタタスク

Oracle Solaris Cluster 構成を管理するほかのコマンドを次のリストに示します。詳細は、マニュアルページを参照してください。

|  |   |
|--|---|
| <code>ccp(1M)</code>                   | クラスタへの遠隔コンソールアクセスを開始します。                            |
| <code>if_mpadm(1M)</code>              | IP ネットワークマルチパスグループ内のあるアダプタから別のアダプタに IP アドレスを切り換えます。 |
| <code>claccess(1CL)</code>             | ノードを追加するために 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>clsnmp(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>clvxdm(1CL)</code>        | Veritas Volume Manager (VxVM) を Oracle Solaris Cluster ノード上で初期化し、状況に応じてルートディスクをカプセル化します。 |
| <code>clzonecluster(1CL)</code> | ゾーンクラスタの作成と変更を行います。  |

さらに、コマンドを使用して 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。使用するコマンドは、クラスタで使用している特定のボリュームマネージャー (Veritas Volume Manager または Solaris ボリュームマネージャー) によって異なります。

## 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構成要素間のケーブルや接続部にラベルを付けておくと、管理作業が簡単になります。

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

### 管理コンソールの使用

「管理コンソール」として専用のワークステーション、または管理ネットワークを介して接続されたワークステーションを使用すると動作中のクラスタを管理できます。通常は、Cluster Control Panel (CCP) と、グラフィカルユーザーインターフェイス (GUI) ツールを管理コンソールにインストールして実行します。CCP の詳細は、[26 ページの「クラスタに遠隔ログインする」](#)を参照してください。Sun Management Center 用の Cluster Control Panel モジュールと Oracle Solaris Cluster Manager GUI ツールをインストールする方法については、『[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。

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

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

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。

注 - グローバルクラスタ投票ノードのみから実行する 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 1-2 Oracle Solaris Cluster 管理ツール

| 作業  | ツール   | 参照先   |
|---|---|---|
| クラスタへの遠隔ログイン                              | ccp コマンドを使用してクラスタコントロールパネル (CCP) を起動します。次に、cconsole、crlogin、cssh、ctelnet の中からいずれかのアイコンを選択します。   | 26 ページの「クラスタに遠隔ログインする」<br>27 ページの「クラスタコントロールに安全に接続する」   |
| 対話形式でのクラスタの構成                             | clzonecluster(1CL) ユーティリティーか clsetup(1CL) ユーティリティーを起動します。   | 28 ページの「クラスタ構成ユーティリティーにアクセスする」  |
| Oracle Solaris Cluster のリリース番号とバージョン情報の表示 | clnode(1CL) コマンドと show-rev -v -node サブコマンドおよびオプションを使用します。   | 29 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」  |
| 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。<br><ul style="list-style-type: none"> <li>■ clresource(1CL)</li> <li>■ clresourcegroup(1CL)</li> <li>■ clresourcetype(1CL)</li> </ul>    | 31 ページの「構成されているリソースタイプ、リソースグループ、リソースを表示する」  |
| クラスタコンポーネントをグラフィカルに監視                     | Oracle Solaris Cluster Manager を使用します。  | オンラインヘルプを参照   |
| いくつかのクラスタコンポーネントをグラフィカルに管理                | Sun Management Center 用の Oracle Solaris Cluster Manager または Oracle Solaris Cluster モジュールを使用します (Sun Management Center は、SPARC ベースシステム上の Oracle Solaris Cluster でのみ利用可)。 | Oracle Solaris Cluster Manager の場合は、オンラインヘルプを参照<br>Sun Management Center の場合は、Sun Management Center のマニュアルを参照 |



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

| 作業                                    | ツール  | 参照先   |
|---------------------------------------|--|---|
| クラスタコンポーネントの状態を確認します。                 | <code>cluster(1CL)</code> コマンドを <code>status</code> サブコマンドとともに使用します。   | 33 ページの「クラスタコンポーネントの状態を確認する」                    |
| パブリックネットワーク上の IP ネットワークマルチパスグループの状態確認 | グローバルクラスタの場合<br>は、 <code>-m</code> オプションを指定して、 <code>clnode(1CL) status</code> コマンドを使用します。<br><br>ゾーンクラスタの場合<br>は、 <code>clzonecluster(1CL) show</code> コマンドを使用します。                    | 35 ページの「パブリックネットワークの状態を確認する」                    |
| クラスタ構成を表示します。                         | グローバルクラスタの場合<br>は、 <code>cluster(1CL)</code> コマンドを <code>show</code> サブコマンドとともに使用します。<br><br>ゾーンクラスタの場合<br>は、 <code>clzonecluster(1CL)</code> コマンドを <code>show</code> サブコマンドとともに使用します。 | 36 ページの「クラスタ構成を表示する」                            |
| 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<br>は、 <code>clzonecluster(1CL)</code> コマンドを <code>show</code> サブコマンドとともに使用します。  | <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<br>は、 <code>cluster(1CL)</code> クラスタ(1CL) コマンドを <code>check</code> サブコマンドとともに使用します。<br><br>ゾーンクラスタの場合<br>は、 <code>clzonecluster(1CL) verify</code> コマンドを使用します。            | 45 ページの「基本的なクラスタ構成を検証する」                        |
| 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。  | 52 ページの「Oracle Solaris Cluster のコマンドログの内容を表示する」 |
| 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。   | 『Solaris のシステム管理 (上級編)』の「システムメッセージの表示」          |
| Solaris Volume Manager の状態の監視         | <code>metastat</code> コマンドを使用します。  | 『Solaris ボリュームマネージャの管理』                         |

## ▼ クラスタに遠隔ログインする

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

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

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

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

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

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

- `SUNWccn` パッケージを管理コンソール上にインストールします。
- 管理コンソールの `PATH` 変数に、Oracle Solaris Cluster ツールのディレクトリ `/opt/SUNWcluster/bin` と `/usr/cluster/bin` が含まれることを確認します。ツールのディレクトリには、`$CLUSTER_HOME` 環境変数を設定することで別の場所を指定できます。
- 端末集配信装置を使用している場合は、`clusters` ファイル、`serialports` ファイル、`nsswitch.conf` ファイルを構成します。これらのファイルは、`/etc` 内ファイルまたは `NIS/NIS+` データベースのどちらでもかまいません。詳細は、`clusters(4)` と `serialports(4)` のマニュアルページを参照してください。

1 **Sun Enterprise 10000** サーバープラットフォームを使用している場合は、システムサービスプロセッサ (SSP) にログインします。

a. `netcon` コマンドを使用して接続する。

- b. 接続が完了したなら、**Shift+@**キーを入力してコンソールのロックを解除し、書き込み権を取得します。
- 2 管理コンソールから**CCP**起動パッドを起動します。  

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

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

## ▼ クラスタコンソールに安全に接続する

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

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

---

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

---

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

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

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

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

## ▼ クラスタ構成ユーティリティーにアクセスする

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

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

この手順では、長形式の 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上でスーパーユーザーになります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 2 構成ユーティリティーを起動します。

```
phys-schost# clsetup
```

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

```
phys-schost# clsetup
```

メインメニューが表示されます。

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

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

ユーティリティーで実行可能な操作は、次のオプションで確認できます。

```
clzc:sczone> ?
```

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

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

## ▼ Oracle Solaris Cluster のパッチ情報を表示する

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

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

この手順では、長形式の 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 のパッチ情報の表示:

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

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

### 例 1-1 Oracle Solaris Cluster のパッチ情報の表示

次に、パッチ 110648-05 についての情報を表示した例を示します。

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

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

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

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

この手順では、長形式の 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 1-2 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、すべてのパッケージのクラスタのリリース情報とバージョン情報の例を示します。

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

phys-schost#% clnode show-rev -v
Oracle Solaris Cluster 3.3 for Solaris 10 sparc
SUNWscu: 3.3.0,REV=2010.06.14.03.44
SUNWsccomu: 3.3.0,REV=2010.06.14.03.44
SUNWsczr: 3.3.0,REV=2010.06.14.03.44
SUNWsccomzu: 3.3.0,REV=2010.06.14.03.44
SUNWsczu: 3.3.0,REV=2010.06.14.03.44
SUNWscsckr: 3.3.0,REV=2010.06.14.03.44
SUNWscscku: 3.3.0,REV=2010.06.14.03.44
SUNWscr: 3.3.0,REV=2010.06.14.03.44
SUNWscrtlh: 3.3.0,REV=2010.06.14.03.44
SUNWscnmr: 3.3.0,REV=2010.06.14.03.44
SUNWscnmu: 3.3.0,REV=2010.06.14.03.44
SUNWscdev: 3.3.0,REV=2010.06.14.03.44
SUNWscgds: 3.3.0,REV=2010.06.14.03.44
SUNWscsmf: 3.3.0,REV=2010.06.14.03.44
SUNWscman: 3.3.0,REV=2010.05.21.18.40
SUNWscsal: 3.3.0,REV=2010.06.14.03.44
SUNWscsam: 3.3.0,REV=2010.06.14.03.44
SUNWscvm: 3.3.0,REV=2010.06.14.03.44
SUNWmdmr: 3.3.0,REV=2010.06.14.03.44
SUNWmdmu: 3.3.0,REV=2010.06.14.03.44
SUNWscmasa: 3.3.0,REV=2010.06.14.03.44
SUNWscmasar: 3.3.0,REV=2010.06.14.03.44
SUNWscmasasen: 3.3.0,REV=2010.06.14.03.44
SUNWscmasazu: 3.3.0,REV=2010.06.14.03.44
SUNWscmasau: 3.3.0,REV=2010.06.14.03.44
SUNWscmautil: 3.3.0,REV=2010.06.14.03.44
SUNWscmautilr: 3.3.0,REV=2010.06.14.03.44
SUNWjfreechart: 3.3.0,REV=2010.06.14.03.44
SUNWscspmr: 3.3.0,REV=2010.06.14.03.44
SUNWscspmu: 3.3.0,REV=2010.06.14.03.44
SUNWscderby: 3.3.0,REV=2010.06.14.03.44
SUNWsc telemetry: 3.3.0,REV=2010.06.14.03.44
SUNWscgrepavs: 3.2.3,REV=2009.10.23.12.12
SUNWscgrepsrdf: 3.2.3,REV=2009.10.23.12.12
SUNWscgreptc: 3.2.3,REV=2009.10.23.12.12
SUNWscghb: 3.2.3,REV=2009.10.23.12.12
SUNWscgctl: 3.2.3,REV=2009.10.23.12.12
SUNWscims: 6.0,REV=2003.10.29
SUNWscics: 6.0,REV=2003.11.14
SUNWscapc: 3.2.0,REV=2006.12.06.18.32
SUNWscdns: 3.2.0,REV=2006.12.06.18.32
SUNWschadb: 3.2.0,REV=2006.12.06.18.32
SUNWschtt: 3.2.0,REV=2006.12.06.18.32
SUNWscslas: 3.2.0,REV=2006.12.06.18.32
SUNWscskb5: 3.2.0,REV=2006.12.06.18.32
SUNWscnfs: 3.2.0,REV=2006.12.06.18.32
SUNWscor: 3.2.0,REV=2006.12.06.18.32
```

```

SUNWscs1mq: 3.2.0,REV=2006.12.06.18.32
SUNWscsap: 3.2.0,REV=2006.12.06.18.32
SUNWscslc: 3.2.0,REV=2006.12.06.18.32
SUNWscsapdb: 3.2.0,REV=2006.12.06.18.32
SUNWscsapenq: 3.2.0,REV=2006.12.06.18.32
SUNWscsaprepl: 3.2.0,REV=2006.12.06.18.32
SUNWscsapscs: 3.2.0,REV=2006.12.06.18.32
SUNWscsapwebas: 3.2.0,REV=2006.12.06.18.32
SUNWscsbl: 3.2.0,REV=2006.12.06.18.32
SUNWscsyb: 3.2.0,REV=2006.12.06.18.32
SUNWscwls: 3.2.0,REV=2006.12.06.18.32
SUNWsc9ias: 3.2.0,REV=2006.12.06.18.32
SUNWscPostgreSQL: 3.2.0,REV=2006.12.06.18.32
SUNWsczone: 3.2.0,REV=2006.12.06.18.32
SUNWscdhc: 3.2.0,REV=2006.12.06.18.32
SUNWscsceb: 3.2.0,REV=2006.12.06.18.32
SUNWscmqi: 3.2.0,REV=2006.12.06.18.32
SUNWscmqs: 3.2.0,REV=2006.12.06.18.32
SUNWscmys: 3.2.0,REV=2006.12.06.18.32
SUNWscsge: 3.2.0,REV=2006.12.06.18.32
SUNWscsaa: 3.2.0,REV=2006.12.06.18.32
SUNWscsag: 3.2.0,REV=2006.12.06.18.32
SUNWscsmb: 3.2.0,REV=2006.12.06.18.32
SUNWscsps: 3.2.0,REV=2006.12.06.18.32
SUNWscTomcat: 3.2.0,REV=2006.12.06.18.32

```

## ▼ 構成されているリソースタイプ、リソースグループ、リソースを表示する

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

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

この手順では、長形式の 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。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

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

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

- `resource`
- `resource group`
- `resourcetype`

### 例 1-3 構成されているリソースタイプ、リソースグループ、リソースの表示

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

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

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

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

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

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

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

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



## ▼ クラスタコンポーネントの状態を確認する

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

---

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

---

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

この手順では、長形式の 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状態を確認します。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

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

### 例 1-4 クラスタコンポーネントの状態確認

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

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

--- Node Status ---

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

=== Cluster Transport Paths ===

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

=== Cluster Quorum ===
```

--- Quorum Votes Summary ---

| Needed | Present | Possible |
|--------|---------|----------|
| -----  | -----   | -----    |
| 3      | 3       | 4        |

--- Quorum Votes by Node ---

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

--- Quorum Votes by Device ---

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

=== Cluster Device Groups ===

--- Device Group Status ---

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

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

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

=== Cluster Resource Groups ===

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

=== Cluster Resources ===

| Resource Name | Node Name     | Status  | Message |
|---------------|---------------|---------|---------|
| -----         | -----         | -----   | -----   |
| test_1        | phys-schost-1 | Offline | Offline |

```

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

```

```

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

```

=== Zone Clusters ===

--- Zone Cluster Status ---

```

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

```

## ▼ パブリックネットワークの状態を確認する

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

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

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

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

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

### 例 1-5 パブリックネットワークの状態を調べる

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

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

Node Name          Group Name      Status  Adapter  Status
-----
phys-schost-1     test-rg        Online  qfe1     Online
phys-schost-2     test-rg        Online  qfe1     Online
```

## ▼ クラスタ構成を表示する

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

`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。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
% cluster show
```

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

ゾーンクラスタのみに関する構成情報を表示するには、`clzonecluster show` コマンドも使用できます。ゾーンクラスタのプロパティには、ゾーンクラスタ名、IP タイプ、自動起動、ゾーンパスなどがあります。`show` サブコマンドは、ゾーンクラスタの内部で実行され、そのゾーンクラスタのみが対象になります。ゾーンクラスタ

ノードから `clzonecluster show` コマンドを実行すると、そのゾーンクラスタから認識可能なオブジェクトのみの状態が得られます。

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

### 例 1-6 グローバルクラスタの構成を表示する

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

```
phys-schost# cluster show

=== Cluster ===

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

=== Host Access Control ===

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

=== Cluster Nodes ===

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

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

Transport Adapter:           qfe3
  Adapter State:              Enabled
  Adapter Transport Type:     dlpi
```

```

Adapter Property(device_name):          qfe
Adapter Property(device_instance):      3
Adapter Property(lazy_free):            1
Adapter Property(dlpi_heartbeat_timeout): 10000
Adapter Property(dlpi_heartbeat_quantum): 1000
Adapter Property(nw_bandwidth):         80
Adapter Property(bandwidth):            10
Adapter Property(ip_address):           172.16.1.1
Adapter Property(netmask):              255.255.255.128
Adapter Port Names:                     0
Adapter Port State(0):                  Enabled

```

```

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

```

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

```

SNMP MIB Name:                          Event
State:                                   Disabled
Protocol:                                SNMPv2

```

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

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

```

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

```

```

Node Name:                               phys-schost-2:za
Node ID:                                  2
Type:                                     cluster
Enabled:                                  yes
privatehostname:                         clusternode2-priv
reboot_on_path_failure:                  disabled
globalzoneshares:                       1
defaultpsetmin:                          2
quorum_vote:                             1
quorum_defaultvote:                     1
quorum_resv_key:                         0x43CB1E1800000002
Transport Adapter List:                   hme0, qfe3

```

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

```

Transport Adapter:                      hme0
Adapter State:                          Enabled

```

```

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

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

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

SNMP MIB Name:                          Event
State:                                  Disabled
Protocol:                               SNMPv2

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

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

=== Transport Cables ===

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

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

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

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

```

```

Cable State:                               Enabled

=== Transport Switches ===

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

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

=== Quorum Devices ===

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

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

=== Device Groups ===

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

=== Registered Resource Types ===

Resource Type:                              SUNW.LogicalHostname:2
RT_description:                             Logical Hostname Resource Type
RT_version:                                  2
API_version:                                 2
RT_basedir:                                  /usr/cluster/lib/rgm/rt/hafoip
Single_instance:                             False
Proxy:                                       False

```



```

Init_nodes:                All potential masters
Installed_nodes:          <All>
Failover:                 True
Pkglist:                  SUNWscu
RT_system:                True

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

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

Resource Type:            SUNW.haderby
RT_description:           haderby server for 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:                  SUNWscderby
RT_system:                False

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:                  SUNWscctelemetry
RT_system:                False

```

=== Resource Groups and Resources ===

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

--- Resources for Group HA\_RG ---

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

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

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

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

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

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

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

```

Enabled{phys-schost-2}:          True
Monitored{phys-schost-1}:       True
Monitored{phys-schost-2}:       True

=== DID Device Instances ===

DID Device Name:                 /dev/did/rdisk/d1
Full Device Path:                phys-schost-1:/dev/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:                            netapp
User ID:                          root

Nas Device:                      nas2
Type:                            netapp
User ID:                          llai

```

### 例 1-7 ゾーンクラスタの構成を表示する

次に、ゾーンクラスタ構成のプロパティの例を示します。

```

% clzonecluster show
=== Zone Clusters ===

Zone Cluster Name:                sczone

```

```

zonename:                sczone
zonewidth:                /zones/sczone
autoboot:                TRUE
ip-type:                 shared
enable_priv_net:        TRUE

--- Solaris Resources for sczone ---

Resource Name:          net
address:                172.16.0.1
physical:               auto

Resource Name:          net
address:                172.16.0.2
physical:               auto

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

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

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

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

--- Zone Cluster Nodes for sczone ---

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

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

```

`clnasdevice show` サブコマンドまたは Oracle Solaris Cluster Manager を使用して、グローバルまたはゾーンクラスタ用に設定された 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

`cluster(1CL)` 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せん。`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。

Oracle Solaris Cluster 3.3 5/11 リリース以降は、`cluster check` コマンドに新しいタイプの検査が導入され機能強化が施されています。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、`-k` キーワード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(1CL)` コマンドをグローバルクラスタの投票ノードで実行すると、ゾーンクラスタが正しく機能するために必要な構成を検証する一連のチェックが実行されます。すべてのチェックでエラーがなかった場合、`clzonecluster verify` はシェルプロンプトに戻ります(その場合は、ゾーンクラスタを安全にインストールできます)。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上でスーパーユーザーになります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。

```
phys-schost# su
```

- 2 最新版の検査プログラムがあるかどうかを確認します。

「[My Oracle Support \(オラクルサポート\)](#)」の「Patches & Updates (購入および更新)」タブに移動します。「Advanced Search (高度な検索)」を使用して「Product (製品)」として「Solaris Cluster」を選択し、「Description (説明)」フィールドで「Check (チェック)」を指定して、検査プログラムが収録されている Oracle Solaris Cluster パッチの場所を指定します。クラスタ上にまだインストールしていないパッチがあれば適用します。

- 3 基本検証検査を実行します。

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

```
-v          詳細モード
```

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

このコマンドによって、利用可能なすべての基本検査が実行されます。クラスタ機能はいっさい影響を受けません。

- 4 対話型検証検査を実行します。

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

```
-k interactive 対話型検証検査の実行を指定します。
```

このコマンドによって、利用可能なすべての対話型検査が実行され、必要なクラスタ情報の入力を促すプロンプトが表示されます。クラスタ機能はいっさい影響を受けません。

## 5 機能検証検査を実行します。

- a. 利用可能なすべての機能検査が非冗長モードで一覧表示されます。

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

- b. 本稼働環境でクラスタの利用またはサービスを中断するアクションが、どの機能検査によって実行されるかを確認します。

たとえば、機能検査によってはノードのパニック、あるいは別のノードへのフェイルオーバーが発生する場合があります。

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

`-C checkID` 特定の検査を指定します。

- c. 実行の対象となる機能検査によってクラスタの動作が中断される場合は、そのクラスタがまだ本稼働段階に達していないものであることを確認してください。

- d. 機能検査を実行します。

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

`-k functional` 機能検証検査の実行を指定します。

検査のプロンプトに応答し、検査の実行を確認し、実行すべき情報またはアクションについても確認します。

- e. 実行する必要のある残りの各機能検査について、**手順 c** および **手順 d** を繰り返します。

---

注 - 記録管理のために、実行する検査ごとに、他と重複しない `outputdir` サブディレクトリ名を指定してください。 `outputdir` の名前を再利用してしまうと、再利用された `outputdir` サブディレクトリの既存の内容が、新しい検査の出力で上書されます。

---

- ## 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」を参照してください。

### 例 1-8 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 1-9 対話型検証検査の一覧表示

次の例は、クラスタに対して実行できるすべての対話型検証検査の一覧です。出力例は可能な検査のサンプルであり、実際の利用可能な検査は各構成に応じて異なります。

```
# 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 1-10 機能検証検査の実行

次の例では、最初に機能検査の詳細が表示されています。次に、検査 F6968101 の一覧表示に関する詳細説明が示されています。この例を見ると、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。このクラスタは稼働しなくなっています。この場合、機能検査を実行すると、`funct.test.F6968101.12Jan2011` サブディレクトリにログの詳細が出力されます。出力例は可能な検査のサンプルであり、実際の利用可能な検査は各構成に応じて異なります。

```
# 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 1-11 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 が無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 3.x nodes.
ANALYSIS : The global filesystem /etc/vfstab entries are not consistent across
all nodes in this cluster.
Analysis indicates:
FileSystem '/global/phys-schost-1' is on 'phys-schost-1' but missing from 'phys-schost-2'.
RECOMMEND: Ensure each node has the correct /etc/vfstab entry for the
filesystem(s) in question.
...
#

```

## ▼ グローバルマウントポイントを確認する

`cluster(ICL)` 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。

---

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

---

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

```
% su
```

- 2 グローバルクラスタ構成を検証します。

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

### 例 1-12 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 3.x nodes.
ANALYSIS : The global filesystem /etc/vfstab entries are not consistent across
all nodes in this cluster.
Analysis indicates:
FileSystem '/global/phys-schost-1' is on 'phys-schost-1' but missing from 'phys-schost-2'.
RECOMMEND: Ensure each node has the correct /etc/vfstab entry for the
filesystem(s) in question.
...
#
# cat /var/cluster/logs/cluster_check/Dec5/cluster_check-results.phys-schost-1.txt
...
=====
= ANALYSIS DETAILS =
=====
-----
CHECK ID : 1398
SEVERITY : HIGH
FAILURE : An unsupported server is being used as an Oracle Solaris Cluster 3.x node.
ANALYSIS : This server may not been qualified to be used as an Oracle Solaris Cluster 3.x node.
```

Only servers that have been qualified with Oracle Solaris Cluster 3.x are supported as Oracle Solaris Cluster 3.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 3.x.

...  
#

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

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

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

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

- `claccess`
- `cldevice`
- `cldevicegroup`
- `clinterconnect`
- `clnasdevice`
- `clnode`
- `clquorum`
- `clreslogicalhostname`
- `clresource`
- `clresourcegroup`
- `clresourcetype`
- `clressharedaddress`
- `clsetup`
- `clsnmpghost`
- `clsnmpmib`
- `clnsmuser`
- `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す。`commandlog` ファイルのアーカイブポリシーを変更するには、クラスタ内の各ノード上で `crontab` コマンドを使用します。詳細は、[crontab\(1\)](#) のマニュアルページを参照してください。

Oracle Solaris Cluster ソフトウェアは任意の時点で、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 1-13 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。

- 55 ページの「RBAC の設定と Oracle Solaris Cluster での使用」
- 56 ページの「Oracle Solaris Cluster RBAC の権限プロファイル」
- 57 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」
- 61 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マニュアルを選んでください。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。

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

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

Oracle Solaris Cluster Manager と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投票ノードおよび非投票ノードの両方に適用されます。

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

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(1HA)</code></p> <p><code>scha_resource_get(1HA)</code></p> <p><code>scha_resource_setstatus(1HA)</code></p> <p><code>scha_resourcegroup_get(1HA)</code></p> <p><code>scha_resourcetype_get(1HA)</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、およびそのほかの読み取り操作の実行、ならびに Oracle Solaris Cluster Manager GUI へのアクセス。 |
| 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>                                    | Oracle Solaris Cluster Manager にアクセスするだけでなく、一覧表示、表示、エクスポート、状態、およびその他の読み取り操作の実行。                       |
|                        | <code>solaris.cluster.admin</code>                                   | クラスタオブジェクトの状態の変更。   |
| システム管理者                | この既存の 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

この作業では、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。

### ▼ 管理役割ツールを使用して役割を作成する方法

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

#### 1 管理役割ツールを起動します。

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

- 2 「管理役割を追加 (Add Administrative)」ウィザードが起動します。  
「アクション (Action)」メニューから「管理役割を追加 (Add Administrative Role)」を選択して、「管理役割を追加 (Add Administrative)」ウィザードを起動します。

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

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

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

---

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

---

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

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

- 5 「**Finish (完了)**」をクリックします。

- 6 端末ウィンドウを開き、**root** になります。

- 7 ネームサービスキャッシュデーモンを起動して停止します。

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

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

## ▼ コマンド行から役割を作成する方法

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

- 2 次のいずれかの役割の作成方法を選択します。

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

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

### 3 ネームサービスキャッシュデーモンを起動して停止します。

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

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

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

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

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

Authenticating as user: primaryadmin

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

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

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

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

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

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

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

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

## ユーザーのRBACプロパティの変更

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

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

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

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

- 1 ユーザーアカウントツールを起動します。

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

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

---

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

---

## ▼ コマンド行からユーザーのRBACプロパティーを変更する方法

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

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

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

## クラスタの停止と起動

---

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

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

この章の関連手順の詳細な説明については、84 ページの「非クラスタモードでノードを起動する」と表 3-2 を参照してください。

### クラスタの停止と起動の概要

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

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

注- グローバルクラスタ全体を正しく停止するには、`cluster shutdown` コマンドを使用します。Oracle Solaris の `shutdown` コマンドは `clnode(1CL) evacuate` コマンドとともに使用して、個々のノードを停止します。詳細は、65 ページの「クラスタを停止する」または73 ページの「クラスタ内の1つのノードの停止と起動」を参照してください。

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

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

注- 必要であれば、ノードを非クラスタモードで (つまり、ノードがクラスタメンバーシップを取得しないように) 起動できます。非クラスタモードは、クラスタソフトウェアをインストールしたり、特定の管理手順を実行する際に役立ちます。詳細は、84 ページの「非クラスタモードでノードを起動する」を参照してください。

表 3-1 作業リスト: クラスタの停止と起動

| 作業  | 参照先                 |
|---|---------------------|
| クラスタの停止   | 65 ページの「クラスタを停止する」  |
| すべてのノードを起動してクラスタを起動<br>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。 | 67 ページの「クラスタを起動する」  |
| クラスタの再起動  | 69 ページの「クラスタを再起動する」 |



## ▼ クラスタを停止する

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



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

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

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

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

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

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

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

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

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

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

### 例 3-1 ゾーンクラスタの停止

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

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

### 例 3-2 SPARC: グローバルクラスタの停止

次に、正常なグローバルクラスタの動作を停止して、すべてのノードを停止し、**ok** プロンプトが表示されたときのコンソールの出力例を示します。ここでは、**-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、[67 ページ](#)の「[クラスタを起動する](#)」を参照してください。

## ▼ クラスタを起動する

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

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

この手順では、長形式の 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「[ゾーンクラスタの設定](#)」の手順を実行します。

---

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

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot
```

- x86 ベースのシステム上で、次のコマンドを実行します。

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

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

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

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

---

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

---

- ゾーンクラスタが1つの場合は、ゾーンクラスタ全体を起動できます。  
`phys-schost# clzonecluster boot zoneclustername`
- ゾーンクラスタが複数ある場合は、すべてのゾーンクラスタを起動できます。 `zoneclustername` の代わりに + を使用してください。

- 2 ノードが問題なく起動し、オンラインであることを確認します。

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

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

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

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

注- ノードの `/var` ファイルシステムが満杯になると、そのノード上では 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、[87 ページの「満杯の /var ファイルシステムを修復する」](#)を参照してください。

### 例 3-4 SPARC: グローバルクラスタの起動

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

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

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

## ▼ クラスタを再起動する

グローバルクラスタを停止するために `cluster shutdown` コマンドを実行してから、各ノード上で `boot` コマンドを使用してグローバルクラスタを起動します。ゾーンクラスタを停止するために `clzonecluster halt` コマンドを使用してから、`clzonecluster boot` コマンドを使用してゾーンクラスタを起動します。`clzonecluster reboot` コマンドを使用することもできます。詳細は、[cluster\(ICL\)](#)、[boot\(IM\)](#)、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 製品のマニュアルを参照してください。
- 2 クラスタ内の任意のノードで、スーパーユーザーになるか、**RBAC** の承認 **solaris.cluster.admin** を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 クラスタを停止します。

- グローバルクラスタを停止します。

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

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

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

各ノードが停止します。ゾーンクラスタ内で **cluster shutdown** コマンドを使用し、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トとの動作中の接続が必要です。

---

- 4 各ノードを起動します。

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

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

```
ok boot
```

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

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

```

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

```

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

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

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

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

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

- 5 ノードが問題なく起動し、オンラインであることを確認します。

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

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

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

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

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

注-ノードの /var ファイルシステムが満杯になると、そのノード上では 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、87 ページの「満杯の /var ファイルシステムを修復する」を参照してください。

### 例 3-5 ゾーンクラスタの再起動

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

```

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

phys-schost#
phys-schost# clzonecluster status

=== Zone Clusters ===

--- Zone Cluster Status ---

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

```

### 例 3-6 SPARC: グローバルクラスタの再起動

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

```

phys-schost# cluster shutdown -g0 -y
Wed Mar 10 13:47:32 phys-schost-1 cl_runtime:
WARNING: CMM monitoring disabled.
phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
...
The system is down.
syncing file systems... done
Program terminated
ok boot
Rebooting with command: boot
...
Hostname: phys-schost-1
Booting as part of a cluster
...

```



```

NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-2 (incarnation # 937690106) has become reachable.
NOTICE: Node phys-schost-3 (incarnation # 937690290) has become reachable.
NOTICE: cluster has reached quorum.
...
NOTICE: Cluster members: phys-schost-1 phys-schost-2 phys-schost-3.
...
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
checking ufs filesystems
...
reservation program successfully exiting
Print services started.
volume management starting.
The system is ready.
phys-schost-1 console login:

```

## クラスタ内の1つのノードの停止と起動

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

グローバルクラスタノードを停止するには、`clnode evacuate` コマンドを 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。

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

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

表 3-2 作業マップ: ノードの停止と起動

| 作業   | ツール  | 参照先                        |
|--|--|----------------------------|
| ノードの停止。  | グローバルクラスタノードの場合は、 <code>clnode(1CL)</code><br><code>evacuate</code> および <code>shutdown</code> を使用。<br>ゾーンクラスタノードの場合は、 <code>clzonecluster(1CL)</code><br><code>halt</code> を使用。  | 74 ページの「ノードを停止する」          |
| ノードの起動。<br>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。                  | グローバルクラスタノードの場合は、 <code>boot</code> または <code>b</code> を使用。<br>ゾーンクラスタノードの場合は、 <code>clzonecluster(1CL)</code><br><code>boot</code> を使用。   | 78 ページの「ノードを起動する」          |
| クラスタ上のノードをいったん停止してから再起動。<br>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。 | グローバルクラスタノードの場合は、 <code>clnode</code><br><code>evacuate</code> および <code>shutdown</code> を使用してから、 <code>boot</code> または <code>b</code> を使用。<br>ゾーンクラスタノードの場合は、 <code>clzonecluster(1CL)</code><br><code>reboot</code> を使用。                 | 80 ページの「ノードを再起動する」         |
| ノードがクラスタメンバーシップを取得しないようにノードを起動。  | グローバルクラスタノードの場合は、 <code>clnode</code><br><code>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<br>(SPARC または x86 の GRUB メニューエントリ編集で)。<br><br>基になるグローバルクラスタが非クラスタモードで起動される場合は、ゾーンクラスタノードも自動的に非クラスタモードになります。 | 84 ページの「非クラスタモードでノードを起動する」 |

## ▼ ノードを停止する

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

この手順では、長形式の 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 製品のマニュアルを参照してください。
- 2 停止するクラスタノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.admin` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 特定のゾーンクラスタメンバーを停止する場合は、手順 4～6 をスキップし、グローバルクラスタノードから次のコマンドを実行します。  
`phys-schost# clzonecluster halt -n physical-name zoneclustername`  
特定のゾーンクラスタノードを指定すると、そのノードのみが停止します。`halt` コマンドは、デフォルトではすべてのノード上のゾーンクラスタを停止します。
- 4 すべてのリソースグループ、リソース、およびデバイスグループを、停止するノードから別のグローバルクラスタノードに切り替えます。  
停止するグローバルクラスタノードで、次のようにコマンドを入力します。`clnode evacuate` コマンドは、非大域ゾーンを含むすべてのリソースグループとデバイスグループを、指定ノードから次に優先されるノードに切り替えます (ゾーンクラスタノード内で `clnode evacuate` を実行することもできます)。  
`phys-schost# clnode evacuate node`  
`node` リソースグループとデバイスグループを切り替えるノードを指定します。
- 5 ノードを停止します。  
停止するグローバルクラスタノードを指定します。  
`phys-schost# shutdown -g0 -y -i0`  
SPARC ベースのシステムではグローバルクラスタノードが `ok` プロンプトを表示し、x86 ベースのシステムでは GRUB メニューで「Press any key to continue」というメッセージが表示されていることを確認します。
- 6 必要であればノードの電源を切ります。

### 例 3-7 SPARC: グローバルクラスタノードの停止

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

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

### 例 3-8 x86: グローバルクラスタノードの停止

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

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

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

phys-schost-1#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
/etc/rc0.d/K05initrgm: Calling 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、78 ページの「ノードを起動する」を参照してください。

## ▼ ノードを起動する

グローバルクラスタまたはゾーンクラスタの他のアクティブなノードを停止または再起動する場合は、起動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、停止または再起動するクラスタ内の他のノードからサービスを引き継ぎません。非大域ゾーンの起動については、『Oracle Solaris のシステム管理 (Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン)』の第 20 章「非大域ゾーンのインストール、起動、停止、アンインストール、および複製(手順)」を参照してください。

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

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

この手順では、長形式の 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。GRUB メニューは次のようになっています。

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

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

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

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

- ゾーンクラスタがある場合は、起動するノードを指定できます。

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

## 2 ノードが問題なく起動し、オンラインであることを確認します。

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

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

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

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

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

---

注-ノードの /var ファイルシステムが満杯になると、そのノード上では 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、87 ページの「満杯の /var ファイルシステムを修復する」を参照してください。

---

### 例 3-10 SPARC: グローバルクラスタノードの起動

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

```
ok boot
Rebooting with command: boot
...
Hostname: phys-schost-1
Booting as part of a cluster
...
NOTICE: Node phys-schost-1: attempting to join cluster
...
NOTICE: Node phys-schost-1: joined cluster
...
The system is coming up. Please wait.
```

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

## ▼ ノードを再起動する

グローバルクラスタまたはゾーンクラスタの他のアクティブなノードを停止または再起動するには、再起動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、停止または再起動するクラスタ内の他のノードからサービスを引き継げません。非大域ゾーンの再起動については、『Oracle Solaris のシステム管理 (Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン)』の第 20 章「非大域ゾーンのインストール、起動、停止、アンインストール、および複製 (手順)」を参照してください。

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

この手順では、長形式の 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、ノードが再起動されます。Failover\_mode プロパティがそれ以外の値に設定されている場合、ノードは再起動されません。

---

- 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 製品のマニュアルを参照してください。
- 2 停止するノードで、スーパーユーザーになるか、**RBAC** の承認 **solaris.cluster.admin** を提供する役割になりますグローバルクラスタのノードから、次の手順のステップをすべて実行します。



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

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

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、ノードを個別に再起動するには、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、ノードを個別に再起動するには、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。GRUB メニューは次のようになっています。

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

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

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

- 停止し、再起動するゾーンクラスタノードを指定します。

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

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

- 4 ノードが問題なく起動し、オンラインであることを確認します。
  - グローバルクラスタノードがオンラインであることを確認します。  
`phys-schost# cluster status -t node`
  - ゾーンクラスタノードがオンラインであることを確認します。  
`phys-schost# clzonecluster status`

### 例 3-11 SPARC: グローバルクラスタノードの再起動

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

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

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

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

### 例 3-12 x86: グローバルクラスタノードの再起動

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

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

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

### 例 3-13 ゾーンクラスタノードの再起動

次の例は、ゾーンクラスタ上のノードを再起動する方法を示しています。

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

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

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

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

| Name | Node Name | Zone HostName | Status | Zone Status |
|------|-----------|---------------|--------|-------------|
| ---- | -----     | -----         | -----  | -----       |

```
sparse-sczone  schost-1  sczone-1  Online  Running
                schost-2  sczone-2  Online  Running
                schost-3  sczone-3  Online  Running
                schost-4  sczone-4  Online  Running
```

phys-schost#

## ▼ 非クラスタモードでノードを起動する

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

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

この手順では、長形式の 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 `clzonecluster halt` コマンドをグローバルクラスタのノード上で実行して、ゾーンクラスタノードを停止します。 `clnode evacuate` および `shutdown` コマンドを使用して、グローバルクラスタノードを停止します。

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

- 特定のグローバルクラスタを停止します。

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

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

- グローバルクラスタノードから特定のゾーンクラスタノードを停止します。

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

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

- 3 **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起動します。
  - SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -xs
```

- x86 ベースのシステム上で、次のコマンドを実行します。

- a. **GRUB** メニューで矢印キーを使用して該当する **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 メニューは次のようになっています。

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

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

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

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

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

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a)                  |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.  
Press 'b' to boot, 'e' to edit the selected command in the  
boot sequence, 'c' for a command-line, 'o' to open a new line  
after ('O' for before) the selected line, 'd' to remove the  
selected line, or escape to go back to the main menu.

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

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

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

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

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

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

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

---

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

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

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

WARNING: CMM monitoring disabled.
phys-schost-1#
...
rg_name = schost-sa-1 ...
offline node = phys-schost-2 ...
num of node = 0 ...
phys-schost-1#
```

```
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
Print services stopped.
syslogd: going down on signal 15
...
The system is down.
syncing file systems... done
WARNING: node phys-schost-1 is being shut down.
Program terminated

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

## 満杯の /var ファイルシステムを修復する

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

### ▼ 満杯の /var ファイルシステムを修復する

/var ファイルシステムが満杯になったことがノードによって報告され、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、『[Solaris のシステム管理 \(上級編\)](#)』の「[システムメッセージの表示](#)」を参照してください。

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





# データ複製のアプローチ

---

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

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、Veritas Volume Manager、ZFS、または raw ディスクデバイスグループの作成および管理については、第5章の127 ページの「[デバイスグループの管理](#)」を参照してください。

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

この章で説明する内容は次のとおりです。

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

## データ複製についての理解

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

- ホストベースのデータ複製では、ソフトウェアを使用して、地理的に離れたクラスタ間でディスクボリュームをリアルタイムに複製します。リモートミラー複製を使用すると、主クラスタのマスターボリュームのデータを、地理的に離れた二次クラスタのマスターボリュームに複製できます。リモートミラービットマップは、主ディスク上のマスターボリュームと、二次ディスク上のマスターボリュームの差分を追跡します。クラスタ間(およびクラスタとクラスタの外にあるホストとの間)の複製に使用されるホストベースの複製ソフトウェアには、Sun StorageTek Availability Suite 4 などがあります。

ホストベースのデータ複製は、特別なストレージレイではなくホストリソースを使用するため、比較的安価なデータ複製ソリューションです。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 9iRAC、Oracle Parallel Server など)。2つのクラスタ間でホストベースのデータ複製を使用する方法については、『[Oracle Solaris Cluster Geographic Edition Data Replication Guide for Sun StorageTek 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 375 ページの「[Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成](#)」を参照してください。

- ストレージベースのデータ複製は、ストレージコントローラ上でソフトウェアを使用して、データ複製の作業をクラスタノードからストレージデバイスに移動させます。このソフトウェアはノードの処理能力を一部解放し、クラスタの要求にサービスを提供します。クラスタ内またはクラスタ間でデータを複製できるストレージベースのソフトウェアには、Hitachi TrueCopy、Hitachi Universal Replicator、および EMC SRDF などがあります。ストレージベースのデータ複製は、構内クラスタ構成において特に重要になることがあります。この方法の場合、必要なインフラストラクチャーを簡素化できます。構内クラスタ環境でストレージベースのデータ複製を使用する方法の詳細については、92 ページの「[クラスタ内でのストレージベースのデータ複製の使用](#)」を参照してください。

複数のクラスタ間でのストレージベースの複製と、そのプロセスを自動化する Oracle Solaris Cluster Geographic Edition 製品については、『[Oracle Solaris Cluster Geographic Edition Data Replication Guide for Hitachi TrueCopy and Universal Replicator](#)』および『[Oracle Solaris Cluster Geographic Edition Data Replication Guide for EMC Symmetrix Remote Data Facility](#)』を参照してください。また、このタイプのクラスタ構成の完全な構成例については、付録 A の 375 ページの「[Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成](#)」も参照してください。

## サポートされるデータ複製方式

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

- ホストベースの複製
  - Sun StorageTek Availability Suite (Oracle Solaris 10 以降)

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 375 ページの「[Sun StorageTek Availability Suite ソフトウェアを使用したホストベースのデータ複製の構成](#)」を参照してください。

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

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

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

## クラスタ内でのストレージベースのデータ複製の使用

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

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

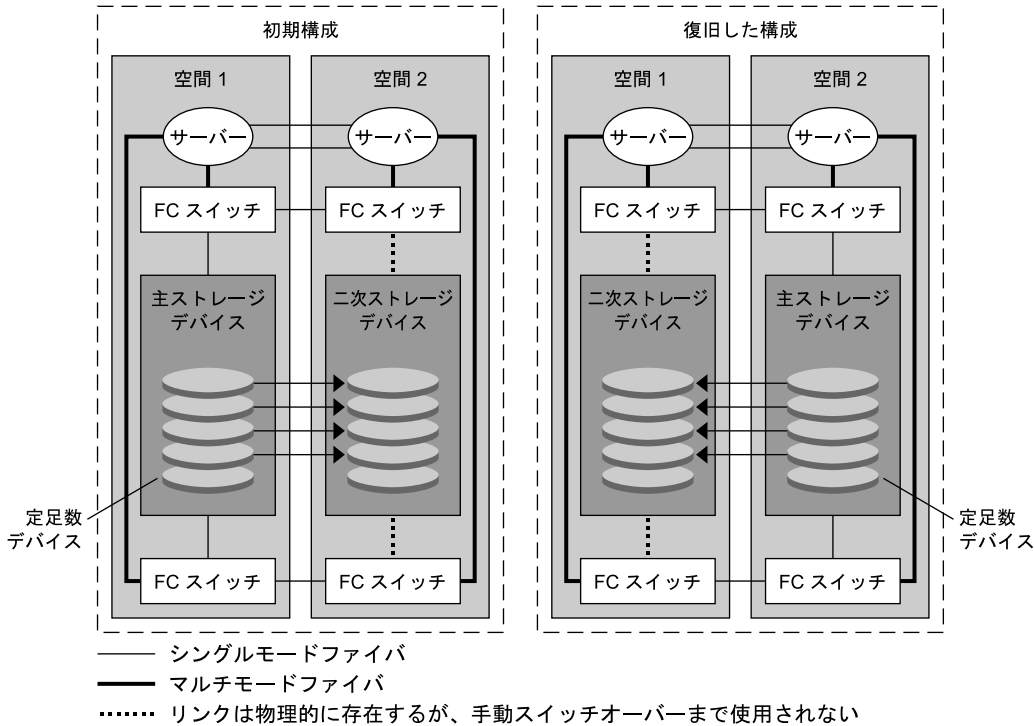
この節では、構内クラスタで使用されるストレージベースのデータ複製を説明します。図 4-1 に、2つのストレージアレイ間でデータが複製される、2か所に設置されたクラスタ構成の例を示します。この例では、第一の場所に主ストレージアレイがあり、これが両方の場所のノードにデータを提供します。また主ストレージアレイは、複製するデータを二次ストレージアレイに提供します。

---

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

---

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



使用するアプリケーションの種類によっては、Hitachi TrueCopy または Hitachi Universal Replicator によるストレージベースのデータ複製を Oracle Solaris Cluster 環境で同期的または非同期に実行できます。構内クラスタで自動フェイルオーバーを実行する場合は、TrueCopy を同期的に使用します。Oracle Solaris Cluster では、EMC SRDF との同期レプリケーションがストレージベースでサポートされています。EMC SRDF に対する非同期レプリケーションはサポート対象外です。

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

リモートストレージデバイスとの通信が失われた場合は、主クラスタ上で動作しているアプリケーションがブロックされるのを回避できるように Fence\_level を never または async に指定しておいてください。Fence\_level が data または status に指定されている場合、リモートストレージデバイスにコピーできなかった更新が、主ストレージデバイスによって拒否されます。

## クラスタ内でストレージベースのデータ複製を使用する際の要件と制限

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

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

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

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

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

## ストレージベースのデータ複製を使用する際のベストプラクティス

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

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

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



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

---

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

- 97 ページの「グローバルデバイスとグローバルな名前空間の管理の概要」
- 101 ページの「ストレージベースの複製されたデバイスの管理」
- 124 ページの「クラスタファイルシステムの管理の概要」
- 127 ページの「デバイスグループの管理」
- 170 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」
- 175 ページの「クラスタファイルシステムの管理」
- 182 ページの「ディスクパス監視の管理」

この章の関連手順の詳細は、表 5-4 を参照してください。

グローバルデバイス、グローバルな名前空間、デバイスグループ、ディスクパスの監視、およびクラスタファイルシステムに関連する概念については、『[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方法は、クラスタにインストールされているボリューム管理ソフトウェアによって決まります。Solaris Volume Manager は「クラスタ対応」なので、Solaris Volume Manager の `metaset(1M)` コマンドを使用してデバイスグループを追加、登録、および削除できます。Veritas Volume Manager (VxVM) を使用している場合、VxVM コマンドを使用してディスクグループを作成し、`clsetup` ユーティリティを使用して、ディスクグループを Oracle Solaris Cluster のデバイスグループとして登録します。VxVM デバイスグループを削除するには、`clsetup` コマンドと VxVM のコマンドの両方を使用します。

---

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

---

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、グループの主ノードであるクラスタから実行する必要があります。

グローバルな名前空間はインストール中に自動的に設定され、Oracle Solaris OS の再起動中に自動的に更新されるため、通常、グローバルデバイス名前空間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 Volume Manager のグローバルデバイスのアクセス権

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

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

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

## グローバルデバイスでの動的再構成

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

- 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、主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、主ノードのアクティブでないデバイスか、二次ノードの任意のデバイス上でだけです。
- DR 操作が終了すると、クラスタのデータアクセスが前と同じように続けられます。
- 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、193 ページの「定足数デバイスへの動的再構成」を参照してください。



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

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

表 5-1 作業マップ: ディスクデバイスとテープデバイスでの動的再構成

| 作業  | 説明  |
|---|---|
| 1. アクティブなデバイスグループに影響するような DR 操作を現在の主ノードに実行する必要がある場合、DR 削除操作をデバイス上で実行する前に、主ノードと二次ノードの切り替えを実行 | 167 ページの「デバイスグループの主ノードを切り替える」   |
| 2. 削除するデバイス上で DR 削除操作を実行  | 「Solaris 10 on Sun Hardware」コレクションの『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』 |

## Veritas Volume Manager による管理に関する注意事項

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

---

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

---

- 各 VxVM ディスクグループには、クラスタ全体で一意のマイナー番号が与えられています。デフォルトでは、ディスクグループを作成したときに、VxVM によって 1000 の倍数の乱数がディスクグループのベースマイナー番号として選択されます。少数のディスクグループしかないほとんどの構成では、このマイナー番号で十分一意性を保証できます。ただし、新たに作成したディスクグループのマイナー番号が、以前別のクラスタノードにインポートしたディスクグループのマイナー番号と衝突することがあります。この場合、Oracle Solaris Cluster デバイスグループは登録できません。この問題を解消するには、新しいディスクグループに一意の値である新しいマイナー番号を付けたうえで、Oracle Solaris Cluster デバイスグループとして登録してください。
- ミラー化したボリュームを設定している場合、ダーティリージョンログ (DRL) を使用すると、ノードに障害が発生してからボリュームが回復するまでの時間を短縮できます。入出力のスループットが低下することになりますが、DRL の使用を強くお勧めします。
- VxVM は、chmod コマンドをサポートしません。VxVM でグローバルデバイスのアクセス権を変更する方法については、VxVM の管理者ガイドを参照してください。
- Oracle Solaris Cluster 3.3 ソフトウェアは、同一ノードからの複数パスの VxVM Dynamic Multipathing (DMP) 管理をサポートしていません。
- VxVM を使用して Oracle RAC 用の共有ディスクグループを設定する場合、『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。Oracle RAC の共有ディスクグループの作成は、ほかのディスクグループの作成とは異なります。Oracle RAC 用の共有ディスクグループをインポートするには、`vxvg -s` を使用する必要があります。Oracle RAC 用の共有ディスクグループをクラスタフレームワークに登録してはいけません。ほかの VxVM ディスクグループを作成する方法については、[139 ページ](#)の「[ディスクの初期化時に新しいディスクグループを作成 \(Veritas Volume Manager\)](#)」を参照してください。

## ストレージベースの複製されたデバイスの管理

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

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

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

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



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

ここでは、次の手順について説明します。

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

## Hitachi TrueCopy で複製されたデバイスの管理

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

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

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

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

| 作業                  | 参照先  |
|---------------------|--|
| Hitachi 複製グループを構成する | 102 ページの「Hitachi TrueCopy 複製グループを構成する」   |
| DID デバイスを構成する       | 104 ページの「Hitachi TrueCopy を使用して DID デバイスを複製用に構成する」   |
| 複製されたグループを登録する      | 135 ページの「デバイスグループを追加および登録する (Solaris Volume Manager)」または 148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」 |
| 構成を確認する             | 106 ページの「Hitachi TrueCopy で複製されたグローバルデバイスグループ構成を確認する」  |

## ▼ Hitachi TrueCopy 複製グループを構成する

始める前に

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



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

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

```
horcm 9970/udp
```

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

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

- 5 まだ複製のペアを作成していない場合は、この時点で作成します。  
**paircreate** コマンドを使用して、希望のフェンスレベルを持つ複製のペアを作成します。複製のペアの作成方法の手順については、TrueCopy のマニュアルを参照してください。

- 6 複製されたデバイスを使用して構成された各ノード上で、**pairdisplay** コマンドを使用することでデータ複製が正しく設定されていることを確認します。**ASYNC** の **fence\_level** を持つ **Hitachi TrueCopy** または **Hitachi Universal Replicator** デバイスグループは、システムの他のデバイスと **ctgid** を共有することはできません。

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

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

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

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

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

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

次の手順に進む前に、初期データコピーが完了するのを待ちます。

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

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

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

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

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

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

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

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

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

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

この手順では、長形式の 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 すべてのノード上で `horcmd` デーモンが実行中であることを確認します。  
実行されていない場合は、次のコマンドでデーモンが起動されます。デーモンがすでに実行されている場合は、システムによりメッセージが表示されます。

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

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

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

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

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

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



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

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

主複製が格納されているリモートノードの名前を指定します。



-S

現在のノード以外のソースノードを指定します。

*secondary replica-nodename*

二次複製が格納されているリモートノードの名前を指定します。

---

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

---

- 5 DID インスタンスが結合されていることを確認します。

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

- 6 TrueCopy 複製が設定されていることを確認します。

```
# cldevice show logical_DID_device
```

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

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




---

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

---

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

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

-d 主複製に対応するリモート DID インスタンス。

*destination-instance*

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

- b. DID の再マッピングが正しく行われたことを確認します。

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

DID インスタンスの 1 つは表示されないはずです。

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

```
# cldevice list -v
```

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

- 135 ページの「デバイスグループを追加および登録する (Solaris Volume Manager)」または148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」  
デバイスグループを登録する際には、必ず TrueCopy 複製グループと同じ名前を指定します。
- 106 ページの「Hitachi TrueCopy で複製されたグローバルデバイスグループ構成を確認する」

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

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

- 135 ページの「デバイスグループを追加および登録する (Solaris Volume Manager)」
- 137 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」
- 138 ページの「複製デバイスグループ (ZFS) の追加と登録方法」
- 139 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」



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

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

この手順では、長形式の 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 主デバイスグループが、主複製が含まれるノードと同じノードに対応することを確認します。

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

- 2 デバイスグループに複製のプロパティが設定されていることを確認します。

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

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

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

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

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

```
# pairedisplay -g group-name
```

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

### 例: Oracle Solaris Cluster 向けの TrueCopy 複製グループの構成

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

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

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

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

例 5-1 ノード 1 上の TrueCopy 構成ファイル

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

例 5-2 ノード 2 上の TrueCopy 構成ファイル

```

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

例 5-3 ノード 3 上の TrueCopy 構成ファイル

```

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

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

例 5-4 ノード 1 上の pairdisplay コマンドの出力

```

# pairdisplay -g VG01
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#.P/S,Status,Fence, Seq#,P-LDEV# M
VG01 pair1(L) (CL1-A , 0, 29)61114 29..S-VOL PAIR NEVER ,----- 58 -
VG01 pair1(R) (CL1-C , 0, 9)20064 58..P-VOL PAIR NEVER ,61114 29 -
VG01 pair2(L) (CL1-A , 0, 30)61114 30..S-VOL PAIR NEVER ,----- 59 -
VG01 pair2(R) (CL1-C , 0, 10)20064 59..P-VOL PAIR NEVER ,61114 30 -
VG01 pair3(L) (CL1-A , 0, 31)61114 31..S-VOL PAIR NEVER ,----- 60 -
VG01 pair3(R) (CL1-C , 0, 11)20064 60..P-VOL PAIR NEVER ,61114 31 -
    
```

例 5-5 ノード 2 上の pairdisplay コマンドの出力

```

# pairdisplay -g VG01
Group PairVol(L/R) (Port#,TID,LU),Seq#,LDEV#.P/S,Status,Fence, Seq#,P-LDEV# M
VG01 pair1(L) (CL1-A , 0, 29)61114 29..S-VOL PAIR NEVER ,----- 58 -
VG01 pair1(R) (CL1-C , 0, 9)20064 58..P-VOL PAIR NEVER ,61114 29 -
VG01 pair2(L) (CL1-A , 0, 30)61114 30..S-VOL PAIR NEVER ,----- 59 -
VG01 pair2(R) (CL1-C , 0, 10)20064 59..P-VOL PAIR NEVER ,61114 30 -
VG01 pair3(L) (CL1-A , 0, 31)61114 31..S-VOL PAIR NEVER ,----- 60 -
VG01 pair3(R) (CL1-C , 0, 11)20064 60..P-VOL PAIR NEVER ,61114 31 -
    
```

例 5-6 ノード 3 上の pairdisplay コマンドの出力

```

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

例 5-6 ノード 3 上の pairdisplay コマンドの出力 (続き)

```
VG01 pair2(L) (CL1-C , 0, 10)20064 59..P-VOL PAIR NEVER ,61114 30 -
VG01 pair2(R) (CL1-A , 0, 30)61114 30..S-VOL PAIR NEVER ,----- 59 -
VG01 pair3(L) (CL1-C , 0, 11)20064 60..P-VOL PAIR NEVER ,61114 31 -
VG01 pair3(R) (CL1-A , 0, 31)61114 31..S-VOL PAIR NEVER ,----- 60 -
```

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

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

```
# pairdisplay -fd -g VG01
Group PairVol(L/R) Device_File ,Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
VG01 pair1(L) c6t500060E800000000000000E0EBA0000001Dd0s2 61114 29..S-VOL PAIR NEVER ,----- 58 -
VG01 pair1(R) c5t50060E8000000000000004E600000003Ad0s2 20064 58..P-VOL PAIR NEVER ,61114 29 -
VG01 pair2(L) c6t500060E800000000000000E0EBA0000001Ed0s2 61114 30..S-VOL PAIR NEVER ,----- 59 -
VG01 pair2(R) c5t50060E8000000000000004E600000003Bd0s2 0064 59..P-VOL PAIR NEVER ,61114 30 -
VG01 pair3(L) c6t500060E800000000000000E0EBA0000001Fd0s2 61114 31..S-VOL PAIR NEVER ,----- 60 -
VG01 pair3(R) c5t50060E8000000000000004E600000003Cd0s2 20064 60..P-VOL PAIR NEVER ,61114 31 -
```

例 5-8 使用されているディスクを示す、ノード 2 上の pairdisplay コマンドの出力

```
# pairdisplay -fd -g VG01
Group PairVol(L/R) Device_File ,Seq#,LDEV#.P/S,Status,Fence,Seq#,P-LDEV# M
VG01 pair1(L) c5t500060E800000000000000E0EBA0000001Dd0s2 61114 29..S-VOL PAIR NEVER ,----- 58 -
VG01 pair1(R) c5t50060E8000000000000004E600000003Ad0s2 20064 58..P-VOL PAIR NEVER ,61114 29 -
VG01 pair2(L) c5t500060E800000000000000E0EBA0000001Ed0s2 61114 30..S-VOL PAIR NEVER ,----- 59 -
VG01 pair2(R) c5t50060E8000000000000004E600000003Bd0s2 20064 59..P-VOL PAIR NEVER ,61114 30 -
VG01 pair3(L) c5t500060E800000000000000E0EBA0000001Fd0s2 61114 31..S-VOL PAIR NEVER ,----- 60 -
VG01 pair3(R) c5t50060E8000000000000004E600000003Cd0s2 20064 60..P-VOL PAIR NEVER ,61114 31 -
```

例 5-9 使用されているディスクを示す、ノード 3 上の pairdisplay コマンドの出力

```
# pairdisplay -fd -g VG01
Group PairVol(L/R) Device_File ,Seq#,LDEV#.P/S,Status,Fence ,Seq#,P-LDEV# M
VG01 pair1(L) c5t50060E8000000000000004E600000003Ad0s2 20064 58..P-VOL PAIR NEVER ,61114 29 -
VG01 pair1(R) c6t500060E800000000000000E0EBA0000001Dd0s2 61114 29..S-VOL PAIR NEVER ,----- 58 -
VG01 pair2(L) c5t50060E8000000000000004E600000003Bd0s2 20064 59..P-VOL PAIR NEVER ,61114 30 -
VG01 pair2(R) c6t500060E800000000000000E0EBA0000001Ed0s2 61114 30..S-VOL PAIR NEVER ,----- 59 -
VG01 pair3(L) c5t50060E8000000000000004E600000003Cd0s2 20064 60..P-VOL PAIR NEVER ,61114 31 -
VG01 pair3(R) c6t500060E800000000000000E0EBA0000001Fd0s2 61114 31..S-VOL PAIR NEVER ,----- 60 -
```

これらの例は、次のディスクが使用されていることを示しています。

- ノード 1 で、次のコマンドを実行します。
  - c6t500060E800000000000000E0EBA0000001Dd0s2
  - c6t500060E800000000000000E0EBA0000001Ed0s2
  - c6t500060E800000000000000E0EBA0000001Fd0s2
- ノード 2:

- c5t500060E80000000000000000EEBA0000001Dd0s2
- c5t500060E80000000000000000EEBA0000001Ed0s2
- c5t500060E80000000000000000EEBA0000001Fd0s2
- ノード 3:
  - c5t50060E80000000000000004E600000003Ad0s2
  - c5t50060E80000000000000004E600000003Bd0s2
  - c5t50060E80000000000000004E600000003Cd0s2

これらのディスクに対応する DID デバイスを確認するには、次の例に示すように、`cldevice list` コマンドを使用します。

例 5-10 使用されているディスクに対応する DID の表示

```
# cldevice list -v
```

```
DID Device Full Device Path
-----
1 node-1:/dev/rdisk/c0t0d0 /dev/did/rdisk/d1
2 node-1:/dev/rdisk/c0t6d0 /dev/did/rdisk/d2
11 node-1:/dev/rdisk/c6t500060E80000000000000000EEBA00000020d0 /dev/did/rdisk/d11
11 node-2:/dev/rdisk/c5t500060E80000000000000000EEBA00000020d0 /dev/did/rdisk/d11
12 node-1:/dev/rdisk/c6t500060E80000000000000000EEBA0000001Fd0 /dev/did/rdisk/d12
12 node-2:/dev/rdisk/c5t500060E80000000000000000EEBA0000001Fd0 /dev/did/rdisk/d12
13 node-1:/dev/rdisk/c6t500060E80000000000000000EEBA0000001Ed0 /dev/did/rdisk/d13
13 node-2:/dev/rdisk/c5t500060E80000000000000000EEBA0000001Ed0 /dev/did/rdisk/d13
14 node-1:/dev/rdisk/c6t500060E80000000000000000EEBA0000001Dd0 /dev/did/rdisk/d14
14 node-2:/dev/rdisk/c5t500060E80000000000000000EEBA0000001Dd0 /dev/did/rdisk/d14
18 node-3:/dev/rdisk/c0t0d0 /dev/did/rdisk/d18
19 node-3:/dev/rdisk/c0t6d0 /dev/did/rdisk/d19
20 node-3:/dev/rdisk/c5t50060E80000000000000004E6000000013d0 /dev/did/rdisk/d20
21 node-3:/dev/rdisk/c5t50060E80000000000000004E600000003Dd0 /dev/did/rdisk/d21
22 node-3:/dev/rdisk/c5t50060E8000000000000004E600000003Cd0 /dev/did/rdisk/d2223
23 node-3:/dev/rdisk/c5t50060E8000000000000004E600000003Bd0 /dev/did/rdisk/d23
24 node-3:/dev/rdisk/c5t50060E8000000000000004E600000003Ad0 /dev/did/rdisk/d24
```

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

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

例 5-11 DID インスタンスの結合

```
# cldevice replicate -D node-3
Remapping instances for devices replicated with node-3...
VG01 pair1 L node-1:/dev/rdisk/c6t500060E80000000000000000EEBA0000001Dd0
```

例 5-11 DID インスタンスの結合 (続き)

```
VG01 pair1 R node-3:/dev/rdisk/c5t50060E8000000000000004E60000003Ad0
Combining instance 14 with 24
VG01 pair2 L node-1:/dev/rdisk/c6t500060E80000000000000EEBA000001Ed0
VG01 pair2 R node-3:/dev/rdisk/c5t50060E800000000000004E60000003Bd0
Combining instance 13 with 23
VG01 pair3 L node-1:/dev/rdisk/c6t500060E80000000000000EEBA000001Fd0
VG01 pair3 R node-3:/dev/rdisk/c5t50060E800000000000004E60000003Cd0
Combining instance 12 with 22
```

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

例 5-12 結合された DID の表示

```
# cldevice list -v
DID Device Full Device Path
-----
1 node-1:/dev/rdisk/c0t0d0 /dev/did/rdisk/d1
2 node-1:/dev/rdisk/c0t6d0 /dev/did/rdisk/d2
11 node-1:/dev/rdisk/c6t500060E80000000000000EEBA0000020d0 /dev/did/rdisk/d11
11 node-2:/dev/rdisk/c5t500060E80000000000000EEBA0000020d0 /dev/did/rdisk/d11
18 node-3:/dev/rdisk/c0t0d0 /dev/did/rdisk/d18
19 node-3:/dev/rdisk/c0t6d0 /dev/did/rdisk/d19
20 node-3:/dev/rdisk/c5t50060E800000000000004E600000013d0 /dev/did/rdisk/d20
21 node-3:/dev/rdisk/c5t50060E800000000000004E60000003Dd0 /dev/did/rdisk/d21
22 node-1:/dev/rdisk/c6t500060E80000000000000EEBA000001Fd0 /dev/did/rdisk/d122
22 node-2:/dev/rdisk/c5t500060E80000000000000EEBA000001Fd0 /dev/did/rdisk/d12
22 node-3:/dev/rdisk/c5t50060E800000000000004E60000003Cd0 /dev/did/rdisk/d22
23 node-1:/dev/rdisk/c6t500060E80000000000000EEBA000001Ed0 /dev/did/rdisk/d13
23 node-2:/dev/rdisk/c5t500060E80000000000000EEBA000001Ed0 /dev/did/rdisk/d13
23 node-3:/dev/rdisk/c5t50060E800000000000004E60000003Bd0 /dev/did/rdisk/d23
24 node-1:/dev/rdisk/c6t500060E80000000000000EEBA000001Dd0 /dev/did/rdisk/d24
24 node-2:/dev/rdisk/c5t500060E80000000000000EEBA000001Dd0 /dev/did/rdisk/d24
24 node-3:/dev/rdisk/c5t50060E800000000000004E60000003Ad0 /dev/did/rdisk/d24
```

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

例 5-13 Solaris ボリュームマネージャデバイスグループの作成

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

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

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

Drive Dbase
d22  Yes
d23  Yes
d24  Yes
```

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

例 5-14 スイッチオーバーとフェイルオーバーの効率化

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

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

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

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

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



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

| 作業                                   | 参照先  |
|--------------------------------------|--|
| ストレージデバイスとノードに SRDF ソフトウェアをインストールする  | EMC ストレージデバイスに付属するマニュアル。   |
| EMC 複製グループを構成する                      | 113 ページの「EMC SRDF 複製グループを構成する」   |
| DID デバイスを構成する                        | 114 ページの「EMC SRDF を使用して DID デバイスを複製用に構成する」   |
| 複製されたグループを登録する                       | 135 ページの「デバイスグループを追加および登録する (Solaris Volume Manager)」または 148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」 |
| 構成を確認する                              | 116 ページの「EMC SRDF で複製されたグローバルデバイスグループ構成を確認する」  |
| キャンバスクラスタの主ルームが完全に失敗したあとに手動でデータを復元する | 122 ページの「主ルームの完全な失敗後に EMC SRDF データを復元する」   |

## ▼ EMC SRDF 複製グループを構成する

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

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



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

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

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

- 3 まだ複製のペアを作成していない場合は、この時点で作成します。  
複製のペアを作成するには、`symrdf` コマンドを使用します。複製のペアの作成方法の手順については、SRDFのマニュアルを参照してください。
- 4 複製されたデバイスによって構成された各ノードで、データの複製が正しく設定されていることを確認します。  
`# /usr/symcli/bin/symsg show group-name`
- 5 デバイスグループのスイッチを実行します。
  - a. 主複製と二次複製が同期していることを確認します。  
`# /usr/symcli/bin/symrdf -g group-name verify -synchronized`
  - b. どのノードに主複製が含まれ、どのノードに二次複製が含まれているかを判別するには、`symsg show` コマンドを使用します。  
`# /usr/symcli/bin/symsg show group-name`  
RDF1 デバイスのノードには主複製が含まれ、RDF2 デバイス状態のノードには二次複製が含まれます。
  - c. 二次複製を有効にします。  
`# /usr/symcli/bin/symrdf -g group-name failover`
  - d. RDF1 デバイスと RDF2 デバイスをスイッチします。  
`# /usr/symcli/bin/symrdf -g group-name swap -refresh R1`
  - e. 複製ペアを有効にします。  
`# /usr/symcli/bin/symrdf -g group-name establish`
  - f. 主ノードと二次複製が同期していることを確認します。  
`# /usr/symcli/bin/symrdf -g group-name verify -synchronized`
- 6 もともと主複製があったノードで上記5つの手順をすべて繰り返します。

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

### ▼ EMC SRDF を使用して DID デバイスを複製用に構成する

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

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

この手順では、長形式の 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 RDF1 デバイスおよび RDF2 デバイスに対応する DID デバイスを判別します。

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

---

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

---

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

```
# cldevice list -v
```

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

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

---

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

---

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

---

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

```
# scdidadm -b device
```

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

---

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

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

- DID** インスタンスが結合されていることを確認します。

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

- SRDF** 複製が設定されていることを確認します。

```
# cldevice show device
```

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

```
# cldevice list -v
```

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

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

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

- 135 ページの「デバイスグループを追加および登録する (Solaris Volume Manager)」
- 137 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」
- 138 ページの「複製デバイスグループ (ZFS) の追加と登録方法」
- 139 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」



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

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

この手順では、長形式の 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 主デバイスグループが、主複製が含まれるノードと同じノードに対応することを確認します。

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

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

デバイスグループがオフラインのときは、オンラインにします。

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

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

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

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

### 例: Oracle Solaris Cluster 向けの SRDF 複製グループの構成

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

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

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

例5-15 複製ペアの作成

すべてのノードで次のコマンドを実行します。

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

Symmetrix ID: 000187990182

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

RDF1 側のすべてのノードで、次のように入力します。

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

RDF2 側のすべてのノードで、次のように入力します。

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

例5-16 データ複製設定の確認

クラスタ内の1つのノードから、次のように入力します。

```
# symdg show dg1
```

Group Name: dg1

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

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

例 5-16 データ複製設定の確認 (続き)

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

Device Group RDF Information
...
# symrdf -g dg1 establish

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

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

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

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

#
# symrdf -g dg1 query

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

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

例 5-16 データ複製設定の確認 (続き)

Legend for MODES:

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

#

例 5-17 使用されているディスクに対応する DID の表示

RDF1 側と RDF2 側で同じ手順を実行します。

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

RDF1 側で次のように入力します。

```
# symdg show dg1
Group Name: dg1
Group Type : RDF1 (RDFA)
...
Standard (STD) Devices (1):
{
-----
LdevName          PdevName          Sym          Cap
Dev Att. Sts      (MB)
-----
DEV001            /dev/rdisk/c5t6006048000018790002353594D303637d0s2 0067      RW      4315
}
Device Group RDF Information
...
```

対応する DID を取得するには、次のように入力します。

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

対応する DID の一覧を表示するには、次のように入力します。

```
# cldevice show d217
=== DID Device Instances ===
DID Device Name: /dev/did/rdisk/d217
Full Device Path: pmoney2:/dev/rdisk/c5t6006048000018790002353594D303637d0
Full Device Path: pmoney1:/dev/rdisk/c5t6006048000018790002353594D303637d0
Replication: none
default_fencing: global
```



例 5-17 使用されているディスクに対応する DID の表示 (続き)

#

RDF2 側で次のように入力します。

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

```
# symdg show dg1
```

```
Group Name: dg1
```

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

```
...
```

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

```
{
```

```
-----
LdevName          PdevName          Sym          Cap
Dev Att. Sts      (MB)
-----
DEV001            /dev/rdisk/c5t6006048000018799018253594D303637d0s2 0067      WD      4315
}
```

```
Device Group RDF Information
```

```
...
```

対応する DID を取得するには、次のように入力します。

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

対応する DID の一覧を表示するには、次のように入力します。

```
# cldevice show d108
```

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

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

#

例 5-18 DID インスタンスの結合

RDF2 側から次のように入力します。

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

例 5-19 結合された DID の表示

クラスタ内の任意のノードから、次のように入力します。

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

=== DID Device Instances ===

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

#
```

▼ 主ルームの完全な失敗後に **EMC SRDF** データを復元する

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

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

始める前に 手動でフェイルオーバーを実行する前に、EMC 複製グループおよび DID デバイスを構成し、EMC 複製グループを登録する必要があります。Solaris ボリュームマネージャー デバイスグループの作成についての詳細は、135 ページの「[デバイスグループを追加および登録する \(Solaris Volume Manager\)](#)」を参照してください。Veritas Volume Manager デバイスグループの作成については、144 ページの「[ディスクをカプセル化する際に新しいディスクグループを作成する \(Veritas Volume Manager\)](#)」を参照してください。

---

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

---

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

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

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

---

ヒント-split 状態にあるデバイスグループは同期されません。

---

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

## クラスタファイルシステムの管理の概要

クラスタファイルシステムを管理するのに特別な 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。また、起動時に自動

的にマウントすることもできます。クラスタファイルシステムは、グローバルクラスタ内の投票ノードからのみ認識できます。クラスタファイルシステムのデータを非投票ノードからアクセス可能にする必要がある場合は、`zoneadm(1M)` または `HASStoragePlus` を使用して非投票ノードにデータをマッピングします。

---

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

---

## クラスタファイルシステムの制限事項

次に、クラスタファイルシステム管理に適用される制限事項を示します。

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

## VxFS サポートについてのガイドライン

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

- クイック入出力
- スナップショット
- 記憶装置チェックポイント
- VxFS 固有のマウントオプション:
  - `convosync` (Convert O\_SYNC)
  - `mincache`

- qlog、delaylog、tmplog
- Veritas クラスタファイルシステム (requires VxVM クラスタ機能および Veritas クラスタサーバーが必要)。VxVM クラスタ機能は、x86 ベースのシステム上ではサポートされていません。

キャッシュアドバイザーは使用可能、効果が認められるのは特定のノードのみ

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

VxFS を使用して高可用性クラスタファイルシステムを作成するための次のガイドラインは、Oracle Solaris 3.3 構成に固有のものです。

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

VxFS クラスタファイルシステムを管理するための次のガイドラインは、Oracle Solaris 3.3 ソフトウェアに固有のものではありません。しかし、これらのガイドラインは UFS クラスタファイルシステムを管理する方法とは異なります。

- VxFS クラスタファイルシステム上にあるファイルは、クラスタ内にある任意のノードから管理できます。例外は `ioctls` で、`ioctls` だけは主ノードから実行する必要があります。管理コマンドが `ioctl` に関連するかどうか分からない場合は、主ノードからコマンドを発行します。
- VxFS クラスタファイルシステムが二次ノードにフェイルオーバーされると、フェイルオーバー時に実行中であったすべての標準システム呼び出し操作は、新しい主ノードで透過的に再実行されます。ただし、フェイルオーバー時に実行していた `ioctl` 関連の操作は失敗します。VxFS クラスタファイルシステムのフェイルオーバーのあとで、このクラスタファイルシステムの状態を調べます。フェイルオーバー以前に古い主ノードから実行された管理コマンドには修正処理が必要になることもあります。詳細については、VxFS のマニュアルを参照してください。

## デバイスグループの管理

クラスタの要件の変化により、クラスタ上のデバイスグループの追加、削除、または変更が必要となる場合があります。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ーフェースがあります。`clsetup` は `cluster` コマンドを生成します。生成されるコマンドについては、各説明の後にある例を参照してください。次の表に、デバイスグループを管理するための作業を示し、またこの節の適切な手順へのリンクを示します。



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

注-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 5-4 作業マップ: デバイスグループの管理

| 作業   | 参照先  |
|--|--|
| <code>cldevice populate</code> コマンドを使用することにより、再構成の再起動を行わずにグローバルデバイス名前空間を更新する               | 129 ページの「グローバルデバイス名前空間を更新する」   |
| 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   | 130 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」   |
| 既存のグローバルデバイス名前空間を移動する  | 132 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」<br>133 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」 |
| <code>metaset</code> コマンドを使用することにより、Solaris Volume Manager ディスクセットを追加し、それらをデバイスグループとして登録する | 135 ページの「デバイスグループを追加および登録する (Solaris Volume Manager)」  |
| <code>cldevicegroup</code> コマンドを使用することにより、 <code>raw</code> ディスクデバイスグループを追加および登録する         | 137 ページの「デバイスグループ ( <code>raw</code> ディスク) を追加および登録する」   |

表 5-4 作業マップ: デバイスグループの管理 (続き)

| 作業   | 参照先   |
|--|---|
| 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                                | 138 ページの「複製デバイスグループ (ZFS) の追加と登録方法」   |
| 推奨の方法を使用することにより、デバイスグループとして新規ディスクグループを追加および登録する                                    | 139 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」   |
| metaset コマンドおよび metaclear コマンドを使用することにより、構成から Solaris Volume Manager デバイスグループを削除する | 140 ページの「デバイスグループを削除して登録を解除する (Solaris Volume Manager)」   |
| cldevicegroup、metaset、および clsetup コマンドを使用することにより、すべてのデバイスグループからノードを削除する            | 141 ページの「すべてのデバイスグループからノードを削除する」  |
| metaset コマンドを使用することにより、Solaris Volume Manager デバイスグループからノードを削除する                   | 142 ページの「デバイスグループからノードを削除する (Solaris Volume Manager)」   |
| VxVM コマンドおよび clsetup を使用することにより、Veritas Volume Manager ディスクグループをデバイスグループとして追加する    | 139 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」<br>144 ページの「ディスクをカプセル化する際に新しいディスクグループを作成する (Veritas Volume Manager)」<br>146 ページの「新しいボリュームを既存のデバイスグループに追加する (Veritas Volume Manager)」<br>147 ページの「既存のディスクグループをデバイスグループに変換する (Veritas Volume Manager)」<br>147 ページの「デバイスグループに新しいマイナー番号を割り当てる (Veritas Volume Manager)」<br>148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」<br>152 ページの「ローカルディスクグループをデバイスグループに変換する (VxVM)」<br>153 ページの「デバイスグループをローカルディスクグループに変換する (VxVM)」<br>151 ページの「ディスクグループの構成変更を登録する (Veritas Volume Manager)」 |



表 5-4 作業マップ: デバイスグループの管理 (続き)

| 作業   | 参照先  |
|--|--|
| clsetup (cldevicegroup コマンドを生成する) コマンドを使用することにより、構成から Veritas Volume Manager デバイスグループを削除する | 154 ページの「デバイスグループからボリュームを削除する (Veritas Volume Manager)」<br>155 ページの「デバイスグループを削除して登録を解除する (Veritas Volume Manager)」 |
| clsetup を使用して cldevicegroup を生成することにより、ノードを Veritas Volume Manager デバイスグループに追加する           | 156 ページの「デバイスグループにノードを追加する (Veritas Volume Manager)」   |
| clsetup を使用して cldevicegroup を生成することにより、Veritas Volume Manager デバイスグループからノードを削除する           | 157 ページの「デバイスグループからノードを削除する (Veritas Volume Manager)」  |
| 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                                    | 159 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」   |
| 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                                 | 161 ページの「デバイスグループのプロパティを変更する」  |
| 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                                      | 166 ページの「デバイスグループ構成の一覧を表示する」   |
| clsetup を使用して cldevicegroup を生成することにより、デバイスグループの二次ノードの希望数を変更する                             | 163 ページの「デバイスグループの二次ノードの希望数を設定する」  |
| cldevicegroup switch コマンドを使用することにより、デバイスグループの主ノードを切り替える                                    | 167 ページの「デバイスグループの主ノードを切り替える」  |
| metaset コマンドまたは vxodg 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                                      | 168 ページの「デバイスグループを保守状態にする」   |

## ▼ グローバルデバイス名前空間を更新する

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

---

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

---

- 1 クラスタの任意のノードで、スーパーユーザーになるか、**RBAC**の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 クラスタの各ノードで `devfsadm(1M)` コマンドを実行します。  
このコマンドは、すべてのノードで同時に実行できます。
- 3 名前空間を再構成します。  
`# cldevice populate`
- 4 各ノードで、ディスクセットを作成する前に、`cldevice populate` コマンドが完了していることを確認してください。  
ノードの1つで `cldevice` コマンドを実行すると、このコマンドはリモートから自分自身をすべてのノードで呼び出します。`cldevice populate` コマンドが処理を終了したかどうかを確認するには、クラスタの各ノードで次のコマンドを実行します。  
`# ps -ef | grep cldevice populate`

#### 例 5-21 グローバルデバイス名前空間を更新する

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

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

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再起動します。  
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、[84 ページの「非クラスタモードでノードを起動する」](#)を参照してください。
- 3 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起動します。  
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

す。Oracle Solaris 10 OS があらかじめインストールされているシステムに Oracle Solaris Cluster ソフトウェアをインストールする場合、この機能が役立ちます。

---

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

---

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

- 132 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」
- 133 ページの「lofi デバイスから専用パーティションにグローバルデバイス名前空間を移行する」

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

- 1 名前空間の場所を変更したいグローバルクラスタ投票ノードのスーパーユーザーになります。
- 2 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再起動します。  
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、84 ページの「非クラスタモードでノードを起動する」を参照してください。
- 3 `/.globaldevices` という名前のファイルがノードに存在しないことを確認します。ファイルが存在する場合は、削除します。
- 4 lofi デバイスを作成します。

```
# mkfile 100m /.globaldevices# lofiadm -a /.globaldevices
# LOFI_DEV='lofiadm /.globaldevices'
# newfs 'echo ${LOFI_DEV} | sed -e 's/lofi/rlofi/g'' < /dev/null# lofiadm -d /.globaldevices
```

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

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

`lofi` デバイスは現在 `/globaldevices` に作成され、グローバルデバイスファイルシステムとしてマウントされています。
- 8 パーティションから `lofi` デバイスへ移行したいグローバルデバイス名前空間のある他のノードでもこのステップを繰り返します。
- 9 1つのノードから、グローバルデバイス名前空間を生成します。
 

```
# /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

- 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再起動します。  
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、[84 ページの「非クラスタモードでノードを起動する」](#)を参照してください。
- 3 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 -nnode ID`
    - 次の形式を使用して、`/etc/vfstab` ファイルに新しいエントリを作成します。  
`blockdevice rawdevice /global/.devices/node@nodeID ufs 2 no global`  
 たとえば、使用するパーティションが `/dev/did/rdisk/d5s3` の場合、`/etc/vfstab` ファイルに追加する新しいエントリは、`/dev/did/dsk/d5s3 /dev/did/rdisk/d5s3 /global/.devices/node@3 ufs 2 no global` となります。
  - 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 クラスタモードにブートします。クラスタの1つのノードから、`cldevice populate` コマンドを実行してグローバルデバイス名前空間を生成します。  
`# /usr/cluster/bin/cldevice populate`  
 クラスタのすべてのノードで処理が完了したことを確認してから、ノードに対する作業を実行してください。  
`# ps -ef | grep 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、Veritas Volume Manager、または raw ディスクのデバイスグループを追加および登録することができます。

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

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

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

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



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

- 1 ディスクセットを作成するディスクに接続されたノードのいずれかで、スーパーユーザーになるか、RBAC の承認 **solaris.cluster.modify** を提供する役割になります。
- 2 **Solaris Volume Manager** 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、デフォルトで二次ノードになります。デバイスグループが作成されたあと、clsetup ユーティリティを使用することで、二次ノードの希望数を変更できます。ディスクのフェイルオーバーの詳細については、163 ページの「デバイスグループの二次ノードの希望数を設定する」を参照してください。

- 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/rdsk/dNの形式の完全なDIDデバイス名を使用してください。

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

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

- 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/rdsk/dN
```

-s setname     デバイスグループ名と同じである、ディスクセット名を指定します。

-a             ディスクセットにドライブを追加します。



---

注 - ディスクセットにドライブを追加するときは、下位デバイス名 (cNtXdY) は使用しないでください。下位レベルデバイス名はローカル名であり、クラスタ全体で一貫ではないため、この名前を使用するとディスクセットがスイッチオーバーできなくなる可能性があります。

---

#### 7 新しいディスクセットとドライブの状態を検査します。

```
# metaset -s setname
```

### 例 5-22 Solaris Volume Manager デバイスグループの追加

次の例は、ディスクドライブ /dev/did/rdisk/d1 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

Oracle Solaris Cluster ソフトウェアでは、他のボリュームマネージャに加え、raw ディスクデバイスグループを使用できます。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
- cldg コマンドの -v オプションを使用せずにデバイスグループのリストを作成したい



---

注意 - 複製したデバイスにデバイスグループを作成する場合、作成するデバイスグループ名 (Solaris Volume Manager、Veritas Volume Manager、または raw ディスク) は複製したデバイスグループの名前と同じにする必要があります。

---

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

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

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

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

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

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

## ▼ 複製デバイスグループ (ZFS) の追加と登録方法

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



注意 - 他社のデータ複製技術による ZFS のフルサポートは保留中です。ZFS サポートに関する最新情報は、最新の Oracle Solaris Cluster リリースノートを参照してください。

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

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

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

- 2 ステップ1で削除したデバイスグループの DID に対応する DID のデバイスグループを作成します。

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

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

- 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
```

- 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。

```
# clrgr create -n pnode1,pnode2 migrate_truecopydg-rg
```

- ステップ4で作成したリソースグループに **hasp-rs** リソースを作成し、**globaldevicepaths** プロパティを **raw** ディスクのデバイスグループに設定します。このデバイスグループは、ステップ2で作成しました。

```
# clr create -t HAStoragePlus -x globaldevicepaths=mypool -g \
migrate_truecopydg-rg hasp2migrate_mypool
```

- アプリケーションリソースグループをローカルゾーンで実行する場合、適切なローカルゾーンを有するノードリストのあるリソースグループを新たに作成してください。ローカルゾーンに対応するグローバルゾーンは、ステップ4で作成したリソースグループのノードリストに含まれている必要があります。このリソースからステップ4で作成したリソースグループで、**rg\_affinities** プロパティの **+++** 値を設定します。

```
# clrgr create -n pnode1:zone-1,pnode2:zone-2 -p \
RG_affinities=+++migrate_truecopydg-rg sybase-rg
```

- ステップ3で作成した **zpool** の **HAStoragePlus** リソース (**hasp-rs**) を、ステップ4または6で作成したリソースグループに作成します。**resource\_dependencies** プロパティをステップ5で作成した **hasp-rs** リソースに設定します。

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

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

## ▼ ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)

---

注- 次の手順は、ディスクを初期化する場合にのみ必要となります。ディスクをカプセル化する場合は、144 ページの「ディスクをカプセル化する際に新しいディスクグループを作成する (Veritas Volume Manager)」の手順を使用してください。

---

VxVM ディスクグループを追加したら、デバイスグループを登録する必要があります。

VxVM を使用して Oracle RAC 用の共有ディスクグループを設定する場合、『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。

- 1 追加しようとしているディスクグループを構成するディスクに物理的に接続されている任意のクラスタノード上でスーパーユーザーになります。
- 2 **VxVM** のディスクグループとボリュームを作成します。  
ディスクグループとボリュームは任意の方法で作成してください。

---

注-ミラー化したボリュームを設定している場合、ダーティーリージョンログ (DRL) を使用すると、ノードに障害が発生してからボリュームが回復するまでの時間を短縮できます。ただし、DRL を使用すると I/O スループットが低下することがあります。

---

この手順を完了する方法については、Veritas Volume Manager のマニュアルを参照してください。

- 3 **VxVM** ディスクグループを **Oracle Solaris Cluster** デバイスグループとして登録します。  
148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」を参照してください。

Oracle RAC 用の共有ディスクグループをクラスタフレームワークに登録してはいけません。

## デバイスグループ名を維持する

デバイスグループに対して様々な管理タスクを実行することができます。

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

デバイスグループは Oracle Solaris Cluster に登録されている Solaris Volume Manager ディスクセットです。Solaris Volume Manager デバイスグループを削除するには、`metaclear` と `metaset` コマンドを使用します。これらのコマンドは、Oracle Solaris Cluster デバイスグループと同じ名前を持つデバイスグループを削除して、ディスクグループの登録を解除します。

ディスクセットを削除する方法については、Solaris Volume Manager のマニュアルを参照してください。

## ▼ すべてのデバイスグループからノードを削除する

すべてのデバイスグループの潜在的な主ノードからクラスタノードを削除する場合は、この手順を使用します。

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

この手順では、長形式の 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主ノードとして削除するノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。
- 2 削除するノードがメンバーになっているデバイスグループ (複数可) を確認します。各デバイスグループの `Device group node list` からこのノード名を検索します。  
`# cldevicegroup list -v`
- 3 手順2で特定したデバイスグループの中に、デバイスグループタイプが `SVM` のものがある場合、そのタイプの各デバイスグループに対して142ページの「[デバイスグループからノードを削除する \(Solaris Volume Manager\)](#)」の手順を実行します。
- 4 手順2で特定したデバイスグループの中に、デバイスグループタイプが `VxVM` のものがある場合、そのタイプの各デバイスグループに対して157ページの「[デバイスグループからノードを削除する \(Veritas Volume Manager\)](#)」の手順を実行します。
- 5 削除するノードがメンバーになっている `raw` デバイスディスクグループを特定します。  
`# cldevicegroup list -v`
- 6 手順5で表示されたデバイスグループの中に、デバイスグループタイプが `Disk` または `Local_Disk` のものがある場合、これらの各デバイスグループに対して、159ページの「[raw ディスクデバイスグループからノードを削除する](#)」の手順を実行します。

- 7 すべてのデバイスグループの潜在的な主ノードのリストからノードが削除されていることを確認します。

ノードがどのデバイスグループの潜在的な主ノードのリストにも存在しなければ、このコマンドは何も返しません。

```
# cldevicegroup list -v nodename
```

## ▼ デバイスグループからノードを削除する (Solaris Volume Manager)

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



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

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

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

- 1 ノードがまだデバイスグループのメンバーであり、かつ、このデバイスグループが **Solaris Volume Manager** デバイスグループであることを確認します。

Solaris Volume Manager のデバイスグループは、デバイスグループタイプが SDS/SVM のものです。

```
phys-schost-1% cldevicegroup show devicegroup
```

- 2 どのノードがデバイスグループの現在の主ノードであるかを特定します。

```
# 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
```

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

- デバイスグループからノードが削除されたことを確認します。  
デバイスグループ名は `metaset` に指定したディスクセット名と一致します。

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

### 例 5-23 デバイスグループからノードを削除する (Solaris Volume Manager)

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

```
[Determine the Solaris Volume Manager
 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
```

## ▼ ディスクをカプセル化する際に新しいディスクグループを作成する (Veritas Volume Manager)

注- 次の手順は、ディスクをカプセル化する場合にのみ必要となります。ディスクを初期化する場合は、139 ページの「ディスクの初期化時に新しいディスクグループを作成 (Veritas Volume Manager)」の手順を使用します。

ルート以外のディスクを Oracle Solaris Cluster デバイスグループに変換するには、そのディスクを VxVM ディスクグループとしてカプセル化してから、そのディスクグループを Oracle Solaris Cluster デバイスグループとして登録します。

ディスクのカプセル化は、VxVM ディスクグループを初めて作成するときのみサポートされています。VxVM ディスクグループを作成して、Oracle Solaris Cluster デバイスグループとして登録したあとは、そのディスクグループには、初期化してもよいディスクだけを登録します。

VxVM を使用して Oracle RAC 用の共有ディスクグループを設定する場合、『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。

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

この手順では、長形式の 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 `/etc/vfstab` ファイルに、カプセル化されたディスクのファイルシステムのエントリがある場合は、`mount at boot` オプションを必ず `no` に設定します。  
ディスクがカプセル化されて Oracle Solaris Cluster デバイスグループとして登録されたあとは、この設定を `yes` に設定し直します。
- 3 ディスクをカプセル化します。  
`vxdiskadm` のメニューまたはグラフィカルユーザーインターフェースを使用して、ディスクをカプセル化します。VxVM では、2つの空きパーティションのほかに、ディスクの始点または終端に未割当てのシリンダが必要です。また、スライス2をディスク全体に設定する必要もあります。詳細は、`vxdiskadm` のマニュアルページを参照してください。
- 4 ノードを停止して再起動します。  
`clnode evacuate` コマンドは、グローバルクラスタ内のすべての非投票ノードを含むすべてのリソースグループとデバイスグループを、指定のノードから次に優先されるノードにスイッチオーバーします。`shutdown` コマンドを使用して、ノードを停止して再起動します。  

```
# clnode evacuate node[...]
# shutdown -g0 -y -i6
```
- 5 必要であれば、すべてのリソースグループとデバイスグループを元のノードにスイッチバックします。  
リソースグループとデバイスグループが、もともと主ノードにフェイルバックするように構成されていた場合、この手順は必要ありません。  

```
# cldevicegroup switch -n node devicegroup
# clresourcegroup switch -z zone -n node resourcegroup
```

`node` ノードの名前。

`zone` リソースグループをマスターできる `node` 上の非投票ノードの名前。リソースグループを作成した際に非投票ノードを指定した場合にのみ、`zone` を指定します。
- 6 VxVM ディスクグループを Oracle Solaris Cluster デバイスグループとして登録します。  
148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」を参照してください。  
Oracle RAC 用の共有ディスクグループをクラスタフレームワークに登録してはいけません。
- 7 手順2 で `mount at boot` オプションを `no` に設定した場合は、`yes` に戻してください。

## ▼ 新しいボリュームを既存のデバイスグループに追加する (Veritas Volume Manager)

新しいボリュームを既存の VxVM デバイスグループに追加する場合、オンラインデバイスグループの一次ノードから、手順を実行します。

---

注- ボリュームを追加したあとで、[151 ページ](#)の「[ディスクグループの構成変更を登録する \(Veritas Volume Manager\)](#)」の手順に従って構成変更の内容を登録する必要があります。

---

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

この手順では、長形式の 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.administer` を提供する役割になります。
- 2 新しいボリュームを追加するデバイスグループの主ノードを確認します。  
`# cldevicegroup status`
- 3 デバイスグループがオフラインである場合、デバイスグループをオンラインにします。  
`# cldevicegroup switch -n nodename devicegroup`  
*nodename*      デバイスグループの切り替え先であるノードの名前を指定します。このノードが新しい主ノードになります。  
*devicegroup*    切り替えるデバイスグループを指定します。
- 4 主ノード (デバイスグループを現在マスターしているノード) から、ディスクグループに VxVM ボリュームを作成します。  
VxVM ボリュームの作成方法は、Veritas Volume Manager のマニュアルを参照してください。
- 5 VxVM ディスクグループに加えた変更を同期化し、グローバルな名前空間を更新します。  
`# cldevicegroup sync`  
[151 ページ](#)の「[ディスクグループの構成変更を登録する \(Veritas Volume Manager\)](#)」。

## ▼ 既存のディスクグループをデバイスグループに変換する (Veritas Volume Manager)

既存の VxVM ディスクグループを Oracle Solaris Cluster デバイスグループに変換するには、ディスクグループを現在のノードにインポートしてから、そのディスクグループを Oracle Solaris Cluster デバイスグループとして登録します。

- 1 クラスタ内の任意のノード上でスーパーユーザーになります。
- 2 VxVM ディスクグループを現在のノードにインポートします。  
`# vxdg import diskgroup`
- 3 VxVM ディスクグループを Oracle Solaris Cluster デバイスグループとして登録します。  
148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」を参照してください。

## ▼ デバイスグループに新しいマイナー番号を割り当てる (Veritas Volume Manager)

マイナー番号がほかのディスクグループと衝突してデバイスグループの登録が失敗する場合、新しいディスクグループに未使用の新しいマイナー番号を割り当てます。新しいマイナー番号を割り当てたあとで、登録手順を再度実行し、ディスクグループを Oracle Solaris Cluster デバイスグループとして登録します。

- 1 クラスタ内の任意のノード上でスーパーユーザーになります。
- 2 使用中のマイナー番号を確認します。  
`# ls -l /global/.devices/node@nodeid/dev/vx/dsk/*`
- 3 新しいディスクグループのベースとなるマイナー番号として、使用されていない別の 1000 の倍数を選択します。
- 4 ディスクグループに新しいマイナー番号を割り当てます。  
`# vxdg reminor diskgroup base-minor-number`
- 5 VxVM ディスクグループを Oracle Solaris Cluster デバイスグループとして登録します。  
148 ページの「ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)」を参照してください。

## 例 5-24 デバイスグループに新しいマイナー番号を割り当てる

次の例は、マイナー番号 16000 - 16002 と 4000 - 4001 が使用されていることを示しています。ここでは、`vx dg remminor` コマンドを使用して、ベースとなるマイナー番号 5000 を新しいデバイスグループに割り当てています。

```
# ls -l /global/.devices/node@nodeid/dev/vx/dsk/*

/global/.devices/node@nodeid/dev/vx/dsk/dg1
brw----- 1 root   root   56,16000 Oct  7 11:32 dg1v1
brw----- 1 root   root   56,16001 Oct  7 11:32 dg1v2
brw----- 1 root   root   56,16002 Oct  7 11:32 dg1v3

/global/.devices/node@nodeid/dev/vx/dsk/dg2
brw----- 1 root   root   56,4000 Oct  7 11:32 dg2v1
brw----- 1 root   root   56,4001 Oct  7 11:32 dg2v2
# vx dg remminor dg3 5000
```

## ▼ ディスクグループをデバイスグループとして登録する (Veritas Volume Manager)

次の手順では、`clsetup` ユーティリティを使用して、関連付けられている VxVM ディスクグループを Oracle Solaris Cluster デバイスグループとして登録します。

---

注- デバイスグループをクラスタに登録したあとは、VxVM コマンドを使用して VxVM ディスクグループをインポートまたはエクスポートしないでください。VxVM ディスクグループやボリュームに変更を加えた場合は、[151 ページの「ディスクグループの構成変更を登録する \(Veritas Volume Manager\)」](#) の手順に従って、デバイスグループの構成変更を登録してください。この手順によって、グローバルな名前空間が正しい状態になります。

---

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

この手順では、長形式の 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 VxVM デバイスグループの登録前に、次の必要条件が満たされていることを確認します。

- クラスタ内の任意のノードでのスーパーユーザー特権。
- デバイスグループとして登録する VxVM ディスクグループの名前。
- デバイスグループをマスターするノードの優先順位。

- デバイスグループの二次ノードの希望数。

優先順位を定義する際には、もっとも優先されるノードに障害が発生し、のちにクラスタに戻った場合に、デバイスグループをそのノードにスイッチバックするかどうかも指定します。

ノードの優先順位とフェイルバックのオプションについての詳細は、[cldevicegroup\(1CL\)](#) を参照してください。

主ノード以外のクラスタノード (スベア) から二次ノードへの移行ノードの優先順位では通常、デバイスグループの二次ノードのデフォルト数は1に設定されます。デフォルトの設定では、主ノードが通常の動作中に複数の二次ノードをチェックすることによって発生する性能の低下を最小限に抑えます。たとえば、4ノードクラスタでは、デフォルトで、1つが主ノード、1つが二次ノード、そして2つがスベアノードに構成されます。[163 ページ](#)の「[デバイスグループの二次ノードの希望数を設定する](#)」も参照してください。

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

- 2 `clsetup` ユーティリティを起動します。

```
# clsetup
```

メインメニューが表示されます。

- 3 VxVM デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプションに対応する数を入力します。  
「デバイスグループメニュー」が表示されます。

- 4 VxVM デバイスグループを登録するために、VxVM ディスクグループをデバイスグループとして登録するためのオプションに対応する数を入力します。

指示に従って、Oracle Solaris Cluster デバイスグループとして登録する VxVM ディスクグループの名前を入力します。

このデバイスグループがストレージベースの複製を使用して複製されている場合、この名前は TrueCopy 複製グループ名と一致する必要があります。

VxVM を使用して Oracle Parallel Server/Oracle RAC 用に共有ディスクグループを設定する場合は、クラスタフレームワークには共有ディスクグループを登録しません。『Veritas Volume Manager Administrator's Reference Guide』に説明されている VxVM のクラスタ機能を使用します。

- 5 デバイスグループを登録しようとしたときに、次のようなエラーが表示された場合は、デバイスグループに新しいマイナー番号を割り当てます。

```
cldevicegroup: Failed to add device group - in use
```

デバイスグループに新しいマイナー番号を割り当てるには、[147 ページ](#)の「[デバイスグループに新しいマイナー番号を割り当てる \(Veritas Volume Manager\)](#)」の手順を使用してください。この手順によって、既存のデバイスグループが使用しているマイナー番号と衝突しない、新しいマイナー番号を割り当てることができます。

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

```
# cldevicegroup sync devicegroup
```

- デバイスグループが登録され、オンラインになったことを確認します。

デバイスグループが適切に登録されている場合、次のコマンドを使用すると、新しいデバイスグループの情報が表示されます。

```
# cldevicegroup status devicegroup
```

---

注-VxVM ディスクグループ、または、クラスタに登録されているボリュームの構成情報を変更した場合、clsetup を使用してデバイスグループを同期化する必要があります。このような構成変更には、ボリュームの追加や削除、既存ボリュームのグループ、所有者、アクセス権の変更などがあります。構成変更後に登録を行うと、グローバルな名前空間が正しい状態になります。[129 ページ](#)の「[グローバルデバイス名前空間を更新する](#)」を参照してください。

---

## 例 5-25 Veritas Volume Manager デバイスグループの登録

次に、clsetup で VxVM デバイスグループ (dg1) を登録する際に生成される cldevicegroup コマンドの例と、その検証手順を示します。この例では、VxVM ディスクグループとボリュームは以前に作成されたものと想定しています。

```
# clsetup
```

```
# cldevicegroup create -t vxvm -n phys-schost-1,phys-schost-2 -p failback=true dg1
```

```
# cldevicegroup status dg1
```

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

```
--- Device Group Status ---
```

| Device Group Name | Primary       | Secondary     | Status |
|-------------------|---------------|---------------|--------|
| dg1               | phys-schost-1 | phys-schost-2 | Online |

参照 VxVM デバイスグループ上にクラスタファイルシステムを作成する場合は、[176 ページ](#)の「[クラスタファイルシステムを追加する](#)」を参照してください。

マイナー番号に問題が発生した場合は、147ページの「デバイスグループに新しいマイナー番号を割り当てる (Veritas Volume Manager)」を参照してください。

## ▼ ディスクグループの構成変更を登録する (Veritas Volume Manager)

VxVM ディスクグループやボリュームの構成情報を変更したときは、Oracle Solaris Cluster デバイスグループに構成変更を登録する必要があります。この登録によって、グローバルな名前空間が正しい状態になります。

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

この手順では、長形式の 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 `clsetup` ユーティリティを起動します。  
`# clsetup`  
メインメニューが表示されます。
- 3 VxVM デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプションに対応する数を入力します。  
「デバイスグループメニュー」が表示されます。
- 4 構成変更を登録するには、VxVM デバイスグループのボリューム情報を同期化するためのオプションに対応する数を入力します。  
指示に従って、構成を変更した VxVM ディスクグループ名を入力します。

### 例 5-26 Veritas Volume Manager ディスクグループの構成の変更の登録

次に、`clsetup` で VxVM デバイスグループ (`dg1`) を登録する際に生成される `cldevicegroup` コマンドの例を示します。この例では、VxVM ディスクグループとボリュームは以前に作成されたものと想定しています。

```
# clsetup
cldevicegroup sync dg1
```

## ▼ ローカルディスクグループをデバイスグループに変換する (VxVM)

ローカル VxVM ディスクグループをグローバルにアクセス可能な VxVM デバイスグループに変更するには、次の手順を実行します。

- 1 各クラスタのノードのスーパーユーザーになります。
- 2 **clsetup** ユーティリティを起動します。  
# **clsetup**
- 3 **localonly** プロパティの設定を解除します。
  - a. 「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。
  - b. 「ローカル VxVM グループの VxVM デバイスグループへの再設定 (Reset a Local VxVM Disk Group to a VxVM Device Group)」メニュー項目を選択します。
  - c. 指示に従い、**localonly** プロパティの設定を解除します。
- 4 ディスクグループをマスターできるノードを指定します。
  - a. **clsetup** ユーティリティのメインメニューに戻ります。
  - b. 「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。
  - c. 「ディスクグループの登録 (Register a Diskgroup)」メニュー項目を選択します。
  - d. 指示に従い、ディスクグループをマスターできるノードを指定します。
  - e. 完了後 **clsetup** ユーティリティを終了します。
- 5 デバイスグループが構成されていることを確認します。  
phys-schost# **cldevicegroup show**



## ▼ デバイスグループをローカルディスクグループに変換する (VxVM)

VxVM デバイスグループを、Oracle Solaris Cluster ソフトウェアにより管理されていないローカル VxVM ディスクグループに変更するには、次の手順を実行します。ローカルディスクグループはそのノードリストに複数のノードを持つことができますが、一度に1つのノードによってのみマスターできます。

- 1 各クラスタのノードのスーパーユーザーになります。
- 2 デバイスグループをオフラインにします。  
`phys-schost# cldevicegroup offline devicegroup`
- 3 デバイスグループを登録解除します。
  - a. `clsetup`ユーティリティを起動します。  
`phys-schost# clsetup`
  - b. 「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。
  - c. 「VxVM ディスクグループの登録解除 (Unregister a VxVM Disk Group)」メニュー項目を選択します。
  - d. 指示に従い、Oracle Solaris Cluster ソフトウェアから登録を解除する VxVM ディスクグループを指定します。
  - e. `clsetup`ユーティリティを終了します。
- 4 ディスクグループが Oracle Solaris Cluster ソフトウェアに登録されていないことを確認します。  
`phys-schost# cldevicegroup status`  
コマンド出力には、登録を解除されたデバイスグループは表示されなくなるはずで  
す。
- 5 ディスクグループをインポートします。  
`phys-schost# vxdg import diskgroup`
- 6 ディスクグループの `localonly` プロパティを設定します。
  - a. `clsetup`ユーティリティを起動します。  
`phys-schost# clsetup`

- b. 「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。
  - c. 「VxVM ディスクグループをローカルディスクグループとして設定 (Set a VxVM Disk Group as a Local Disk Group)」メニュー項目を選択します。
  - d. 指示に従い、`localonly` プロパティを設定し、ディスクグループを排他的にマスタする1つのノードを指定します。
  - e. 完了後 `clsetup` ユーティリティーを終了します。
- 7 ディスクグループが正しくローカルディスクグループとして構成されていることを確認します。

```
phys-schost# vxdg list diskgroup
```

## ▼ デバイスグループからボリュームを削除する (Veritas Volume Manager)

注- デバイスグループからボリュームを削除したあとは、[151 ページ](#)の「ディスクグループの構成変更を登録する (Veritas Volume Manager)」の手順に従って、デバイスグループに構成の変更を登録する必要があります。

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

この手順では、長形式の 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 デバイスグループの主ノードと状態を確認します。  
`# cldevicegroup status devicegroup`
- 3 デバイスグループがオフラインのときは、オンラインにします。  
`# cldevicegroup online devicegroup`

- 4 主ノード (デバイスグループを現在マスターしているノード) から、ディスクグループの VxVM ボリュームを削除します。
 

```
# vxedit -g diskgroup -rf rm volume
```

`-g diskgroup` ボリュームが含まれる VxVM ディスクグループを指定します。

`-rf rm volume` 指定したボリュームを削除します。-r オプションは、処理を再帰的に繰り返す指定です。-f オプションは、有効に設定されているボリュームを削除する場合に必要です。
- 5 `clsetup` ユーティリティを使用して、デバイスグループの構成変更を登録し、グローバルな名前空間を更新します。  
[151 ページの「ディスクグループの構成変更を登録する \(Veritas Volume Manager\)」](#) を参照してください。

## ▼ デバイスグループを削除して登録を解除する (Veritas Volume Manager)

Oracle Solaris Cluster デバイスグループを削除すると、対応する VxVM ディスクグループはエクスポートされますが、削除されるわけではありません。ただし、VxVM ディスクグループが引き続き存在していても、再登録しないかぎりクラスタで使用することはできません。

次の手順では、`clsetup` ユーティリティを使用して、VxVM ディスクグループを削除し、Oracle Solaris Cluster デバイスグループから登録を解除します。

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

この手順では、長形式の 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 デバイスグループをオフラインにします。
 

```
# cldevicegroup offline devicegroup
```
- 3 `clsetup` ユーティリティを起動します。
 

```
# clsetup
```

メインメニューが表示されます。

- 4 VxVM デバイスグループを使用して作業するには、デバイスグループおよびポリシーのオプションに対応する数を入力します。  
「デバイスグループメニュー」が表示されます。
- 5 VxVM ディスクグループを登録解除するには、VxVM デバイスグループを登録解除するためのオプションに対応する数を入力します。  
指示に従って、登録を解除する VxVM ディスクグループの名前を入力します。

#### 例 5-27 Veritas Volume Manager デバイスグループの削除および登録の解除

次に、VxVM デバイスグループ dg1 をオフラインにして、デバイスグループの削除および登録解除の際に `clsetup` により生成される `cldevicegroup` コマンドの例を示します。

```
# cldevicegroup offline dg1
# clsetup
cldevicegroup delete dg1
```

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

この手順では、`clsetup` ユーティリティーを使用してデバイスグループにノードを追加します。

VxVM デバイスグループにノードを追加するには以下が必要です。

- クラスタ内のノードでのスーパーユーザー特権
- ノードの追加先の VxVM デバイスグループの名前
- 追加するノードの名前またはノード ID

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

この手順では、長形式の 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` ユーティリティーを起動します。

```
# clsetup
```

メインメニューが表示されます。

- 3 VxVM デバイスグループを使用して作業するには、デバイスグループおよびボリュームのオプションに対応する数を入力します。  
「デバイスグループメニュー」が表示されます。
- 4 VxVM デバイスグループにノードを追加するには、VxVM デバイスグループへノードを追加するためのオプションに対応する数を入力します。  
指示に従って、デバイスグループ名とノード名を入力します。
- 5 ノードが追加されたことを確認します。  
次のコマンドを実行し、表示される新しいディスクのデバイスグループ情報を確認します。  

```
# cldevicegroup show devicegroup
```

#### 例 5-28 Veritas Volume Manager デバイスグループへのノードの追加

次に、`clsetup` でノード (`phys-schost-3`) を VxVM デバイスグループ (`dg1`) に追加する際に生成される `cldevicegroup` コマンドと、その検証手順の例を示します。

```
# clsetup
cldevicegroup add-node -n phys-schost-3 dg1
# cldevicegroup show dg1

=== Device Groups ===

Device Group Name:          dg1
Type:                      VxVM
failback:                  yes
Node List:                  phys-schost-1, phys-schost-3
preferenced:               no
numsecondaries:            1
diskgroup names:          dg1
```

## ▼ デバイスグループからノードを削除する (Veritas Volume Manager)

Veritas Volume Manager (VxVM) デバイスグループ (ディスクグループ) の潜在的な主ノードリストからクラスタノードを削除する場合は、この手順を使用します。

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

この手順では、長形式の 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 ノードがまだグループのメンバーであり、かつ、そのグループが **VxVM** デバイスグループであることを確認します。  
 デバイスグループタイプ **VxVM** は **VxVM** デバイスグループを示します。  

```
phys-schost-1% cldevicegroup show devicegroup
```
- 2 現在のクラスタメンバーノードで、スーパーユーザーになるか、**RBAC** の承認 **solaris.cluster.read** および **solaris.cluster.modify** を提供する役割になります。
- 3 **clsetup** ユーティリティを起動します。  

```
# clsetup
```

 メインメニューが表示されます。
- 4 デバイスグループを再構成する場合は、デバイスグループおよびボリュームのオプションに対応する数を入力します。
- 5 **VxVM** デバイスグループからノードを削除する場合は、**VxVM** デバイスグループからノードを削除するためのオプションに対応する数を入力します。  
 プロンプトに従って、デバイスグループからクラスタノードを削除します。次の情報を入力するよう求められます。
  - **VxVM** のデバイスグループ
  - ノード名
- 6 ノードが **VxVM** デバイスグループ (複数可) から削除されていることを確認します。  

```
# cldevicegroup show devicegroup
```

### 例 5-29 デバイスグループからノードを削除する (VxVM)

この例では、**dg1** という **VxVM** のデバイスグループから **phys-schost-1** というノードを削除します。

```
[Determine the VxVM device group for the node:]
# cldevicegroup show dg1

=== Device Groups ===

Device Group Name:          dg1
Type:                      VxVM
failback:                  no
Node List:                  phys-schost-1, phys-schost-2
preferenced:               no
numsecondaries:            1
diskgroup names:          dg1
[Become superuser and start the clsetup utility:]
# clsetup
Select Device groups and volumes>Remove a node from a VxVM device group.
```

Answer the questions when prompted.

You will need the following information.

| Name:                  | Example:      |
|------------------------|---------------|
| VxVM device group name | dg1           |
| node names             | phys-schost-1 |

[Verify that the `cldevicegroup` command executed properly:]

```
cldevicegroup remove-node -n phys-schost-1 dg1
```

Command completed successfully.

Dismiss the `clsetup` Device Groups Menu and Main Menu.

[Verify that the node was removed:]

```
# cldevicegroup show dg1
```

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

```
Device Group Name:          dg1
Type:                      VxVM
failback:                  no
Node List:                  phys-schost-2
preferenced:                no
numsecondaries:             1
device names:               dg1
```

## ▼ raw ディスクデバイスグループからノードを削除する

raw ディスクデバイスグループの潜在的な主ノードリストからクラスタノードを削除する場合は、この手順を使用します。

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

この手順では、長形式の 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 5-30 raw デバイスグループからノードを削除する

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

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

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

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

```
Device Group Name:      dsk/d2
Type:                   VxVM
failback:               true
Node List:              pbrave2
preferenced:            false
localonly:              false
autogen                 true
numsecondaries:         1
diskgroup name:        vxdg1
```

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



```

numsecondaries:                1
diskset name:                  ms1
(dsk/d4) Device group node list: phys-schost-2
(dsk/d2) Device group node list: phys-schost-1, phys-schost-2
(dsk/d1) Device group node list: phys-schost-1, phys-schost-2
[Disable the localonly flag for each local disk on the node:]
phys-schost-1# cldevicegroup set -p localonly=false dsk/d4
[Verify that the localonly flag is disabled:]
phys-schost-1# cldevicegroup show -n phys-schost-2 -t rawdisk +
(dsk/d4) Device group type:      Disk
(dsk/d8) Device group type:      Local_Disk
[Remove the node from all raw-disk device groups:]

phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d4
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d2
phys-schost-1# cldevicegroup remove-node -n phys-schost-2 dsk/d1

```

## ▼ デバイスグループのプロパティを変更する

デバイスグループの主所有権を確立する方法は、`preferenced` という所有権設定属性の設定に基づきます。この属性を設定していない場合は、ほかで所有されていないデバイスグループの主所有者が、そのグループ内のディスクへのアクセスを試みる最初のノードになります。一方、この属性を設定してある場合は、ノードが所有権の確立を試みる優先順位を指定する必要があります。

`preferenced` 属性を無効にすると、`failback` 属性も自動的に無効に設定されます。ただし、`preferenced` 属性を有効または再有効にする場合は、`failback` 属性を有効にするか無効にするかを選択できます。

`preferenced` 属性を有効または再有効にした場合は、主所有権の設定一覧でノードの順序を確立し直す必要があります。

次の手順では、`clsetup` を使用し、Solaris Volume Manager または VxVM デバイスグループの、`preferenced` 属性と `failback` 属性を設定または設定解除します。

始める前に この手順を実行するには、属性値を変更するデバイスグループの名前が必要です。

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

この手順では、長形式の 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に対応する数を入力します。

「デバイスグループメニュー」が表示されます。

- 4 デバイスグループの重要なプロパティを変更するには、**VxVM**または**Solaris Volume Manager** デバイスグループの重要なプロパティを変更するためのオプションに対応する数を入力します。

「デバイスグループのプロパティ変更メニュー」が表示されます。

- 5 デバイスグループのプロパティを変更するには、**preference**、**failback**などのプロパティを変更するオプションに対応する数を入力します。

指示に従って、デバイスグループの **preferenced** および **failback** オプションを設定します。

- 6 デバイスグループの属性が変更されたことを確認します。

次のコマンドを実行し、表示されるデバイスグループ情報を確認します。

```
# cldevicegroup show -v devicegroup
```

### 例 5-31 デバイスグループのプロパティの変更

次に、**clsetup** でデバイスグループ (**dg-schost-1**) の属性値を設定したときに生成される **cldevicegroup** コマンドの例を示します。

```
# cldevicegroup set -p preferenced=true -p failback=true -p numsecondaries=1 \
-p nodelist=phys-schost-1,phys-schost-2 dg-schost-1
# cldevicegroup show dg-schost-1
```

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

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

## ▼ デバイスグループの二次ノードの希望数を設定する

`numsecondaries` プロパティは、主ノードに障害が発生した場合にグループをマスターできる、デバイスグループ内のノード数を指定します。デバイスサービスの二次ノードのデフォルト数は1です。この値には、1からデバイスグループ内で動作している主ノード以外のプロバイダノード数までの任意の整数を設定できます。

この設定は、クラスタの性能と可用性のバランスをとるための重要な要因になります。たとえば、二次ノードの希望数を増やすと、クラスタ内で同時に複数の障害が発生した場合でも、デバイスグループが生き残る可能性が増えます。しかし、二次ノード数を増やすと、通常の動作中の性能が一様に下がります。通常、二次ノード数を減らすと、性能が上がりますが、可用性が下がります。しかし、二次ノード数を増やしても、必ずしも、当該のファイルシステムまたはデバイスグループの可用性が上がるわけではありません。詳細は、『[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、二次ノードの実数と希望数の間に整合性がない場合、二次ノードはデバイスグループに追加されるか、またはデバイスグループから削除されます。

この手順では、`clsetup` ユーティリティを使用して、すべてのタイプのデバイスグループの `numsecondaries` プロパティを設定します。デバイスグループを構成する際のデバイスグループのオプションの詳細については、`cldevicegroup(1CL)` を参照してください。

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

この手順では、長形式の 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 二次ノードの希望数を変更するには、**numsecondaries** プロパティを変更するためのオプションに対応する数を入力します。  
指示に従って、デバイスグループに構成したい二次ノードの希望数を入力します。対応する `cldevicegroup` コマンドが実行され、ログが出力され、ユーティリティーは前のメニューに戻ります。
- 6 デバイスグループの構成を検証します。

```
# cldevicegroup show dg-schost-1
=== Device Groups ===
```

```
Device Group Name:          dg-schost-1
Type:                      VxVm      This might also be SDS or Local_Disk.
failback:                  yes
Node List:                  phys-schost-1, phys-schost-2 phys-schost-3
preferenced:                yes
numsecondaries:             1
diskgroup names:           dg-schost-1
```

---

注-VxVM ディスクグループ、または、クラスタに登録されているボリュームの構成情報を変更した場合、`clsetup` を使用してデバイスグループを再登録する必要があります。このような構成変更には、ボリュームの追加や削除、既存ボリュームのグループ、所有者、アクセス権の変更などがあります。構成変更後に登録を行うと、グローバルな名前空間が正しい状態になります。[129 ページの「グローバルデバイス名前空間を更新する」](#) を参照してください。

---

- 7 デバイスグループの属性が変更されたことを確認します。  
次のコマンドを実行して、表示されるデバイスグループ情報を確認します。

```
# cldevicegroup show -v devicegroup
```

**例 5-32 二次ノードの希望数の変更 (Solaris ポリリュームマネージャー)**

次に、デバイスグループ (dg-schost-1) の二次ノードの希望数を構成するときには、`clsetup` によって生成される `cldevicegroup` コマンドの例を示します。この例では、ディスクグループとポリリュームは以前に作成されているものと想定しています。

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

=== Device Groups ===

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

**例 5-33 二次ノードの希望数の設定 (Veritas Volume Manager)**

次に、デバイスグループ (dg-schost-1) の二次ノードの希望数を 2 に設定するときには、`clsetup` によって生成される `cldevicegroup` コマンドの例を示します。デバイスグループを作成したあとで二次ノードの希望数を変更する方法については、[163 ページの「デバイスグループの二次ノードの希望数を設定する」](#)を参照してください。

```
# cldevicegroup set -p numsecondaries=2 dg-schost-1
# cldevicegroup show dg-schost-1
=== Device Groups ===

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

**例 5-34 二次ノードの希望数のデフォルト値への設定**

次に、ヌル文字列値を使用して、二次ノードのデフォルト数を構成する例を示します。デバイスグループは、デフォルト値が変更されても、デフォルト値を使用するように構成されます。

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

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

```
Device Group Name:          dg-schost-1
Type:                      SVM
failback:                  yes
Node List:                  phys-schost-1, phys-schost-2 phys-schost-3
preferenced:                yes
numsecondaries:            1
diskset names:             dg-schost-1
```

## ▼ デバイスグループ構成の一覧を表示する

構成の一覧を表示するために、スーパーユーザーになる必要はありません。ただし、`solaris.cluster.read` の権限は必要です。

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

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

- 次に示されている方法のどれかを選択してください。

Oracle Solaris Cluster Manager GUI      詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

`cldevicegroup show`                      `cldevicegroup show` を使用して、クラスタ内のすべてのデバイスグループの構成を一覧表示します。

`cldevicegroup show devicegroup`          `cldevicegroup show devicegroup` を使用して、1つのデバイスグループの構成を一覧表示します。

`cldevicegroup status devicegroup`        `cldevicegroup status devicegroup` を使用して、1つのデバイスグループのステータスを判別します。

`cldevicegroup status +`                    `cldevicegroup status +` を使用して、クラスタ内のすべてのデバイスグループのステータスを判別します。

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

### 例 5-35 すべてのデバイスグループのステータスの一覧表示

```
# cldevicegroup status +
=== Cluster Device Groups ===
```

```

--- Device Group Status ---

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

```

### 例 5-36 特定のデバイスグループの構成の一覧表示

```

# cldevicegroup show dg-schost-1

=== Device Groups ===

Device Group Name:          dg-schost-1
Type:                      SVM
failback:                  yes
Node List:                  phys-schost-2, phys-schost-3
preferenced:               yes
numsecondaries:            1
diskset names:             dg-schost-1

```

## ▼ デバイスグループの主ノードを切り替える

次の手順は、アクティブでないデバイスグループを起動する(オンラインにする)ときにも使用できます。

Oracle Solaris Cluster Manager GUI を使用すると、アクティブでないデバイスグループをオンラインにしたり、デバイスグループの主ノードを切り替えることができます。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

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

この手順では、長形式の 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主ノードを切り替えます。  

```
# cldevicegroup switch -n nodename devicegroup
```

`-n nodename` 切り替え先のノードの名前を指定します。このノードが新しい主ノードになります。

`devicegroup` 切り替えるデバイスグループを指定します。

- 3 デバイスグループが新しい主ノードに切り替わったことを確認します。  
デバイスグループが適切に登録されている場合、次のコマンドを使用すると、新しいデバイスグループの情報が表示されます。

```
# cldevice status devicegroup
```

### 例 5-37 デバイスグループの主ノードの切り替え

次に、デバイスグループの主ノードを切り替えて変更結果を確認する例を示します。

```
# cldevicegroup switch -n phys-schost-1 dg-schost-1
```

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

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

```
--- Device Group Status ---
```

| Device Group Name | Primary       | Secondary     | Status |
|-------------------|---------------|---------------|--------|
| dg-schost-1       | phys-schost-1 | phys-schost-2 | Online |

## ▼ デバイスグループを保守状態にする

デバイスグループを保守状態にすることによって、デバイスのいずれかにアクセスされたときに、デバイスグループが自動的にオンラインになることを防ぎます。デバイスグループを保守状態にする必要があるのは、修理手順において、修理が終わるまで、すべての入出力活動を停止する必要がある場合などです。また、デバイスグループを保守状態にすることによって、別のノード上のディスクセットまたはディスクグループを修復していても、当該ノード上のデバイスグループはオンラインにならないため、データの損失を防ぎます。

破損したディスクセットを復元する方法については、[302 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 Volume Manager の場合:

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



注意 - Solaris Volume Manager ディスクセットの所有権を取得する場合、デバイスグループが保守状態にあるときは、`metaset -C take` コマンドを使用する必要があります。`metaset -t` を使用すると、所有権の取得作業の一部として、デバイスグループがオンラインになります。VxVM ディスクグループをインポートする場合、ディスクグループをインポートするときは、`-t` フラグを使用する必要があります。`-t` フラグを使用することで、当該ノードが再起動した場合に、ディスクグループが自動的にインポートされることを防ぎます。

Veritas Volume Manager の場合:

```
# vxvg -t import disk-group-name
```

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



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

- Solaris Volume Manager の場合:

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

- Veritas Volume Manager の場合:

```
# vxvg deport diskgroupname
```

- 5 デバイスグループをオンラインにします。

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

### 例 5-38 デバイスグループを保守状態にする

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

```
[Place the device group in maintenance state.]
# cldevicegroup disable dg-schost-1
# cldevicegroup offline dg-schost-1
[If needed, manually import the disk set or disk group.]
For Solaris Volume Manager:
# metaset -C take -f -s dg-schost-1
For Veritas Volume Manager:
# vxdg -t import dg1

[Complete all necessary repair procedures.]

[Release ownership.]
For Solaris Volume Manager:
# metaset -C release -s dg-schost-1
For Veritas Volume Manager:
# vxdg deport dg1

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

## ストレージデバイス用の SCSI プロトコル設定の管理

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 リザーベーションが割り当てられます。次の手順に従って、複数のデバイスの設定を確認し、必要に応じてデバイスの設定を上書きします。

- 171 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」
- 171 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」
- 172 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」
- 174 ページの「単一ストレージデバイスのフェンシングプロトコルを変更する」

## ▼ すべてのストレージデバイスのデフォルトのグローバルな SCSI プロトコル設定を表示する

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

この手順では、長形式の 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 5-39 すべてのストレージデバイスのデフォルトのグローバルな SCSI プロトコル設定の表示

次の例に、クラスタ上のすべてのストレージデバイスの SCSI プロトコル設定を示します。

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

```
=== Cluster ===
```

```
Cluster Name:                               racerxx
installmode:                                disabled
heartbeat_timeout:                          10000
heartbeat_quantum:                           1000
private_netaddr:                             172.16.0.0
private_netmask:                             255.255.248.0
max_nodes:                                   64
max_privatenets:                             10
global_fencing:                              pathcount
Node List:                                   phys-racerxx-1, phys-racerxx-2
```

## ▼ 単一ストレージデバイスの SCSI プロトコルを表示する

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

この手順では、長形式の 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 5-40 単一デバイスの SCSI プロトコルの表示

次の例に、デバイス `/dev/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

フェンシングは、クラスタに接続されているすべてのストレージデバイスに対して、グローバルにオンまたはオフに設定できます。あるストレージデバイスのデフォルトのフェンシングが `pathcount`、`prefer3`、または `nofencing` に設定されている場合、そのデバイスのデフォルトのフェンシング設定は、グローバル設定よりも優先されます。ストレージデバイスのデフォルトのフェンシング設定が `global` に設定されている場合、ストレージデバイスはグローバル設定を使用します。たとえば、ストレージデバイスのデフォルト設定が `pathcount` である場合、ここでの手順を使用してグローバルな SCSI プロトコル設定を `prefer3` に変更しても、設定は変更されません。単一デバイスのデフォルト設定を変更するには、[174 ページの「単一ストレージデバイスのフェンシングプロトコルを変更する」](#) の手順を使用します。



注意-フェンシングを誤ってオフに設定すると、アプリケーションのフェイルオーバー時にデータの破損する可能性が生じやすくなります。フェンシングをオフに設定する場合は、そのような状況でもデータが破損しないかどうか十分に検査してください。共有ストレージデバイスが SCSI プロトコルをサポートしていない場合や、クラスタのストレージへのアクセスをクラスタ外のホストに対して許可する場合は、フェンシングをオフに設定できません。

定足数デバイスのデフォルトのフェンシング設定を変更するには、デバイスの構成を解除し、フェンシング設定を変更して、定足数デバイスを再構成します。フェンシングをオフに設定したあとで、定足数デバイスを含むデバイスについては定期的にオンに戻す場合は、定足数サーバーサービスを利用して定足数を構成することを検討してください(そうすることで、定足数の動作を中断せずに済みます)。

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

この手順では、長形式の 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。

```
cluster set -p global_fencing={pathcount | prefer3 | nofencing | nofencing-noscrub}
```

|                   |   |
|-------------------|---|
| -p global_fencing | すべての共有デバイスの現在のグローバルなデフォルトフェンシングアルゴリズムを設定します。  |
| prefer3           | パスが2より多いデバイスに対して SCSI-3 プロトコルを使用します。  |
| pathcount         | 共有デバイスに接続されている DID パスの数でフェンシングプロトコルを決定します。pathcount 設定は、定足数デバイスで使用されます。   |
| nofencing         | フェンシングをオフに設定します(すべてのストレージデバイスについてフェンシングステータスを設定します)。  |
| nofencing-noscrub | ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスが消去され、クラスタの外側にあるシステムからストレージへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大 |

な問題があるストレージデバイスに対してのみ使用してください。

#### 例 5-41 すべてのストレージデバイスのデフォルトのグローバルなフェンシングプロトコル設定の設定

次の例では、クラスタ上のすべてのストレージデバイスのフェンシングプロトコルを、SCSI-3 プロトコルに設定します。

```
# cluster set -p global_fencing=prefer3
```

## ▼ 単一ストレージデバイスのフェンシングプロトコルを変更する

フェンシングプロトコルは、1つのストレージデバイスに対して設定することもできます。

---

注- 定足数デバイスのデフォルトのフェンシング設定を変更するには、デバイスの構成を解除し、フェンシング設定を変更して、定足数デバイスを再構成します。フェンシングをオフに設定したあとで、定足数デバイスを含むデバイスについては定期的にオンに戻す場合は、定足数サーバーサービスを利用して定足数を構成することを検討してください(そうすることで、定足数の動作を中断せずに済みます)。

---

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

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



---

注意- フェンシングを誤ってオフに設定すると、アプリケーションのフェイルオーバー時にデータの破損する可能性が生じやすくなります。フェンシングをオフに設定する場合は、そのような状況でもデータが破損しないかどうか十分に検査してください。共有ストレージデバイスが SCSI プロトコルをサポートしていない場合や、クラスタのストレージへのアクセスをクラスタ外のホストに対して許可する場合は、フェンシングをオフに設定できません。

---

- 1 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 プロパティーを変更します。  |
| pathcount          | 共有デバイスに接続されている DID パスの数でフェンシングプロトコルを決定します。   |
| scsi3              | SCSI-3 プロトコルを使用します。  |
| global             | グローバルなデフォルトのフェンシング設定を使用します。global 設定は、定足数デバイス以外のデバイスで使用されます。   |
| nofencing-noscrub  | 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。<br>ディスク消し込みにより、持続的なすべての SCSI 予約情報からデバイスが消去され、クラスタの外側にあるシステムからストレージデバイスへのアクセスが可能になります。nofencing-noscrub オプションは、SCSI 予約に重大な問題があるストレージデバイスに対してのみ使用してください。 |
| device             | デバイスパスの名前またはデバイス名を指定します。   |

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

### 例 5-42 単一デバイスのフェンシングプロトコルの設定

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

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

次の例では、d11 デバイスのデフォルトフェンシングをオフに設定します。

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

## クラスタファイルシステムの管理

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

表 5-5 作業リスト:クラスタファイルシステムの管理

| 作業  | 参照先                            |
|---|--------------------------------|
| クラスタファイルシステムを Oracle Solaris Cluster の最初のインストール後に追加する | 176 ページの「クラスタファイルシステムを追加する」    |
| クラスタファイルシステムを削除する                                     | 180 ページの「クラスタファイルシステムを削除する」    |
| クラスタ内のグローバルマウントポイントをチェックして、ノード間の一貫性が保たれているかどうかを確認する   | 182 ページの「クラスタ内のグローバルマウントを確認する」 |

## ▼ クラスタファイルシステムを追加する

次の作業は、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がデバイスグループ (Solaris Volume Manager デバイスグループまたは VxVM デバイスグループ)、またはブロックディスクドライブであること。

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。  
クラスタに非大域ゾーンが設定されているときは、大域ゾーンでこの手順を実行します。

---

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

---

- 2 ファイルシステムを作成する。



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。

- UFS ファイルシステムの場合は、**newfs(1M)** コマンドを使用します。

```
phys-schost# newfs raw-disk-device
```

下の表に、引数 *raw-disk-device* の名前の例を挙げます。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  |
| Veritas Volume Manager | /dev/vx/rdisk/oradg/vol01 | oradg ディスクセット内の raw デバイス vol01 |
| なし                     | /dev/global/rdisk/d1s3    | raw ディスクデバイス d1s3              |

- Veritas File System (VxFS) ファイルシステムの場合、**VxFS** ドキュメントに記載されている手順を実行してください。
- 3 クラスタ内の各ノードで、クラスタファイルシステムのマウントポイントのディレクトリを作成します。  
そのノードからはクラスタファイルシステムにアクセスしない場合でも、マウントポイントはノードごとに必要です。

---

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

---

```
phys-schost# mkdir -p /global/device-group/mountpoint/
```

|                     |                                       |
|---------------------|---------------------------------------|
| <i>device-group</i> | 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。 |
| <i>mountpoint</i>   | クラスタファイルシステムのマウント先のディレクトリ名を指定します。     |

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

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

---

注-クラスタに非大域ゾーンが設定されている場合は、大域ゾーンのクラスタファイルシステムを必ず大域ゾーンのルートディレクトリのパスにマウントしてください。

---

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

たとえば、`phys-schost-1` がディスクデバイス `d0` を `/global/oracle/` にマウントし、`phys-schost-2` がディスクデバイス `d1` を `/global/oracle/logs/` にマウントすると仮定します。この構成では、`phys-schost-1` が起動され、`/global/oracle/` がマウントされたあとにのみ、`phys-schost-2` を起動し、`/global/oracle/logs/` をマウントできます。

- 5 クラスタの任意のノード上で、設定確認ユーティリティを実行します。

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

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

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

## 6 クラスタファイルシステムをマウントします。

```
phys-schost# mount /global/device-group/mountpoint/
```

- UFSの場合は、クラスタ内の任意のノードから、クラスタファイルシステムをマウントします。
- VxFSの場合、*device-group*の現在のマスタからクラスタファイルシステムをマウントして、ファイルシステムを確実にマウントします。  
さらに、VxFSファイルシステムを正しく確実にマウント解除するには、*device-group*の現在のマスターからファイルシステムをマウント解除します。

---

注 - Oracle Solaris Cluster 環境で VxFS クラスタファイルシステムを管理するには、管理コマンドは VxFS クラスタファイルシステムがマウントされているノードからのみ実行します。

---

## 7 クラスタ内にある各ノード上で、クラスタファイルシステムがマウントされていることを確認します。

df コマンドまたは mount コマンドのいずれかを使用し、マウントされたファイルシステムの一覧を表示します。詳細は、[df\(1M\)](#) マニュアルページまたは [mount\(1M\)](#) マニュアルページを参照してください。

クラスタファイルシステムは大域ゾーンおよび非大域ゾーンの両方からアクセスできます。

### 例 5-43 UFS クラスタファイルシステムの作成

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

```
phys-schost# newfs /dev/md/oracle/rdisk/d1
...
phys-schost# mkdir -p /global/oracle/d1
phys-schost# vi /etc/vfstab
#device          device          mount    FS      fsck    mount    mount
#to mount        to fsck         point   type    pass   at boot options
#
/dev/md/oracle/dsk/d1 /dev/md/oracle/rdisk/d1 /global/oracle/d1 ufs 2 yes global,logging
...
phys-schost# cluster check -k vfstab
phys-schost# mount /global/oracle/d1
phys-schost# mount
...
```

/global/oracle/d1 on /dev/md/oracle/dsk/d1 read/write/setuid/global/logging/largefiles  
on Sun Oct 3 08:56:16 2005

## ▼ クラスタファイルシステムを削除する

クラスタファイルシステムを削除するには、単に、そのクラスタファイルシステムのマウントを解除します。データも削除する場合は、配下のディスクデバイス(またはメタデバイスかボリューム)をシステムから削除します。

---

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

---

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

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

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 マウントされているクラスタファイルシステムを確認します。

```
# mount -v
```

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

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

```
-c          ファイルシステムのマウントポイントとなっているファイル  
           と、マウントされているファイルシステム内のファイルがすべて  
           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マニュアルを参照してください。

#### 例 5-44 クラスタファイルシステムの削除

次に、Solaris Volume Manager メタデバイスまたはボリューム **/dev/md/oracle/rdsk/d1** にマウントされた UFS クラスタファイルシステムを削除する例を示します。

```
# mount -v
...
/global/oracle/d1 on /dev/md/oracle/dsk/d1 read/write/setuid/global/logging/largefiles
# fuser -c /global/oracle/d1
/global/oracle/d1: 4006c
# fuser -c -k /global/oracle/d1
/global/oracle/d1: 4006c
# fuser -c /global/oracle/d1
/global/oracle/d1:
# umount /global/oracle/d1

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

[Save and exit.]

クラスタファイルシステム上のデータを削除するには、配下のデバイスを削除します。詳細については、ボリューム管理ソフトウェアのマニュアルを参照してください。

## ▼ クラスタ内のグローバルマウントを確認する

`cluster(1CL)` コーティリティは `/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、二次ディスクパス障害の通知を受け取ることができます。この節では、ディスクパスの監視に必要な管理作業を行うための手順を説明します。ディスクパス監視デーモンの概念については、『[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)` マニュアルページを参照してください。 `scdpmd` デーモンの調整に関する詳細は、`scdpmd.conf(4)` マニュアルページを参照してください。デーモンがレポートするログエラーに関しては、`syslogd(1M)` マニュアルページも参照してください。

---

注-`cldevice` コマンドを使ってノードに入出力デバイスを追加すると、監視を行っていた監視リストにディスク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 5-6 作業マップ: ディスクパス監視の管理

| 作業   | 参照先  |
|--|--|
| ディスクパスを監視します。  | 183 ページの「ディスクパスを監視する」  |
| ディスクパスの監視を解除します。   | 185 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。  | 186 ページの「障害のあるディスクパスを表示する」   |
| ファイルからディスクパスを監視します。  | 187 ページの「ファイルからディスクパスを監視する」  |
| 監視しているすべての共有ディスクパスが失敗したときのノードの自動再起動を有効化または無効化します。  | 189 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再起動を有効にする」<br>190 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再起動を無効にする」 |
| 不正なディスクパス状態を解決します。起動時に監視対象の DID デバイスを利用できず、DID インスタンスが DID ドライバにアップロードされない場合、不正なディスクパス状態が報告されることがあります。 | 186 ページの「ディスクパスの状態エラーを解決する」  |

cldevice コマンドを実行する以下のセクションの手順にはディスクパス引数が含まれます。ディスクパス引数はノード名とディスク名からなります。ただし、ノード名は必須ではありません。指定しないと、all が使用されます。

## ▼ ディスクパスを監視する

この作業は、クラスタのディスクパスを監視するときに行います。



注意 - DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

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 5-45 単一ノードのディスクパスを監視

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

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

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

#### 例 5-46 すべてのノードのディスクパスを監視

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

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

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

#### 例 5-47 CCR からディスク構成を読み直す

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

```
# cldevice monitor +
# cldevice status

Device Instance           Node           Status
-----
/dev/did/rdisk/d1         schost-1       Ok
/dev/did/rdisk/d2         schost-1       Ok
/dev/did/rdisk/d3         schost-1       Ok
                          schost-2       Ok
/dev/did/rdisk/d4         schost-1       Ok
```



|                   |          |    |
|-------------------|----------|----|
|                   | schost-2 | Ok |
| /dev/did/rdisk/d5 | schost-1 | Ok |
|                   | schost-2 | Ok |
| /dev/did/rdisk/d6 | schost-1 | Ok |
|                   | schost-2 | Ok |
| /dev/did/rdisk/d7 | schost-2 | Ok |
| /dev/did/rdisk/d8 | schost-2 | Ok |

## ▼ ディスクパスの監視を解除する方法

ディスクパスの監視を解除する場合は、この手順を使用します。



注意 - DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

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。
- 3 各ノードで、適切なディスクパスの監視を解除します。

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

### 例 5-48 ディスクパスの監視解除

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

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

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

## ▼ 障害のあるディスクパスを表示する

クラスタに障害のあるディスクパスを表示する場合は、次の手順を使用します。



注意 - DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

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

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

### 例 5-49 障害のあるディスクパスを表示する

次の例では、全クラスタ内の障害のあるディスクパスを表示します。

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

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

## ▼ ディスクパスの状態エラーを解決する

次のイベントが発生すると、DPM が障害の発生したパスがオンラインになっても、そのパスの状態を更新しない可能性があります。

- 監視対象パスの障害によって、ノードが再起動する。
- 再起動したノードがオンラインに戻るまで、監視対象の DID パスの下のデバイスがオンラインに戻らない。

起動時に監視対象の DID デバイスを利用できず、このため DID インスタンスが DID ドライバにアップロードされないため、不正なディスクパス状態が報告されます。このような状態が発生する場合は、手動で DID 情報を更新します。

- 1 1つのノードからグローバルデバイス名前空間を更新します。

```
# cldevice populate
```

- 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状態が **OK** になっていることを確認します。

```
# cldevice status disk-device
```

| Device Instance | Node          | Status |
|-----------------|---------------|--------|
| -----           | ----          | -----  |
| dev/did/dsk/dN  | phys-schost-1 | Ok     |

## ▼ ファイルからディスクパスを監視する

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

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



注意 - DPM は、Sun Cluster 3.1 10/03 ソフトウェアより前にリリースされたバージョンが動作するノードではサポートされません。ローリングアップグレードが行われているときには DPM コマンドを使用しないでください。すべてのノードをアップグレードしたら、DPM コマンドを使用する前にこれらのノードをオンラインにする必要があります。

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

この手順では、長形式の 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 5-50 ファイルからディスクパスを監視する

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

最初に、現在のクラスタ構成をエクスポートします。

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

`deviceconfig` XML ファイルは、`phys-schost-2` と `d3` の間の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再起動を有効にする

この機能を有効にすると、次の条件が満たされる場合、ノードは自動的に再起動します。

- ノード上ですべての監視対象の共有ディスクパスが失敗した。
- 少なくとも1つのディスクがクラスタ内の異なるノードからアクセス可能である。

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

ノードが自動再起動したあと、ノード上のすべての監視対象共有ディスクパスがアクセス不能のままである場合、そのノードは再び自動再起動しません。しかし、ノードが再起動したが失敗したあとに、利用可能になったディスクパスがある場合、そのノードは再び自動再起動します。

reboot\_on\_path\_failure プロパティを有効にすると、ローカルディスクパスの状態は、ノードの再起動が必要かどうか決定するときには考慮されません。監視された共有ディスクのみが影響を受けます。

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

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

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

## ▼ すべての監視共有ディスクパスが失敗した場合にノードの自動再起動を無効にする

この機能を無効にすると、あるノード上のすべての監視共有ディスクパスに障害が発生しても、ノードは自動的に再起動しません。

- 1 クラスタ内の任意のノードで、スーパーユーザーになるか、**solaris.cluster.modify RBAC**の承認を提供する役割になります。
- 2 クラスタ内のすべてのノードに対して、ノードへのすべての監視共有ディスクパスに障害が発生した場合の、ノードの自動再起動を無効にします。

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

## 定足数の管理

---

この章では、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。

- 191 ページの「定足数デバイスの管理」
- 217 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(1CL)` コマンドを使用すると、定足数デバイスの管理手順をすべて実行できます。また、`clsetup(1CL)` 対話型ユーティリティーや Oracle Solaris Cluster Manager GUI を使用しても、いくつかの管理手順を実行できます。このセクションの管理手順は、可能なかぎり `clsetup` ユーティリティーを使用して説明してあります。GUI を使用して定足数手順を実行する方法については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。定足数デバイスを使用して作業する際は、次のガイドラインに注意してください。

- 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、状態と投票数の情報を提供します。

このセクションで示す例は、主に 3 ノードクラスタです。

表 6-1 作業リスト:定足数の管理

| 作業   | 説明                              |
|--|---------------------------------|
| <code>clsetup(1CL)</code> を使用することで、定足数デバイスをクラスタに追加する                                 | 194 ページの「定足数デバイスの追加」            |
| <code>clsetup</code> を使用する ( <code>clquorum</code> を生成することにより、クラスタから定足数デバイスを削除する)    | 206 ページの「定足数デバイスを削除する」          |
| <code>clsetup</code> を使用する ( <code>clquorum</code> を生成することにより、クラスタから最後の定足数デバイスを削除する) | 207 ページの「クラスタから最後の定足数デバイスを削除する」 |
| 追加と削除の手順を使用することで、クラスタ内の定足数デバイスを交換する  | 209 ページの「定足数デバイスを交換する」          |
| 追加と削除の手順を使用することで、定足数デバイスのリストを変更する  | 210 ページの「定足数デバイスのノードリストを変更する」   |



表 6-1 作業リスト:定足数の管理 (続き)

| 作業   | 説明                         |
|--|----------------------------|
| clsetup を使用する (clquorum を生成する) ことにより、定足数デバイスを保守状態にする<br><br>(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。) | 212 ページの「定足数デバイスを保守状態にする」  |
| clsetup を使用する (clquorum を生成する) ことにより、定足数構成をデフォルト状態にリセットする  | 213 ページの「定足数デバイスを保守状態から戻す」 |
| clquorum(1CL) コマンドを使用することで、定足数デバイスと投票数を一覧表示する  | 215 ページの「定足数構成を一覧表示する」     |

## 定足数デバイスへの動的再構成

クラスタ内の定足数デバイス上で動的再構成 (DR) を実行するときには、いくつかの問題を考慮する必要があります。

- 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 作業マップ:定足数デバイスへの動的再構成

| 作業                                   | 説明                     |
|--------------------------------------|------------------------|
| 1. 削除する定足数デバイスと交換する、新しい定足数デバイスを有効に設定 | 194 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                 | 206 ページの「定足数デバイスを削除する」 |

表 6-2 作業マップ: 定足数デバイスへの動的再構成 (続き)

| 作業                          | 説明  |
|-----------------------------|---|
| 3. 削除する定足数デバイス上で DR 削除操作を実行 | 「Solaris 10 on Sun Hardware」コレクションの『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』 |

## 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を確認する方法、推奨される定足数構成、障害回避などについては、『[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。

- 次からの共有 LUN
  - 共有 SCSI ディスク
  - SATA (Serial Attached Technology Attachment) ストレージ
  - Sun NAS
  - Oracle 社製 Sun ZFS Storage Appliance
- Oracle Solaris Cluster Quorum Server
- ネットワークアプライアンス NAS

これらのデバイスを追加する方法については、次の節で説明しています。

- 195 ページの「共有ディスク定足数デバイスを追加する」
- 199 ページの「Network Appliance ネットワーク接続ストレージ (NAS) 定足数デバイスを追加する」
- 202 ページの「定足数サーバー定足数をデバイスとして追加する」

---

注-複製されたディスクを定足数デバイスとして構成することはできません。複製されたディスクを定足数デバイスとして追加しようとすると、次のエラーメッセージが表示され、コマンドはエラーコードとともに終了します。

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

---

共有ディスク定足数デバイスは、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設定すると、デュアルポートのディスクを定足数デバイスとして構成して、SCSI-2 または SCSI-3 (デフォルトは SCSI-2) を使用できます。フェンシングがオンに設定され、共有デバイスが3つ以上のノードに接続されている場合は、SCSI-3 プロトコル (2 ノードを超える場合のデフォルトのプロトコル) を使用する定足数デバイスとして共有ディスクを構成できません。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設定した場合は、ソフトウェア定足数プロトコルを使用する定足数デバイスとしてディスクを構成できます。これは、そのディスクが SCSI-2 と SCSI-3 のどちらのプロトコルをサポートしている場合でも有効です。ソフトウェアの定足数は、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、ユーザーデータを持つディスクまたはデバイスグループのメンバーであるディスクを使用できます。共有ディスクがある定足数サブシステムで使用されているプロトコルは、`cluster show` コマンドの出力の、共有ディスクの `access-mode` 値で確認します。

これらの作業は、Oracle Solaris Cluster Manager GUI を使用しても実行できます。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

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

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設定し、ソフトウェア定足数プロトコルを使用します。

この手順を実行するには、ノードが共有するデバイス ID (Device Identifier、DID) によりディスクドライブを確認します。cldevice show コマンドを使用して、DID 名の一覧を参照します。詳細は、cldevice(1CL) のマニュアルページを参照してください。クラスタのすべてのノードがオンラインか確認してから、新しい定足数デバイスを追加します。

次の手順を実行して、SCSI または SATA デバイスを構成します。

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

この手順では、長形式の 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 clsetup ユーティリティーを起動します。  
# clsetup  
clsetup のメインメニューが表示されます。
- 3 定足数のオプションに対応する番号を入力します。  
「定足数メニュー」が表示されます。
- 4 定足数デバイスを追加するためのオプションに対応する番号を入力し、追加する定足数デバイスを確認するメッセージ clsetup ユーティリティーのプロンプトが表示されたら「yes」と入力します。  
追加する定足数デバイスの種類を確認するメッセージが表示されます。
- 5 共有ディスク定足数デバイスのオプションに対応する番号を入力します。  
どのグローバルデバイスを使用するかを確認するメッセージが表示されます。
- 6 使用しているグローバルデバイスを入力します。  
指定したグローバルデバイスに新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。
- 7 「yes」と入力し、新しい定足数デバイスの追加を続行します。  
新しい定足数デバイスが正常に追加されると、clsetup ユーティリティーではその旨のメッセージが表示されます。

- 8 定足数デバイスが追加されていることを確認します。

```
# clquorum list -v
```

### 例 6-1 共有ディスク定足数デバイスの追加

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

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

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

```
# clsetup
```

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

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

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

```
[Information:                               Example:]
[Directly attached shared disk             shared_disk]
[Global device                             d20]
```

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

```
clquorum add d20
```

```
Command completed successfully.
```

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

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

```
# clquorum list -v
```

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

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

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 Sun NAS GUI を使用して、Sun NAS ファイラ上で iSCSI デバイスを設定します。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マニュアル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。Sun NAS デバイスがある場合は、次のコマンドを実行します。
  - a. 約 50 M バイトのサイズのファイルボリュームを作成します。
  - b. 各ノードで、iSCSI アクセスリストを作成します。
    - i. クラスタの名前を iSCSI アクセスリスト名として使用します。
    - ii. 各クラスタノードのイニシエータノード名をアクセスリストに追加します。CHAP および IQN は不要です。
  - c. iSCSI LUN を構成します。

バッキングファイルボリュームの名前を LUN の名前として使用できます。各ノードのアクセスリストを LUN に追加します。
- 2 各クラスタノードで、iSCSI LUN を検出して、iSCSI アクセスリストを静的構成に設定します。

```
# iscsiadm modify discovery -s enable

# iscsiadm list discovery
Discovery:
  Static: enabled
  Send Targets: disabled
  iSNS: disabled

# iscsiadm add static-config iqn.LUNName,IPAddress_of_NASDevice
# devfsadm -i iscsi
# cldevice refresh
```
- 3 1つのクラスタノードから DID を iSCSI LUN 用に構成します。

```
# /usr/cluster/bin/cldevice populate
```
- 4 iSCSI を使用するクラスタに設定した 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。

## 例 6-2 Sun NAS または Sun ZFS Storage Appliance の NAS 定足数デバイスの追加

次の例は、Sun NAS 定足数デバイスを追加する際に `clsetup` によって生成される `clquorum` コマンドと、検証ステップを示しています。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マニュアル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。

```
Add an iSCSI device on the Sun NAS filer.
Use the Sun NAS GUI to create a file volume that is approximately 50mb in size.
File Volume Operations -> Create File Volume
For each node, create an iSCSI access list.
iSCSI Configuration -> Configure Access List
Add the initiator node name of each cluster node to the access list.
*** Need GUI or command syntax for this step. ***
Configure the iSCSI LUN
iSCSI Configuration -> Configure iSCSI LUN
On each of the cluster nodes, discover the iSCSI LUN and set the iSCSI access list to static configuration.
iscsiadm modify discovery -s enable
iscsiadm list discovery
Discovery:
  Static: enabled
  Send Targets: enabled
  iSNS: disabled
iscsiadm add static-config
iqn.1986-03.com.sun0-1:000e0c66efe8.4604DE16.thinquorum,10.11.160.20
devsadm -i iscsi
From one cluster node, configure the DID devices for the iSCSI LUN.
/usr/cluster/bin/scldevice populate
/usr/cluster/bin/scldevice populate
Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.

[Add the NAS device as a quorum device
using the DID device:]
clquorum add d20

Command completed successfully.
```

## ▼ Network Appliance ネットワーク接続ストレージ (NAS) 定足数デバイスを追加する

Network Appliance (NetApp) ネットワーク接続ストレージ (Network-Attached Storage、NAS) デバイスを定足数デバイスとして使用する場合は、次の要件があります。

- NetApp の iSCSI ライセンスをインストールする必要があります。
- クラスタ化されたファイラを定足数デバイスとして使用する場合は、そのファイラに iSCSI LUN を構成する必要があります。

- 時間の同期をとるために NTP を使用するには、NetApp NAS ユニットの構成する必要があります。
- クラスタ化されたファイラに選択されている NTP サーバーのうち少なくとも 1 つは、Oracle Solaris Cluster ノードの NTP サーバーでなければなりません。
- クラスタを起動する場合は、常にクラスタノードを起動する前に NAS デバイスを起動する必要があります。  
誤った順序でデバイスを起動すると、ノードは定足数デバイスを検出できません。このような状況でノードが停止した場合、クラスタはサービスに対応できなくなる可能性があります。サービスの中断が発生した場合は、クラスタ全体を起動し直すか、NetApp NAS 定足数デバイスを削除して追加し直す必要があります。
- クラスタは、各 NAS デバイスを単一の定足数デバイスにしか使用できません。  
定足数デバイスがさらに必要な場合は、ほかの共有ストレージを構成できません。同じ NAS デバイスを使用するほかのクラスタは、そのデバイスの別の LUN をそれらの定足数デバイスとして使用できます。

Oracle Solaris Cluster 環境で NetApp NAS ストレージデバイスをインストールする方法については、Oracle Solaris Cluster のマニュアル『[Oracle Solaris Cluster 3.3 With Network-Attached Storage Device Manual](#)』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 すべての **Oracle Solaris Cluster** ノードがオンライン状態であり、クラスタ化された **NetApp** ファイラと通信が行えることを確認します。
- 2 クラスタの任意のノードで、スーパーユーザーになるか、**RBAC** の承認 **solaris.cluster.modify** を提供する役割になります。
- 3 **clsetup** ユーティリティを起動します。  
# clsetup  
clsetup のメインメニューが表示されます。
- 4 定足数のオプションに対応する番号を入力します。  
「定足数メニュー」が表示されます。
- 5 定足数デバイスを追加するためのオプションに対応する番号を入力します。定足数デバイスを追加することを確認するには、「**yes**」と入力します。  
追加する定足数デバイスの種類を確認するメッセージが表示されます。



- 6 **netapp\_nas** 定足数デバイスのオプションに対応する番号を入力します。**netapp\_nas** 定足数デバイスを追加することを確認するには、「**yes**」と入力します。  
新しい定足数デバイスの名前を入力するように、**clsetup** ユーティリティーのプロンプトが表示されます。
- 7 追加する定足数デバイスの名前を入力します。  
定足数デバイスの名前は任意に選択できます。この名前は、今後の管理コマンドの処理だけに使用されるものです。  
新しい定足数デバイスのファイラの名前を入力するように、**clsetup** ユーティリティーのプロンプトが表示されます。
- 8 新しい定足数デバイスのファイラの名前を入力します。  
この名前には、ネットワークアクセスが可能なファイラ名またはファイラのアドレスを指定してください。  
ファイラの LUNID を指定するように、**clsetup** ユーティリティーのプロンプトが表示されます。
- 9 ファイラの定足数デバイス LUN の ID を入力します。  
ファイラに新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。
- 10 「**yes**」と入力し、新しい定足数デバイスの追加を続行します。  
新しい定足数デバイスが正常に追加されると、**clsetup** ユーティリティ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 6-3 NetApp NAS 定足数デバイスの追加

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

```

Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on any cluster node.
[Start the clsetup utility:]
# clsetup
[Select Quorum>Add a quorum device]
[Answer the questions when prompted.]
[You will need the following information.]
  [Information:           Example:]
  [Quorum Device         Netapp_nas quorum device]
  [Name:                  qd1]
  [Filer:                 nas1.sun.com]
  [LUN ID:                0]
[Verify that the clquorum command was completed successfully:]

```

```
clquorum add -t netapp_nas -p filer=nas1.sun.com,-p lun_id=0 qd1
Command completed successfully.
[Quit the clsetup Quorum Menu and Main Menu.]
[Verify that the quorum device is added:]
# clquorum list -v
Quorum      Type
-----
qd1          netapp_nas
scphyshost-1 node
scphyshost-2 node
```

## ▼ 定足数サーバー定足数をデバイスとして追加する

始める前に 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、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「定足数サーバーソフトウェアをインストールおよび構成する」を参照してください。

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

この手順では、長形式の 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に対応する番号を入力します。

「定足数メニュー」が表示されます。

- 5 定足数デバイスを追加するためのオプションに対応する番号を入力します。定足数デバイスを追加することを確認するには、「**yes**」と入力します。

追加する定足数デバイスの種類を確認するメッセージが表示されます。

- 6 **quorum\_server** 定足数デバイスのオプションに対応する番号を入力します。**quorum\_server** 定足数デバイスを追加することを確認するには、「**yes**」と入力します。

新しい定足数デバイスの名前を入力するように、**clsetup** ユーティリティーのプロンプトが表示されます。

- 7 追加する定足数デバイスの名前を入力します。

定足数デバイスの名前は任意に選択できます。この名前は、今後の管理コマンドの処理だけに使用されるものです。

新しい定足数デバイスのファイラの名前を入力するように、**clsetup** ユーティリティーのプロンプトが表示されます。

- 8 定足数サーバーのホストの名前を入力します。

この名前で、定足数サーバーが動作するマシンの IP アドレス、またはネットワーク上のマシンのホスト名を指定します。

ホストの IPv4 または IPv6 構成に応じて、マシンの IP アドレスを `/etc/hosts` ファイル、`/etc/inet/ipnodes` ファイル、またはその両方で指定します。

---

注- 指定したマシンはすべてのクラスターノードから到達可能で、定足数サーバーをマシン上で実行してある必要があります。

---

**clsetup** ユーティリティーは、定足数サーバーのポート番号を入力するようメッセージを表示します。

- 9 クラスターノードとやり取りする際に定足数サーバーが使用するポート番号を入力します。

新しい定足数デバイスを追加するか確認を求めるメッセージが表示されます。

- 10 「**yes**」と入力し、新しい定足数デバイスの追加を続行します。

新しい定足数デバイスが正常に追加されると、**clsetup** ユーティリティーではその旨のメッセージが表示されます。

- 11 定足数デバイスが追加されていることを確認します。

```
# clquorum list -v
```

#### 例 6-4 定足数サーバー定足数デバイスの追加

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

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

[Start the `clsetup` utility:]

**# clsetup**

[Select Quorum > Add a quorum device]

[Answer the questions when prompted.]

[You will need the following information.]

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

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

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

Command completed successfully.

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

[Verify that the quorum device is added:]

**# clquorum list -v**

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

**# clquorum status**

=== Cluster Quorum ===

-- Quorum Votes Summary --

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

-- Quorum Votes by Node --

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

-- Quorum Votes by Device --

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

## 定足数デバイスの削除または交換

この節では、定足数デバイスを削除または交換するための次の手順を説明します。

- 206 ページの「定足数デバイスを削除する」
- 207 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」
- 209 ページの「定足数デバイスを交換する」

### ▼ 定足数デバイスを削除する

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

定足数デバイスを削除すると、そのデバイスは定足数確立の投票に参加できなくなります。2 ノードクラスタでは、定足数デバイスが少なくとも1つは構成されている必要があります。構成されているデバイスが、クラスタの最後の定足数デバイスの場合は、`clquorum(1CL)` はデバイスを構成から削除できません。ノードを削除する場合は、そのノードに接続されている定足数デバイスをすべて削除してください。

---

注- 削除するデバイスがクラスタの最後の定足数デバイスの場合は、[207 ページ](#)の「クラスタから最後の定足数デバイスを削除する」の手順を参照してください。

---

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

この手順では、長形式の 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(1CL)` ユーティリティーを実行します。  
`# clsetup`  
メインメニューが表示されます。
- 4 定足数のオプションに対応する番号を入力します。

- 5 定足数デバイスを削除するオプションに対応する番号を入力します。  
削除プロセス中に表示される質問に答えます。
- 6 `clsetup`を終了します。
- 7 定足数デバイスが削除されたことを確認します。  
`# clquorum list -v`

### 例6-5 定足数デバイスの削除

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

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

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

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

**注意事項** 定足数サーバー定足数デバイスの削除中に、クラスタと定足数サーバーホストの間の通信が失われた場合、定足数サーバーホストに関する無効な構成情報をクリーンアップする必要があります。このクリーンアップの実行に関する説明は、[221 ページ](#)の「[期限切れの定足数サーバークラスタ情報のクリーンアップ](#)」を参照してください。

### ▼ クラスタから最後の定足数デバイスを削除する

この手順では、`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、[206 ページ](#)の「[定足数デバイスを削除する](#)」の手順に従ってください。

定足数デバイスを追加する処理では、ノードが再構成されるため、障害のあった定足数デバイスに影響が及び、マシンでパニックが発生します。F(強制)オプションを使用すると、マシンでパニックを発生させることなく、障害があった定足数デバイスを削除できます。clquorum(1CL) コマンドでは、構成からデバイスを削除することができます。不具合が発生した定足数デバイスを削除したあと、clquorum add コマンドで新しいデバイスを追加することができます。194 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、そのデバイスを削除します。

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

---

注-また、削除するノードを保持状態とし、定足数デバイスを clquorum removequorum コマンドを使用して削除することができます。clsetup(1CL) クラスタ管理メニューは、クラスタがインストールモードのときは使用できません。詳細は、278 ページの「ノードを保守状態にする」を参照してください。

---

- 3 定足数デバイスが削除されたことを確認します。

```
# clquorum list -v
```

### 例 6-6 最後の定足数デバイスの削除

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

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



```
scphyshost-2 node
scphyshost-3 node
```

## ▼ 定足数デバイスを交換する

この作業は、既存の定足数デバイスをほかの定足数デバイスに交換する場合に行います。定足数デバイスは、類似したデバイスタイプに交換することも(例: NAS デバイスをほかの NAS デバイスに置き換える)、あるいは類似点がないデバイスに交換することも(例: NAS デバイスを共有ディスクに置き換える)こともできます。

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

この手順では、長形式の 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、[194 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、[206 ページの「定足数デバイスを削除する」](#)を参照してください。
- 3 定足数デバイスが障害が発生したディスクである場合は、ディスクを取り替えます。  
『[Oracle Solaris Cluster 3.3 Hardware Administration Manual](#)』で、使用しているディスク装置のハードウェア作業を参照してください。

## 定足数デバイスの保守

この節では、定足数デバイスを保守するための次の手順を説明します。

- [210 ページの「定足数デバイスのノードリストを変更する」](#)
- [212 ページの「定足数デバイスを保守状態にする」](#)
- [213 ページの「定足数デバイスを保守状態から戻す」](#)
- [215 ページの「定足数構成を一覧表示する」](#)
- [216 ページの「定足数デバイスを修復する」](#)
- [217 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

**clsetup(1CL)** ユーティリティを使用すると、既存の定足数デバイスのノードリストにノードを追加したり、ノードリストからノードを削除したりできます。定足数デバイスのノードリストを変更するには、定足数デバイスを削除し、削除した定足数デバイスへのノードの物理的な接続を変更して、定足数デバイスをクラスタ構成に追加し直す必要があります。定足数デバイスを追加すると、**clquorum(1CL)** は自動的に、ディスクが接続されているすべてのノードについて、ノードからディスクへのパスを構成します。

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

この手順では、長形式の 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 定足数オプションに対応する番号を入力します。  
「定足数メニュー」が表示されます。
- 5 定足数デバイスを削除するオプションに対応する番号を入力します。  
画面の指示に従います。削除するディスクの名前を問い合わせられます。
- 6 定足数デバイスへのノード接続を追加または削除します。
- 7 定足数デバイスを追加するオプションに対応する番号を入力します。  
画面の指示に従います。定足数デバイスとして使用するディスクの名前を問い合わせられます。
- 8 定足数デバイスが追加されていることを確認します。  

```
# clquorum list -v
```

## 例6-7 定足数デバイスノードリストの変更

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

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

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

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

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

```
# clsetup
```

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

```
.
```

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

```
.
```

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

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

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

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

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

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

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

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

```
.
```

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

```
.
```

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

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

```
Information      Example:
quorum device name  d2
```

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

```
clquorum add d2
```

```
Command completed successfully.
```

```
Quit the clsetup utility.
```

```
[Verify that the correct nodes have paths to the quorum device.  
In this example, note that phys-schost-3 has been added to the  
enabled hosts list.]
```

```
# clquorum show d2 | grep Hosts  
=== Quorum Devices ===
```

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

```
[Verify that the modified quorum device is online.]
```

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

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

| Device Name | Present | Possible | Status |
|-------------|---------|----------|--------|
| -----       | -----   | -----    | -----  |
| d2          | 1       | 1        | OnLine |

## ▼ 定足数デバイスを保守状態にする

`clquorum(1CL)`コマンドを使用して定足数デバイスを保守状態にします。現在、`clsetup(1CL)`ユーティリティにこの機能はありません。この手順は、Oracle Solaris Cluster Manager GUIを使用しても実行できます。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

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、278 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 6-8 定足数デバイスを保守状態にする

次に、定足数デバイスを保守状態にし、結果を検証する例を示します。

```
# clquorum disable d20
# clquorum status d20

=== Cluster Quorum ===

--- Quorum Votes by Device ---

Device Name      Present      Possible      Status
-----
d20                1             1             Offline
```

参照 定足数デバイスを有効にし直す方法については、213 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、278 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  $1(2-1)$  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、279 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、このノードを再起動します。

- 4 定足数投票数を確認します。

```
# clquorum show +
```

### 例 6-9 定足数投票数 (定足数デバイス) のリセット

次に、定足数デバイスの投票数をリセットしてデフォルト設定に戻し、結果を検証する例を示します。

```
# clquorum enable d20
# clquorum show +
```

```
=== Cluster Nodes ===
```

```
Node Name:                phys-schost-2
Node ID:                   1
Quorum Vote Count:        1
Reservation Key:           0x43BAC41300000001
```

```

Node Name:                phys-schost-3
Node ID:                  2
Quorum Vote Count:       1
Reservation Key:          0x43BAC41300000002

```

```
=== Quorum Devices ===
```

```

Quorum Device Name:      d3
Enabled:                 yes
Votes:                   1
Global Name:             /dev/did/rdisk/d20s2
Type:                    shared_disk
Access Mode:             scsi2
Hosts (enabled):        phys-schost-2, phys-schost-3

```

## ▼ 定足数構成を一覧表示する

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

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

---

注- 定足数デバイスに対するノード接続の数を増減させる場合、定足数が自動的に再計算されることはありません。すべての定足数デバイスをいったん削除し、その後それらを構成に追加し直すと、正しい定足数が再設定されます。2 ノードクラスタの場合、定足数デバイスを取り外して、もとの定足数デバイスに戻す前に一時的に新しい定足数デバイスを追加します。次に一時的に追加した定足数デバイスを取り外します。

---

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

この手順では、長形式の 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(1CL)** を使用して、定足数構成を一覧表示します。

```
% clquorum show +
```

### 例 6-10 定足数構成の一覧表示

```
% clquorum show +
```

```
=== Cluster Nodes ===
```

```
Node Name:                phys-schost-2
Node ID:                  1
Quorum Vote Count:       1
Reservation Key:         0x43BAC41300000001
```

```
Node Name:                phys-schost-3
Node ID:                  2
Quorum Vote Count:       1
Reservation Key:         0x43BAC41300000002
```

=== Quorum Devices ===

```
Quorum Device Name:      d3
Enabled:                 yes
Votes:                   1
Global Name:             /dev/did/rdsd/d20s2
Type:                    shared_disk
Access Mode:             scsi2
Hosts (enabled):         phys-schost-2, phys-schost-3
```

## ▼ 定足数デバイスを修復する

この作業は、動作が不正な定足数デバイスを交換する場合に行なってください。

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

この手順では、長形式の 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、[194 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、[206 ページの「定足数デバイスを削除する」](#)を参照してください。

### 2 ディスクデバイスを交換します。

ディスクデバイスを交換する方法については、『[Oracle Solaris Cluster 3.3 Hardware Administration Manual](#)』に記載された、ディスク装置のハードウェア作業の説明を参照してください。



- 3 交換したディスクを新しい定足数デバイスとして追加します。

ディスクを新しい定足数デバイスとして追加する方法については、194 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、206 ページの「[定足数デバイスを削除する](#)」を参照してください。

---

## 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 秒に設定されています。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](#)」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](#)」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。

- 218 ページの「[Quorum Server Software の起動および停止](#)」
- 218 ページの「[定足数サーバーを起動する](#)」
- 219 ページの「[定足数サーバーを停止する](#)」
- 220 ページの「[定足数サーバーに関する情報の表示](#)」
- 221 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 Quorum Server のインストールと構成については、『Oracle Solaris Cluster ソフトウェアのインストール』の「定足数サーバーソフトウェアをインストールおよび構成する」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。

定足数サーバーの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「定足数サーバーソフトウェアをインストールおよび構成する」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、217 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

- 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 6-11 すべての構成済み定足数サーバーの起動

次の例では、構成されているすべての定足数サーバーを起動します。

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

## 例 6-12 特定の定足数サーバーの起動

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

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

## ▼ 定足数サーバーを停止する

- 1 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 6-13 すべての構成済み定足数サーバーの停止

次の例では、構成されているすべての定足数サーバーを停止します。

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

## 例 6-14 特定の定足数サーバーの停止

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

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

## 定足数サーバーに関する情報の表示

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

### ▼ 定足数サーバーに関する情報を表示する

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

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

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

#### 例 6-15 1つの定足数サーバーの構成の表示

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

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

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

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

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

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

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

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

```

Node ID:                2
Registration key:       0x439a2efb00000002

Node ID:                3
Registration key:       0x439a2efb00000003

Node ID:                4
Registration key:       0x439a2efb00000004

```

### 例 6-16 複数の定足数サーバーの構成の表示

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

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

### 例 6-17 動作しているすべての定足数サーバーの構成の表示

次の例では、動作しているすべての定足数サーバーの構成情報を表示します。

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

## 期限切れの定足数サーバークラスタ情報のク リーンアップ

quorumserver のタイプの定足数デバイスを削除するには、206 ページの「定足数デバイスを削除する」で説明されているように、clquorum remove コマンドを使用します。通常の動作では、このコマンドは定足数サーバーホストに関する定足数サーバーの情報も削除します。ただし、クラスタが定足数サーバーホストとの通信を失うと、定足数デバイスを削除しても、この情報がクリーンアップされません。

定足数サーバークラスタ情報は、次の状況で無効になります。

- clquorum remove コマンドを使用してクラスタ定足数デバイスを削除せずに、クラスタの運用を停止した場合。
- 定足数サーバーホストが停止している間に、quorum\_server タイプの定足数デバイスをクラスタから削除した場合。



注意 - タイプ quorumserver の定足数デバイスがまだクラスタから削除されていない場合、この手順を使用して無効な定足数サーバーを削除すると、クラスタ定足数に障害が発生する可能性があります。

## ▼ 定足数サーバーの構成情報をクリーンアップする

始める前に 206 ページの「定足数デバイスを削除する」で説明されているとおりに、定足数サーバーの定足数デバイスを削除します。



注意-クラスタがまだこの定足数サーバーを使用している場合、この手順を実行するとクラスタ定足数に障害が発生します。

- 1 定足数サーバーホストでスーパーユーザーになります。
- 2 **clquorumserver clear** コマンドを使用して、構成ファイルをクリーンアップします。  

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

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

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

**-I clusterID** クラスタ ID です。

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

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

定足数サーバーは、ポート番号かインスタンス名で識別できません。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 6-18 定足数サーバー構成からの期限切れのクラスタ情報のクリーンアップ

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

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





# クラスタインターコネク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 10 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、[241 ページの「パブリックネットワークの管理」](#)の節を参照してください。

この章では、次のトピックの手順について説明します。

- [225 ページの「クラスタインターコネクの管理」](#)
- [241 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とパブリックネットワークの背景情報や概要については、『[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の管理

このセクションでは、クラスタトランスポートアダプタ、クラスタトランスポートケーブルなどのクラスタインターコネクの再構成手順について説明します。これらの手順では、Oracle Solaris Cluster ソフトウェアをインストールする必要があります。

通常、`clsetup`ユーティリティーを使用すると、クラスタインターコネクットのクラスタトランスポートを管理できます。詳細は、`clsetup(1CL)`のマニュアルページを参照してください。クラスタインターコネクットコマンドはすべて、グローバルクラスタ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をサービスする手順については、『Oracle Solaris Cluster 3.3 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 作業リスト:クラスタインターコネクットの管理

| 作業  | 参照先  |
|---|--|
| <code>clsetup(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                           | 28 ページの「クラスタ構成ユーティリティーにアクセスする」                                 |
| <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          | 227 ページの「クラスタインターコネクットの状態を確認する」                                |
| <code>clsetup</code> を使用することで、クラスタトランスポートケーブル、トランスポートアダプタ、またはスイッチを追加する        | 228 ページの「Cluster Transport Cable トランスポートアダプタ、トランスポートスイッチを追加する」 |
| <code>clsetup</code> を使用することで、クラスタトランスポートケーブル、トランスポートアダプタ、またはトランスポートスイッチを削除する | 231 ページの「Cluster Transport Cable トランスポートアダプタ、トランスポートスイッチを削除する」 |
| <code>clsetup</code> を使用することで、クラスタトランスポートケーブルを有効にする                           | 234 ページの「Cluster Transport Cable を有効にする」                       |
| <code>clsetup</code> を使用することで、クラスタトランスポートケーブルを無効にする                           | 235 ページの「Cluster Transport Cable を無効にする」                       |
| トランスポートアダプタのインスタンス番号の確認   | 237 ページの「トランスポートアダプタのインスタンス番号を確認する」                            |
| IP アドレスまたは既存のクラスタのアドレス範囲の変更   | 238 ページの「既存のクラスタのプライベートネットワークアドレスまたはアドレス範囲を変更する」               |

## クラスタインターコネクトでの動的再構成

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 作業マップ:パブリックネットワークインタフェースでの動的再構成

| 作業                                 | 参照先   |
|------------------------------------|---|
| 1. アクティブなインターコネクトからインタフェースを無効にして削除 | 242 ページの「パブリックネットワークインタフェースでの動的再構成」   |
| 2. パブリックネットワークインタフェース上で DR 操作を実行   | 「Solaris 10 on Sun Hardware」コレクションの『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』 |

### ▼ クラスタインターコネクトの状態を確認する

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

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状態を確認します。  
% clinterconnect status
- 2 一般的な状態メッセージについては、以下の表を参照してください。

| 状態メッセージ      | 説明および可能な処置  |
|--------------|---|
| Path online  | パスが現在正常に機能しています。処置は必要ありません。   |
| Path waiting | パスが現在初期化中です。処置は必要ありません。   |
| Faulted      | パスが機能していません。これは、パスが一時的に待機状態とオンライン状態の間にある状態の可能性があります。再び clinterconnect status を実行してもメッセージが繰り返される場合は、適切な処置を行ってください。 |

### 例 7-1 クラスタインターコネクットの状態を確認する

次に、正常に機能しているクラスタインターコネクットの状態の例を示します。

```
% clinterconnect status
-- Cluster Transport Paths --
      Endpoint                Endpoint                Status
      -----                -
Transport path: phys-schost-1:qfe1 phys-schost-2:qfe1 Path online
Transport path: phys-schost-1:qfe0 phys-schost-2:qfe0 Path online
Transport path: phys-schost-1:qfe1 phys-schost-3:qfe1 Path online
Transport path: phys-schost-1:qfe0 phys-schost-3:qfe0 Path online
Transport path: phys-schost-2:qfe1 phys-schost-3:qfe1 Path online
Transport path: phys-schost-2:qfe0 phys-schost-3:qfe0 Path online
```

## ▼ Cluster Transport Cable トランスポートアダプタ、トランスポートスイッチを追加する

クラスタ固有のトランスポートの要件については、『Oracle Solaris Cluster 3.3 Hardware Administration Manual』の「Interconnect Requirements and Restrictions」を参照してください。

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

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 3.3 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に対応する番号を入力します。
  - 5 トランスポートケーブルを追加するためのオプションに対応する番号を入力します。  
指示に従い、必要な情報を入力します。
  - 6 トランスポートアダプタをノードに追加するためのオプションに対応する番号を入力します。  
指示に従い、必要な情報を入力します。  
クラスタインターコネクットで次のアダプタのいずれかを使用する予定の場合、関連するエントリを各クラスタノードの /etc/system ファイルに追加します。このエントリは、次のシステム再起動後に有効になります。
- | アダプタ | エントリ                          |
|------|-------------------------------|
| ce   | set ce:ce_taskq_disable=1     |
| ipge | set ipge:ipge_taskq_disable=1 |
| ixge | set ixge:ixge_taskq_disable=1 |
- 7 トランスポートスイッチを追加するためのオプションに対応する番号を入力します。  
指示に従い、必要な情報を入力します。

- 8 クラスタトランスポートケーブルトランスポートアダプタ、トランスポートスイッチが追加されたことを確認します。

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

### 例 7-2 クラスタトランスポートケーブル、トランスポートアダプタ、トランスポートスイッチの追加

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

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

[Select either Add a transport cable,
Add a transport adapter to a node,
or Add a transport switch.]
[Answer the questions when prompted.]
  [You Will Need: ]
[Information:      Example:]
  node names      phys-schost-1
  adapter names   qfe2
  switch names    hub2
  transport type  dlpi
[Verify that the clinterconnect
  command completed successfully:]Command completed successfully.
Quit the clsetup Cluster Interconnect Menu and Main Menu.
[Verify that the cable, adapter, and switch are added:]
# clinterconnect show phys-schost-1:qfe2,hub2
===Transport Cables===
Transport Cable:                phys-schost-1:qfe2@0,hub2
Endpoint1:                      phys-schost-2:qfe0@0
Endpoint2:                      ethernet-1@2 ??? Should this be hub2?
State:                          Enabled

# clinterconnect show phys-schost-1:qfe2
=== Transport Adepters for qfe2
Transport Adapter:                qfe2
Adapter State:                   Enabled
Adapter Transport Type:          dlpi
Adapter Property (device_name):  ce
Adapter Property (device_instance): 0
Adapter Property (lazy_free):     1
Adapter Property (dlpi_heartbeat_timeout): 10000
Adpater Property (dlpi_heartbeat_quantum): 1000
Adapter Property (nw_bandwidth):  80
Adapter Property (bandwidth):     70
Adapter Property (ip_address):    172.16.0.129
Adapter Property (netmask):       255.255.255.128
Adapter Port Names:              0
Adapter Port SState (0):         Enabled
```

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

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

次の手順 クラスタトランスポートケーブルのインターコネクットのステータスを確認するには、[227 ページの「クラスタインターコネクットの状態を確認する」](#)を参照してください。

## ▼ Cluster Transport Cable トランスポートアダプタ、トランスポートスイッチを削除する

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

次の手順を使用して、クラスタトランスポートケーブル、クラスタトランスポートアダプタ、およびトランスポートスイッチをノード構成から削除します。ケーブルを無効にした場合、このケーブルの2つのエンドポイントは構成されたままになります。トランスポートケーブルの終端として使用されているアダプタは削除できません。



注意-各クラスタノードには、他のすべてのクラスタノードに対する(機能している)トランスポートパスが少なくとも1つずつ必要です。2つのノードは必ず接続されており、お互いに分離されているノードは存在しません。ケーブルを無効にする前には、必ず、ノードのクラスタインターコネクットの状態を確認してください。状態が冗長な場合、つまり別の接続が使用できる場合だけ、ケーブル接続を無効にします。ノードの最後の機能しているケーブルを無効にすると、そのノードはクラスタメンバーシップから外れます。

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

この手順では、長形式の 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状態を確認します。

```
# clinterconnect status
```



---

注意-2 ノードクラスタのいずれかのノードを削除しようとして「パス障害 (Path faulted)」などのエラーメッセージが表示された場合、この手順を続ける前に問題を調査してください。このような問題は、ノードパスが利用できないことを示しています。残りの動作中のパスを削除すると、このノードはクラスタメンバーシップから外れ、クラスタが再構成されます。

---

- 3 **clsetup** ユーティリティを起動します。

# **clsetup**

メインメニューが表示されます。

- 4 クラスタインターコネクットメニューにアクセスするためのオプションに対応する番号を入力します。

- 5 トランスポートケーブルを無効にするためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。

- 6 トランスポートケーブルを削除するためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。

---

注- 物理的にケーブル接続を解除する場合は、ポートと宛先デバイスをつないでいるケーブルを切り離します。

---

- 7 トランスポートアダプタをノードから削除するためのオプションに対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。

---

注- アダプタをノードから物理的に取り外す場合のハードウェアサービス手順については、『[Oracle Solaris Cluster 3.3 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に対応する番号を入力します。

指示に従い、必要な情報を入力します。アプリケーションのノード名、アダプタ名、およびスイッチ名を知っておく必要があります。



注- ポートがトランスポートケーブルの終端として使用されている場合、スイッチは削除できません。

- 9 ケーブル、アダプタ、またはスイッチが削除されたことを確認します。

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

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

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

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

```
[Become superuser on any node in the cluster.]
[Start the utility:]
# clsetup
[Select Cluster interconnect.]
[Select either Remove a transport cable,
Remove a transport adapter to a node,
or Remove a transport switch.]
[Answer the questions when prompted.]
  You Will Need:
    Information           Example:
    node names            phys-schost-1
    adapter names         qfe1
    switch names          hub1
[Verify that the clinterconnect
command was completed successfully:]
Command completed successfully.
[Quit the clsetup utility Cluster Interconnect Menu and Main Menu.]
[Verify that the cable, adapter, or switch is removed:]
# clinterconnect show phys-schost-1:qfe2,hub2
===Transport Cables===
Transport Cable:                phys-schost-2:qfe2@0,hub2
  Cable Endpoint1:              phys-schost-2:qfe0@0
  Cable Endpoint2:              ethernet-1@2 ??? Should this be hub2???
  Cable State:                  Enabled

# clinterconnect show phys-schost-1:qfe2
=== Transport Adepters for qfe2
Transport Adapter:              qfe2
  Adapter State:                Enabled
  Adapter Transport Type:       dlpi
  Adapter Property (device_name): ce
  Adapter Property (device_instance): 0
  Adapter Property (lazy_free): 1
  Adapter Property (dlpi_heartbeat_timeout): 10000
  Adpater Property (dlpi_heartbeat_quantum): 1000
```

```

Adapter Property (nw_bandwidth):      80
Adapter Property (bandwidth):         70
Adapter Property (ip_address):        172.16.0.129
Adapter Property (netmask):           255.255.255.128
Adapter Port Names:                   0
Adapter Port State (0):                Enabled

# clinterconnect show phys-schost-1:hub2
=== Transport Switches ===
Transport Switch:                      hub2
Switch State:                           Enabled
Switch Type:                             switch
Switch Port Names:                       1 2
Switch Port State(1):                    Enabled
Switch Port State(2):                    Enabled

```

## ▼ Cluster Transport Cable を有効にする

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

このオプションを使用し、既存のクラスタトランスポートケーブルを有効にします。

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

この手順では、長形式の 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に対応する番号を入力し、**Return** キーを押します。
- 4 トランスポートケーブルを有効にするためのオプションに対応する番号を入力し、**Return** キーを押します。  
 プロンプトが表示されたなら、指示に従います。ケーブルのいずれかの終端のノード名およびアダプタ名の両方を入力する必要があります。
- 5 ケーブルが有効になっていることを確認します。  

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

## 例 7-4 クラスタトランスポートケーブルを有効にする

次の例に、ノード `phys-schost-2` にあるアダプタ `qfe-1` のクラスタトランスポートケーブルを有効にする方法を示します。

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

[Answer the questions when prompted.]
[You will need the following information.]
  You Will Need:
Information:                                Example:
node names                                phys-schost-2
adapter names                             qfe1
switch names                              hub1
[Verify that the scinterconnect
command was completed successfully:]

clinterconnect enable phys-schost-2:qfe1

Command completed successfully.
[Quit the clsetup Cluster Interconnect Menu and Main Menu.]
[Verify that the cable is enabled:]
# clinterconnect show phys-schost-1:qfe2,hub2
Transport cable:  phys-schost-2:qfe1@0 ethernet-1@2    Enabled
Transport cable:  phys-schost-3:qfe0@1 ethernet-1@3    Enabled
Transport cable:  phys-schost-1:qfe0@0 ethernet-1@1    Enabled
```

## ▼ Cluster Transport Cable を無効にする

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

クラスタトランスポートケーブルを無効にし、クラスタインターコネクtpassを一時的に停止する必要があることがあります。一時的な停止は、クラスタインターコネクtpassで発生する問題の解決や、クラスタインターコネクtpass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トの状態を確認してください。状態が冗長な場合、つまり別の接続が使用できる場合だけ、ケーブル接続を無効にします。ノードの最後の機能しているケーブルを無効にすると、そのノードはクラスタメンバーシップから外れます。

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

この手順では、長形式の 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トの状態を確認します。

```
# clinterconnect status
```



注意-2 ノードクラスタのいずれかのノードを削除しようとして「パス障害(Path faulted)」などのエラーメッセージが表示された場合、この手順を続ける前に問題を調査してください。このような問題は、ノードパスが利用できないことを示しています。残りの動作中のパスを削除すると、このノードはクラスタメンバーシップから外れ、クラスタが再構成されます。

- 3 `clsetup` ユーティリティを起動します。

```
# clsetup
```

メインメニューが表示されます。
- 4 クラスタインターコネクトメニューにアクセスするためのオプションに対応する番号を入力し、**Return** キーを押します。
- 5 トランスポートケーブルを無効にするためのオプションに対応する番号を入力し、**Return** キーを押します。  
指示に従い、必要な情報を入力します。このクラスタインターコネクトのすべてのコンポーネントは無効になります。ケーブルのいずれかの終端のノード名およびアダプタ名の両方を入力する必要があります。
- 6 ケーブルが無効になっていることを確認します。

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

## 例 7-5 クラスタトランスポートケーブルを無効にする

次の例に、ノード `phys-schost-2` にあるアダプタ `qfe-1` のクラスタトランスポートケーブルを無効にする方法を示します。

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

[Answer the questions when prompted.]
[You will need the following information.]
 [ You Will Need:]
Information:          Example:
node names           phys-schost-2
adapter names       qfe1
switch names        hub1
[Verify that the clinterconnect
command was completed successfully:]
Command completed successfully.
[Quit the clsetup Cluster Interconnect Menu and Main Menu.]
[Verify that the cable is disabled:]
# clinterconnect show -p phys-schost-1:qfe2,hub2
Transport cable:  phys-schost-2:qfe1@0 ethernet-1@2   Disabled
Transport cable:  phys-schost-3:qfe0@1 ethernet-1@3   Enabled
Transport cable:  phys-schost-1:qfe0@0 ethernet-1@1   Enabled
```

## ▼ トランスポートアダプタのインスタンス番号を確認する

`clsetup` コマンドを使用して正しいトランスポートアダプタの追加と削除を行うには、トランスポートアダプタのインスタンス番号を確認する必要があります。アダプタ名は、アダプタの種類とアダプタのインスタンス番号を組み合わせたものです。

- 1 スロット番号にもとづき、アダプタの名前を確認してください。  
次の画面は例であり、個々のハードウェアと一致しない可能性があります。

```
# prtdiag
...
===== IO Cards =====
                Bus Max
                Bus  Dev,
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

プライベートネットワークアドレスまたは使用されるネットワークアドレスの範囲、またはその両方を変更するには、次の手順に従います。

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

- 1 各クラスタノード上で次のサブステップを実行することで、すべてのクラスタノードを再起動し、非クラスタモードにします。

- a. 非クラスタモードで起動するクラスタノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。

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

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

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

- 2 1つのノードから、`clsetup` ユーティリティを起動します。

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

- 3 IPアドレス範囲を変更するためのオプションに対応する番号を入力し、**Return** キーを押します。  
c1setup 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。  
c1setup 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。  
c1setup ユーティリティーは、デフォルトのネットマスクをそのまま使用してもよいかどうかを尋ねます。次の手順に進み、応答を入力します。
  - デフォルトのプライベートネットワーク IP アドレスを変更するには、次のサブステップを実行します。
    - a. c1setup 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。  
c1setup 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。  
c1setup 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。  
続いて、次の手順に進みます。

- IPアドレス範囲を変更するには、次のサブステップを実行します。
  - a. **clsetup**ユーティリティーの、デフォルトのアドレス範囲をそのまま使用してもよいかどうかに関する質問に対しては「no」と入力し、**Return**キーを押します。  
デフォルトのネットマスクを使用しない場合、**clsetup**ユーティリティーは、ユーザーがクラスタで構成する予定のノード、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再起動し、クラスタモードに戻します。
  - a. ノードを起動します。
    - SPARC ベースのシステム上で、次のコマンドを実行します。  
ok boot
    - x86 ベースのシステム上で、次のコマンドを実行します。  
GRUB メニューが表示された時点で、適切な Oracle Solaris エントリを選択し、Enter キーを押します。GRUB メニューは次のようになっています。  
  
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)  
+-----+  
| Solaris 10 /sol\_10\_x86 |  
| Solaris failsafe |  
+-----+



```

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

```

- 10 ノードが問題なく起動し、オンラインであることを確認します。

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

## パブリックネットワークの管理

Oracle Solaris Cluster ソフトウェアでは、パブリックネットワークでのインターネットプロトコルネットワークマルチパス (Internet Protocol network Multipathing, IPMP) の Oracle Solaris ソフトウェア実装がサポートされています。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 10 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 のマニュアル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

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 アドレスを設定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。
- あらかじめ IP アドレスをグループ内の削除する予定のアダプタから代替アダプタにスイッチオーバーせずに、IP ネットワークマルチパスグループのアダプタを構成解除 (`unplumb` する) または停止しないようにします (つまり、`if_mpadm(1M)` コマンドを使用)。
- 個々のマルチパスグループから削除する前に、アダプタを別のサブネットに配線しないようにします。
- 論理アダプタ操作は、マルチパスグループで監視中の場合でもアダプタに対して行うことができます。
- クラスタ内の各ノードについて、最低 1 つのパブリックネットワーク接続を維持しなければなりません。クラスタは、パブリックネットワーク接続がないとアクセスできません。
- クラスタ上の IP ネットワークマルチパスグループの状態を表示するには、`clinterconnect status` コマンドを使用します。

IP ネットワークマルチパスの詳細は、Oracle Solaris OS システム管理マニュアルセットの該当するマニュアルを参照してください。

表 7-3 作業リスト:パブリックネットワークの管理

| Oracle Solaris OS のリリース | 参照先   |
|-------------------------|---|
| Oracle Solaris 10 OS    | 『Solaris のシステム管理 (IP サービス)』の「IP ネットワークマルチパス (トピック)」 |

クラスタソフトウェアをインストールする手順については、『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をサービスする手順については、『Oracle Solaris Cluster 3.3 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(1M)` コマンドを使用して、削除するアダプタからマルチパスグループ内の別のアダプタに IP アドレスを切り替えます。
- アクティブなネットワークインタフェースを適切に無効にせずにパブリックネットワークインタフェースカードを削除しようとした場合、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パスグループの場合、無効にしたネットワークアダプタ上で DR 削除操作を実行している間に残りのネットワークアダプタに障害が発生すると、可用性に影響が生じます。これは、DR 操作の間は、残りのネットワークアダプタのフェイルオーバー先が存在しないためです。

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

表 7-4 作業マップ:パブリックネットワークインタフェースでの動的再構成

| 作業   | 参照先   |
|--|---|
| 1. <code>if_mpadm</code> を使用して、IP アドレスをマルチパスグループ内の削除する予定のアダプタから代替アダプタへの切り換えを実行 | <code>if_mpadm(1M)</code> のマニュアルページ<br>『Solaris のシステム管理 (IP サービス)』のパート VI 「IPMP」  |
| 2. <code>ifconfig</code> コマンドを使用して、マルチパスグループからアダプタを削除                          | <code>ifconfig(1M)</code> のマニュアルページ<br>『Solaris のシステム管理 (IP サービス)』のパート VI 「IPMP」  |
| 3. パブリックネットワークインタフェース上で DR 操作を実行   | 「Solaris 10 on Sun Hardware」コレクションの『Sun Enterprise 10000 DR Configuration Guide』と『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』 |



## ノードの追加と削除

---

この章では、クラスタへのノードの追加とノードの削除方法を説明します。

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

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

### クラスタへのノードの追加

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

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の IP アドレスを設定しない場合、発生する問題は次の2点です。

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 デバイスを設定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作業を示します。作業は、示されている順に実行してください。

表 8-1 作業マップ: 既存のグローバルクラスタまたはゾーンクラスタへのノードの追加

| 作業   | 参照先   |
|--|---|
| ホストアダプタのノードへの取り付けと、既存のクラスタインターコネクトが新しいノードをサポートできることの確認   | 『Oracle Solaris Cluster 3.3 Hardware Administration Manual』   |
| 共有記憶装置の追加  | 『Oracle Solaris Cluster 3.3 Hardware Administration Manual』   |
| <code>clsetup</code> を使用した、承認済みノードリストへのノードの追加  | 246 ページの「ノードを認証ノードリストに追加する」   |
| 新しいクラスタノードへのソフトウェアのインストールと構成   | 『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  | 245 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」 |

## ▼ ノードを認証ノードリストに追加する

既存のグローバルクラスタまたはゾーンクラスタに Oracle Solaris ホストまたは仮想マシンを追加する前に、プライベートクラスタインターコネクトとの物理的な接続が動作しているかどうかも含め、必要なハードウェアがすべてノードに正しくインストールされ構成されていることを確認してください。

ハードウェアのインストールについては、『Oracle Solaris Cluster 3.3 Hardware Administration Manual』または各サーバーに付属するハードウェアマニュアルを参照してください。

この手順によって、マシンは自分自身をクラスタ内にインストールします。つまり、自分のノード名を当該クラスタの認証ノードリストに追加します。

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

この手順では、長形式の 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 の作業マップに記載されている必要なハードウェアのインストールと構成作業をすべて正しく完了していることを確認します。

- 3 **clsetup** ユーティリティを起動します。

```
phys-schost# clsetup
```

メインメニューが表示されます。

---

注-ノードをゾーンクラスタに追加するために、**clzonecluster** ユーティリティを使用します。ゾーンをゾーンクラスタに手動で追加する方法は、手順 9 を参照してください。

---

- 4 新規ノードメニューを表示するためのオプションに対応する番号を入力し、**Return** キーを押します。
- 5 承認済みリストを変更するためのオプションに対応する番号を入力し、**Return** キーを押します。自分自身を追加できるマシンの名前を指定します。  
指示に従って、ノードの名前をクラスタに追加します。追加するノードの名前が問い合わせられます。
- 6 作業が正常に行われたことを確認します。  
作業が正常に行われた場合、**clsetup** ユーティリティは「コマンドが正常に完了しました」というメッセージを表示します。
- 7 新しいマシンがクラスタに追加されないようにするために、新しいマシンを追加する要求を無視するようクラスタに指示する、オプションに対応する番号を入力します。**Return** キーを押します。  
**clsetup** のプロンプトに従います。このオプションを設定すると、クラスタは、自分自身をクラスタに追加しようとする新しいマシンからのパブリックネットワーク経由の要求をすべて無視します。
- 8 **clsetup** ユーティリティを終了します。

- 9 ノードをゾーンクラスタに手動で追加するには、**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、**bge0** は両方のマシンのパブリックネットワークアダプタです。

```
clzc:sczone>add node
clzc:sczone:node>set physical-host=phys-cluster-1
clzc:sczone:node>set hostname=hostname1
clzc:sczone:node>add net
clzc:sczone:node:net>set address=hostname1
clzc:sczone:node:net>set physical=bge0
clzc:sczone:node:net>end
clzc:sczone:node>end
clzc:sczone>add node
clzc:sczone:node>set physical-host=phys-cluster-2
clzc:sczone:node>set hostname=hostname2
clzc:sczone:node>add net
clzc:sczone:node:net>set address=hostname2
clzc:sczone:node:net>set physical=bge0
clzc:sczone:node:net>end
clzc:sczone:node>end
```

ノードを構成する手順の詳細は、『[Oracle Solaris Cluster ソフトウェアのインストール](#)』の「[ゾーンクラスタの設定](#)」を参照してください。

- 10 新しいクラスタノード上でソフトウェアをインストールして構成します。  
cluster create または JumpStart ソフトウェアのいずれかを使用して、新しいノードのインストールと構成を完了します。詳細は、『[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。

#### 例 8-1 認証ノードリストへのグローバルクラスタノードの追加

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

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

claccess allow -h phys-schost-3

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

参照 [clsetup\(1CL\)](#)

クラスタノードを追加する作業の一連の手順については、[表 8-1](#)、「[作業マップ: クラスタノードの追加](#)」を参照してください。



ノードを既存のリソースグループに追加する方法については、『Oracle Solaris Cluster Data Services Planning and Administration Guide』を参照してください。

## グローバルクラスタ内の非投票ノード (ゾーン) の作成

ここでは、非投票ノード (単にゾーンと呼ばれる) をグローバルクラスタノード上に作成するための次の情報と手順を説明します。

### ▼ グローバルクラスタ内に非投票ノードを作成する

- 1 非投票ノードを作成するグローバルクラスタノードでスーパーユーザーになります。  
大域ゾーンで作業してください。
- 2 各ノードで、サービス管理機能 (**Service Management Facility**、**SMF**) のマルチユーザーサービスがオンラインになっていることを確認します。  
ノードのサービスがまだオンラインでない場合は、次のステップに進む前に状態がオンラインに変わるまで待ちます。

```
phys-schost# svcs multi-user-server node
STATE      STIME      FMRI
online     17:52:55  svc:/milestone/multi-user-server:default
```

- 3 新しいゾーンを構成、インストール、および起動します。

---

注 - グローバルクラスタの非投票ノードでリソースグループ機能をサポートするには、`autoboot` プロパティを `true` に設定します。

---

Solaris のドキュメントの手順に従ってください。

- a. 『Oracle Solaris のシステム管理 (Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン)』の第 18 章「非大域ゾーンの計画と構成 (手順)」の手順を実行します。
  - b. 『Oracle Solaris のシステム管理 (Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン)』の「ゾーンのインストールと起動」の手順を実行します。
  - c. 『Oracle Solaris のシステム管理 (Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン)』の「ゾーンの起動方法」の手順を実行します。
- 4 ゾーンが **ready** 状態であることを確認します。

```
phys-schost# zoneadm list -v
ID  NAME      STATUS      PATH
```

```
0 global running /
1 my-zone ready /zone-path
```

- 5 (省略可能) 共有 IP ゾーンでは、プライベート IP アドレスとプライベートホスト名をゾーンに割り当てます。

次のコマンドは、クラスタのプライベート IP アドレスの範囲から、使用可能な IP アドレスを選択し、割り当てます。またこのコマンドは、指定されたプライベートホスト名、またはホスト別名をゾーンに割り当て、割り当てられたプライベート IP アドレスにそれをマッピングします。

```
phys-schost# clnode set -p zprivatehostname=hostalias node:zone
```

**-p** プロパティを指定します。

**zprivatehostname=hostalias** ゾーンプライベートホスト名、またはホスト別名を指定します。

**node** ノードの名前。

**zone** グローバルクラスタの非投票ノードの名前。

- 6 初期内部ゾーン構成を実行します。

『Oracle Solaris のシステム管理 (Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン)』の「初期内部ゾーン構成を実行する」の手順に従います。次のどちらかの方法を選択します。

- ゾーンにログインします。
- /etc/sysidcfg ファイルを使用します。

- 7 非投票ノードで、**nsswitch.conf** ファイルを変更します。

これらの変更により、クラスタ固有のホスト名と IP アドレスの検索をゾーンが解決できるようになります。

- a. ゾーンにログインします。

```
phys-schost# zlogin -c zonename
```

- b. 編集するため **/etc/nsswitch.conf** ファイルを開きます。

```
sczone# vi /etc/nsswitch.conf
```

- c. **hosts** エントリと **netmasks** エントリのルックアップの先頭に、**cluster** スイッチを追加し、その後に **files** スイッチを追加します。

変更されたエントリは次のようになるはずです。

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

```

- d. ほかのすべてのエントリでは、**files** スイッチが、エントリに一覧表示される最初のスイッチになるようにする必要があります。
  - e. ゾーンを終了します。
- 8 排他的 IP ゾーンを作成した場合は、ゾーン上の各 `/etc/hostname.interface` ファイルで IPMP グループを構成します。
- ゾーン内のデータサービストラフィックに使用されているパブリックネットワークアダプタごとに、IPMP グループを設定します。この情報は、大域ゾーンから継承されません。IPMP グループをクラスタ内に構成する方法については、『[Oracle Solaris Cluster ソフトウェアのインストール](#)』の「パブリックネットワーク」を参照してください。
- 9 ゾーンにより使用されるすべての論理ホスト名リソースの名前とアドレスのマッピングを設定します。
- a. ゾーン上の `/etc/inet/hosts` ファイルに名前とアドレスのマッピングを追加します。  
この情報は、大域ゾーンから継承されません。
  - b. ネームサーバーを使用している場合は、名前とアドレスのマッピングを追加します。

## クラスタからのノードの削除

このセクションでは、グローバルクラスタまたはゾーンクラスタ上のノードを削除する手順について説明します。グローバルクラスタから特定のゾーンクラスタを削除することもできます。次の表に、ノードを既存のクラスタから削除するときに行う作業を示します。作業は、示されている順に実行してください。



注意 - RAC 構成の場合、この手順のみを使用してノードを削除すると、再起動中のノードでパニックが発生する可能性があります。RAC 構成からノードを削除する方法については、『[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 用サポートを削除する](#)」を参照してください。このプロセスが終了したら、次の適切な手順に従ってください。

表 8-2 作業マップ: ノードの削除

| 作業   | 参照先  |
|--|--|
| 削除するノードからすべてのリソースグループとデバイスグループを移動                      | <code>clnode evacuate node</code>  |
| 許可されたホストをチェックして、ノードを削除できることを確認                         | <code>claccess show node</code><br><code>claccess allow -h node-to-remove</code>   |
| ノードを削除できない場合に、クラスタ構成へのアクセス権をノードに付与                     |  |
| すべてのデバイスグループからノードを削除                                   | 142 ページの「デバイスグループからノードを削除する (Solaris Volume Manager)」  |
| 削除するノードに接続されているすべての定足数デバイスを削除                          | 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。<br><br>206 ページの「定足数デバイスを削除する」<br><br>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。<br><br>207 ページの「クラスタから最後の定足数デバイスを削除する」 |
| 削除するノードを非クラスタモードにする                                    | 278 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  | 252 ページの「ゾーンクラスタからノードを削除する」  |
| クラスタソフトウェア構成からノードを削除                                   | 253 ページの「クラスタソフトウェア構成からノードを削除する」   |
| (省略可能) 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 | 282 ページの「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」  |
| ゾーンクラスタ全体の削除   | 295 ページの「ゾーンクラスタを削除する」   |

## ▼ ゾーンクラスタからノードを削除する

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

- 1 グローバルクラスタのノードのスーパーユーザーになります。

- 2 ノードとそのゾーンクラスタを指定して、削除するゾーンクラスタノードを停止します。

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

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

- 3 ゾーンクラスタノードをアンインストールします。

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

- 4 ゾーンクラスタノードを構成から削除します。

次のコマンドを使用します。

```
phys-schost# clzonecluster configure zoneclustername
```

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

- 5 ノードがゾーンクラスタから削除されたことを確認します。

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

## ▼ クラスタソフトウェア構成からノードを削除する

この手順を実行して、ノードをグローバルクラスタから削除します。

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

この手順では、長形式の 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の構成から削除していること、および、このノードを保守状態にしていることを確認します。
- 2 削除するノードで、スーパーユーザーになるか、**RBAC** の承認 `solaris.cluster.modify` を提供する役割になります。グローバルクラスタのノードから、次の手順のステップをすべて実行します。
- 3 削除するグローバルクラスタノードを非クラスタモードで起動します。ゾーンクラスタノードの場合は、この手順を実行する前に、[252 ページの「ゾーンクラスタからノードを削除する」](#)の手順を実行します。
  - SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
shutdown -g -y -i0
```

Press any key to continue

- a. GRUB メニューで矢印キーを使用して該当する **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 メニューは次のようになっています。

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

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

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

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

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

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a)
| kernel /platform/i86pc/multiboot
| module /platform/i86pc/boot_archive
+-----+
```

Use the ^ and v keys to select which entry is highlighted.  
Press 'b' to boot, 'e' to edit the selected command in the  
boot sequence, 'c' for a command-line, 'o' to open a new line  
after ('O' for before) the selected line, 'd' to remove the  
selected line, or escape to go back to the main menu.

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

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

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

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。

画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a)
|
```

```
| kernel /platform/i86pc/multiboot -x |
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b**と入力して、ノードを非クラスタモードで起動します。

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

---

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

---

- 4 削除するノードで、ノードをクラスタから削除します。

```
phys-schost# clnode remove -F
```

`clnode remove` コマンドが失敗し、無効なノード参照が存在したままになっている場合は、アクティブなノードで `clnode clear -F nodename` を実行します。

---

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

---

- 5 別のクラスタノードから、ノードの削除を確認します。

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

- 6 ノードの削除を完了します。

- 削除するノードから **Oracle Solaris Cluster** ソフトウェアをアンインストールする場合は、[282 ページの「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」](#)に進んでください。
- 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 3.3 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例を示します。 `clnode remove` コマンドは、クラスタから削除するノード (`phys-schost-2`) から非クラスタモードで実行されます。

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

**参照** 削除するノードから 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、282 ページの「[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](#)」を参照してください。

ハードウェア作業については、『[Oracle Solaris Cluster 3.3 Hardware Administration Manual](#)』を参照してください。

クラスタノードを削除する作業の概要については、[表 8-2](#) を参照してください。

ノードを既存のクラスタに追加する方法については、246 ページの「[ノードを認証ノードリストに追加する](#)」を参照してください。

## ▼ グローバルクラスタから非投票ノード(ゾーン)を削除する

- 1 非投票ノードを作成したグローバルクラスタノードでスーパーユーザーになります。
- 2 システムから非投票ノードを削除します。  
『[Oracle Solaris のシステム管理 \(Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン\)](#)』の「[システムから非大域ゾーンを削除する](#)」の手順に従います。

## ▼ 2 ノード接続より大きなクラスタでアレイと単一ノード間の接続を削除する

3 ノードまたは 4 ノード接続のクラスタでストレージアレイを単一クラスタノードから取り外すには、次の手順を使用します。



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

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

- 1 取り外す予定のストレージアレイに関連付けられているすべてのデータベーステーブル、データサービス、ボリュームのバックアップを作成します。
- 2 切断する予定のノードで動作しているリソースグループとデバイスグループを判別します。

```
phys-schost# clresourcegroup status
phys-schost# cldevicegroup status
```

- 3 必要であれば、切断する予定のノードからすべてのリソースグループとデバイスグループを移動します。



注意 ((SPARCのみ)) - Oracle RACソフトウェアをクラスタで実行している場合、グループをノードから移動する前に、動作している Oracle RAC データベースのインスタンスを停止します。手順については、『Oracle Database Administration Guide』を参照してください。

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

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

- 4 デバイスグループを保守状態にします。  
Veritas 共有ディスクグループへの入出力活動を休止させる手順については、VxVM のマニュアルを参照してください。  
デバイスグループを保守状態にする手順については、[278 ページの「ノードを保守状態にする」](#)を参照してください。
- 5 デバイスグループからノードを削除します。  
VxVM または raw ディスクを使用している場合は、cldevicegroup(1CL) コマンドを使用して、デバイスグループを削除します。
- 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 -z zone -n node + | resourcegroup
```



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

- 13 ノードの `/devices` と `/dev` エントリを更新して、デバイスの名前空間を更新します。

```
phys-schost# devfsadm -C
cldevice refresh
```

- 14 デバイスグループをオンラインに戻します。

Veritas 共有ディスクグループをオンラインにする手順については、Veritas Volume Manager のマニュアルを参照してください。

デバイスグループをオンラインにする方法については、[279 ページの「ノードを保守状態から戻す」](#)を参照してください。

## ▼ エラーメッセージを修正する

クラスタノードの削除手順のいずれかを実行中に発生したエラーメッセージを修正するには、次の手順を実行します。

- 1 グローバルクラスタへのノードの再結合を試みます。この手順は、グローバルクラスタ上のみで実行します。
 

```
phys-schost# boot
```
- 2 ノードがクラスタに正常に再結合されているかどうかを確認します。
  - 再結合されていない場合は、[手順 b](#)に進みます。
  - 再結合されている場合は、次の各手順を行なってノードをデバイスグループから削除します。
    - a. ノードが正常にクラスタに再結合された場合は、残っているデバイスグループからノードを削除します。  
[141 ページの「すべてのデバイスグループからノードを削除する」](#)の作業を行います。
    - b. すべてのデバイスグループからノードを削除したあと、[282 ページの「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」](#)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
```
- 4 282 ページの「[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](#)」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。

- 261 ページの「クラスタの管理の概要」
- 294 ページの「ゾーンクラスタ管理タスクの実行」
- 300 ページの「障害追跡」

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

### クラスタの管理の概要

このセクションでは、グローバルクラスタやゾーンクラスタ全体の管理作業を実行する方法を説明します。次の表に、これらの管理作業と、関連する手順を示します。クラスタの管理作業は通常は大域ゾーンで行います。ゾーンクラスタを管理するには、そのゾーンクラスタをホストするマシンが 1 台以上クラスタモードで起動していることが必要です。すべてのゾーンクラスタノードが起動し動作している必要はありません。現在クラスタ外にあるノードがクラスタに再結合すると、構成の変更点が 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 作業リスト:クラスタの管理

| 作業  | 参照先  |
|---|--|
| クラスタへのノードの追加または削除   | 第 8 章「ノードの追加と削除」   |
| クラスタ名を変更  | 262 ページの「クラスタ名を変更する」<br>264 ページの「クラスタ名の変更 (Veritas CVM を使用する<br>場合)」                 |
| ノード ID およびそれらの対応する<br>ノード名の一覧の表示  | 265 ページの「ノード ID をノード名にマップする」   |
| クラスタへの新しいノードの追加を許<br>可または拒否   | 265 ページの「新しいクラスタノード認証で作業する」  |
| Network Time Protocol (NTP) を使用し<br>てクラスタの時刻を変更   | 267 ページの「クラスタの時刻をリセットする」   |
| ノードを停止し、SPARC ベースのシス<br>テムでは OpenBoot PROM ok プロンプ<br>ト、x86 ベースのシステムでは GRUB<br>メニューで「Press any key to<br>continue」というメッセージを表示 | 269 ページの「SPARC: ノードで OpenBoot PROM (OBP) を<br>表示する」                                  |
| プライベートホスト名の追加または変<br>更  | 273 ページの「グローバルクラスタ上の非投票ノードのプ<br>ライベートホスト名を追加する」<br>270 ページの「ノードのプライベートホスト名を変更す<br>る」 |
| クラスタノードを保守状態に変更   | 278 ページの「ノードを保守状態にする」  |
| ノード名の変更   | 275 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  | 279 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<br>ンインストール  | 282 ページの「Oracle Solaris Cluster ソフトウェアをクラ<br>スタノードからアンインストールする」                      |
| SNMP Event MIB の追加および管理   | 287 ページの「SNMP イベント MIB を有効にする」<br>290 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設定  | 292 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<br>ケーション用ゾーンクラスタの準<br>備、ゾーンクラスタの削除   | 294 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

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 **clsetup** ユーティリティーを起動します。  
phys-schost# **clsetup**  
メインメニューが表示されます。
- 3 クラスタ名を変更するには、クラスタその他のプロパティのオプションに対応する番号を入力します。  
「クラスタその他のプロパティ」メニューが表示されます。
- 4 メニューから選択を行って、画面の指示に従います。
- 5 **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再起動します。  
phys-schost# **reboot**

### 例 9-1 クラスタ名の変更

次の例に、新しいクラスタ名 dromedary へ変更するために、**clsetup(1CL)** ユーティリティーから生成される **cluster(1CL)** コマンドを示します。

```
phys-schost# cluster rename -c dromedary
```

## ▼ クラスタ名の変更 (Veritas CVM を使用する場合)

CVM (Veritas Cluster Volume Manager) ディスクグループのいずれかに Oracle Solaris Cluster をお使いの場合、初回のインストール後にクラスタ名を変更することができます。

- 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 **Oracle RAC 11g**または**10g**をお使いの場合、次の手順を実行します。
  - a. すべてのノード上の **CRS (Oracle Cluster Ready Services)** を停止して無効にします。
 

```
/etc/init.d/init.crs stop
/etc/init.d/init.crs disable
```
  - b. **Oracle Solaris Cluster SUNW.scalable\_rac\_server\_proxy** リソースおよび **SUNW.ScalDevGroup** リソースを無効にします。これにより、リブート後にリソースがオンラインになるのを回避できます。
 

```
# clrs disable resource_name
```
- 3 各 CVM ディスクグループ内のすべてのボリュームを停止します。
 

```
# vxvol -g dg_name stopall
```
- 4 CVM ディスクグループをデポートします。
 

```
# vxdg deport dg_name
```
- 5 クラスタの名前を変更します。
 

```
# cluster rename -c new_name
```
- 6 クラスタ内のすべてのノードを再起動します。
- 7 ディスクグループを共有ディスクグループとしてインポートします。
 

```
# vxdg -s import dg_name
```
- 8 各ディスクグループ内のボリュームを起動します。
 

```
# vxvol -g dg_name startall
```
- 9 すべてのノード上の **Oracle CRS** を有効にして起動します。
 

```
/etc/init.d/init.crs enable
/etc/init.d/init.crs start
```
- 10 先に無効にした **Oracle Solaris Cluster** リソースを有効にして起動します。
 

```
# clrs enable +
```



## ▼ ノード ID をノード名にマップする

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(1CL)** コマンドを使用して、グローバルクラスタに対するクラスタ構成情報を一覧表示します。

```
phys-schost# clnode show | grep Node
```

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

```
phys-schost# zlogin sczone clnode -v | grep Node
```

### 例 9-2 ノード名のノード ID へのマップ

次の例は、グローバルクラスタに対するノード ID の割り当てを示しています。

```
phys-schost# clnode show | grep Node
=== Cluster Nodes ===
Node Name:          phys-schost1
Node ID:            1
Node Name:          phys-schost2
Node ID:            2
Node Name:          phys-schost3
Node ID:            3
```

## ▼ 新しいクラスタノード認証で作業する

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、ノードに加わるには、すべての必要な暗号化鍵を構成する必要があります。詳細は、[keyserv\(1M\)](#) と [publickey\(4\)](#) のマニュアルページを参照してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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(1CL)** ユーティリティーを起動します。  
phys-schost# **clsetup**  
メインメニューが表示されます。
- 3 クラスタ認証で作業するため、新規ノードのオプションに対応する番号を入力します。  
「新規ノード」メニューが表示されます。
- 4 メニューから選択を行って、画面の指示に従います。

### 例 9-3 新しいマシンがグローバルクラスタに追加されないようにする

clsetup ユーティリティーにより、claccess コマンドを生成します。次の例は、新しいマシンがクラスタに追加されないようにする claccess コマンドを示しています。

```
phys-schost# claccess deny -h hostname
```

### 例 9-4 すべての新しいマシンがグローバルクラスに追加されることを許可する

clsetup ユーティリティーにより、claccess コマンドを生成します。次の例は、すべての新しいマシンをクラスタに追加できるようにする claccess コマンドを示しています。

```
phys-schost# claccess allow-all
```

### 例 9-5 グローバルクラスタに追加される新しいマシンを指定する

clsetup ユーティリティーにより、claccess コマンドを生成します。次の例は、1台の新しいマシンをクラスタに追加できるようにする claccess コマンドを示しています。

```
phys-schost# claccess allow -h hostname
```

### 例 9-6 認証を標準 UNIX に設定する

clsetup ユーティリティにより、claccess コマンドを生成します。次の例は、クラスタに参加している新規ノードの標準 UNIX 認証に対し、リセットを行う claccess コマンドを示しています。

```
phys-schost# claccess set -p protocol=sys
```

### 例 9-7 認証を DES に設定する

clsetup ユーティリティにより、claccess コマンドを生成します。次の例は、クラスタに参加している新規ノードの DES 認証を使用する claccess コマンドを示しています。

```
phys-schost# claccess set -p protocol=des
```

DES 認証を使用する場合、クラスタにノードが加わるには、すべての必要な暗号化鍵を構成します。詳細は、[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

Oracle Solaris Cluster ソフトウェアは、Network Time Protocol (NTP) を使用し、クラスタノード間で時刻を同期させています。グローバルクラスタの時刻の調整は、ノードが時刻を同期するときに、必要に応じて自動的に行われます。詳細は、『[Oracle Solaris Cluster Concepts Guide](#)』および『[Network Time Protocol User's Guide](#)』を参照してください。



注意 - NTP を使用する場合、クラスタの稼動中はクラスタの時刻を調整しないでください。[date\(1\)](#)、[rdate\(1M\)](#)、[xntpd\(1M\)](#)、[svcadm\(1M\)](#) などのコマンドを、対話的に使用したり、[cron\(1M\)](#) スクリプト内で使用して時刻を調整しないでください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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起動します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
# shutdown -g -y -i0
```

```
Press any key to continue
```

- 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 メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86          |
| Solaris failsafe                |
|                                 |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

GRUB ベースの起動についての詳細は、『[Solaris のシステム管理 \(基本編\)](#)』の「[GRUB を使用して x86 システムをブートする \(作業マップ\)](#)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、**e** を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a)                  |
| kernel /platform/i86pc/multiboot |
| module /platform/i86pc/boot_archive |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.
```

- c. コマンドに **-x** を追加して、システムを非クラスタモードで起動するように指定します。

```
[ Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits. ]
```

```
grub edit> kernel /platform/i86pc/multiboot -x
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a)                               |
| kernel /platform/i86pc/multiboot -x         |
| module /platform/i86pc/boot_archive        |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-x` オプションを追加します。

---

- 5 単一のノードで、**date** コマンドを実行して時刻を設定します。  

```
phys-schost# date HHMM.SS
```
- 6 ほかのマシンで、**rdate(1M)** コマンドを実行し、時刻をこのノードに同期化します。  

```
phys-schost# rdate hostname
```
- 7 各ノードを起動し、クラスタを再起動します。  

```
phys-schost# reboot
```
- 8 すべてのクラスタノードで変更が行われたことを確認します。  
各ノードで、**date** コマンドを実行します。  

```
phys-schost# date
```

## ▼ SPARC: ノードで OpenBoot PROM (OBP) を表示する

OpenBoot™ PROM 設定を構成または変更する必要がある場合は、この手順を使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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内の指定された投票ノードまたは非投票ノードから、次に優先される投票ノードまたは非投票ノードに、すべてのリソースグループを切り替えます。

```
phys-schost# clnode evacuate node
# shutdown -g0 -y
```



注意 - クラスタノードを停止する場合は、クラスタコンソール上で `send brk` を使用してはいけません。

- 3 **OBP** コマンドを実行します。

## ▼ ノードのプライベートホスト名を変更する

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 IP アドレスを割り当てます。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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 **Network Time Protocol (NTP)** デーモンを停止します。

Network Time Protocol (NTP) デーモンを停止するには、svcadm コマンドを使用します。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(1CL)** ユーティリティーを実行して、適切なノードのプライベートホスト名を変更します。

クラスタ内の1つのノードでのみユーティリティーを実行します。

---

注-新しいプライベートホスト名を選択するときには、その名前がクラスタノード内で一意であることを確認してください。

---

- 4 プライベートホスト名用のオプションに対応する番号を入力します。
- 5 プライベートホスト名を変更するためのオプションに対応する番号を入力します。表示される質問に答えます。変更しようとしているプライベートホスト名のノード名 (clusternode<nodeid> -priv) および新しいプライベートホスト名を入力してください。

- 6 ネームサービスキャッシュをフラッシュします。  
 クラスタの各ノードで次の手順を実行します。フラッシュすることによって、クラスタアプリケーションとデータサービスが古いプライベートホスト名にアクセスしないようにします。  

```
phys-schost# nscd -i hosts
```
- 7 NTP 構成ファイルでプライベートホスト名を変更した場合、各ノード上で NTP 構成ファイル (`ntp.conf` または `ntp.conf.cluster`) を更新します。
  - a. 任意のエディタを使用してください。  
 この手順をインストール時に行う場合は、構成するノードの名前を削除する必要があります。デフォルトのテンプレートには 16 のノードが事前構成されています。通常 `ntp.conf.cluster` ファイルは各クラスタノード上で同じです。
  - b. すべてのクラスタノードから新しいプライベートホスト名に `ping` を実行できることを確認します。
  - c. NTP デーモンを再起動します。  
 クラスタの各ノードで次の手順を実行します。  
 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 ntp
```
- 8 手順 1 で無効にしたデータサービスリソースとそのほかのアプリケーションをすべて有効にします。  

```
phys-schost# clresource 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。

```
[Disable all applications and data services as necessary.]
phys-schost-1# /etc/init.d/xntpd stop
phys-schost-1# clnode show | grep node
...
private hostname:                clusternode1-priv
private hostname:                clusternode2-priv
private hostname:                clusternode3-priv
...
phys-schost-1# clsetup
phys-schost-1# nscd -i hosts
```



```
phys-schost-1# vi /etc/inet/ntp.conf
...
peer clusternode1-priv
peer clusternode4-priv
peer clusternode3-priv
phys-schost-1# ping clusternode4-priv
phys-schost-1# /etc/init.d/xntpd start
[Enable all applications and data services disabled at the beginning of the procedure.]
```

## ▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を追加する

インストール完了後、グローバルクラスタ上の非投票ノードのプライベートホスト名を追加するには、次の手順を使用します。この章の手順の `phys-schost#` は、グローバルクラスタプロンプトを表します。この手順は、グローバルクラスタ上のみで実行します。

- 1 `clsetup(1CL)` ユーティリティーを実行して、適切なゾーンにプライベートホスト名を追加します。

```
phys-schost# clsetup
```

- 2 プライベートホスト名のオプションに対応する番号を入力し、**Return** キーを押します。
- 3 ゾーンプライベートホスト名を追加するためのオプションに対応する番号を入力し、**Return** キーを押します。

表示される質問に答えます。グローバルクラスタの非投票ノードには、デフォルトのプライベートホスト名はありません。ホスト名を入力する必要があります。

## ▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を変更する

インストール完了後、非投票ノードのプライベートホスト名を変更するには、次の手順を使用します。

プライベートホスト名は、クラスタの初期インストール時に割り当てられます。プライベートホスト名の形式は、`clusternode<nodeid>-priv` です (`clusternode3-priv` など)。名前がすでにドメイン内で使用されている場合にかぎり、プライベートホスト名を変更します。



---

注意-新しいプライベートホスト名にはIPアドレスを割り当てないでください。クラスタソフトウェアがIPアドレスを割り当てます。

---

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

- 1 グローバルクラスタ内のすべてのノード上で、プライベートホスト名をキャッシュする可能性があるデータサービスリソースやアプリケーションをすべて無効にします。

```
phys-schost# clresource disable resource1, resource2
```

無効にするアプリケーションには次のようなものがあります。

- HA-DNS と HA-NFS サービス (構成している場合)
- プライベートホスト名を使用するためにカスタム構成されたアプリケーション
- クライアントがプライベートインターコネクト経由で使用しているアプリケーション

`clresource` コマンドの使用については、`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 `clsetup(1CL)` ユーティリティを実行して、グローバルクラスタ上の適切な非投票ノードのプライベートホスト名を変更します。

```
phys-schost# clsetup
```

この手順は、クラスタ内の1つのノードからのみ実行する必要があります。

---

注-新しいプライベートホスト名を選択するときには、その名前がクラスタ内で一意であることを確認してください。

---

- 3 プライベートホスト名のオプションに対応する番号を入力し、**Return** キーを押します。
- 4 ゾーンプライベートホスト名を追加するためのオプションに対応する番号を入力し、**Return** キーを押します。

グローバルクラスタのプライベートホスト名の非投票ノードには、デフォルトは存在しません。ホスト名を入力する必要があります。

- 5 ゾーンプライベートホスト名を変更するためのオプションに対応する番号を入力します。  
表示される質問に答えます。ここでは、プライベートホスト名を変更する非投票ノードの名前 (clusternode<nodeid> -priv) と新しいプライベートホスト名を入力してください。
- 6 ネームサービスキャッシュをフラッシュします。  
クラスタの各ノードで次の手順を実行します。フラッシュすることによって、クラスタアプリケーションとデータサービスが古いプライベートホスト名にアクセスしないようにします。  

```
phys-schost# nscd -i hosts
```
- 7 **手順1**で無効にしたデータサービスリソースとそのほかのアプリケーションをすべて有効にします。

## ▼ グローバルクラスタ上の非投票ノードのプライベートホスト名を削除する

グローバルクラスタ上の非投票ノードのプライベートホスト名を削除するには、次の手順を使用します。この手順は、グローバルクラスタ上のみで実行します。

- 1 **clsetup(1CL)**ユーティリティーを実行して、適切なゾーン上のプライベートホスト名を削除します。
- 2 ゾーンのプライベートホスト名用のオプションに対応する番号を入力します。
- 3 ゾーンプライベートホスト名を削除するためのオプションに対応する番号を入力します。
- 4 削除する非投票ノードのプライベートホスト名の名前を入力します。

## ▼ ノード名を変更する

Oracle Solaris Cluster 構成の一部であるノードの名前を変更できます。Oracle Solaris のホスト名は、ノード名の変更前に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 クラスタでノード名を変更する場合は、さらに実行する必要がある手順があります。名前変更作業を実行するクラスタが保護グループの主クラスタであるときに保護グループのアプリケーションをオンラインにする場合は、名前変更作業中に保護グループを二次クラスタに切り替えることができます。Geographic Edition クラスタおよびノードについては、『Oracle Solaris Cluster Geographic Edition System Administration Guide』の第5章「Administering Cluster Partnerships」を参照してください。
- 3 『Solaris のシステム管理 (上級編)』の「システムのホスト名の変更方法」の手順を完了して(ただし、手順の最後にある再起動は実行しない)、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起動します。  
ok> `boot -x`
- 5 Oracle Solaris のホスト名を変更したノード上で、非クラスタモードの状態での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。名前変更は、一度に1ノードずつ行います。  
# `clnode rename -n newnodename oldnodename`
- 6 クラスタ上で動作するアプリケーションで、以前のホスト名への既存の参照をすべて更新します。
- 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;/etc/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を含む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` プロパティを変更することもできます。このオプションの手順については、277 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

275 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。

```
# clr disable apache-lh-res
```

- c. 新しいホスト名リストを指定します。

```
# clr 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。

```
# clr 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 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。

```
# clr 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はずす場合は、そのノードを保守状態にします。保守状態のノードは、サービス対象中に定足数確立の投票に参加しません。クラスタノードを保守状態にするには、`clnode(1CL) evacuate` および `cluster(1CL) shutdown` コマンドを使用して、ノードを停止しておく必要があります。

---

注- ノードを1つだけ停止する場合は、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。

クラスタノードを保守状態にするには、`clquorum(1CL) disable` コマンドを使用します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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 すべてのリソースグループとデバイスグループをノードから退避します。`clnode evacuate` コマンドは、すべての非投票ノードを含むすべてのリソースグループとデバイスグループを、指定ノードから次に優先されるノードに切り替えます。

```
phys-schost# clnode evacuate node
```

- 3 退避させたノードを停止します。

```
phys-schost# shutdown -g0 -y-i 0
```

- 4 クラスタ内の別のノード上で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になり、手順3で停止したノードを保守状態にします。

```
phys-schost# clquorum disable node
```

node 保守モードにするノードの名前を指定します。

- 5 グローバルクラスタノードが保守状態にあることを確認します。

```
phys-schost# clquorum status node
```

保守状態にしたノードの Status は offline で、その Present と Possible の定足数投票数は 0 (ゼロ) である必要があります。

### 例 9-9 グローバルクラスタノードを保守状態にする

次に、クラスタノードを保守状態にして、その結果を確認する例を示します。cnode 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# cnode 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、279 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、投票数も増えているはずですが、

```
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。GRUB メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                     |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

```
phys-schost-1# clquorum status
```

```
--- Quorum Votes Summary ---
```

| Needed | Present | Possible |
|--------|---------|----------|
| 4      | 6       | 6        |

```
--- Quorum Votes by Node ---
```

| Node Name     | Present | Possible | Status |
|---------------|---------|----------|--------|
| phys-schost-2 | 1       | 1        | Online |
| phys-schost-3 | 1       | 1        | Online |

```
--- Quorum Votes by Device ---
```

| Device Name          | Present | Possible | Status |
|----------------------|---------|----------|--------|
| /dev/did/rdisk/d3s2  | 1       | 1        | Online |
| /dev/did/rdisk/d17s2 | 0       | 1        | Online |
| /dev/did/rdisk/d31s2 | 1       | 1        | Online |

## ▼ 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

完全に確立されたクラスタ構成からグローバルクラスタノードを切り離す前に、そのノードから Oracle Solaris Cluster ソフトウェアをアンインストールするには、この手順を実行します。この手順では、クラスタに存在する最後のノードからソフトウェアをアンインストールできます。

---

注-クラスタにまだ結合されていない、あるいはまだインストールモードであるノードから 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「Oracle Solaris Cluster ソフトウェアをアンインストールしてインストール問題を解決する」に進みます。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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 クラスタノードを削除するための前提作業(作業マップを参照)がすべて完了していることを確認します。

[表 8-2](#) を参照してください。

---

注-この手順を続ける前に、`clnode remove` を使用してクラスタ構成からノードを削除します。

---

- 2 アンインストールするグローバルクラスタノード以外の、アクティブなグローバルクラスタメンバー上で、スーパーユーザーになります。この手順は、グローバルクラスタノードから実行します。
- 3 アクティブなクラスタメンバーから、アンインストールを行うノードをクラスタのノード認証リストに追加します。

```
phys-schost# claccess allow -h hostname
```

-h ノードの認証リストに追加するノードの名前を指定します。

または、`clsetup(1CL)` ユーティリティーを使用できます。手順については、[246 ページ](#)の「ノードを認証ノードリストに追加する」を参照してください。

- 4 アンインストールするノードでスーパーユーザーになります。

- 5 ゾーンクラスタがある場合は、アンインストールします。

```
phys-schost# clzonecluster uninstall -F zoneclustername
```

具体的な手順は、295 ページの「ゾーンクラスタを削除する」を参照してください。

- 6 ノードにグローバルデバイス名前空間用の専用パーティションがある場合、グローバルクラスタノードを非クラスタモードで再起動します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
# shutdown -g0 -y -i0ok boot -x
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
# shutdown -g0 -y -i0
...
<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@7,1/
sd@0,0:a
Boot args:

Type   b [file-name] [boot-flags] <ENTER> to boot with options
or     i <ENTER>                       to enter boot interpreter
or     <ENTER>                          to boot with defaults

<<< timeout in 5 seconds >>>
Select (b)oot or (i)nterpreter: b -x
```

- 7 `/etc/vfstab` ファイルから、グローバルにマウントされるすべてのファイルシステムエントリを削除します。ただし、`/global/.devices` グローバルマウントを除きます。

- 8 このノード上で **Oracle Solaris Cluster** ソフトウェアを再インストールする場合は、**Sun Java Enterprise System (Java ES)** 製品のレジストリから **Oracle Solaris Cluster** のエントリを削除します。

Java ES 製品のレジストリに Oracle Solaris Cluster ソフトウェアがインストールされたという記録が含まれていると、Java ES のインストーラは Oracle Solaris Cluster のコンポーネントを灰色で表示し、再インストールを許可しません。

- a. **Java ES** のアンインストーラを起動します。

次のコマンドを実行します。`ver` は Oracle Solaris Cluster ソフトウェアのインストール元である Java ES ディストリビューションのバージョンです。

```
# /var/sadm/prod/SUNWentsysver/uninstall
```

- b. プロンプトに従い、アンインストールする **Oracle Solaris Cluster** を選択します。

`uninstall` コマンドの使用については、『**Sun Java Enterprise System 5 Update 1 Installation Guide for UNIX**』の第 8 章「Uninstalling」を参照してください。

- 9 このクラスタ上で **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. 切り離し手順については、ホストアダプタやサーバーに添付されているマニュアルを参照してください。

---

ヒント-ループバックファイルインターフェース (lofi) デバイスを使用する場合、Java ES アンインストーラが自動的に `/globaldevices` という lofi ファイルを削除します。グローバルデバイス名前空間の lofi への移行に関する詳細は、[131 ページの「グローバルデバイス名前空間を移行する」](#) を参照してください。

---

## ノードのアンインストールに伴う問題の解決

ここでは、`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/dg1 is still configured as a global mount.

clnode: It is not safe to uninstall with these outstanding errors.
clnode: Refer to the documentation for complete uninstall instructions.
clnode: Uninstall failed.
```

このエラーを修正するためには、[282 ページの「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」](#)に戻って、その手順を繰り返す必要があります。手順 7 コマンドを再度実行する前に、この Step 7 が正しく行われているか確認してください。

## デバイスグループに削除されていないリストがある場合

次のエラーメッセージは、削除したノードが依然としてデバイスグループにリストされていることを示しています。

```
Verifying that no device services still reference this node ... failed
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.
```

## 存在しないアンインストールスクリプト

削除する Sun Cluster または Oracle Solaris Cluster ソフトウェアをインストールまたはアップグレードする際に `installer` プログラムを使用しなかった場合、そのソフトウェアバージョンで使用するアンインストールスクリプトは存在しません。代わりに、次の手順を実行してソフトウェアをアンインストールします。

### ▼ アンインストールスクリプトを使用しないで **Sun Cluster 3.1** および **3.2** ソフトウェアをアンインストールする

- 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 **Sun Cluster** パッケージに関連付けられていないディレクトリに移動します。

```
# cd /directory
```
- 3 ノードから **Sun Cluster** ソフトウェアをアンインストールします。

```
# scinstall -r
```
- 4 `productregistry` ファイルの名前を変更し、今後ソフトウェアの再インストールができるようにします。

```
# mv /var/sadm/install/productregistry /var/sadm/install/productregistry.sav
```

## Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理

この節では、Simple Network Management Protocol (簡易ネットワーク管理プロトコル、SNMP) イベント Management Information Base (管理情報ベース、MIB) を作成、設定、および管理する方法を説明します。またこのセクションでは、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 `clsnmpost` 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

| 作業                                   | 参照先  |
|--------------------------------------|--|
| SNMP イベント MIB の有効化                   | 287 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                   | 287 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                    | 288 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 | 288 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                          | 289 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                         | 290 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                         | 291 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# clnmpmib set -n node -p version=value MIB
```

`-n node`

変更するイベント MIB がある `node` を指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

`-p version=value`

MIB で使用する SNMP プロトコルのバージョンを指定します。 `value` は次のように指定します。

- `version=SNMPv2`
- `version=snmpv2`
- `version=2`
- `version=SNMPv3`
- `version=snmpv3`
- `version=3`

*MIB*

サブコマンドが適用される単数または複数の MIB の名前を指定します。この場合、`event` を指定してください。このオペランドを指定しない場合は、デフォルトのプラス記号 (+) がサブコマンドで使用されます (すべての MIB を意味します)。 *MIB* オペランドを使用する場合は、ほかのコマンドラインオプションの後ろに、スペース区切りリストで MIB を指定します。

## ▼ SNMP ホストがノード上の SNMP トラップを受信できるようにする

この手順では、ノード上の SNMP ホストを、MIB のトラップ通知を受信するホストのリストに追加する方法を説明します。

`phys-schost#` プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。



この手順では、長形式の 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# clsnmphost add -c SNMPcommunity [-n node] host
```

*-c SNMPcommunity*

ホスト名とともに使用される SNMP コミュニティ名を指定します。

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# clnmiphost remove -c SNMPcommunity -n node host
```

remove

指定のノードから指定の SNMP ホストを削除します。

-c *SNMPcommunity*

SNMP ホストを削除する SNMP コミュニティの名前を指定します。

-n *node*

構成から削除される SNMP ホストの *node* の名前を指定します。ノード名またはノード ID を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

*host*

構成から削除されるホストの名前、IP アドレス、または IPv6 アドレスを指定します。

指定の SNMP コミュニティ内のすべてのホストを削除するには、-c オプション付きの *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# clnmpuser create -n node -a authentication \  
-f password user
```

-n *node*

SNMP ユーザーが追加されるノードを指定します。ノード ID またはノード名を指定できます。このオプションを指定しないと、デフォルトで現在のノードが使用されます。

-a *authentication*

ユーザーの承認に使用する認証プロトコルを指定します。認証プロトコルの値は SHA または MD5 です。

`-f password` SNMP ユーザーパスワードを含むファイルを指定します。新しいユーザーを作成する際にこのオプションを指定しないと、コマンドはパスワードを求めるプロンプトを表示します。このオプションは、`add` サブコマンドとだけ有効です。

ユーザーパスワードは、次の形式で、独立した行の上に指定します。

`user:password`

パスワードには次に示す文字または空白文字を含めることはできません。

- ;(セミコロン)
- :(コロン)
- \ (バックスラッシュ)
- \n (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。
- 強い制限値 - 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\(1CL\)](#) のマニュアルページを参照してください。

高い優先度を持つようにリソースグループを設定すると、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す。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` フラグを設定することもできます。

```
# 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ーを指定できます。

強い正または負のアフィニティーは負荷分散より優先されます。強いアフィニティーや強い負荷制限が無効になることはありません。強いアフィニティー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ーを指定しています。

```
# 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、ノードまたはその非大域ゾーンで定義された負荷制限設定がすべて含まれます。

## ゾーンクラスタ管理タスクの実行

ゾーンクラスタに関するほかの管理作業(ゾーンパスの移動、アプリケーションを実行するためのゾーンクラスタの準備、ゾーンクラスタの複製など)を実行できます。これらのコマンドは、必ずグローバルクラスタの投票ノードから実行してください。

---

注-グローバルクラスタ内の投票ノードからのみ実行する 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 そのほかのゾーンクラスタ作業

| 作業                     | 参照先  |
|------------------------|--|
| 新規ゾーンパスへのゾーンパスの移動      | <code>clzonecluster move -f zonepath zoneclustername</code>  |
| アプリケーション実行用のゾーンクラスタの準備 | <code>clzonecluster ready -n nodename zoneclustername</code>   |
| ゾーンクラスタの複製             | <code>clzonecluster clone -Z source- zoneclustername [-m copymethod] zoneclustername</code><br><br>clone サブコマンドを使用する前に、複製元のゾーンクラスタを停止してください。複製先のゾーンクラスタは、構成済みであることが必要です。 |
| ゾーンクラスタの削除             | 295 ページの「ゾーンクラスタを削除する」   |
| ゾーンクラスタからファイルシステムを削除   | 296 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」   |

表 9-3 そのほかのゾーンクラスタ作業 (続き)

| 作業  | 参照先   |
|---|---|
| ゾーンクラスタからストレージデバイスを削除                             | 298 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ト                          | 284 ページの「ノードのアンインストールに伴う問題の解決」                              |
| Oracle Solaris Cluster SNMP イベント MIB の作成、設定、および管理 | 286 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
- VxFS ローカルファイルシステム
- QFS スタンドアロンファイルシステム
- QFS 共有ファイルシステム (Oracle RAC のサポートに使用する場合のみ)
- (データセットとしてエクスポートされた) ZFS
- 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
- VxFS ローカルファイルシステム
- QFS スタンドアロンファイルシステム
- QFS 共有ファイルシステム (Oracle RAC のサポートに使用する場合のみ)
- UFS クラスタファイルシステム
- VxFS 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。グローバルクラスタのノードから、このプロシージャの一部のステップが実行されます。他のステップは、ゾーンクラスタのノードから実行されます。
- 2 削除するファイルシステムに関連するリソースを削除します。
  - a. 削除するゾーンクラスタのファイルシステム用に構成されている **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 **SUNW.qfs** タイプの **Oracle Solaris Cluster** リソースがあれば、そのリソースを特定し、削除します。

```
phys-schost# clresource delete -F fs_global_resources
```

-F オプションを指定すると、前もって無効にしているリソースも含め、指定したリソースがすべて強制的に削除されるため、このオプションは注意して使用してください。すべての指定リソースが、ほかのリソースのリソース関係設定から削除されるため、クラスタ内のサービスが失われることがあります。削除されていない依存リソースは、無効な状態やエラー状態になる可能性があります。詳細は、**clresource(1CL)** のマニュアルページを参照してください。

---

ヒント-削除したリソースのリソースグループがあとで空になると、そのリソースグループを安全に削除できます。

---

- 3 ファイルシステムのマウントポイントディレクトリのパスを調べます。たとえば、次のように使用します。

```
phys-schost# clzonecluster configure zoneclustername
```

- 4 ファイルシステムをゾーンクラスタの構成から削除します。

```
phys-schost# clzonecluster configure zoneclustername
```

```
clzc:zoneclustername> remove fs dir=filesystemdirectory
```

```
clzc:zoneclustername> commit
```

ファイルシステムのマウントポイントは、**dir=** で指定します。

- 5 ファイルシステムが削除されたことを確認します。

```
phys-schost# clzonecluster show -v zoneclustername
```

### 例 9-12 ゾーンクラスタ内の高可用性ファイルシステムの削除

この例は、**sczone** というゾーンクラスタ内に構成された、マウントポイントディレクトリ (**/local/ufs-1**) のあるファイルシステムを削除する方法を示しています。リソースは **hasp-rs** で、そのタイプは **HAStoragePlus** です。

```
phys-schost# clzonecluster show -v sczone
```

```
...
Resource Name:                fs
  dir:                        /local/ufs-1
  special:                     /dev/md/ds1/dsk/d0
  raw:                         /dev/md/ds1/rdisk/d0
  type:                         ufs
  options:                     [logging]
...
```

```
phys-schost# clresource delete -F -Z sczone hasp-rs
```

```

phys-schost# clzonecluster configure sczone
clzc:sczone> remove fs dir=/local/ufs-1
clzc:sczone> commit
phys-schost# clzonecluster show -v sczone

```

### 例 9-13 ゾーンクラスタ内の高可用性 ZFS ファイルシステムの削除

この例は、HAzpool という ZFS プール内の ZFS ファイルシステムを削除する方法を示しています。このファイルシステムは、sczone ゾーンクラスタ内に構成されており、リソースは hasp-rs で、そのタイプは SUNW.HAStoragePlus です。

```

phys-schost# clzonecluster show -v sczone
...
Resource Name:                dataset
name:                          HAzpool
...
phys-schost# clresource delete -F -Z sczone hasp-rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove dataset name=HAzpool
clzc:sczone> commit
phys-schost# clzonecluster show -v sczone

```

## ▼ ゾーンクラスタからストレージデバイスを削除する

ストレージデバイス (SVM ディスクセット、DID デバイスなど) をゾーンクラスタから削除することが可能です。この手順は、ゾーンクラスタからストレージデバイスを削除する場合に実行します。

- 1 ゾーンクラスタをホストするグローバルクラスタのノードで、スーパーユーザーになります。グローバルクラスタのノードから、このプロシージャの一部のステップが実行されます。ほかのステップは、ゾーンクラスタのノードから実行することが可能です。
- 2 削除するデバイスに関連するリソースを削除します。削除するゾーンクラスタのデバイス用に構成されている **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 SVM ディスクセットを削除する方法を示しています。apachedg ディスクセットのセット番号は 3 です。このデバイスは、クラスタに構成された zc\_rs のリソースにより使用されます。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name:      device
  match:             /dev/md/apachedg/*dsk/*
Resource Name:      device
  match:             /dev/md/shared/3/*dsk/*
...
phys-schost# clresource delete -F -Z sczone zc_rs

phys-schost# ls -l /dev/md/apachedg
lrwxrwxrwx 1 root root 8 Jul 22 23:11 /dev/md/apachedg -> shared/3
phys-schost# clzonecluster configure sczone
clzc:sczone> remove device match=/dev/md/apachedg/*dsk/*
clzc:sczone> remove device match=/dev/md/shared/3/*dsk/*
clzc:sczone> commit
clzc:sczone> end
phys-schost# clzonecluster reboot sczone
phys-schost# clzonecluster show -v sczone
```

#### 例 9-15 DID デバイスをゾーンクラスタから削除する

この例は、DID デバイス d10 および d11 を削除する方法を示しています。このデバイスは、sczone というゾーンクラスタに構成されています。このデバイスは、クラスタに構成された zc\_rs のリソースにより使用されます。

```
phys-schost# clzonecluster show -v sczone
...
Resource Name:      device
  match:             /dev/did/*dsk/d10*
Resource Name:      device
  match:             /dev/did/*dsk/d11*
...
```

```
phys-schost# clresource delete -F -Z sczone zc_rs
phys-schost# clzonecluster configure sczone
clzc:sczone> remove device match=/dev/did/*dsk/d10*
clzc:sczone> remove device match=/dev/did/*dsk/d11*
clzc:sczone> commit
clzc:sczone> end
phys-schost# clzonecluster reboot sczone
phys-schost# clzonecluster show -v sczone
```

## 障害追跡

この節では、テスト用に使用できる障害追跡手順について説明します。

### 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習得するには、Oracle サポートから提供されている診断ツールキット (SUNWscdtk) パッケージ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起動するグローバルクラスタノードを退避します。

```
phys-schost# clresourcegroup evacuate -n targetnode
```

- 3 **HASStorage** または **HASStoragePlus** リソースを含み、あとで非クラスタモードにするメタセットの影響を受けるデバイスまたはファイルシステムを含む、1つまたは複数のリソースグループをオフラインにします。

```
phys-schost# clresourcegroup offline resourcegroupname
```

- 4 オフラインにしたリソースグループ内のすべてのリソースを無効にします。

```
phys-schost# clresource disable resourcename
```

- 5 リソースグループを非管理状態に切り替えます。

```
phys-schost# clresourcegroup unmanage resourcegroupname
```

- 6 対応する1つまたは複数のデバイスグループをオフラインにします。

```
phys-schost# cldevicegroup offline devicegroupname
```

- 7 1つまたは複数のデバイスグループを無効にします。

```
phys-schost# cldevicegroup disable devicegroupname
```

- 8 パッシブノードを非クラスタモードで起動します。

```
phys-schost# reboot -x
```

- 9 続ける前にパッシブノードで起動プロセスが完了していることを確認します。

```
phys-schost# svcs -x
```

- 10 メタセット内のディスクに **SCSI3** 予約があるかどうかを調べます。メタセット内のすべてのディスクで次のコマンドを実行します。

```
phys-schost# /usr/cluster/lib/sc/scsi -c inkeys -d /dev/did/rdisk/dids2
```

- 11 ディスク上に **SCSI3** 予約がある場合は、それらをスクラブします。

```
phys-schost# /usr/cluster/lib/sc/scsi -c scrub -d /dev/did/rdisk/dids2
```

- 12 メタセットを退避したノードに移します。

```
phys-schost# metaset -s name -C take -f
```

- 13 メタセット上で定義されたデバイスを含む1つまたは複数のファイルシステムをマウントします。

```
phys-schost# mount device mountpoint
```

- 14 アプリケーションを起動して、必要なテストを実行します。テストが終了したら、アプリケーションを停止します。

- 15 ノードを再起動し、起動プロセスが終了するまで待ちます。

```
phys-schost# reboot
```

- 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
```

- 1つまたは複数のリソースグループを起動します。

```
phys-schost# clresourcegroup online -eM resourcegroupname
```

## 破損したディスクセットの復元

ディスクセットが破損した場合や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ある場合は、次の手順に従います。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。303 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起動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 サポートから提供されている診断ツールキット (SUNWscdtk) パッケージ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。詳細は、[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 (SUNWscdtk) パッケージを使用して、**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パッチ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なっています。

---

- 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エ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再び追加します。

```
/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。たとえば、次のように使用します。

```
# /usr/sbin/fmthard -s /etc/lvm/d4.vtoc /dev/global/rdisk/d4s2
```

```
# /usr/sbin/fmthard -s /etc/lvm/d8.vtoc /dev/global/rdisk/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/rdisk/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。

- 307 ページの「CPU 制御の概要」
- 309 ページの「CPU 制御の構成」

## CPU 制御の概要

Oracle Solaris Cluster ソフトウェアを使用すると、CPU の使用率を制御できます。

CPU 制御機能は、Oracle Solaris OS で利用可能な機能に基づいて構築されています。ゾーン、プロジェクト、リソースプール、プロセッサセット、およびスケジューリングクラスについては、『[Oracle Solaris のシステム管理 \(Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン\)](#)』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   | 参照先  |
|--|--|
| <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とゾーンに割り当て、<code>project.cpu-shares</code> および <code>zone.cpu-shares</code> の値を指定します。</p> <p>この手順は、グローバルクラスタの非投票ノードが構成されているかどうかに関係なく実行できます。</p>   | <p>309 ページの「グローバルクラスタの投票ノードで CPU 使用率を制御する」</p>                     |
| <p>リソースグループはデフォルトのプロセッサセットを使用することによりグローバルクラスタの非投票ノードで動作する。</p> <p>CPU シェアをリソースグループとゾーンに割り当て、<code>project.cpu-shares</code> および <code>zone.cpu-shares</code> の値を指定します。</p> <p>この手順は、プロセッサセットのサイズを制御する必要がない場合に実行します。</p>  | <p>311 ページの「デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する」</p> |
| <p>リソースグループは専用のプロセッサセットを使用してグローバルクラスタの非投票ノードで動作する。</p> <p>CPU シェアをリソースグループに割り当て、<code>project.cpu-shares</code>、<code>zone.cpu-shares</code> の値、および専用のプロセッサセット内のプロセッサの最大数を提供する</p> <p>専用のプロセッサセット内のプロセッサセットの最小数を設定します。</p> <p>CPU シェアと、プロセッサセットのサイズを制御したい場合に、この手順を実行します。この制御は、専用のプロセッサセットを使用することにより、グローバルクラスタの非投票ノードでのみ実行できます。</p> | <p>314 ページの「専用のプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する」</p>    |

## 公平配分スケジューラ

CPU シェアをリソースグループに割り当てる手順の最初のステップは、システムのスケジューラを公平配分スケジューラ (FSS) に設定することです。デフォルトでは、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。

- 309 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」
- 311 ページの「デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードのCPU使用率を制御する」
- 314 ページの「専用のプロセッサセットを使用してグローバルクラスタの非投票ノードのCPU使用率を制御する」

### ▼ グローバルクラスタの投票ノードでCPU使用率を制御する

グローバルクラスタの投票ノードで実行されるリソースグループにCPUシェアを割り当てるには、この手順を実行します。

リソースグループにCPUシェアが割り当てられている場合、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、次の作業を実行します。

- 投票ノードに割り当てられているCPUシェア (`zone.cpu-shares`) の数を、指定されたCPUシェアの数だけ増やします (まだ行われていない場合)。
- 投票ノードに `SCSLM_resourcegroup_name` という名前のプロジェクトを作成します (まだ行われていない場合)。このプロジェクトはリソースグループに固有で、指定された数のCPUシェア (`project.cpu-shares`) が割り当てられています。
- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

CPU制御機能の構成についての詳細は、[rg\\_properties\(5\)](#)のマニュアルページを参照してください。

- 1 システムのデフォルトのスケジューラを、公平配分スケジューラ (FSS) に設定します。

```
# dispadmin -d FSS
```

次の再起動時に、FSS がデフォルトのスケジューラになります。この構成をすぐに有効にするには、`priocntl` コマンドを使用します。

```
# priocntl -s -C FSS
```

`prionctl` コマンドと `dispadmin` コマンドを組み合わせることで、FSS がすぐにデフォルトのスケジューラになり、再起動後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [prionctl\(1\)](#) のマニュアルページを参照してください。

注 - FSS がデフォルトのスケジューラでない場合、CPU シェアの割り当ては有効になりません。

- 2 各ノードで CPU 制御を使用するため、グローバルクラスタの投票ノードに対するシェア数と、デフォルトのプロセッサセットで使用可能な CPU の最小数を構成します。

これらのパラメータを設定することで、投票ノードで動作中のプロセスを、非投票ノードで動作中のプロセスと CPU を獲得する際に発生する競合から保護します。 `globalzonestshares` および `defaultpsetmin` プロパティに値を割り当てないと、これらのプロパティはデフォルト値をとります。

```
# clnode set [-p globalzonestshares=integer] \
[-p defaultpsetmin=integer] \
node
```

`-p defaultpsetmin=defaultpsetmininteger` デフォルトのプロセッサセットで利用可能な CPU 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。これらのプロパティを設定しないと、非投票ノードで `RG_SLM_PSET_TYPE` プロパティの恩恵を受けることができません。

- 3 これらのプロパティを正しく設定したことを確認します。

```
# clnode show node
```

指定するノードに対して、`clnode` コマンドは、設定されているプロパティ、およびこれらのプロパティに設定されている値を出力します。`clnode` を使用して CPU 制御プロパティを設定しないと、これらはデフォルト値をとります。

- 4 CPU 制御機能を構成します。

```
# cresourcegroup 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 を設定する手順の一部を自動化します。

`-p RG_SLM_CPU_SHARES=value` リソースグループ固有のプロジェクトに割り当てられる CPU シェアの数 (`project.cpu-shares`) を指定し、投票ノードに割り当てられる CPU シェアの数 (`zone.cpu-shares`) を決定します。

`resource_group_name` リソースグループの名前を指定します。

この手順では、`RG_SLM_PSET_TYPE` プロパティは設定しません。投票ノードでは、このプロパティは値 `default` をとります。

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

## 5 構成の変更を有効にします。

```
# clresourcegroup online -M resource_group_name
```

`resource_group_name` リソースグループの名前を指定します。

---

注-SCSLM `resource_group_name` プロジェクトは削除または変更しないでください。手動で、たとえば `project.max-lwps` プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

## ▼ デフォルトのプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する

グローバルクラスタの非投票ノードのリソースグループに対して CPU シェアを割り当てるが、専用のプロセッサセットを作成する必要がない場合は、この手順を実行します。

リソースグループに CPU シェアが割り当てられている場合、Oracle Solaris Cluster ソフトウェアは、非投票ノードのリソースグループの 1 つのリソースを起動する際に、次の作業を実行します。

- `SCSLM_resource_group_name` という名前のプールを作成します (まだ行われていない場合)。
- `SCSLM_pool_zone_name` プールを、デフォルトのプロセッサセットに関連付けます。
- 非投票ノードを `SCSLM_pool_zone_name` プールに動的にバインドします。
- 非投票ノードに割り当てられている CPU シェア (`zone.cpu-shares`) の数を、指定された CPU シェアの数だけ増やします (まだ行われていない場合)。

- 非投票ノードに `SCSLM_resourcegroup_name` という名前のプロジェクトを作成します (まだ行われていない場合)。このプロジェクトはリソースグループに固有で、指定された数の CPU シェア (`project.cpu_shares`) が割り当てられます。
- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

CPU 制御機能の構成についての詳細は、[rg\\_properties\(5\)](#) のマニュアルページを参照してください。

- 1 システムのデフォルトのスケジューラを、公平配分スケジューラ (**FSS**) に設定します。

```
# dispadmin -d FSS
```

次の再起動時に、FSS がデフォルトのスケジューラになります。この構成をすぐに有効にするには、`priocntl` コマンドを使用します。

```
# priocntl -s -C FSS
```

`priocntl` コマンドと `dispadmin` コマンドを組み合わせることで、FSS がすぐにデフォルトのスケジューラになり、再起動後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [priocntl\(1\)](#) のマニュアルページを参照してください。

---

注-FSS がデフォルトのスケジューラでない場合、CPU シェアの割り当ては有効になりません。

---

- 2 各ノードで CPU 制御を使用するため、グローバルクラスタの投票ノードに対するシェア数と、デフォルトのプロセッサセットで使用可能な CPU の最小数を構成します。

これらのパラメータを設定することで、投票ノードで動作中のプロセスを、グローバルクラスタの非投票ノードで動作中のプロセスと CPU を獲得する際に発生する競合から保護します。`globalzonestshares` および `defaultpsetmin` プロパティに値を割り当てないと、これらのプロパティはデフォルト値をとります。

```
# clnode set [-p globalzonestshares=integer] \  
[-p defaultpsetmin=integer] \  
node
```

```
-p globalzonestshares=integer
```

投票ノードに割り当てられるシェアの数を設定します。デフォルト値は 1 です。

```
-p defaultpsetmin=defaultpsetmininteger
```

デフォルトのプロセッサセットで利用可能な CPU の最小数を設定します。デフォルト値は 1 です。

```
node
```

プロパティを設定するノードを指定します。



これらのプロパティを設定する際には、投票ノードのプロパティを設定していません。

- 3 これらのプロパティを正しく設定したことを確認します。

```
# clnode show node
```

指定するノードに対して、`clnode` コマンドは、設定されているプロパティ、およびこれらのプロパティに設定されている値を出力します。`clnode` を使用して CPU 制御プロパティを設定しないと、これらはデフォルト値をとります。

- 4 CPU 制御機能を構成します。

```
# clresourcegroup create -p RG_SLM_TYPE=automated \
  [-p RG_SLM_CPU_SHARES=value] resource_group_name
```

|   |  |
|---|--|
| <code>-p RG_SLM_TYPE=automated</code>   | 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 を設定する手順の一部を自動化します。  |
| <code>-p RG_SLM_CPU_SHARES=value</code> | リソースグループ固有のプロジェクトに割り当てられる CPU シェアの数 ( <code>project.cpu-shares</code> ) を指定し、グローバルクラスタの非投票ノードに割り当てられる CPU シェアの数 ( <code>zone.cpu_shares</code> ) を決定します。 |
| <code>resource_group_name</code>        | リソースグループの名前を指定します。   |

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

デフォルト以外のプールがゾーン構成内にある場合、またはゾーンがデフォルト以外のプールに動的にバインドされている場合は、非投票ノードで `RG_SLM_TYPE` を `automated` に設定できません。ゾーン構成とプールのバインディングについては、それぞれ [zonecfg\(1M\)](#) と [poolbind\(1M\)](#) のマニュアルページを参照してください。ゾーン構成を次のように表示します。

```
# zonecfg -z zone_name info pool
```

---

注 -HASStoragePlus、LogicalHostname などのリソースは、非投票ノードで起動するように構成されていても、`GLOBAL_ZONE` プロパティが `TRUE` に設定されている場合は、投票ノードで起動されます。`RG_SLM_TYPE` プロパティを `automated` に設定していても、このリソースは CPU シェア構成の恩恵を受けることはなく、`RG_SLM_TYPE` が手動に設定されているリソースグループと同様に扱われます。

---

この手順では、`RG_SLM_PSET_TYPE` プロパティは設定しません。Oracle Solaris Cluster はデフォルトのプロセッサセットを使用します。

## 5 構成の変更を有効にします。

```
# clresourcegroup online -M resource_group_name
```

`resource_group_name` リソースグループの名前を指定します。

RG\_SLM\_PSET\_TYPE に default を設定すると、Oracle Solaris Cluster はプール SCSLM\_pool\_zone\_name を作成しますが、プロセッサセットは作成しません。この場合、SCSLM\_pool\_zone\_name はデフォルトのプロセッサセットに関連付けられます。

オンラインリソースグループが、非投票ノード内の CPU 制御に対して設定されなくなった場合、非投票ノードの CPU シェア値はゾーン構成内の zone.cpu-shares の値をとります。このパラメータの値はデフォルトで 1 です。ゾーン構成の詳細は、[zonecfg\(1M\)](#) のマニュアルページを参照してください。

---

注 - SCSLM\_resource\_group\_name プロジェクトは削除または変更しないでください。手動で、たとえば project.max-lwps プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

## ▼ 専用のプロセッサセットを使用してグローバルクラスタの非投票ノードの CPU 使用率を制御する

リソースグループを専用のプロセッサセットで実行する場合は、この手順を実行します。

リソースグループが専用のプロセッサセットで実行するよう構成されている場合、Oracle Solaris Cluster ソフトウェアは、グローバルクラスタ非投票ノードでリソースグループのリソースを起動する際に、次の作業を実行します。

- SCSLM\_pool\_zone\_name という名前のプールを作成します (まだ行われていない場合)。
- 専用のプロセッサセットを作成します。プロセッサセットのサイズは、RG\_SLM\_CPU\_SHARES および RG\_SLM\_PSET\_MIN プロパティを使用することで決定されます。
- SCSLM\_pool\_zone\_name プールを、作成されたプロセッサセットに関連付けます。
- 非投票ノードを SCSLM\_pool\_zone\_name プールに動的にバインドします。
- 非投票ノードに割り当てられている CPU シェアの数、指定された CPU シェアの数だけ増やします (まだ行われていない場合)。
- 非投票ノードに SCSLM\_resource\_group\_name という名前のプロジェクトを作成します (まだ行われていない場合)。このプロジェクトはリソースグループに固有で、指定された数の CPU シェア (project.cpu\_shares) が割り当てられます。

- `SCSLM_resourcegroup_name` プロジェクトのリソースを起動する。

## 1 システムのスケジューラを、公平配分スケジューラ (FSS) に設定します。

```
# dispadmin -d FSS
```

次の再起動時に、FSS がデフォルトのスケジューラになります。この構成をすぐに有効にするには、`priocntl` コマンドを使用します。

```
# priocntl -s -C FSS
```

`priocntl` コマンドと `dispadmin` コマンドを組み合わせることで、FSS がすぐにデフォルトのスケジューラになり、再起動後もそのままになります。スケジューリングクラスの設定についての詳細は、[dispadmin\(1M\)](#) および [priocntl\(1\)](#) のマニュアルページを参照してください。

---

注 - FSS がデフォルトのスケジューラでない場合、CPU シェアの割り当ては有効になりません。

---

## 2 各ノードで CPU 制御を使用するため、グローバルクラスタの投票ノードに対するシェア数と、デフォルトのプロセッサセットで使用可能な CPU の最小数を構成します。

これらのパラメータを設定することで、投票ノードで動作中のプロセスを、非投票ノードで動作中のプロセスと CPU を獲得する際に発生する競合から保護します。`globalzonestshares` および `defaultpsetmin` プロパティに値を割り当てないと、これらのプロパティはデフォルト値をとります。

```
# clnode set [-p globalzonestshares=integer] \
[-p defaultpsetmin=integer] \
node
```

`-p defaultpsetmin=defaultpsetmininteger` デフォルトのプロセッサセットで利用可能な CPU の最小数を設定します。デフォルトは 1 です。

`-p globalzonestshares=integer` 投票ノードに割り当てられるシェアの数を設定します。デフォルトは 1 です。

`node` プロパティを設定するノードを指定します。

これらのプロパティを設定する際には、投票ノードのプロパティを設定しています。

## 3 これらのプロパティを正しく設定したことを確認します。

```
# clnode show node
```

指定するノードに対して、`clnode` コマンドは、設定されているプロパティー、およびこれらのプロパティーに設定されている値を出力します。`clnode` を使用して CPU 制御プロパティーを設定しないと、これらはデフォルト値をとります。

#### 4 CPU制御機能を構成します。

```
# clresourcegroup create -p RG_SLM_TYPE=automated \  
  [-p RG_SLM_CPU_SHARES=value] \  
-p -y RG_SLM_PSET_TYPE=value \  
[-p RG_SLM_PSET_MIN=value] resource_group_name
```

|   |   |
|---|---|
| <code>-p RG_SLM_TYPE=automated</code>   | CPU の使用状況を管理できるようにし、システム資源管理用に Oracle Solaris OS を設定する手順の一部を自動化します。   |
| <code>-p RG_SLM_CPU_SHARES=value</code> | リソースグループ固有のプロジェクトに割り当てられる CPU シェアの数 ( <code>project.cpu-shares</code> ) を指定し、非投票ノードに割り当てられる CPU シェアの数 ( <code>zone.cpu-shares</code> ) とプロセッサセット内のプロセッサの最大数を決定します。  |
| <code>-p RG_SLM_PSET_TYPE=value</code>  | 専用のプロセッサセットの作成を可能にします。専用のプロセッサセットを使用するには、このプロパティーを <code>strong</code> または <code>weak</code> に設定します。値 <code>strong</code> と <code>weak</code> は相互に排他的です。つまり、同じゾーン内のリソースグループを、 <code>strong</code> と <code>weak</code> が混在するように構成することはできません。 |
| <code>-p RG_SLM_PSET_MIN=value</code>   | プロセッサセット内のプロセッサの最小数を判別します。  |
| <code>resource_group_name</code>        | リソースグループの名前を指定します。  |

このステップによりリソースグループが作成されます。また、`clresourcegroup set` コマンドを使用して既存のリソースグループを変更することもできます。

デフォルト以外のプールがゾーン構成内にある場合、またはゾーンがデフォルト以外のプールに動的にバインドされている場合は、非投票ノードで `RG_SLM_TYPE` を `automated` に設定できません。ゾーン構成とプールのバインディングについては、それぞれ [zoncfg\(1M\)](#) と [poolbind\(1M\)](#) のマニュアルページを参照してください。ゾーン構成を次のように表示します。

```
# zoncfg -z zone_name info pool
```

---

注-HAStoragePlus、LogicalHostnameなどのリソースは、非投票ノードで起動するように構成されていても、GLOBAL\_ZONE プロパティがTRUEに設定されている場合は、投票ノードで起動されます。RG\_SLM\_TYPE プロパティを automated に設定していても、このリソースはCPUシェアと専用のプロセッサセット構成の恩恵を受けることはなく、RG\_SLM\_TYPE が手動に設定されているリソースグループと同様に扱われます。

---

## 5 構成の変更を有効にします。

*resource\_group\_name* リソースグループの名前を指定します。

---

注-SCSLM *resource\_group\_name* プロジェクトは削除または変更しないでください。手動で、たとえば `project.max-lwps` プロパティを構成することにより、プロジェクトにさらにリソース制御を追加できます。詳細は、[projmod\(1M\)](#) のマニュアルページを参照してください。

---

リソースグループがオンラインの間に RG\_SLM\_CPU\_SHARES と RG\_SLM\_PSET\_MIN に行われた変更は、動的に考慮されます。しかし、RG\_SLM\_PSET\_TYPE に strong が設定されている場合、および、変更を受け入れるためのCPUが十分に存在しない場合、RG\_SLM\_PSET\_MIN に要求された変更は適用されません。この場合は、警告メッセージが表示されます。次のスイッチオーバーでは、RG\_SLM\_PSET\_MIN に対して構成した値を受け入れる十分なCPUが使用できない場合、不十分なCPU数によるエラーが発生する可能性があります。

オンラインリソースグループが、非投票ノード内のCPU制御に対して設定されなくなった場合、非投票ノードのCPUシェア値は `zone.cpu-shares` の値をとります。このパラメータの値はデフォルトで1です。



# Oracle Solaris Cluster ソフトウェアと ファームウェアのパッチ

---

この章では、Oracle Solaris Cluster 構成のパッチの追加および削除手順について説明します。この章で説明する手順は次のとおりです。

- 319 ページの「Oracle Solaris Cluster へのパッチの適用の概要」
- 321 ページの「Oracle Solaris Cluster ソフトウェアへのパッチの適用」

## Oracle Solaris Cluster へのパッチの適用の概要

クラスタの性質上、クラスタを正しく動作させるには、すべてのクラスタメンバーノードが同じパッチレベルにある必要があります。Oracle Solaris Cluster パッチをノードに適用するときは、パッチをインストールする前に、クラスタメンバーシップからノードを一時的に削除するか、全体のクラスタを停止しておく必要があります。この節では、これらの手順について説明します。

Oracle Solaris Cluster パッチを適用する前に、パッチの README ファイルを確認します。また、ストレージデバイスのアップグレード要件を確認して、必要なパッチ方法を判別します。

---

注 - Oracle Solaris Cluster パッチを適用する場合は、この章の説明よりも新しい注意事項がないかどうか、README ファイルと SunSolve を参照してください。

---

すべてのクラスタノードにパッチをインストールする作業は、次のいずれかの状況に該当します。

再起動パッチ(ノード)

パッチを提供するには、`boot -sx` または `shutdown -g -y -i0` コマンドを使用して、ノードをシングルユーザーモードで起動してから、クラスタに結合するために再起動します。まず、任意のリソースグループまたはデバイスグループを、パッチを適用するノードから別のクラスタメンバーに切り替え、ノードをオフライン状

態にする必要があります。また、クラスタ全体が停止しないように、パッチまたはファームウェアは1つのクラスタノードに適用します。

たとえ個々のノードが一時的に使用不可能になっても、クラスタ自体はこの種類のパッチアプリケーションの間は使用可能のままです。パッチを適用したノードは、他のノードが同じパッチレベルになくても、メンバーノードとしてクラスタに結合できます。

#### 再起動パッチ(クラスタ)

ソフトウェアまたはファームウェアのパッチを適用するには、`boot -sx` または `shutdown -g -y -i0` コマンドを使用して、クラスタを停止して各ノードをシングルユーザーモードで起動する必要があります。次に、ノードを再起動してクラスタに結合します。このようなパッチでは、パッチ適用中にクラスタを使用できません。

#### 非再起動パッチ

ノードをオフライン状態にする必要はありません(引き続きリソースグループやデバイスグループのマスターとして動作可能)。また、パッチの適用時にノードをオフライン状態に、または再起動する必要もありません。ただし、パッチは一度に1つのノードに適用し、次のノードに適用する前に、パッチが動作することを確認する必要があります。

---

注-パッチの適用によって配下のクラスタプロトコルが変更されることはありません。

---

パッチをクラスタに適用するには `patchadd` コマンドを、パッチを削除するには(可能な場合) `patchrm` コマンドをそれぞれ使用します。

## Oracle Solaris Cluster パッチの適用に関する注意事項

Oracle Solaris Cluster パッチをより効率的に適用するために、次の点に注意してください。

- パッチを適用する前に、必ずパッチの README ファイルを参照してください。
- ストレージデバイスのアップグレード要件を確認して、必要なパッチ方法を判別します。



- クラスタを実際の環境で実行する前に、すべてのパッチ (必須および推奨) を適用します。
- ハードウェアのファームウェアレベルを確認し、必要と思われる必須ファームウェアアップデートをインストールします。
- クラスタメンバーとして機能するノードには、すべて同じパッチを適用する必要があります。
- クラスタサブシステムパッチの状態を最新の状態に保ちます。これらのパッチには、たとえば、ボリューム管理、ストレージデバイスのファームウェア、クラスタトランスポートなどが含まれます。
- 定期的に (四半期に一度など) パッチレポートを確認し、推奨パッチを Oracle Solaris Cluster 構成に適用します。
- ご購入先が推奨するパッチを適用します。
- 主要なパッチを更新したならフェイルオーバーをテストします。クラスタの動作が低下または悪化した場合に備えて、パッチを取り消す準備をしておきます。

## Oracle Solaris Cluster ソフトウェアへのパッチの適用

表 11-1 作業リスト:クラスタへのパッチの適用

| 作業   | 参照先   |
|--|---|
| ノードを停止せずに、非再起動 Oracle Solaris Cluster パッチを一度に 1 つのノードだけに適用 | 329 ページの「非再起動 Oracle Solaris Cluster パッチを適用する」              |
| クラスタメンバーを非クラスタモードにした後で、再起動 Oracle Solaris Cluster パッチを適用   | 321 ページの「再起動パッチを適用する (ノード)」<br>326 ページの「再起動パッチを適用する (クラスタ)」 |
| フェイルオーバーゾーン構成をしている場合に、シングルユーザーモードでノードにパッチを適用               | 331 ページの「フェイルオーバーゾーン構成をしている場合に、シングルユーザーモードでノードにパッチを適用する」    |
| Oracle Solaris Cluster パッチを削除                              | 335 ページの「Oracle Solaris Cluster パッチの変更」                     |

### ▼ 再起動パッチを適用する (ノード)

パッチプロセス中にクラスタ自体の稼働状態を維持するには、クラスタ内の各ノードに個別にパッチを適用します。この手順で、パッチを適用する前に、`boot -sx` または `shutdown -g -y -i0` コマンドを使用して、まずクラスタのノードを停止してシングルユーザーモードで起動する必要があります。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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 パッチを適用する前に、**Oracle Solaris Cluster** 製品の Web サイトで、インストール前後の特別な注意事項があるかどうかを確認してください。
- 2 パッチの適用先であるノード上で、スーパーユーザーになるか、**RBAC** の承認 **solaris.cluster.admin** を提供する役割になります。
- 3 パッチを適用するノードのリソースグループおよびデバイスグループの一覧を表示します。

```
# clresourcegroup status -Z all -n node[...]
```

*node* パッチを適用するグローバルクラスタノードの名前、またはパッチを適用するノード上に存在するゾーンクラスタノードの名前。

```
# cldevicegroup status -n node
```

*node* パッチを適用するグローバルクラスタノードの名前。

---

注-デバイスグループは、ゾーンクラスタに関連付けられていません。

---

- 4 すべてのリソースグループ、リソース、デバイスグループを、パッチを適用するノードから他のクラスタメンバーに切り替えます。

```
# clnode evacuate -n node
```

*evacuate* グローバルクラスタのすべての非投票ノードも含め、すべてのデバイスグループとリソースグループを退避させます。

*-n node* リソースグループとデバイスグループを切り替えるノードを指定します。

- 5 ノードを停止します。

```
# shutdown -g0 [-y]
[-i0]
```

- 6 ノードを非クラスタのシングルユーザーモードで起動します。
  - SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -sx
```
  - x86 ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# shutdown -g -y -i0
```

```
Press any key to continue
```

- a. GRUB メニューで矢印キーを使用して該当する 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 メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                       |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

GRUB ベースの起動についての詳細は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、**e** を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a)                                       |
| kernel /platform/i86pc/multiboot                   |
| module /platform/i86pc/boot_archive                 |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.
```

- c. コマンドに **-sx** を追加して、システムを非クラスタモードで起動することを指定します。

```
[ Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits. ]
```

```
grub edit> kernel /platform/i86pc/multiboot -sx
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a)                                       |
| kernel /platform/i86pc/multiboot -sx                 |
+-----+
```

```
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-sx` オプションを追加します。

---

- 7 ソフトウェアパッチまたはファームウェアパッチを適用します。

```
# patchadd -M patch-dir patch-id
```

*patch-dir*            パッチのディレクトリの場所を指定します。

*patch-id*            特定のパッチのパッチ番号を指定します。

---

注-パッチディレクトリに、この章の手順よりも新しい注意事項がないかどうかを必ず確認してください。

---

- 8 パッチが正常にインストールされていることを確認します。

```
# showrev -p | grep patch-id
```

- 9 ノードを再起動してクラスタに結合します。

```
# reboot
```

- 10 パッチが機能しており、ノードとクラスタが正常に動作していることを確認します。

- 11 残りのすべてのクラスタノードで、**手順 2** から **手順 10** を繰り返します。

- 12 必要に応じて、リソースグループとデバイスグループを切り替えます。

すべてのノードを再起動した後、最後に再起動したノードのリソースグループとデバイスグループはオンラインになりません。

```
# cldevicegroup switch -n node + | devicegroup ...
```

```
# clresourcegroup switch -n node[:zone][,...] + | resource-group ...
```

*node*    リソースグループとデバイスグループの切り替え先のノードの名前。

**zone** リソースグループをマスターできるグローバルクラスタ非投票ノード (node) の名前。リソースグループを作成した際に非投票ノードを指定した場合にのみ、zone を指定します。

```
# clresourcegroup switch -Z zoneclustername -n zcnode[...] +|resource-group ...
```

**zoneclustername** リソースグループの切り替え先となるゾーンクラスタの名前。

**zcnode** リソースグループをマスターできるゾーンクラスタノードの名前。

---

注- デバイスグループは、ゾーンクラスタに関連付けられていません。

---

- 13 scversions** コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。

```
# /usr/cluster/bin/scversions
```

結果として、次に示すメッセージのどちらか一方が表示されます。

```
Upgrade commit is needed.
```

```
Upgrade commit is NOT needed. All versions match.
```

- 14** コミットが必要な場合は、パッチソフトウェアをコミットします。

```
# scversions -c
```

---

注- scversions を実行すると、状況に応じて1つ以上のCMM再構成が発生します。

---

### 例 11-1 再起動パッチの適用 (ノード)

次に、ノードに Oracle Solaris Cluster 再起動パッチを適用する例を示します。

```
# clresourcegroup status -n rg1
...Resource Group      Resource
-----
rg1                      rs-2
rg1                      rs-3
...
# cldevicegroup status -n nodedg-schost-1
...
Device Group Name:                dg-schost-1
...
# clnode evacuate phys-schost-2
# shutdown -g0 -y -i0
...
```

ノードを非クラスタのシングルユーザーモードで起動します。

- SPARC:次のように入力します。

```
ok boot -sx
```

- x86:ノードを非クラスタのシングルユーザーモードで起動します。続きの手順で起動ステップを確認します。

```
# patchadd -M /var/tmp/patches 234567-05
...
# showrev -p | grep 234567-05

...
# reboot
...
# cldevicegroup switch -n phys-schost-1 dg-schost-1
# clresourcegroup switch -n phys-schost-1 schost-sa-1
# scversions
Upgrade commit is needed.
# scversions -c
```

参照 パッチを取り消す必要がある場合は、335 ページの「Oracle Solaris Cluster パッチの変更」を参照してください。

## ▼ 再起動パッチを適用する(クラスタ)

この手順で、パッチを適用する前に、`boot -sx` または `shtudown -g -y -i0` コマンドを使用して、まずクラスタを停止して各ノードをシングルユーザーモードで起動する必要があります。

- 1 パッチを適用する前に、**Oracle Solaris Cluster** 製品の Web サイトで、インストール前後の特別な注意事項があるかどうかを確認してください。

- 2 クラスタ内にある任意のノード上でスーパーユーザーになります。

- 3 クラスタを停止します。

```
# cluster shutdown -y -g grace-period "message"
```

-y 確認プロンプトで *yes* と答えます。

-g *grace-period* 停止までの待機時間を秒単位で指定します。デフォルトの猶予期間は 60 秒です。

*message* 送信する警告メッセージを指定します。*message* が複数の単語の場合は、引用符で囲みます。

- 4 各ノードを非クラスタのシングルユーザーモードで起動します。  
各ノードのコンソールで、次のコマンドを実行します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -sx
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# shutdown -g -y -i0
```

```
Press any key to continue
```

- a. GRUB メニューで矢印キーを使用して該当する 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 メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                     |
|                                                       |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

GRUB ベースの起動についての詳細は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、**e** を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
```

```
+-----+
| root (hd0,0,a)                                       |
| kernel /platform/i86pc/multiboot                    |
| module /platform/i86pc/boot_archive                 |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.
```

- c. コマンドに **-sx** を追加して、システムを非クラスタモードで起動することを指定します。

```
[ Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits. ]
```

```
grub edit> kernel /platform/i86pc/multiboot -sx
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。

画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a)                                     |
| kernel /platform/i86pc/multiboot -sx              |
| module /platform/i86pc/boot_archive               |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b** と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-sx` オプションを追加します。

---

- 5 ソフトウェアパッチまたはファームウェアパッチを適用します。

一度に1つのノードずつ、次のコマンドを実行します。

```
# patchadd -M patch-dir patch-id
patch-dir      パッチのディレクトリの場所を指定します。
patch-id       特定のパッチのパッチ番号を指定します。
```

---

注-パッチディレクトリに、この章の手順よりも新しい注意事項がないかどうかを必ず確認してください。

---

- 6 パッチが各ノードに正常にインストールされていることを確認します。

```
# showrev -p | grep patch-id
```

- 7 パッチをすべてのノードに適用したなら、ノードを再起動してクラスタに結合します。

各ノードで次のコマンドを実行します。

```
# reboot
```

- 8 `scversions` コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。

```
# /usr/cluster/bin/scversions
```



結果として、次に示すメッセージのどちらか一方が表示されます。

```
Upgrade commit is needed.
```

```
Upgrade commit is NOT needed. All versions match.
```

- 9 コミットが必要な場合は、パッチソフトウェアをコミットします。

```
# scversions -c
```

---

注 - scversions を実行すると、状況に応じて1つ以上のCMM再構成が発生します。

---

- 10 パッチが機能しており、ノードとクラスタが正常に動作していることを確認します。

### 例 11-2 再起動パッチの適用 (クラスタ)

次に、クラスタに Oracle Solaris Cluster 再起動パッチを適用する例を示します。

```
# cluster shutdown -g0 -y
...
```

クラスタを非クラスタのシングルユーザーモードで起動します。

- SPARC: 次のように入力します。

```
ok boot -sx
```

- x86: 各ノードを非クラスタのシングルユーザーモードで起動します。ステップの続きの手順を確認します。

```
...
# patchadd -M /var/tmp/patches 234567-05
(Apply patch to other cluster nodes)
...
# showrev -p | grep 234567-05
# reboot
# scversions
Upgrade commit is needed.
# scversions -c
```

参照 パッチを取り消す必要がある場合は、[335 ページ](#)の「Oracle Solaris Cluster パッチの変更」を参照してください。

## ▼ 非再起動 Oracle Solaris Cluster パッチを適用する

パッチを一度にクラスタの1つのノードだけに適用します。非再起動パッチを適用するときは、パッチを適用するノードを停止する必要はありません。

- 1 パッチを適用する前に、**Oracle Solaris Cluster** 製品の **Web** ページで、インストール前後の特別な注意事項があるかどうかを確認してください。

- 2 ひとつのノードにパッチを適用します。

```
# patchadd -M patch-dir patch-id
```

*patch-dir*           パッチのディレクトリの場所を指定します。

*patch-id*           特定のパッチのパッチ番号を指定します。

- 3 パッチが正常にインストールされていることを確認します。

```
# showrev -p | grep patch-id
```

- 4 パッチが機能しており、ノードとクラスタが正常に動作していることを確認します。

- 5 残りのクラスタノードで、**手順2**から**手順4**を繰り返します。

- 6 **scversions** コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。

```
# /usr/cluster/bin/scversions
```

結果として、次に示すメッセージのどちらか一方が表示されます。

```
Upgrade commit is needed.
```

```
Upgrade commit is NOT needed. All versions match.
```

- 7 コミットが必要な場合は、パッチソフトウェアをコミットします。

```
# scversions -c
```

---

注 - **scversions** を実行すると、状況に応じて1つ以上のCMM再構成が発生します。

---

### 例 11-3 非再起動 Oracle Solaris Cluster パッチの適用

```
# patchadd -M /tmp/patches 234567-05
```

```
...
```

```
# showrev -p | grep 234567-05
```

```
# scversions
```

```
Upgrade commit is needed.
```

```
# scversions -c
```

参照   パッチを取り消す必要がある場合は、[335 ページ](#)の「[Oracle Solaris Cluster パッチの変更](#)」を参照してください。

## ▼ フェイルオーバーゾーン構成をしている場合に、シングルユーザーモードでノードにパッチを適用する

フェイルオーバーゾーン構成をしている場合に、シングルユーザーモードでノードにパッチを適用するには、次の作業を実行します。このパッチ方法は、Oracle Solaris Cluster ソフトウェアによるフェイルオーバー構成で Solaris コンテナ用 Oracle Solaris Cluster データサービスを使用する場合に必要です。

- 1 この手順で、手動で取得されるゾーンパスを含むディスクセットの一部となる共有ストレージとして使用される LUN のいずれかに、定足数デバイスが構成されていないことを確認します。
  - a. 定足数デバイスがゾーンパスを含むディスクセットで使用されているかどうかを確認し、さらに定足数デバイスが SCSI2 または SCSI3 予約を使用するかどうかを確認します。
 

```
# clquorum show
```
  - b. 定足数デバイスがディスクセットの LUN 内にある場合は、ゾーンパスを含むディスクセットの一部ではない定足数デバイスとして、新しい LUN を追加します。
 

```
# clquorum add new-didname
```
  - c. 古い定足数デバイスを削除します。
 

```
# clquorum remove old-didname
```
  - d. 古い定足数デバイスで SCSI2 予約が使用されている場合は、古い定足数デバイスから SCSI2 予約をスクラブして、SCSI2 予約が残っていないことを確認します。  
 pgre コマンドの実行方法を習得するには、Oracle サポートから提供されている診断ツールキット (SUNWscdtk) パッケージをインストールして使用する必要があります。

---

注-有効な定足数デバイスで誤って予約キーをスクラブした場合は、定足数デバイスを削除してからもう一度追加して、その定足数デバイスに新しい予約キーを付与します。

---

- 2 パッチを適用するノードを退避させます。

```
# clresourcegroup evacuate -n node1
```

- 3 HA Solaris コンテナリソースを含む1つまたは複数のリソースグループをオフラインにします。

```
# clresourcegroup offline resourcegroupname
```

- 4 オフラインにしたリソースグループ内のすべてのリソースを無効にします。

```
# clresource disable resourcename
```

- 5 オフラインにしたリソースグループをアンマネージします。

```
# clresourcegroup unmanage resourcegroupname
```

- 6 対応する1つまたは複数のデバイスグループをオフラインにします。

```
# cldevicegroup offline cldevicegroupname
```

---

注-ゾーンパスの zpool があるフェイルオーバーゾーンにパッチを適用する場合は、このステップおよび手順7をスキップしてください。

---

- 7 オフラインにしたデバイスグループを無効にします。

```
# cldevicegroup disable devicegroupname
```

- 8 クラスタ外のパッシブノードを起動します。

```
# reboot -- -x
```

---

注-ゾーンパスの zpool があるフェイルオーバーゾーンにパッチを適用する場合は、次のコマンドを使用してください。

---

```
# reboot -- -xs
```

- 9 続ける前に、パッシブノードで SMF 起動方法が完了していることを確認します。

```
# svcs -x
```

---

注-ゾーンパスの zpool があるフェイルオーバーゾーンにパッチを適用する場合は、このステップをスキップしてください。

---

- 10 アクティブノード上で再構成プロセスが完了していることを確認します。

```
# cluster status
```

- 11 ディスクセット内のディスクに SCSI-2 予約があるかどうかを調べて、キーを解放します。次の手順に従って、SCSI-2 予約が存在するかどうかを確認し、SCSI-2 予約を解放します。

- ディスクセット内のすべてのディスクについて、コマンド  
/usr/cluster/lib/sc/scsi -c disfailfast -d /dev/did/rdisk/d#s2 を実行します。

- キーがリストされている場合は、コマンド `/usr/cluster/lib/sc/scsi -c release -d /dev/did/rdisk/d#s2` を実行してキーを解放します。

予約キーの解放が終了したら、手順 12 をスキップして、手順 13 に進みます。

- 12 ディスクセット内のディスクに SCSI-3 予約があるかどうかを調べます。
  - a. ディスクセット内のすべてのディスクで次のコマンドを実行します。
 

```
# /usr/cluster/lib/sc/scsi -c inkeys -d /dev/did/rdisk/didnames2
```
  - b. キーがリストされている場合は、それらをスクラブします。
 

```
# /usr/cluster/lib/sc/scsi -c scrub -d /dev/did/rdisk/didnames2
```

- 13 パッシブノード上のメタセットの所有権を得ます。

```
# metaset -s disksetname -C take -f
```

---

注-ゾーンパスの `zpool` があるフェイルオーバーゾーンにパッチを適用する場合は、次のコマンドを使用してください。

```
# zpool import -R / pool_name
```

---

- 14 ゾーンパスを含む 1 つまたは複数のファイルシステムをパッシブノードにマウントします。
 

```
# mount device mountpoint
```

---

注-ゾーンパスの `zpool` があるフェイルオーバーゾーンにパッチを適用する場合は、このステップおよび手順 15 をスキップしてください。

---

- 15 パッシブノードでシングルユーザーモードに切り替えます。

```
# init s
```

- 16 Solaris コンテナ用の Oracle Solaris Cluster データサービスの制御下でない、起動されているゾーンをすべて停止します。

```
# zoneadm -z zonename halt
```

- 17 (省略可能) 複数のパッチをインストールする場合は、パフォーマンス上の理由から、すべての構成済みゾーンをシングルユーザーモードで起動することを選択できます。

```
# zoneadm -z zonename boot -s
```

- 18 パッチを適用します。

- 19 ノードを再起動して、**SMF** 起動方法が終了するまで待ちます。ノードが再起動したあとではじめて **svcs -a** コマンドを実行します。  

```
# reboot
```

```
# svcs -a
```

最初のノードの準備ができました。
- 20 パッチを適用する 2 番目のノードを退避させます。  

```
# clresourcegroup evacuate -n node2
```
- 21 2 番目のノードで手順 8～13 を繰り返します。
- 22 パッチを適用するゾーンを切り離します。パッチ適用済みゾーンを切り離しておかないと、パッチプロセスは失敗します。  

```
# zoneadm -z zonename detach
```
- 23 パッシブノードでシングルユーザーモードに切り替えます。  

```
# init s
```
- 24 **Solaris** コンテナ用の **Oracle Solaris Cluster** データサービスの制御下でない、起動されているゾーンをすべて停止します。  

```
# zoneadm -z zonename halt
```
- 25 (省略可能) 複数のパッチをインストールする場合は、パフォーマンス上の理由から、すべての構成済みゾーンをシングルユーザーモードで起動することを選択できます。  

```
# zoneadm -z zonename boot -s
```
- 26 パッチを適用します。
- 27 切り離れたゾーンを接続します。  

```
# zoneadm -z zonename attach -F
```
- 28 ノードを再起動して、クラスタモードにします。  

```
# reboot
```
- 29 1 つまたは複数のデバイスグループをオンラインにします。
- 30 リソースグループを起動します。
- 31 **scversions** コマンドを使用し、パッチソフトウェアをコミットする必要があるかどうかを確認してください。  

```
# /usr/cluster/bin/scversions
```

結果として、次に示すメッセージのどちらか一方が表示されます。

```
Upgrade commit is needed.
```

```
Upgrade commit is NOT needed. All versions match.
```

- 32** コミットが必要な場合は、パッチソフトウェアをコミットします。

```
# scversions -c
```

---

注 - `scversions` を実行すると、状況に応じて1つ以上のCMM再構成が発生します。

---

## Oracle Solaris Cluster パッチの変更

クラスタに適用した Oracle Solaris Cluster パッチを削除するには、まず新しい Oracle Solaris Cluster パッチを削除して、次に以前のパッチまたは更新リリースを再適用します。適用した Oracle Solaris Cluster パッチを削除するには、次の手順を参照してください。以前の Oracle Solaris Cluster パッチをもう一度適用するには、次の手順のいずれかを参照してください。

- [321 ページの「再起動パッチを適用する\(ノード\)」](#)
- [326 ページの「再起動パッチを適用する\(クラスタ\)」](#)
- [329 ページの「非再起動 Oracle Solaris Cluster パッチを適用する」](#)

---

注 - Oracle Solaris Cluster パッチを適用する前に、パッチの README ファイルを確認します。

---

### ▼ 非再起動 Oracle Solaris Cluster パッチを削除する

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 非再起動パッチを削除します。

```
# patchrm patchid
```

### ▼ 再起動 Oracle Solaris Cluster パッチを削除する

- 1 クラスタ内にある任意のノード上でスーパーユーザーになります。
- 2 クラスタノードを非クラスタモードで起動します。ノードを非クラスタモードで起動する方法については、[84 ページの「非クラスタモードでノードを起動する」](#)を参照してください。

- 3 再起動パッチを削除します。  
# `patchrm patchid`
- 4 クラスタノードを再起動して、クラスタモードに戻します。  
# `reboot`
- 5 各クラスタノードで手順2～4を繰り返します。



# ◆◆◆ 第 12 章

## クラスタのバックアップと復元

---

この章は次の節から構成されています。

- 337 ページの「クラスタのバックアップ」
- 349 ページの「クラスタファイルの復元の作業マップ」

### クラスタのバックアップ

表 12-1 作業リスト: クラスタファイルのバックアップ

| 作業                                 | 参照先  |
|------------------------------------|--|
| バックアップするファイルシステムの名前の検索             | 338 ページの「バックアップするファイルシステム名を確認する」   |
| フルバックアップを作成するのに必要なテープ数の計算          | 338 ページの「完全なバックアップに必要なテープ数を決定する」   |
| ルートファイルシステムのバックアップの作成              | 339 ページの「ルート(/) ファイルシステムをバックアップする」   |
| ミラーまたはブ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 | 342 ページの「ミラーのオンラインバックアップを実行する (Solaris Volume Manager)」<br>345 ページの「ボリュームのオンラインバックアップを実行する (Veritas Volume Manager)」 |
| クラスタ構成のバックアップ                      | 349 ページの「クラスタ構成をバックアップする」  |
| ストレージディスクのディスクパーティション分割構成のバックアップ   | ストレージディスクのマニュアルを参照   |

---

## ▼ バックアップするファイルシステム名を確認する

この手順を使用し、バックアップするファイルシステムの名前を判定します。

- 1 **/etc/vfstab** ファイルの内容を表示します。  
このコマンドを実行するためにスーパーユーザーまたは同等の役割である必要はありません。

```
# more /etc/vfstab
```

- 2 バックアップするファイルシステムの名前のマウントポイントの列を調べます。  
この名前は、ファイルシステムをバックアップするときに使用します。

```
# more /etc/vfstab
```

### 例 12-1 バックアップするファイルシステム名の確認

次に、**/etc/vfstab** ファイルに記述されている使用可能なファイルシステム名の例を示します。

```
# more /etc/vfstab
#device          device          mount FS fsck  mount  mount
#to mount        to fsck         point type  pass   at boot options
#
#/dev/dsk/cl1d0s2 /dev/rdisk/cl1d0s2 /usr   ufs   1     yes   -
f              -              /dev/fd fd    -     no    -
/proc          -              /proc  proc  -     no    -
/dev/dsk/cl1t6d0s1 -              -      swap  -     no    -
/dev/dsk/cl1t6d0s0 /dev/rdisk/cl1t6d0s0 /      ufs   1     no    -
/dev/dsk/cl1t6d0s3 /dev/rdisk/cl1t6d0s3 /cache ufs   2     yes   -
swap          -              /tmp   tmpfs  -     yes   -
```

## ▼ 完全なバックアップに必要なテープ数を決定する

この手順を使用し、ファイルシステムのバックアップに必要なテープ数を計算します。

- 1 バックアップするクラスタノード上で、スーパーユーザーまたは同等の役割になります。
- 2 バックアップのサイズをバイト単位で予測します。

```
# ufsdump S filesystem
```

S                   バックアップの実行に必要な予測バイト数を表示します。

filesystem         バックアップするファイルシステムの名前を指定します。

- 3 予測サイズをテープの容量で割り、必要なテープの数を確認します。

## 例 12-2 必要なテープ数の判別

次の例では、ファイルシステムのサイズは 905,881,620 バイトなので、4G バイトのテープに収めることができます (905,881,620 ÷ 4,000,000,000)。

```
# ufsdump S /global/phys-schost-1
905881620
```

## ▼ ルート(/)ファイルシステムをバックアップする

この手順を使用し、クラスタノードのルート (/) ファイルシステムをバックアップします。バックアップ手順を実行する前に、クラスタが正常に動作していることを確認してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify` を提供する役割になります。
- 2 実行中の各データサービスを、バックアップを作成するノードからクラスタ内の別のノードに切り替えます。

```
# clnode evacuate node
```

node リソースグループとデバイスグループを切り替えるノードを指定します。

- 3 ノードを停止します。

```
# shutdown -g0 -y -i0
```

- 4 ノードを非クラスタモードで再起動します。

- SPARC ベースのシステム上で、次のコマンドを実行します。

```
ok boot -xs
```

- x86 ベースのシステム上で、次のコマンドを実行します。

```
phys-schost# shutdown -g -y -i0
```

Press any key to continue

- a. GRUB メニューで矢印キーを使用して該当する 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 メニューは次のようになっています。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
| Solaris 10 /sol_10_x86                               |
| Solaris failsafe                                    |
|                                                     |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

GRUB ベースの起動についての詳細は、『Solaris のシステム管理 (基本編)』の「GRUB を使用して x86 システムをブートする (作業マップ)」を参照してください。

- b. ブートパラメータの画面で、矢印キーを使用してカーネルエントリを選択し、**e** を入力してエントリを編集します。

GRUB ブートパラメータの画面は、次のような画面です。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a)                                       |
| kernel /platform/i86pc/multiboot                    |
| module /platform/i86pc/boot_archive                 |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.
```

- c. コマンドに **-x** を追加して、システムが非クラスタモードで起動するように指定します。

```
[ Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits. ]
```

```
grub edit> kernel /platform/i86pc/multiboot -x
```

- d. **Enter** キーを押して変更を受け入れ、ブートパラメータの画面に戻ります。画面には編集されたコマンドが表示されます。

```
GNU GRUB version 0.95 (615K lower / 2095552K upper memory)
+-----+
| root (hd0,0,a)                                       |
| kernel /platform/i86pc/multiboot -x                 |
+-----+
```

```
| module /platform/i86pc/boot_archive |
+-----+
Use the ^ and v keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.-
```

- e. **b**と入力して、ノードを非クラスタモードで起動します。

---

注-カーネル起動パラメータコマンドへのこの変更は、システムを起動すると無効になります。次にノードを再起動する際には、ノードはクラスタモードで起動します。クラスタモードではなく、非クラスタモードで起動するには、これらの手順を再度実行して、カーネル起動パラメータコマンドに `-x` オプションを追加します。

---

- 5 **UFS** スナップショットを作成して、ルート (`/`) ファイルシステムのバックアップを作成します。
- a. ファイルシステムに、バックイングストアファイル用の十分なディスク容量が存在することを確認してください。
- ```
# df -k
```
- b. 同じ場所に同じ名前のバックイングストアファイルが存在しないことを確認します。
- ```
# ls /backing-store-file
```
- c. **UFS** スナップショットを作成します。
- ```
# fssnap -F ufs -o bs=/backing-store-file /file-system
```
- d. スナップショットが作成されたことを確認します。
- ```
# /usr/lib/fs/ufs/fssnap -i /file-system
```
- 6 ファイルシステムのスナップショットのバックアップを作成します。
- ```
# ufsdump 0ucf /dev/rmt/0 snapshot-name
```
- たとえば、次のように使用します。
- ```
# ufsdump 0ucf /dev/rmt/0 /dev/rfssnap/1
```
- 7 スナップショットのバックアップが作成されていることを確認します。
- ```
# ufsrestore ta /dev/rmt/0
```
- 8 ノードをクラスタモードで再起動します。
- ```
# init 6
```

### 例 12-3 ルート (/) ファイルシステムのバックアップ

次の例では、ルート (/) ファイルシステムのスナップショットは /usr ディレクトリ内の /scratch/usr.back.file に保存されています。

```
# fssnap -F ufs -o bs=/scratch/usr.back.file /usr
/dev/fssnap/1
```

## ▼ ミラーのオンラインバックアップを実行する (Solaris Volume Manager)

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ため、ミラー化の状態ではなくなります。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(1M)** コマンドを使用し、バックアップするボリュームの所有権を持つノードを判別します。

```
# metaset -s setname
-s setname      ディスクセット名を指定します。
```

- 3 **-w** オプションを指定して **lockfs(1M)** コマンドを使用し、ファイルシステムへの書き込みをロックします。

```
# lockfs -w mountpoint
```

---

注-ファイルシステムをロックする必要があるのは、UFS ファイルシステムがミラー上にある場合だけです。たとえば、Solaris Volume Manager ボリュームがデータベース管理ソフトやその他の特定のアプリケーションに使用する raw デバイスとして設定されている場合、lockfs コマンドを使用する必要はありません。ただし、ソフトウェアアプリケーション固有の適切なユーティリティーを実行し、任意のバッファをフラッシュしてアクセスをロックしてもかまいません。

---

- 4 **metastat(1M)** コマンドを使用し、サブミラーの名前を判別します。

```
# metastat -s setname -p
-p                md.tab ファイルと同様の形式で状態を表示します。
```

- 5 **metadetach(1M)** コマンドを使用し、ミラーから 1 つのサブミラーをオフラインにします。

```
# metadetach -s setname mirror submirror
```

---

注-読み取り操作は引き続きそのほかのサブミラーから行われます。読み取り操作は引き続きそのほかのサブミラーから実行できますが、オフラインのサブミラーは、ミラーに最初書き込んだ直後から同期がとれなくなります。この不一致は、オフラインのサブミラーをオンラインに戻したときに修正されます。fsck を実行する必要はありません。

---

- 6 **-u** オプションを指定して **lockfs** コマンドを使用し、ファイルシステムのロックを解除して書き込みを続行できるようにします。

```
# lockfs -u mountpoint
```

- 7 ファイルシステムを検査します。

```
# fsck /dev/md/diskset/rdisk/submirror
```

- 8 オフラインのサブミラーをテープなどのメディアにバックアップします。

**ufsdump(1M)** コマンドか、通常使用しているバックアップユーティリティーを使用します。

```
# ufsdump 0ucf dump-device submirror
```

---

注-ブロッケデバイス (/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(1M)** コマンドを使用し、メタデバイスまたはボリュームをオンラインに戻します。

```
# metattach -s setname mirror submirror
```

メタデバイスまたはボリュームをオンラインに戻すと、自動的にミラーとの再同期が行われます。

- 10 **metastat** コマンドを使用し、サブミラーが再同期されていることを確認します。

```
# metastat -s setname mirror
```

#### 例 12-4 ミラーのオンラインバックアップの実行 (Solaris Volume Manager)

次の例では、クラスタノード `phys-schost-1` がメタセット `schost-1` の所有者なので、バックアップ作成手順は `phys-schost-1` から実行します。ミラー `/dev/md/schost-1/dsk/d0` は、サブミラー `d10`、`d20`、`d30` で構成されています。

```
[Determine the owner of the metaset:]
# metastat -s schost-1
Set name = schost-1, Set number = 1
Host          Owner
  phys-schost-1  Yes
...
[Lock the file system from writes:]
# lockfs -w /global/schost-1
[List the submirrors:]
# metastat -s schost-1 -p
schost-1/d0 -m schost-1/d10 schost-1/d20 schost-1/d30 1
schost-1/d10 1 1 d4s0
schost-1/d20 1 1 d6s0
schost-1/d30 1 1 d8s0
[Take a submirror offline:]
# metadetach -s schost-1 d0 d30
[Unlock the file system:]
# lockfs -u /
[Check the file system:]
# fsck /dev/md/schost-1/rdisk/d30
[Copy the submirror to the backup device:]
# ufsdump 0ucf /dev/rmt/0 /dev/md/schost-1/rdisk/d30
DUMP: Writing 63 Kilobyte records
DUMP: Date of this level 0 dump: Tue Apr 25 16:15:51 2000
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/md/schost-1/rdisk/d30 to /dev/rdisk/c1t9d0s0.
...
DUMP: DUMP IS DONE
[Bring the submirror back online:]
# metattach -s schost-1 d0 d30
schost-1/d0: submirror schost-1/d30 is attached
[Resynchronize the submirror:]
# metastat -s schost-1 d0
schost-1/d0: Mirror
  Submirror 0: schost-0/d10
    State: Okay
  Submirror 1: schost-0/d20
    State: Okay
  Submirror 2: schost-0/d30
    State: Resyncing
    Resync in progress: 42% done
    Pass: 1
```



```
Read option: roundrobin (default)
...
```

## ▼ ボリュームのオンラインバックアップを実行する (Veritas Volume Manager)

Veritas Volume Manager では、ミラー化ボリュームはプレックスと認識されます。プレックスは、マウント解除したり、ボリューム全体をオフラインにしなくてもバックアップできます。プレックスは、ボリュームのスナップショットコピーを作成し、この一時ボリュームをバックアップします。システムを停止したり、データへのユーザーアクセスを拒否する必要はありません。

バックアップ手順を実行する前に、クラスタが正常に動作していることを確認してください。

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の Oracle Solaris Cluster コマンドを使用して説明します。多くのコマンドには短縮形もあります。コマンド名の形式の長短を除き、コマンドは同一です。

- 1 クラスタ内の任意のノードにログオンし、クラスタ上のディスクグループの現在の主ノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。

- 2 ディスクグループ情報を表示します。

```
# vxprint -g diskgroup
```

- 3 どのノードに現在インポートされているディスクグループがあるかを判別します(これはそのノードがディスクグループの主ノードであることを示します)。

```
# cldevicegroup status
```

- 4 ボリュームのスナップショットを作成します。

```
# vxassist -g diskgroup snapstart volume
```

---

注-ボリュームのサイズによっては、スナップショットの作成に時間がかかることがあります。

---

- 5 新しいボリュームが作成されたことを確認します。

```
# vxprint -g diskgroup
```

スナップショットの作成が完了すると、選択したディスクグループの State フィールドに Snapdone と表示されます。

- 6 ファイルシステムにアクセスしているデータサービスを停止します。

```
# clresourcegroup offline resource-group
```

---

注-データファイルシステムが正しくバックアップされるように、すべてのデータサービスを停止します。データサービスが実行中でない場合は、[手順6](#)と[手順8](#)を実行する必要はありません。

---

- 7 **bkup-vol** という名前のバックアップボリュームを作成し、それにスナップショットボリュームを添付します。

```
# vxassist -g diskgroup snapshot volume bkup-vol
```

- 8 **clresourcegroup** コマンドを使用して、[手順6](#)で停止されたデータサービスを再起動します。

```
# clresourcegroup online -zone -n node resourcegroup
```

*node* ノードの名前。

*zone* リソースグループをマスターできるグローバルクラスタ非投票ノード (*node*) の名前。リソースグループを作成した際に非投票ノードを指定した場合にのみ、*zone* を指定します。

- 9 そのボリュームが新しいボリューム **bkup-vol** に添付されていることを確認します。

```
# vxprint -g diskgroup
```

- 10 デバイスグループ構成変更を登録します。

```
# cldevicegroup sync diskgroup
```

- 11 バックアップボリュームを確認します。

```
# fsck -y /dev/vx/rdisk/diskgroup/bkup-vol
```

- 12 テープなどのメディアにボリューム **bkup-vol** をバックアップします。

**ufsdump(1M)** コマンドか、通常使用しているバックアップユーティリティを使用します。

```
# ufsdump 0ucf dump-device /dev/vx/dsk/diskgroup/bkup-vol
```

- 13 一時ボリュームを削除します。

```
# vxedit -rf rm bkup-vol
```

- 14 ディスクグループ構成変更を登録します。

```
# cldevicegroup sync diskgroup
```

## 例 12-5 ポリリュームのオンラインバックアップの実行 (Veritas Volume Manager)

次の例では、クラスタノード phys-schost-2 はデバイスグループ schost-1 の主所有者です。そのため、バックアップ手順は phys-schost-2 から実行します。ポリリューム /vo101 がコピーされ、新しいポリリューム bkup-vol と関連付けられます。

```
[Become superuser or assume a role that provides solaris.cluster.admin RBAC authorization on the primary node.]
```

```
[Identify the current primary node for the device group:]
```

```
# cldevicegroup status
```

```
-- Device Group Servers --
                Device Group      Primary           Secondary
                -----            -             -
Device group servers:  rmt/1                -                -
Device group servers:  schost-1          phys-schost-2    phys-schost-1
```

```
-- Device Group Status --
```

```
                Device Group      Status
                -----            -
Device group status:  rmt/1                Offline
Device group status:  schost-1          Online
```

```
[List the device group information:]
```

```
# vxprint -g schost-1
```

```
TY NAME          ASSOC      KSTATE  LENGTH  PLOFFS STATE  TUTIL0  PUTIL0
dg schost-1      schost-1  -        -        -        -        -        -

dm schost-101    c1t1d0s2  -        17678493 -        -        -        -
dm schost-102    c1t2d0s2  -        17678493 -        -        -        -
dm schost-103    c2t1d0s2  -        8378640  -        -        -        -
dm schost-104    c2t2d0s2  -        17678493 -        -        -        -
dm schost-105    c1t3d0s2  -        17678493 -        -        -        -
dm schost-106    c2t3d0s2  -        17678493 -        -        -        -

v vol01          gen        ENABLED  204800  -        ACTIVE  -        -
pl vol01-01      vol01      ENABLED  208331  -        ACTIVE  -        -
sd schost-101-01 vol01-01   ENABLED  104139  0        -        -        -
sd schost-102-01 vol01-01   ENABLED  104139  0        -        -        -
pl vol01-02      vol01      ENABLED  208331  -        ACTIVE  -        -
sd schost-103-01 vol01-02   ENABLED  103680  0        -        -        -
sd schost-104-01 vol01-02   ENABLED  104139  0        -        -        -
pl vol01-03      vol01      ENABLED  LOGONLY  -        ACTIVE  -        -
sd schost-103-02 vol01-03   ENABLED  5        LOG      -        -        -
```

```
[Start the snapshot operation:]
```

```
# vxassist -g schost-1 snapstart vol01
```

```
[Verify the new volume was created:]
```

```
# vxprint -g schost-1
```

```
TY NAME          ASSOC      KSTATE  LENGTH  PLOFFS STATE  TUTIL0  PUTIL0
dg schost-1      schost-1  -        -        -        -        -        -

dm schost-101    c1t1d0s2  -        17678493 -        -        -        -
dm schost-102    c1t2d0s2  -        17678493 -        -        -        -
dm schost-103    c2t1d0s2  -        8378640  -        -        -        -
dm schost-104    c2t2d0s2  -        17678493 -        -        -        -
dm schost-105    c1t3d0s2  -        17678493 -        -        -        -
dm schost-106    c2t3d0s2  -        17678493 -        -        -        -
```

```

v vol01          gen          ENABLED 204800 - ACTIVE - -
pl vol01-01     vol01         ENABLED 208331 - ACTIVE - -
sd schost-101-01 vol01-01     ENABLED 104139 0 - - - -
sd schost-102-01 vol01-01     ENABLED 104139 0 - - - -
pl vol01-02     vol01         ENABLED 208331 - ACTIVE - -
sd schost-103-01 vol01-02     ENABLED 103680 0 - - - -
sd schost-104-01 vol01-02     ENABLED 104139 0 - - - -
pl vol01-03     vol01         ENABLED LOGONLY - ACTIVE - -
sd schost-103-02 vol01-03     ENABLED 5 LOG - - - -
pl vol01-04     vol01         ENABLED 208331 - SNAPDONE - -
sd schost-105-01 vol01-04     ENABLED 104139 0 - - - -
sd schost-106-01 vol01-04     ENABLED 104139 0 - - - -
[Stop data services, if necessary:]
# clresourcegroup offline nfs-rg
[Create a copy of the volume:]
# vxassist -g schost-1 snapshot vol01 bkup-vol
[Restart data services, if necessary:]
# clresourcegroup online -n phys-schost-1 nfs-rg
[Verify bkup-vol was created:]
# vxprint -g schost-1
TY NAME          ASSOC          KSTATE  LENGTH  PLOFFS STATE  TUTIL0  PUTIL0
dg schost-1      schost-1      -        -        -        -        -        -

dm schost-101    clt1d0s2      -        17678493 -        -        -        -
...

v bkup-vol       gen          ENABLED 204800 - ACTIVE - -
pl bkup-vol-01  bkup-vol     ENABLED 208331 - ACTIVE - -
sd schost-105-01 bkup-vol-01  ENABLED 104139 0 - - - -
sd schost-106-01 bkup-vol-01  ENABLED 104139 0 - - - -

v vol01          gen          ENABLED 204800 - ACTIVE - -
pl vol01-01     vol01         ENABLED 208331 - ACTIVE - -
sd schost-101-01 vol01-01     ENABLED 104139 0 - - - -
sd schost-102-01 vol01-01     ENABLED 104139 0 - - - -
pl vol01-02     vol01         ENABLED 208331 - ACTIVE - -
sd schost-103-01 vol01-02     ENABLED 103680 0 - - - -
sd schost-104-01 vol01-02     ENABLED 104139 0 - - - -
pl vol01-03     vol01         ENABLED LOGONLY - ACTIVE - -
sd schost-103-02 vol01-03     ENABLED 5 LOG - - - -
[Synchronize the disk group with cluster framework:]
# cldevicegroup sync schost-1
[Check the file systems:]
# fsck -y /dev/vx/rdisk/schost-1/bkup-vol
[Copy bkup-vol to the backup device:]
# ufsdump 0ucf /dev/rmt/0 /dev/vx/rdisk/schost-1/bkup-vol
DUMP: Writing 63 Kilobyte records
DUMP: Date of this level 0 dump: Tue Apr 25 16:15:51 2000
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/vx/dsk/schost-2/bkup-vol to /dev/rmt/0.
...
DUMP: DUMP IS DONE
[Remove the bkup-volume:]
# vxedit -rf rm bkup-vol
[Synchronize the disk group:]
# cldevicegroup sync schost-1

```

## ▼ クラスタ構成をバックアップする

クラスタ構成をアーカイブし、クラスタ構成の簡単な復元を実現するため、定期的にクラスタ構成をバックアップします。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)` を参照してください。

- 3 クラスタ構成情報が正常に XML ファイルにエクスポートされたことを確認します。

```
# vi configfile
```

## クラスタファイルの復元の作業マップ

`ufsrestore(1M)` コマンドを実行すると、`ufsdump(1M)` コマンドで作成されたバックアップのファイルが、現在の作業ディレクトリを基準として指定されるディスク上の位置にコピーされます。`ufsrestore` を使用すると、レベル 0 のダンプとそれ以降の増分ダンプからファイルシステム階層全体を読み込み直したり、任意のダンプテープから個々のファイルを復元できます。スーパーユーザーまたは同等の役割として `ufsrestore` を実行すると、元の所有者、最終修正時刻、モード (アクセス権) を保持したままファイルを復元できます。

ファイルまたはファイルシステムの復元を開始する前に、次の点を確認してください。

- 必要なテープ
- ファイルシステムを復元する raw デバイス名
- 使用するテープドライブの種類
- テープドライブのデバイス名 (ローカルまたはリモート)
- 障害が発生したディスクのパーティション分割方式。これは、パーティションとファイルシステムを交換用ディスクに正確に複製しなければならないためです。

表 12-2 作業リスト:クラスタファイルの復元

| 作業   | 参照先   |
|--|---|
| Solaris Volume Manager の場合、対話形式でファイルを復元                | 350 ページの「個々のファイルを対話形式で復元する (Solaris Volume Manager)」  |
| Solaris Volume Manager の場合、ルート (/) ファイルシステムを復元         | 350 ページの「ルート (/) ファイルシステムを復元する (Solaris Volume Manager)」<br><br>353 ページの「Solaris Volume Manager ボリューム上に存在していたルート (/) ファイルシステムを復元する」 |
| Veritas Volume Manager の場合、ルート (/) ファイルシステムを復元         | 358 ページの「カプセル化されていないルート (/) ファイルシステムを復元する (Veritas Volume Manager)」   |
| Veritas Volume Manager の場合、カプセル化されたルート (/) ファイルシステムを復元 | 360 ページの「カプセル化されたルート (/) ファイルシステムを復元する (Veritas Volume Manager)」  |

## ▼ 個々のファイルを対話形式で復元する (Solaris Volume Manager)

この手順を使用し、1つまたは複数の個々のファイルを復元します。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

- 1 復元するクラスタノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.admin` を提供する役割になります。
- 2 復元するファイルを使用しているデータサービスをすべて停止します。  
`# clresourcegroup offline resource-group`
- 3 ファイルを復元します。  
`# ufsrestore`

## ▼ ルート (/) ファイルシステムを復元する (Solaris Volume Manager)

障害の発生したルートディスクを交換した後などに、この手順を使用してルート (/) ファイルシステムを新しいディスクに復元します。復元中のノードは起動しなさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

---

注-新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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。復元を行っているノードはオフラインであるため、システムは「RPC: Rpcbnd failure - RPC: Timed out」というエラーを表示します。このエラーを無視し、次のステップを続けます。

```
# metaset -s setname -f -d -h nodelist
```

```
-s setname      ディスクセット名を指定します。
-f             ディスクセットから最後のホストを削除します。
-d            ディスクセットから削除します。
-h nodelist    ディスクセットから削除するノードの名前を指定します。
```

- 3 **root (/)** ファイルシステムと **/usr** ファイルシステムを復元します。  
root ファイルシステムと /usr ファイルシステムを復元するには、『Solaris のシステム管理(デバイスとファイルシステム)』の第 26 章「UFS ファイルとファイルシステムの復元(手順)」の手順に従ってください。Oracle Solaris OS の手順にあるシステムを再起動する手順は省略してください。

---

注 - /global/.devices/node@nodeid ファイルシステムが作成されていることを確認します。

---

- 4 ノードをマルチユーザーモードで再起動します。

```
# reboot
```

- 5 デバイスIDを交換します。

```
# cldevice repair rootdisk
```

- 6 **metadb(1M)** コマンドを使用し、状態データベースの複製を再作成します。

```
# metadb -c copies -af raw-disk-device
```

-c *copies*                    作成する複製の数を指定します。

-f *raw-disk-device*        複製の作成先の raw ディスクデバイス名を指定します。

-a                            複製を追加します。

- 7 復元するノード以外のクラスタノードから、復元するノードをすべてのディスクセットに追加します。

```
phys-schost-2# metaset -s setname -a -h nodelist
```

-a                            ホストを作成してディスクセットに追加します。

ノードがクラスタモードで再起動します。これでクラスタを使用できるようになります。

#### 例 12-6 ルート (/) ファイルシステムの復元 (Solaris Volume Manager)

次に、テープデバイス `/dev/rmt/0` からノード `phys-schost-1` に復元したルート (/) ファイルシステムの例を示します。metaset コマンドは、クラスタの別のノード `phys-schost-2` から実行し、ノード `phys-schost-1` を削除し、後でディスクセット `schost-1` に追加します。そのコマンドはすべて `phys-schost-1` から実行します。新しいブートブロックが `/dev/rdisk/c0t0d0s0` に作成され、3つの状態データベースの複製が `/dev/rdisk/c0t0d0s4` に再作成されます。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC authorization on a cluster node
other than the node to be restored.]
[Remove the node from the metaset:]
phys-schost-2# metaset -s schost-1 -f -d -h phys-schost-1
[Replace the failed disk and boot the node:]
Restore the root (/) and /usr file system using the procedure in the Solaris system
administration documentation
[Reboot:]
# reboot
[Replace the disk ID:]
# cldevice repair /dev/dsk/c0t0d0
[Re-create state database replicas:]
# metadb -c 3 -af /dev/rdisk/c0t0d0s4
[Add the node back to the metaset:]
phys-schost-2# metaset -s schost-1 -a -h phys-schost-1
```



## ▼ Solaris Volume Manager ボリューム上に存在していたルート (/) ファイルシステムを復元する

この手順を使用して、バックアップ実行時に Solaris Volume Manager ボリューム上に存在していたルート (/) ファイルシステムを復元します。この手順は、破損したルートディスクを新しいディスクに交換する場合などに実行します。復元中のノードは起動しなおさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

---

注- 新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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 ディスクセットへのアクセス権があるクラスタノード、ただし復元するノード以外のノード上で、スーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になります。  
復元する以外のノードを使用します。
- 2 接続先のすべてのディスクセットから、復元するノードのホスト名を削除します。  
ディスクセットごとに次のコマンドを1回実行します。

```
# metaset -s setname -d -h hostname
```

|                       |                                       |
|-----------------------|---------------------------------------|
| -s setname            | メタセット名を指定します。                         |
| -f                    | ディスクセットから最後のホストを削除します。                |
| -d                    | メタセットから削除します。                         |
| -h nodelist           | メタセットから削除するノードの名前を指定します。              |
| -h hostname           | ホストの名前を指定します。                         |
| -m mediator_host_list | ディスクセットから追加または削除するメディアータホストの名前を指定します。 |

- 3 ノードが二重列メディアータホストである場合は、メディアータを削除します。  
ノードの接続先ディスクセットごとに次のコマンドを1回実行します。

```
# metaset -ssetname-d -m hostname
```

- 4 ルート(/)ファイルシステムを復元するノードで、障害の発生したディスクを交換します。

ディスク交換手順については、サーバーに付属のマニュアルを参照してください。

- 5 復元するノードを起動します。修復されたホストは**CD-ROM**からシングルユーザーモードで起動されるため、**Solaris Volume Manager**はそのノードで実行されていません。

- Oracle Solaris OS CD を使用している場合は、次の点に注意してください。

- SPARC:次のように入力します。

```
ok boot cdrom -s
```

- x86:CD をシステムのCDドライブに挿入し、システムを停止して、電源を切って入れなおすことにより、システムを起動します。「Current Boot Parameters」画面で**b**または**i**を入力します。

```
<<< Current Boot Parameters >>>
```

```
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@7,1/sd@0,0:a
```

```
Boot args:
```

```
Type b [file-name] [boot-flags] <ENTER> to boot with options
or i <ENTER> to enter boot interpreter
or <ENTER> to boot with defaults
```

```
<<< timeout in 5 seconds >>>
```

```
Select (b)oot or (i)nterpreter: b -s
```

- Solaris JumpStart サーバーを使用している場合は、次の点に注意してください。

- SPARC:次のように入力します。

```
ok boot net -s
```

- x86:CD をシステムのCDドライブに挿入し、システムを停止して、電源を切って入れなおすことにより、システムを起動します。「Current Boot Parameters」画面で**b**または**i**を入力します。

```
<<< Current Boot Parameters >>>
```

```
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@7,1/sd@0,0:a
```

```
Boot args:
```

```
Type b [file-name] [boot-flags] <ENTER> to boot with options
or i <ENTER> to enter boot interpreter
or <ENTER> to boot with defaults
```

```
<<< timeout in 5 seconds >>>
```

```
Select (b)oot or (i)nterpreter: b -s
```

- 6 **format** コマンドを使用し、ルートディスクのすべてのパーティションとスワップ空間を作成します。  
障害の発生したディスクに存在していた元のパーティションの分割方式を再作成します。

- 7 **newfs** コマンドを使用し、必要に応じてルート (/) ファイルシステムやその他のファイルシステムを作成します。  
障害の発生したディスクに存在していた元のファイルシステムを再作成します。

---

注 - /global/.devices/ node@nodeid ファイルシステムが作成されていることを確認します。

---

- 8 ルート (/) ファイルシステムを一時マウントポイントにマウントします。

```
# mount device temp-mountpoint
```

- 9 次のコマンドを使用し、ルート (/) ファイルシステムを復元します。

```
# cd temp-mountpoint
# ufsrestore rvf dump-device
# rm restoresymtable
```

- 10 新しい起動ブロックを新しいディスクにインストールします。

```
# /usr/sbin/installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk
raw-disk-device
```

- 11 `/temp-mountpoint/etc/system` ファイルの MDD ルート情報の行を削除します。

```
* Begin MDD root info (do not edit)
forceload: misc/md_trans
forceload: misc/md_raid
forceload: misc/md_mirror
forceload: misc/md_hotspares
forceload: misc/md_stripe
forceload: drv/pcipsy
forceload: drv/glm
forceload: drv/sd
rootdev:/pseudo/md@0:0,10,blk
* End MDD root info (do not edit)
```

- 12 `/temp-mountpoint/etc/vfstab` ファイルを編集して、ルートエントリを、**Solaris Volume Manager** ポリリュームから、メタデバイスまたはポリリュームの一部であるルートディスク上の各ファイルシステムの対応する正常なスライスに変更します。

Example:

Change from-

```
/dev/md/dsk/d10 /dev/md/rdisk/d10 / ufs 1 no -
```

Change to-

```
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
```

- 13 一時ファイルシステムをマウント解除し、**raw** ディスクデバイスを確認します。
- ```
# cd /
# umount temp-mountpoint
# fsck raw-disk-device
```
- 14 ノードをマルチユーザーモードで再起動します。
- ```
# reboot
```
- 15 デバイス ID を交換します。
- ```
# cldevice repair rootdisk
```
- 16 **metadb** コマンドを使用し、状態データベースの複製を再作成します。
- ```
# metadb -c copies -af raw-disk-device
```
- c *copies*                    作成する複製の数を指定します。
- af *raw-disk-device*        指定した raw ディスクデバイスに初期状態のデータベースの複製を作成します。
- 17 復元したノード以外のクラスタノードから、復元したノードをすべてのディスクセットに追加します。
- ```
phys-schost-2# metaset -s setname -a -h nodelist
```
- a                            メタセットを追加 (作成) します。
- マニュアルに従って、ボリューム/ミラーのルート (/) を設定します。
- ノードがクラスタモードで再起動します。
- 18 ノードが二重列メディアータホストだった場合は、メディアータを再び追加します。
- ```
phys-schost-2# metaset -s setname -a -m hostname
```

### 例 12-7 Solaris Volume Manager ボリューム上に存在していたルート (/) ファイルシステムの復元

次に、テープデバイス /dev/rmt/0 からノード `phys-schost-1` に復元したルート (/) ファイルシステムの例を示します。metaset コマンドは、クラスタの別のノード `phys-schost-2` から実行し、ノード `phys-schost-1` を削除し、後でメタセット `schost-1` に追加します。そのコマンドはすべて `phys-schost-1` から実行します。新しいブートブロックが /dev/rdisk/c0t0d0s0 に作成され、3つの状態データベースの複製が /dev/rdisk/c0t0d0s4 に再作成されます。

```
[Become superuser or assume a role that provides solaris.cluster.modify RBAC
authorization on a cluster node with access to the metaset, other than the node to be restored.]
[Remove the node from the metaset:]
```

```
phys-schost-2# metaset -s schost-1 -d -h phys-schost-1
[Replace the failed disk and boot the node:]
```

Oracle Solaris OS CD からノードを起動するには、次の手順に従います。

- SPARC:次のように入力します。

```
ok boot cdrom -s
```

- x86:CD をシステムの CD ドライブに挿入し、システムを停止して、電源を切って入れなおすことにより、システムを起動します。「Current Boot Parameters」画面で **b** または **i** を入力します。

```
<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci8086,2545@3/pci8086,1460@1d/pci8086,341a@7,1/
sd@0,0:a
Boot args:
```

```
Type  b [file-name] [boot-flags] <ENTER> to boot with options
      i <ENTER>                          to enter boot interpreter
      <ENTER>                             to boot with defaults
```

```
<<< timeout in 5 seconds >>>
Select (b)oot or (i)nterpreter: b -s
```

```
[Use format and newfs to recreate partitions and file systems
.]
[Mount the root file system on a temporary mount point:]
# mount /dev/dsk/c0t0d0s0 /a
[Restore the root file system:]
# cd /a
# ufsrestore rvf /dev/rmt/0
# rm restoresymtable
[Install a new boot block:]
# /usr/sbin/installboot /usr/platform/'uname \
-i'/lib/fs/ufs/bootblk /dev/rdisk/c0t0d0s0

[Remove the lines in / temp-mountpoint/etc/system file for MDD root information:
]
* Begin MDD root info (do not edit)
forceload: misc/md_trans
forceload: misc/md_raid
forceload: misc/md_mirror
forceload: misc/md_hotspares
forceload: misc/md_stripe
forceload: drv/pcipsy
forceload: drv/glm
forceload: drv/sd
rootdev:/pseudo/md@0:0,10,blk
* End MDD root info (do not edit)
[Edit the /temp-mountpoint/etc/vfstab file]
Example:
Change from-
/dev/md/dsk/d10 /dev/md/rdsk/d10 / ufs 1 no -
Change to-
/dev/dsk/c0t0d0s0 /dev/rdsk/c0t0d0s0 /usr ufs 1 no -
```

```
[Unmount the temporary file system and check the raw disk device:]
# cd /
# umount /a
# fsck /dev/rdisk/c0t0d0s0
[Reboot:]
# reboot
[Replace the disk ID:]
# cldevice repair /dev/rdisk/c0t0d0
[Re-create state database replicas:]
# metadb -c 3 -af /dev/rdisk/c0t0d0s4
[Add the node back to the metaset:]
phys-schost-2# metaset -s schost-1 -a -h phys-schost-1
```

## ▼ カプセル化されていないルート (/) ファイルシステムを復元する (Veritas Volume Manager)

この手順を使用して、カプセル化されていないルート (/) ファイルシステムをノードに復元します。復元中のノードは起動しなおさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

---

注- 新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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 ルートファイルシステムを復元するノードで、障害が発生したディスクを交換します。  
ディスク交換手順については、サーバーに付属のマニュアルを参照してください。
- 2 復元するノードを起動します。
  - Oracle Solaris OS CD を使用している場合は、OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。  
ok **boot cdrom -s**
  - Solaris JumpStart サーバーを使用している場合は、OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。  
ok **boot net -s**

- 3 **format** コマンドを使用し、ルートディスクのすべてのパーティションとスワップを作成します。  
障害の発生したディスクに存在していた元のパーティションの分割方式を再作成します。
- 4 **newfs** コマンドを使用し、必要に応じてルート (/) ファイルシステムやその他のファイルシステムを作成します。  
障害の発生したディスクに存在していた元のファイルシステムを再作成します。

---

注 - /global/.devices/ node@nodeid ファイルシステムが作成されていることを確認します。

---

- 5 ルート (/) ファイルシステムを一時マウントポイントにマウントします。  
`# mount device temp-mountpoint`
- 6 バックアップからルート (/) ファイルシステムを復元し、ファイルシステムをマウント解除して確認します。  
`# cd temp-mountpoint`  
`# ufsrestore rvf dump-device`  
`# rm restoresymtable`  
`# cd /`  
`# umount temp-mountpoint`  
`# fsck raw-disk-device`  
これでファイルシステムが復元されます。
- 7 新しい起動ブロックを新しいディスクにインストールします。  
`# /usr/sbin/installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk raw-disk-device`
- 8 ノードをマルチユーザーモードで再起動します。  
`# reboot`
- 9 デバイス ID を更新します。  
`# cldevice repair /dev/rdisk/disk-device`
- 10 **Control-D** キーを押して、マルチユーザーモードで再起動します。  
ノードがクラスタモードで再起動します。これでクラスタを使用できるようになります。

#### 例 12-8 カプセル化されていないルート (/) ファイルシステムの復元 (Veritas Volume Manager)

次に、カプセル化されていないルート (/) ファイルシステムがテープデバイス /dev/rmt/0 からノード phys-schost-1 に復元される例を示します。

[Replace the failed disk and boot the node:]

Oracle Solaris OS CD からノードを起動します。OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。

```
ok boot cdrom -s
...
[Use format and newfs to create partitions and file systems]
[Mount the root file system on a temporary mount point:]
# mount /dev/dsk/c0t0d0s0 /a
[Restore the root file system:]
# cd /a
# ufsrestore rvf /dev/rmt/0
# rm restoresymtable
# cd /
# umount /a
# fsck /dev/rdisk/c0t0d0s0
[Install a new boot block:]
# /usr/sbin/installboot /usr/platform/'uname \
-i'/lib/fs/ufs/bootblk /dev/rdisk/c0t0d0s0

[Reboot:]
# reboot
[Update the disk ID:]
# cldevice repair /dev/rdisk/c0t0d0
```

## ▼ カプセル化されたルート (/) ファイルシステムを復元する (Veritas Volume Manager)

この手順を使用して、カプセル化されたルート (/) ファイルシステムをノードに復元します。復元中のノードは起動しなさないでください。復元手順を実行する前に、クラスタが正常に動作していることを確認してください。

---

注-新しいディスクは、障害が発生したディスクと同じ形式でパーティション分割する必要があります。このため、この手順を始める前に、パーティションの分割方式を確認し、ファイルシステムを適切に再作成しておいてください。

---

phys-schost# プロンプトは、グローバルクラスタのプロンプトを表します。この手順は、グローバルクラスタ上で実行します。

この手順では、長形式の 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 ルートファイルシステムを復元するノードで、障害が発生したディスクを交換します。  
ディスク交換手順については、サーバーに付属のマニュアルを参照してください。
- 2 復元するノードを起動します。
  - Oracle Solaris OS CD を使用している場合は、OpenBoot PROM の `ok` プロンプトで、次のコマンドを入力します。
 

```
ok boot cdrom -s
```
  - Solaris JumpStart サーバーを使用している場合は、OpenBoot PROM の `ok` プロンプトで、次のコマンドを入力します。
 

```
ok boot net -s
```
- 3 `format` コマンドを使用し、ルートディスクのすべてのパーティションとスワップ空間を作成します。  
障害の発生したディスクに存在していた元のパーティションの分割方式を再作成します。
- 4 `newfs` コマンドを使用し、必要に応じてルート (`/`) ファイルシステムやその他のファイルシステムを作成します。  
障害の発生したディスクに存在していた元のファイルシステムを再作成します。

---

注 `~/global/.devices/node@nodeid` ファイルシステムが作成されていることを確認します。

---

- 5 ルート (`/`) ファイルシステムを一時マウントポイントにマウントします。
 

```
# mount device temp-mountpoint
```
- 6 バックアップからルート (`/`) ファイルシステムを復元します。
 

```
# cd temp-mountpoint
# ufsrestore rvf dump-device
# rm restoresymtable
```
- 7 空の `install-db` ファイルを作成します。  
このファイルによって、次回起動時にノードが VxVM インストールモードになります。
 

```
# touch \
/temp-mountpoint/etc/vx/reconfig.d/state.d/install-db
```
- 8 `/temp-mountpoint/etc/system` ファイル内の次のエントリを削除します。
 

```
* rootdev:/pseudo/vxio@0:0
* set vxio:vol_rootdev_is_volume=1
```

- 9 /temp-mountpoint/etc/vfstab ファイルを編集し、すべてのVxVM マウントポイントをルートディスクの標準ディスクデバイス(/dev/dsk/c0t0d0s0 など)に置換します。

Example:

Change from-

```
/dev/vx/dsk/rootdg/rootvol /dev/vx/rdisk/rootdg/rootvol / ufs 1 no -
```

Change to-

```
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
```

- 10 一時ファイルシステムをマウント解除してファイルシステムを確認します。

```
# cd /
# umount temp-mountpoint
# fsck raw-disk-device
```

- 11 起動ブロックを新しいディスクにインストールします。

```
# /usr/sbin/installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk raw-disk-device
```

- 12 ノードをマルチユーザーモードで再起動します。

```
# reboot
```

- 13 **scdidadm(1M)** を使用し、デバイス ID を更新します。

```
# cldevice repair /dev/rdisk/c0t0d0
```

- 14 ディスクをカプセル化して再起動するために、**clvsvm** コマンドを実行します。

- 15 マイナー番号が他のシステムと衝突している場合は、グローバルデバイスをマウント解除し、ディスクグループに別のマイナー番号を割り当てます。

- クラスタノードのグローバルデバイスファイルシステムをマウント解除します。

```
# umount /global/.devices/node@nodeid
```

- クラスタノードの rootdg ディスクグループに別のマイナー番号を割り当てます。

```
# vxdg remminor rootdg 100
```

- 16 ノードを停止し、クラスタモードで再起動します。

```
# shutdown -g0 -i6 -y
```

#### 例 12-9 カプセル化されたルート (/) ファイルシステムの復元 (Veritas Volume Manager)

次に、カプセル化されたルート (/) ファイルシステムがテープデバイス /dev/rmt/0 からノード phys-schost-1 に復元される例を示します。

```
[Replace the failed disk and boot the node:]
```

Oracle Solaris OS CD からノードを起動します。OpenBoot PROM の ok プロンプトで、次のコマンドを入力します。

```
ok boot cdrom -s
...
[Use format and newfs to create partitions and file systems]
[Mount the root file system on a temporary mount point:]
# mount /dev/dsk/c0t0d0s0 /a
[Restore the root file system:]
# cd /a
# ufsrestore rvf /dev/rmt/0
# rm restoresymtable
[Create an empty install-db file:]
# touch /a/etc/vx/reconfig.d/state.d/install-db
[Edit /etc/system on the temporary file system and
remove or comment out the following entries:]
    # rootdev:/pseudo/vxio@0:0
    # set vxio:vol_rootdev_is_volume=1
[Edit /etc/vfstab on the temporary file system:]
Example:
Change from-
/dev/vx/dsk/rootdg/rootvol /dev/vx/rdisk/rootdg/rootvol / ufs 1 no-

Change to-
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
[Unmount the temporary file system, then check the file system:]
# cd /
# umount /a
# fsck /dev/rdisk/c0t0d0s0
[Install a new boot block:]
# /usr/sbin/installboot /usr/platform/'uname \
-i'/lib/fs/ufs/bootblk /dev/rdisk/c0t0d0s0
[Reboot:]
# reboot
[Update the disk ID:]
# cldevice repair /dev/rdisk/c0t0d0
[Encapsulate the disk:]
# vxinstall
Choose to encapsulate the root disk.
[If a conflict in minor number occurs, remind the rootdg disk group:]
# umount /global/.devices/node@nodeid
# vxdg reminor rootdg 100
# shutdown -g0 -i6 -y
```

参照 カプセル化されたルートディスクをミラー化する手順については、『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。



# ◆◆◆ 13

## 第 13 章

# グラフィカルユーザーインターフェースによる Oracle Solaris Cluster の管理

---

この章では、グラフィカルユーザーインターフェース (Graphical User Interface、GUI) ツールの、Oracle Solaris Cluster Manager と Sun Management Center について説明します。これらのツールを使用すると、クラスタをさまざまな面から管理できます。また、Oracle Solaris Cluster Manager を構成および起動する手順も説明します。Oracle Solaris Cluster Manager GUI に含まれるオンラインヘルプでは、さまざまな Oracle Solaris Cluster 管理作業の手順を説明しています。

この章の内容は、次のとおりです。

- 365 ページの「Oracle Solaris Cluster Manager の概要」
- 366 ページの「SPARC: Sun Management Center の概要」
- 367 ページの「Oracle Solaris Cluster Manager の構成」
- 370 ページの「Oracle Solaris Cluster Manager ソフトウェアの起動」

## Oracle Solaris Cluster Manager の概要

Oracle Solaris Cluster Manager は、クラスタ情報のグラフィカルな表示、クラスタコンポーネントの状態の確認、および構成変更の監視が可能な GUI です。Oracle Solaris Cluster Manager では、以下の Oracle Solaris Cluster コンポーネントを対象としたさまざまな管理作業も行えます。

- アダプタ
- ケーブル
- データサービス
- グローバルデバイス
- 相互接続
- 接続点
- ノードの負荷制限
- NAS デバイス
- ノード
- 定足数デバイス

- リソースグループ
- リソース

Oracle Solaris Cluster Manager をインストールおよび使用方法については、次の文書を参照してください。

- **Oracle Solaris Cluster Manager** のインストール: 『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。
- **Oracle Solaris Cluster Manager** の起動: 370 ページの「Oracle Solaris Cluster Manager ソフトウェアの起動」を参照してください。
- ポート番号、サーバーアドレス、セキュリティー証明書、およびユーザーの構成: 367 ページの「Oracle Solaris Cluster Manager の構成」を参照してください。
- **Oracle Solaris Cluster Manager** によるクラスタのインストールと管理: Oracle Solaris Cluster Manager に付属のオンラインヘルプを参照してください。
- **Oracle Solaris Cluster Manager** セキュリティー鍵の再生成: 369 ページの「共通エージェントコンテナのセキュリティー鍵を再生成する」を参照してください。

---

注 - ただし、Oracle Solaris Cluster Manager は現在、Oracle Solaris Cluster のすべての管理作業を実行できるわけではありません。一部の作業には、コマンド行インタフェースを使用する必要があります。

---

## SPARC: Sun Management Center の概要

Sun Management Center (旧 Sun Enterprise SyMON) 用の Oracle Solaris Cluster モジュールの GUI コンソールを使用すると、クラスタリソース、リソースタイプ、リソースグループをグラフィカルに表示できます。また、構成の変更を監視したり、クラスタコンポーネントの状態を検査できます。ただし、Sun Management Center 用の Oracle Solaris Cluster モジュールは、Oracle Solaris Cluster の構成作業を行えません。構成処理には、コマンド行インタフェースを使用する必要があります。詳細については、第 1 章「コマンド行インタフェース」を参照してください。

Sun Management Center 用 Oracle Solaris Cluster モジュールのインストールと起動については、『Oracle Solaris Cluster ソフトウェアのインストール』の第 8 章「Oracle Solaris Cluster モジュールの Sun Management Center へのインストール」を参照してください。

Sun Management Center 用の Oracle Solaris Cluster モジュールは Simple Network Management Protocol (SNMP) に準拠しています。したがって、SNMP に基づくサン以外の管理ステーションは、Oracle Solaris Cluster が作成する管理情報ベース (MIB) をデータ定義として使用できます。

Oracle Solaris Cluster MIB ファイルは、任意のクラスタノード上の `/opt/SUNWsymon/modules/cfg/sun-cluster-mib.mib` にあります。

Oracle Solaris Cluster の MIB ファイルは、モデル化された Oracle Solaris Cluster データの ASN.1 仕様です。この仕様は、Sun Management Center のすべての MIB で使用される仕様と同じです。Oracle Solaris Cluster MIB を使用方法については、『Sun Management Center 3.6 User's Guide』の「SNMP MIBs for Sun Management Centre Modules」にある、ほかの Sun Management Center MIB を使用するための手順を参照してください。

## Oracle Solaris Cluster Manager の構成

Oracle Solaris Cluster Manager は、定足数デバイス、IPMP グループ、インターコネクトコンポーネント、グローバルデバイスなどのあらゆる局面の状態を管理、表示できる GUI です。この GUI は、多くの Oracle Solaris Cluster CLI コマンドの代わりに使用できます。

Oracle Solaris Cluster Manager をクラスタに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。GUI を使用してさまざまな作業を行う方法については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

この節では、初期インストール後、Oracle Solaris Cluster Manager を再構成するための次のような手順について説明します。

- 367 ページの「RBAC の役割の設定」
- 368 ページの「Oracle Solaris Cluster Manager のサーバーアドレスを変更する」
- 369 ページの「共通エージェントコンテナのセキュリティー鍵を再生成する」

### RBAC の役割の設定

Oracle Solaris Cluster Manager では、RBAC を使用してクラスタを管理する権限を持つユーザーが決定されます。Oracle Solaris Cluster ソフトウェアには、いくつかの RBAC 権限プロファイルが含まれています。これらの権限プロファイルをユーザーまたは役割に割り当てることで、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に与えることができます。Oracle Solaris Cluster ソフトウェアの RBAC を設定および管理する方法については、第 2 章「Oracle Solaris Cluster と RBAC」を参照してください。

### ▼ 共通エージェントコンテナを使用して、サービスまたは管理エージェントのポート番号を変更する

共通エージェントコンテナサービスのデフォルトのポート番号 (6789) が実行中の別のプロセスと衝突する場合、`cacaoadm` コマンドを使用し、各クラスタノード上で、衝突しているサービスまたは管理エージェントのポート番号を変更できます。

- 1 すべてのクラスタ上で共通エージェントコンテナ管理デーモンを停止します。  
# /opt/bin/cacaoadm stop
- 2 Sun Java Web Console を停止します。  
# /usr/sbin/smcwebserver stop
- 3 **get-param** サブコマンドを使用して、共通エージェントコンテナサービスにより現在使用されているポート番号を取得します。  
# /opt/bin/cacaoadm get-param *parameterName*  
cacaoadm コマンドを使用して、以下の共通エージェントコンテナサービスのポート番号を変更できます。次のリストは、共通エージェントコンテナで管理できるサービスとエージェント、および対応するパラメータ名の例を示しています。  

|              |                            |
|--------------|----------------------------|
| JMX コネクタポート  | jmxmp-connector-port       |
| SNMP ポート     | snmp-adapter-port          |
| SNMP トラップポート | snmp-adapter-trap-port     |
| コマンドストリームポート | commandstream-adapter-port |
- 4 ポート番号を変更します。  
# /opt/bin/cacaoadm set-param *parameterName=parameterValue*
- 5 クラスタの各ノード上で手順 4 を繰り返します。
- 6 Sun Java Web Console を再起動します。  
# /usr/sbin/smcwebserver start
- 7 すべてのクラスタノード上で共通エージェントコンテナ管理デーモンを再起動します。  
# /opt/bin/cacaoadm start

## ▼ Oracle Solaris Cluster Manager のサーバーアドレスを変更する

クラスタノードのホスト名を変更する場合は、Oracle Solaris Cluster Manager が実行されるアドレスを変更する必要があります。デフォルトのセキュリティー証明書は、Oracle Solaris Cluster Manager がインストールされる時点でノードのホスト名に基づいて生成されます。ノードのホスト名をリセットするには、証明書ファイルである keystore を削除し、Oracle Solaris Cluster Manager を再起動します。Oracle Solaris Cluster Manager は、新しいホスト名を使用して新しい証明書ファイルを自動的に作成します。この手順は、ホスト名を変更したすべてのノード上で行う必要があります。



- 1 `/etc/opt/webconsole` にある証明書ファイル `keystore` を削除します。  

```
# cd /etc/opt/webconsole  
# pkgrm keystore
```
- 2 Oracle Solaris Cluster Manager を再起動します。  

```
# /usr/sbin/smcwebserver restart
```

## ▼ 共通エージェントコンテナのセキュリティー鍵を再生成する

Oracle Solaris Cluster Manager は、強力な暗号化技術を使用して、Oracle Solaris Cluster Manager Web サーバーと各クラスタノード間の安全な通信を確保しています。

Oracle Solaris Cluster Manager が使用する鍵は、各ノードの `/etc/opt/SUNWcacao/security` ディレクトリに格納されています。これらの鍵は、すべてのクラスタノードで同一でなければなりません。

通常の動作では、これらのキーはデフォルトの構成のままとなります。クラスタノードのホスト名を変更する場合は、共通エージェントコンテナのセキュリティー鍵を再生成する必要があります。また、鍵が攻撃の対象となる恐れがある場合 (マシンのルート侵入など) にも鍵の再生成が必要となります。セキュリティー鍵を再生成するには、次の手順を実行します。

- 1 すべてのクラスタ上で共通エージェントコンテナ管理デーモンを停止します。  

```
# /opt/bin/cacaoadm stop
```
- 2 クラスタの1つのノード上で、セキュリティー鍵を再生成します。  

```
phys-schost-1# /opt/bin/cacaoadm create-keys --force
```
- 3 セキュリティー鍵を再生成したノード上で共通エージェントコンテナ管理デーモンを再起動します。  

```
phys-schost-1# /opt/bin/cacaoadm start
```
- 4 `/etc/cacao/instances/default` ディレクトリの `tar` ファイルを作成します。  

```
phys-schost-1# cd /etc/cacao/instances/default  
phys-schost-1# tar cf /tmp/SECURITY.tar security
```
- 5 `/tmp/Security.tar` ファイルを各クラスタノードにコピーします。

- 6 `/tmp/SECURITY.tar` ファイルをコピーした各ノード上で、セキュリティーファイルを解凍します。  
`/etc/opt/SUNWcacao/` ディレクトリに既にセキュリティーファイルがある場合は、すべて上書きされます。  

```
phys-schost-2# cd /etc/cacao/instances/default
phys-schost-2# tar xf /tmp/SECURITY.tar
```
- 7 クラスタの各ノードから `/tmp/SECURITY.tar` ファイルを削除します。  
セキュリティーのリスクを避けるために `tar` ファイルの各コピーを削除する必要があります。  

```
phys-schost-1# rm /tmp/SECURITY.tar
phys-schost-2# rm /tmp/SECURITY.tar
```
- 8 すべてのノード上で 共通エージェントコンテナ管理デーモンを再起動します。  

```
phys-schost-1# /opt/bin/cacaoadm start
```
- 9 **Oracle Solaris Cluster Manager** を再起動します。  

```
# /usr/sbin/smcwebserver restart
```

## Oracle Solaris Cluster Manager ソフトウェアの起動

Oracle Solaris Cluster Manager グラフィカルユーザーインターフェース (Graphical User Interface、GUI) は、Oracle Solaris Cluster ソフトウェアをさまざまな面から簡単に管理する方法を提供します。詳細については、Oracle Solaris Cluster Manager のオンラインヘルプを参照してください。

クラスタを起動すると、Sun Java Web コンソールと共通エージェントコンテナの両方が起動します。Sun Java Web コンソールと共通エージェントコンテナが実行されていることを確認するには、この手順のすぐあとの障害追跡のセクションを参照してください。

### ▼ Oracle Solaris Cluster Manager を起動する

この手順では、クラスタ上で Oracle Solaris Cluster Manager を起動する方法を示します。

- 1 **Oracle Solaris Cluster Manager** にアクセスするときに、クラスタノードの `root` のユーザー名とパスワードを使用するか、異なるユーザー名とパスワードを設定するかを決定します。

- クラスタノードのルートのユーザー名を使用して Oracle Solaris Cluster Manager にアクセスする場合は、[手順5](#)に進みます。
  - 別のユーザー名とパスワードを設定する場合は、[手順3](#)に進んで Oracle Solaris Cluster Manager ユーザーアカウントを設定します。
- 2 クラスタノード上にインストールするクラスタノード上でスーパーユーザーになります。
  - 3 **Oracle Solaris Cluster Manager** 経由でクラスタにアクセスするためのユーザーアカウントを作成します。

`useradd(1M)` コマンドを使用して、ユーザーアカウントをシステムに追加します。root システムアカウントを使用しない場合、Oracle Solaris Cluster Manager にアクセスするには、少なくとも1つのユーザーアカウントを設定する必要があります。Oracle Solaris Cluster Manager のユーザーアカウントは、Oracle Solaris Cluster Manager だけで使用されます。これらのアカウントは、Oracle Solaris OS システムのユーザーアカウントとの関連はありません。RBAC の役割を作成し、それをユーザーアカウントに割り当てる方法については、[57 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」](#)を参照してください。

---

注- ノードにユーザーアカウントが設定されていない場合、そのユーザーはそのノードからは Oracle Solaris Cluster Manager 経由でクラスタにアクセスできません。また、アクセス権を持っている別のクラスタノードからも、そのノードを管理することはできません。

---

- 4 (省略可能) 追加するユーザーアカウントごとに [手順3](#) を繰り返します。
- 5 管理コンソール、またはクラスタの外部に存在する他のマシンから、ブラウザを起動します。
- 6 ブラウザのディスクとメモリーキャッシュのサイズが、0 より大きな値に設定されていることを確認します。
- 7 ブラウザで **Java** および **Javascript** が有効になっていることを確認します。
- 8 ブラウザから、クラスタ内の任意のノード上にある **Oracle Solaris Cluster Manager** のポートに接続します。  
デフォルトのポート番号は 6789 です。  
`https://node:6789/`
- 9 **Web** ブラウザにより提示されたすべての証明書を受け入れます。  
Java Web Console ログインページが表示されます。

- 10 Oracle Solaris Cluster Manager にアクセスするユーザーのユーザー名とパスワードを入力します。
- 11 「Log In」 ボタンをクリックします。  
Java Web Console のアプリケーション起動ページが表示されます。
- 12 Systems カテゴリの下の Oracle Solaris Cluster Manager リンクをクリックします。
- 13 Web ブラウザにより提示されたすべての追加の証明書を受け入れます。
- 14 Oracle Solaris Cluster Manager に接続できない場合は、次のサブステップを実行して、Solaris のインストール中に制限されたネットワークプロファイルが選択されたかどうかを判別し、Java Web コンソールサービスへの外部アクセスを復元します。  
Oracle Solaris のインストール中に制限されたネットワークプロファイルを選択すると、Sun Java Web コンソールサービスの外部アクセスは制限されます。このネットワークは、Oracle Solaris Cluster Manager の GUI を使用するために必要です。

- a. Java Web コンソールサービスが制限されているかどうかを調べます。

```
# svcprop /system/webconsole:console | grep tcp_listen
```

tcp\_listen プロパティの値が true でない場合、Web コンソールサービスは制限されます。

- b. Java Web Console サービスへの外部アクセスを復元します。

```
# svccfg
svc:> select system/webconsole
svc:/system/webconsole> setprop options/tcp_listen=true
svc:/system/webconsole> quit
# /usr/sbin/smcwebserver restart
```

- c. サービスが使用可能になっていることを確認します。

```
# netstat -a | grep 6789
```

サービスが使用可能な場合、コマンド出力により 6789 のエントリが返されます。これは Java Web コンソールへの接続に使用されるポート番号です。

- 注意事項
- この手順の実行後に Oracle Solaris Cluster Manager に接続できない場合、`/usr/sbin/smcwebserver status` を入力して、Sun Java Web コンソールが実行されているかどうかを調べます。Sun Java Web コンソールが実行されていない場合、`/usr/sbin/smcwebserver start` を入力して手動で起動します。それでも Oracle Solaris Cluster Manager に接続できない場合は、`usr/sbin/cacaoadm status` を入力して、共通エージェントコンテナが実行されているかどうかを調べます。共通エージェントコンテナが実行されていない場合は、`/usr/sbin/cacaoadm start` を入力して手動で起動します。

- GUI を実行しているノード以外のノードに関する情報を表示しようとしたときにシステムエラーメッセージが表示された場合は、共通エージェントコンテナのネットワークバインドアドレスパラメータが正しい値である `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
```



## Sun StorageTek Availability Suite ソフトウェアを使用した ホストベースのデータ複製の構成

この付録では、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、クラスタ内のホストベースの複製の構成と操作を簡素化することをお勧めします。90 ページの「データ複製についての理解」を参照してください。

この付録の例は、Sun StorageTek Availability Suite 4.0 ソフトウェアを使用してクラスタ間のホストベースのデータ複製を構成する方法を示しています。この例では、NFS アプリケーション用の完全なクラスタ構成を示し、個別のタスクの実行方法に関する詳細情報を提供します。すべてのタスクはグローバルクラスタの投票ノードで行われます。例には、ほかのアプリケーションやクラスタ構成で必要な手順がすべて含まれているわけではありません。

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、『Solaris のシステム管理 (セキュリティサービス)』を参照してください。各 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 Sun StorageTek Availability Suite ソフトウェアの理解

ここでは、耐障害性について紹介し、Sun StorageTek Availability Suite ソフトウェアが使用するデータ複製方式について説明します。

耐障害性は、主クラスタで障害が発生した場合に代わりのクラスタ上でアプリケーションを復元するシステムの機能です。災害耐性のベースは、データ複製とフェイルオーバーです。フェイルオーバーとは、主クラスタから二次クラスタへの、リソースグループまたはデバイスグループの自動再配置です。主クラスタに障害が発生した場合でも、アプリケーションとデータは二次クラスタで即座に使用できます。

### Sun StorageTek Availability Suite ソフトウェアが使用するデータ複製方式

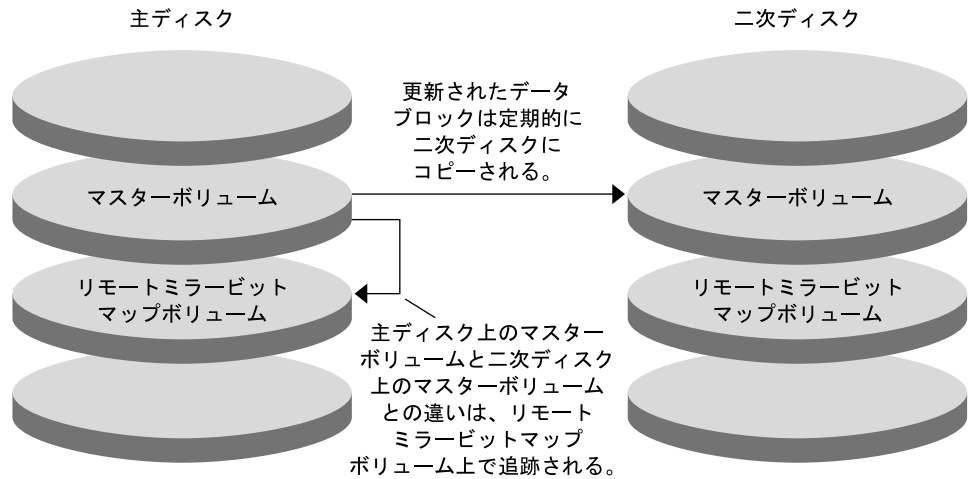
この節では、Sun StorageTek Availability Suite が使用するリモートミラー複製方式とポイントインタイムスナップショット方式について説明します。このソフトウェアは、`sndradm (1RPC)` と `iiadm(III)` コマンドを使用してデータを複製します。

#### リモートミラー複製

図 A-1 はリモートミラー複製を示しています。主ディスクのマスターボリュームのデータは、TCP/IP 接続を経由して二次ディスクのマスターボリュームに複製されます。リモートミラービットマップは、主ディスク上のマスターボリュームと、二次ディスク上のマスターボリュームの差分を追跡します。



図 A-1 リモートミラー複製



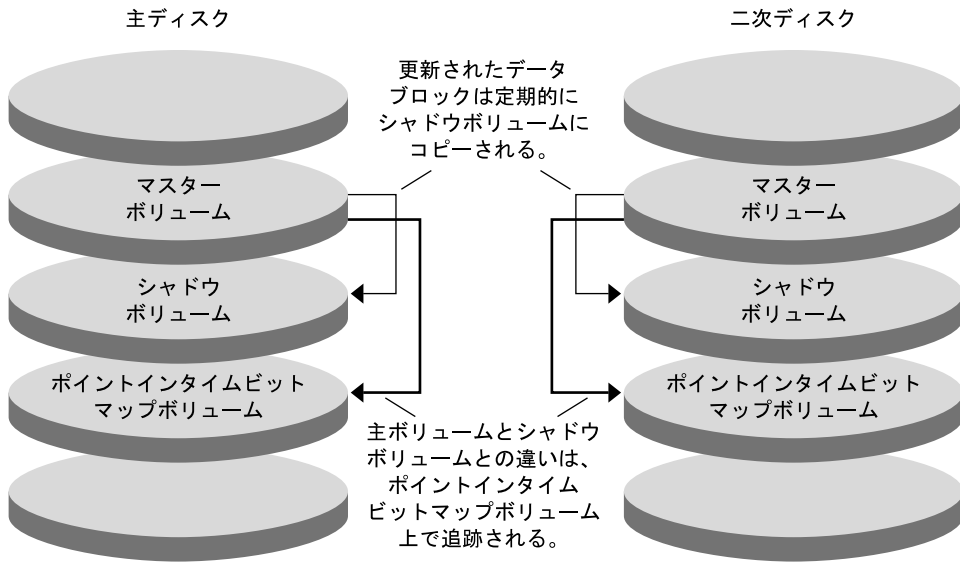
リモートミラー複製は、リアルタイムに同期で実行することも非同期で実行することもできます。各クラスタの各ボリュームセットはそれぞれ、同期複製または非同期複製に構成できます。

- 同期データ複製では、リモートボリュームが更新されるまで、書き込み操作は完了したとは確認されません。
- 非同期データ複製では、リモートボリュームが更新される前に書き込み操作が完了したと確認されます。非同期データ複製は、長い距離や低い帯域幅で大きな柔軟性を発揮します。

## ポイントインタイムスナップショット

図 A-2 は、ポイントインタイムスナップショットを示しています。各ディスクのマスターボリュームのデータは、同じディスクのシャドウボリュームにコピーされます。ポイントインタイムビットマップは、マスターボリュームとシャドウボリューム間の違いを追跡調査します。データがシャドウボリュームにコピーされると、ポイントインタイムビットマップはリセットされます。

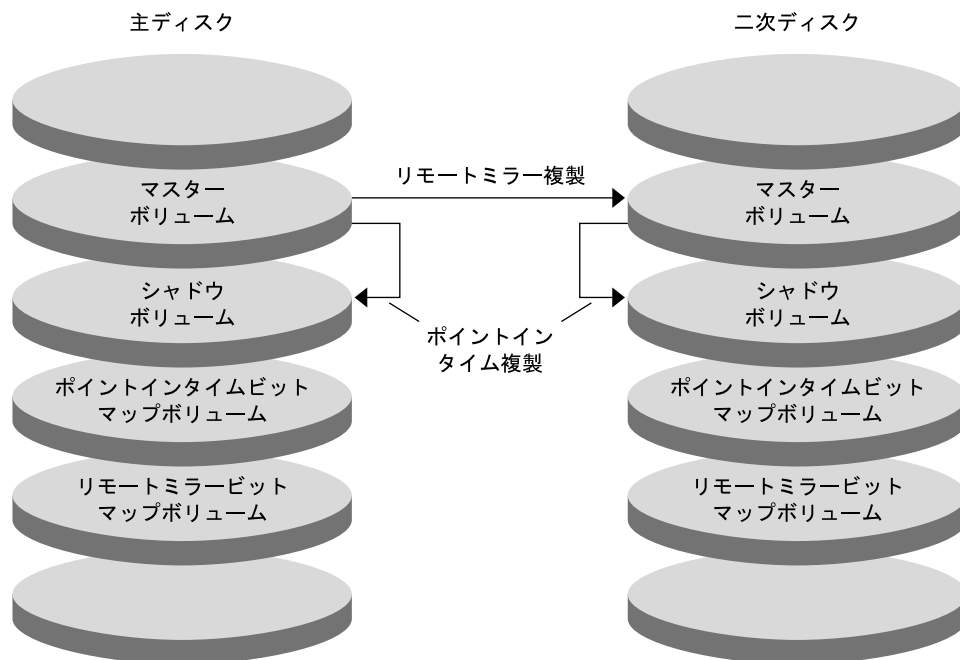
図A-2 ポイントインタイムスナップショット



### 構成例での複製

図A-3に、この構成例でミラー複製とポイントインタイムスナップショットがどのように使用されているかを示します。

図 A-3 構成例での複製



## クラスタ間でホストベースのデータ複製を構成するためのガイドライン

この節では、クラスタ間のデータ複製の構成ガイドラインを提供します。また、複製リソースグループとアプリケーションリソースグループの構成のコツも紹介します。これらのガイドラインは、クラスタのデータ複製を構成する際に使用してください。

この節では、次の項目について説明します。

- 380 ページの「複製リソースグループの構成」
- 381 ページの「アプリケーションリソースグループの構成」
  - 381 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」
  - 383 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」
- 384 ページの「フェイルオーバーの管理のガイドライン」

## 複製リソースグループの構成

複製リソースグループは、Sun StorageTek Availability Suite ソフトウェアが制御するデバイスグループと論理ホスト名リソースを相互に関連付けます。複製リソースグループには、次の特徴があります。

- フェイルオーバーリソースグループである  
フェイルオーバーリソースは、常に単一のノード上で実行されます。フェイルオーバーが発生すると、フェイルオーバーリソースがフェイルオーバーに加わります。
- 論理ホスト名リソースを持つ  
論理ホスト名は、主クラスタがホストでなければなりません。フェイルオーバーの後には、論理ホスト名は二次クラスタがホストでなければなりません。ドメインネームシステム (Domain Name System、DNS) は、論理ホスト名とクラスタを関連付けるために使用されます。

- HAStoragePlus リソースを持つ

HAStoragePlus リソースは、複製リソースグループがスイッチオーバーまたはフェイルオーバーしたときに、デバイスグループをフェイルオーバーします。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ースイッチオーバーと呼ばれます。
- *ZPoolsSearchDir*。この拡張プロパティは、ZFS ファイルシステムを使用するために必要です。

HAStoragePlus についての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 結び付いているデバイスグループに `-stor-rg` を付けた名前になる  
たとえば、`devgrp-stor-rg` などです。
- 主クラスタと二次クラスタでオンラインになる

## アプリケーションリソースグループの構成

高可用性を実現するためには、アプリケーションはアプリケーションリソースグループのリソースとして管理される必要があります。アプリケーションリソースグループは、フェイルオーバーアプリケーションまたはスケーラブルアプリケーション向けに構成できます。

主クラスタ上に構成したアプリケーションリソースとアプリケーションリソースグループは、二次クラスタ上でも構成される必要があります。また、アプリケーションリソースがアクセスするデータは、二次クラスタに複製する必要があります。

この節では、次のアプリケーションリソースグループを構成するためのガイドラインを紹介します。

- 381 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」
- 383 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 リソースにデバイスグループをフェイルオーバーさせる  
デバイスグループは、複製リソースグループとアプリケーションリソースグループに結び付けられています。したがって、アプリケーションリソースグループがフェイルオーバーすると、デバイスグループと複製リソースグループもフェイルオーバーします。アプリケーションリソースグループ、複製リソースグループおよびデバイスグループは、同じノードによって制御されます。  
ただし、デバイスグループや複製リソースグループがフェイルオーバーしても、アプリケーションリソースグループはフェイルオーバーを行いません。
- アプリケーションデータがグローバルマウントされている場合は、アプリケーションリソースグループに HAStoragePlus リソースを必ず入れなければならないわけではありませんが、入れることをお勧めします。
- アプリケーションデータがローカルマウントされている場合は、アプリケーションリソースグループに HAStoragePlus リソースを必ず入れなければならないなりません。

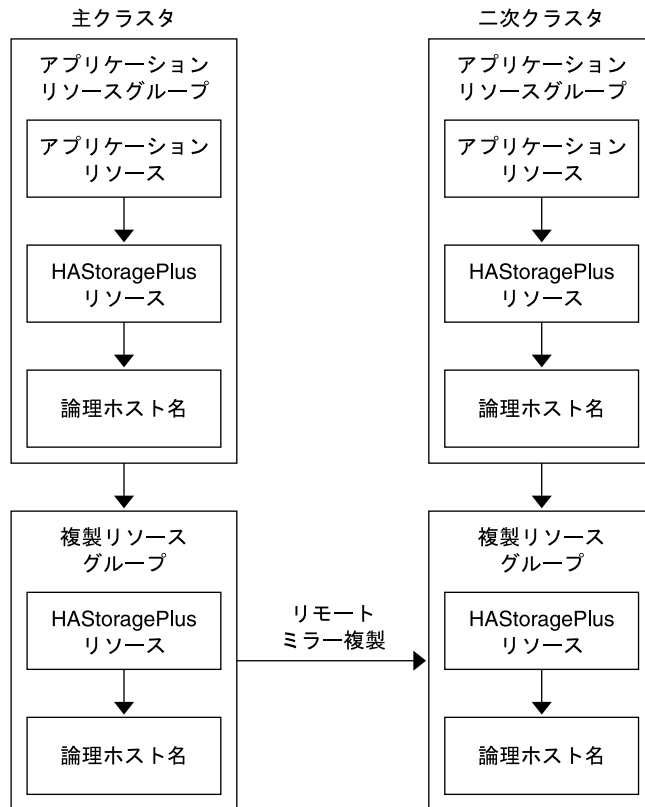
HAStoragePlus リソースがないと、アプリケーションリソースグループがフェイルオーバーしても、複製リソースグループとデバイスグループのフェイルオーバーは行われません。フェイルオーバーの後は、アプリケーションリソースグループ、複製リソースグループおよびデバイスグループは同じノードに制御されません。

HAStoragePlus についての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 主クラスタでオンライン、二次クラスタでオフラインとなる  
二次クラスタが主クラスタをテイクオーバーした場合は、二次クラスタ上のアプリケーションリソースグループをオンラインにします。

図 A-4 に、フェイルオーバーアプリケーションでのアプリケーションリソースグループと複製リソースグループの構成を示します。

図 A-4 フェイルオーバーアプリケーションでのリソースグループの構成



## スケーラブルアプリケーション向けのリソースグループの構成

スケーラブルアプリケーションでは、アプリケーションは複数のノードで実行されて、1つの論理サービスを作成します。スケーラブルアプリケーションを実行しているノードで障害が発生しても、フェイルオーバーは起こりません。アプリケーションは別のノードで引き続き実行されます。

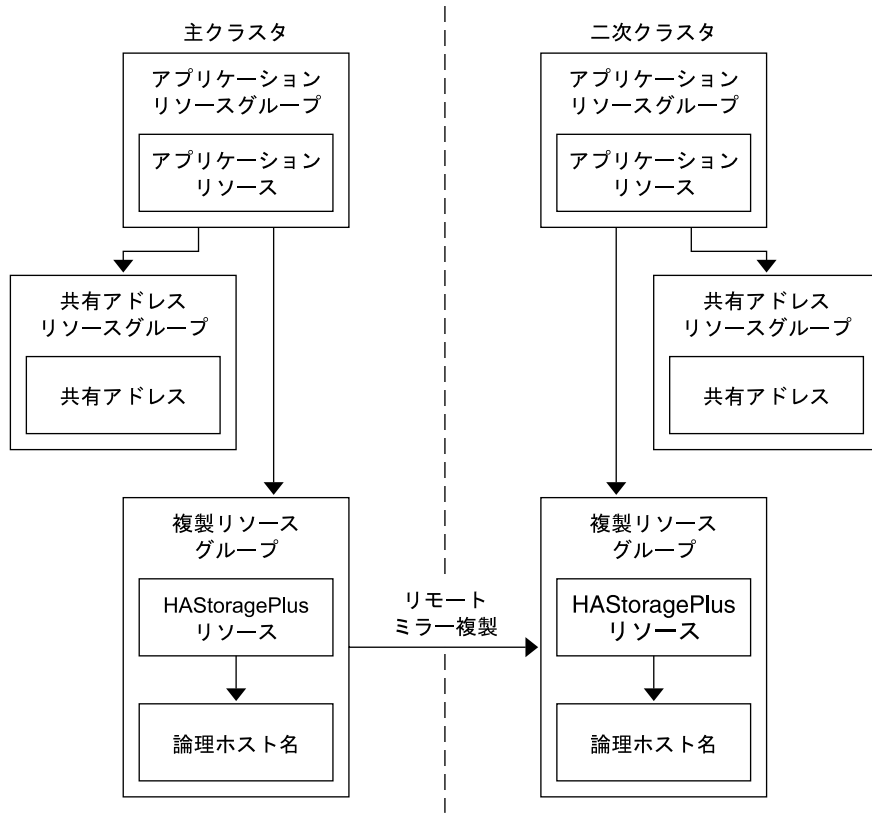
スケーラブルアプリケーションをアプリケーションリソースグループのリソースとして管理している場合は、アプリケーションリソースグループをデバイスグループと結び付ける必要はありません。したがって、アプリケーションリソースグループ向けに HAStoragePlus リソースを作成する必要はありません。

スケーラブルアプリケーション向けリソースグループは、以下の特徴を持っていないければなりません。

- 共有アドレスのリソースグループに依存する  
共有アドレスは、受信データを配信するためにスケーラブルアプリケーションを実行するノードで使用されます。
- 主クラスタでオンライン、二次クラスタでオフラインとなる

図 A-5 に、スケーラブルアプリケーションでのリソースグループの構成を示します。

図 A-5 スケーラブルアプリケーションでのリソースグループの構成



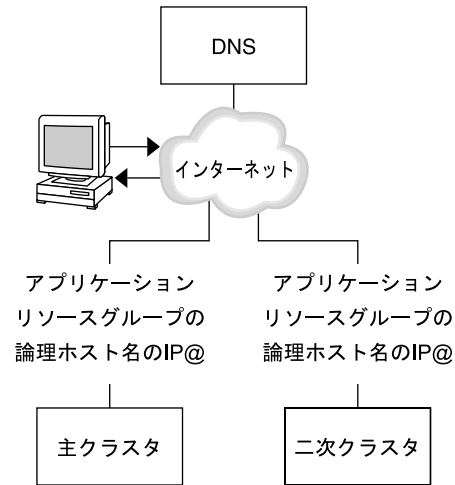
## フェイルオーバーの管理のガイドライン

主クラスタで障害が発生した場合、できるだけ速やかにアプリケーションを二次クラスタにスイッチオーバーする必要があります。二次クラスタがテイクオーバーできるようにするには、DNSを更新する必要があります。

DNSは、クライアントをアプリケーションの論理ホスト名に関連付けます。フェイルオーバーの後、主クラスタへのDNSマッピングを削除し、二次クラスタへのDNSマッピングを作成します。図 A-6 に、DNSがどのようにクライアントをクラスタにマッピングするかを示します。



図 A-6 クライアントからクラスタへの DNS マッピング



DNSを更新するには、`nsupdate` コマンドを使用します。詳細は、[nsupdate\(1M\)](#)のマニュアルページを参照してください。フェイルオーバーの管理方法の例については、[410 ページの「フェイルオーバーの管理方法の例」](#)を参照してください。

修復後は、主クラスタをオンラインに戻せます。元の主クラスタにスイッチバックするには、次の手順を実行します。

1. 主クラスタと二次クラスタを同期させ、主ボリュームが最新のものであることを確認します。
2. クライアントが主クラスタのアプリケーションにアクセスできるように、DNSを更新します。

## 作業マップ: データ複製の構成例

表 A-1 に、Sun StorageTek Availability Suite ソフトウェアを使用して NFS アプリケーション向けにどのようにデータ複製を構成するかを示すこの例での作業を示します。

表 A-1 作業マップ: データ複製の構成例

| 作業  | 参照先   |
|---|---|
| 1. クラスタを接続およびインストールする   | <a href="#">386 ページの「クラスタの接続とインストール」</a>        |
| 2. 主クラスタと二次クラスタで、デバイスグループ、NFS アプリケーション用のファイルシステム、およびリソースグループを構成する | <a href="#">388 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」</a> |

表 A-1 作業マップ: データ複製の構成例 (続き)

| 作業                          | 参照先  |
|-----------------------------|--|
| 3. 主クラスタと二次クラスタでデータ複製を有効にする | 402 ページの「主クラスタで複製を有効にする」<br>404 ページの「二次クラスタで複製を有効にする」        |
| 4. データ複製を実行する               | 405 ページの「リモートミラー複製を実行する」<br>406 ページの「ポイントインタイムスナップショットを実行する」 |
| 5. データ複製の構成を確認する            | 407 ページの「複製が正しく構成されていることを確認する」                               |

## クラスタの接続とインストール

図 A-7 に、構成例で使用するクラスタ構成を示します。構成例の二次クラスタにはノードが1つ含まれていますが、これ以外のクラスタ構成も使用できます。

図 A-7 クラスタ構成例

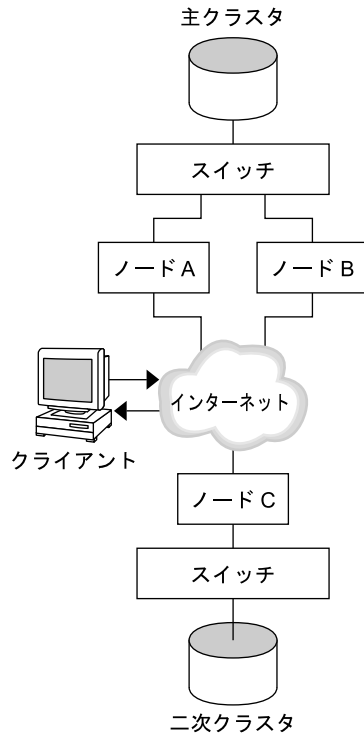


表 A-2 に、構成例で必要となるハードウェアとソフトウェアをまとめました。Oracle Solaris OS、Oracle Solaris Cluster ソフトウェア、ボリューム管理ソフトウェアは、Sun StorageTek Availability Suite ソフトウェアとパッチをインストールする前にクラスタノードにインストールしてください。

表 A-2 必要なハードウェアとソフトウェア

| ハードウェアまたはソフトウェア | 要件  |
|-----------------|---|
| ノードハードウェア       | Sun StorageTek Availability Suite ソフトウェアは、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。<br>使用するハードウェアについては、『 <a href="#">Oracle Solaris Cluster 3.3 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 3.3 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または Veritas Volume Manager (VxVM) 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」および『Oracle Solaris Cluster ソフトウェアのインストール』の第5章「Veritas Volume Manager をインストールして構成する」を参照してください。</p> |
| Sun StorageTek Availability Suite ソフトウェア    | <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 Sun StorageTek Availability Suite ソフトウェアのインストールマニュアルを参照してください。</p> <ul style="list-style-type: none"> <li>■ Sun StorageTek Availability Suite 4.0 – Sun StorageTek Availability のマニュアル</li> </ul>   |
| Sun StorageTek Availability Suite ソフトウェアパッチ | <p>最新のバッチ情報を閲覧するには、<a href="#">My Oracle Support</a> 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

この節では、NFS アプリケーション向けにディスクデバイスグループとリソースグループをどのように構成するかを説明します。追加情報については、380 ページの「複製リソースグループの構成」および381 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。

- 390 ページの「主クラスタでデバイスグループを構成する」
- 391 ページの「二次クラスタでデバイスグループを構成する」
- 392 ページの「主クラスタのファイルシステムを NFS アプリケーション向けに構成する」

- 393 ページの「二次クラスタのファイルシステムを NFS アプリケーション向けに構成する」
- 394 ページの「主クラスタで複製リソースグループを作成する」
- 396 ページの「二次クラスタで複製リソースグループを作成する」
- 397 ページの「主クラスタで NFS アプリケーションリソースグループを作成する」
- 400 ページの「二次クラスタで NFS アプリケーションリソースグループを作成する」
- 407 ページの「複製が正しく構成されていることを確認する」

構成例のために作成されたグループとリソースの名前を次の表に示します。

表 A-3 構成例内のグループとリソースのまとめ

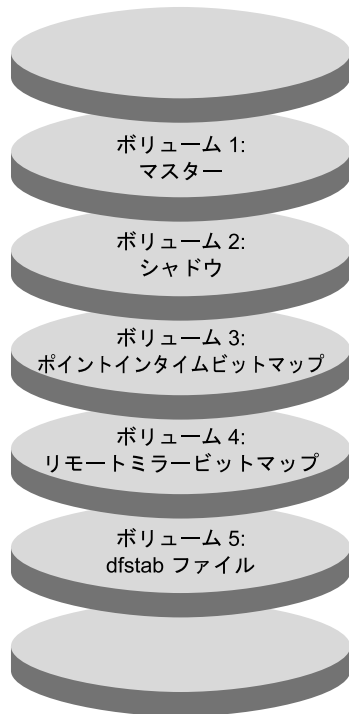
| グループまたはリソース                   | 名前                               | 説明                                   |
|-------------------------------|----------------------------------|--------------------------------------|
| デバイスグループ                      | devgrp                           | デバイスグループ                             |
| 複製リソースグループ<br>とリソース           | devgrp-stor-rg                   | 複製リソースグループ                           |
|                               | lhost-reprg-prim、lhost-reprg-sec | 主クラスタと二次クラスタの複製リソースグループの論理ホスト名       |
|                               | devgrp-stor                      | 複製リソースグループの HAStoragePlus リソース       |
| アプリケーションリ<br>ソースグループとリ<br>ソース | nfs-rg                           | アプリケーションリソースグループ                     |
|                               | lhost-nfsrg-prim、lhost-nfsrg-sec | 主クラスタと二次クラスタのアプリケーションリソースグループの論理ホスト名 |
|                               | nfs-dg-rs                        | アプリケーションの HAStoragePlus リソース         |
|                               | nfs-rs                           | NFS リソース                             |

devgrp-stor-rg 以外のグループとリソースの名前は一例で、必要に応じて変更可能です。複製リソースグループは、*devicegroupname-stor-rg* というフォーマットでなければなりません。

この構成例では VxVM ソフトウェアを使用しています。Solaris ボリュームマネージャ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ソフトウェアの構成」を参照してください。

デバイスグループで作成済みのボリュームを下図に示します。

図A-8 デバイスグループのボリューム



---

注-この手順で定義されたボリュームに、シリンダ0などのディスクラベルのプライベート領域を含めてはいけません。VxVMソフトウェアは、この制限を自動管理します。

---

## ▼ 主クラスタでデバイスグループを構成する

始める前に 次の作業を完成していることを確認してください。

- 次の節のガイドラインと要件を確認します。
  - [376 ページの「クラスタにおける Sun StorageTek Availability Suite ソフトウェアの理解」](#)
  - [379 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」](#)
- [386 ページの「クラスタの接続とインストール」](#)で説明されているように、主クラスタおよび二次クラスタを設定します。

- 1 **nodeA** にスーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になりますしてアクセスします。

**nodeA** は、主クラスタの最初のノードです。どのノードが **nodeA** であるかを確認するには、[図 A-7](#) を参照してください。

- 2 **nodeA** でボリューム **1 vol01** からボリューム **4 vol04** を含むディスクグループを作成します。

VxVM ソフトウェアを使用したディスクグループの構成については、『Oracle Solaris Cluster ソフトウェアのインストール』の第5章「Veritas Volume Manager をインストールして構成する」を参照してください。

- 3 ディスクグループを構成して、デバイスグループを作成します。

```
nodeA# cldevicegroup create -t vxvm -n nodeA nodeB devgrp
```

デバイスグループは `devgrp` と呼ばれます。

- 4 デバイスグループのファイルシステムを作成します。

```
nodeA# newfs /dev/vx/rdisk/devgrp/vol01 < /dev/null
nodeA# newfs /dev/vx/rdisk/devgrp/vol02 < /dev/null
```

`vol03` と `vol04` は raw ボリュームとして使用されるため、ファイルシステムは必要ありません。

次の手順 [391 ページ](#) の「二次クラスタでデバイスグループを構成する」に進みます。

## ▼ 二次クラスタでデバイスグループを構成する

始める前に [手順390 ページ](#) の「主クラスタでデバイスグループを構成する」を完了します。

- 1 **nodeC** にスーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` を提供する役割になりますしてアクセスします。

- 2 **nodeC** でボリューム **1 vol01** からボリューム **4 vol04** までの4つのボリュームを含むディスクグループを作成します。

- 3 ディスクグループを構成して、デバイスグループを作成します。

```
nodeC# cldevicegroup create -t vxvm -n nodeC devgrp
```

デバイスグループは `devgrp` という名前です。

- 4 デバイスグループのファイルシステムを作成します。

```
nodeC# newfs /dev/vx/rdisk/devgrp/vol01 < /dev/null
nodeC# newfs /dev/vx/rdisk/devgrp/vol02 < /dev/null
```

`vol03` と `vol04` は raw ボリュームとして使用されるため、ファイルシステムは必要ありません。

次の手順 392 ページの「主クラスタのファイルシステムを NFS アプリケーション向けに構成する」に進みます。

## ▼ 主クラスタのファイルシステムを NFS アプリケーション向けに構成する

始める前に 手順391 ページの「二次クラスタでデバイスグループを構成する」を完了します。

- 1 **nodeA** および **nodeB** で、スーパーユーザーになるか、**RBAC** の承認 **solaris.cluster.admin** を提供する役割になりますになります。
- 2 **nodeA** と **nodeB** で、**NFS** ファイルシステム向けのマウントポイントディレクトリを作成します。  
たとえば、次のように使用します。  

```
nodeA# mkdir /global/mountpoint
```
- 3 **nodeA** と **nodeB** で、マウントポイントに自動でマウントされるようにマスターボリュームを構成します。  
**nodeA** と **nodeB** の `/etc/vfstab` ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは 1 行で記述してください。  

```
/dev/vx/dsk/devgrp/vol01 /dev/vx/rdisk/devgrp/vol01 \  
/global/mountpoint ufs 3 no global,logging
```

  
デバイスグループで使用されているボリューム名とボリューム番号を確認するには、[図 A-8](#) を参照してください。
- 4 **nodeA** で、**Oracle Solaris Cluster HA for NFS** データサービスが使用するファイルのシステム情報向けのボリュームを作成します。  

```
nodeA# vxassist -g devgrp make vol05 120m disk1
```

  
ボリューム 5 **vol05** には **Oracle Solaris Cluster HA for NFS** データサービスが使用するファイルシステム情報が含まれています。
- 5 **nodeA** で、デバイスグループと **Oracle Solaris Cluster** ソフトウェアを再同期化します。  

```
nodeA# cldevicegroup sync devgrp
```
- 6 **nodeA** で、**vol05** 用のファイルシステムを作成します。  

```
nodeA# newfs /dev/vx/rdsk/devgrp/vol05
```
- 7 **nodeA** と **nodeB** で、**vol05** のマウントポイントを作成します。  
次の例では、マウントポイント `/global/etc` を作成しています。  

```
nodeA# mkdir /global/etc
```



- 8 **nodeA** と **nodeB** で、マウントポイントに自動でマウントされるように **vol05** を構成します。

**nodeA** と **nodeB** の `/etc/vfstab` ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは1行で記述してください。

```
/dev/vx/dsk/devgrp/vol05 /dev/vx/rdisk/devgrp/vol05 \  
/global/etc ufs 3 yes global,logging
```

- 9 **nodeA** に **vol05** をマウントします。

```
nodeA# mount /global/etc
```

- 10 **vol05** がリモートシステムからアクセスできるようにします。

- a. **nodeA** に `/global/etc/SUNW.nfs` というディレクトリを作成します。

```
nodeA# mkdir -p /global/etc/SUNW.nfs
```

- b. **nodeA** に `/global/etc/SUNW.nfs/dfstab.nfs-rs` ファイルを作成します。

```
nodeA# touch /global/etc/SUNW.nfs/dfstab.nfs-rs
```

- c. **nodeA** の `/global/etc/SUNW.nfs/dfstab.nfs-rs` ファイルに次の行を追加します。

```
share -F nfs -o rw -d "HA NFS" /global/mountpoint
```

次の手順 393 ページの「二次クラスタのファイルシステムを NFS アプリケーション向けに構成する」に進みます。

## ▼ 二次クラスタのファイルシステムを NFS アプリケーション向けに構成する

始める前に 手順392 ページの「主クラスタのファイルシステムを NFS アプリケーション向けに構成する」を完了します。

- 1 **nodeC** で、スーパーユーザーになるか、RBAC の承認 `solaris.cluster.admin` を提供する役割になります。

- 2 **nodeC** で、NFS ファイルシステム向けのマウントポイントディレクトリを作成します。

たとえば、次のように使用します。

```
nodeC# mkdir /global/mountpoint
```

- 3 **nodeC** で、マウントポイントに自動でマウントされるようにマスターボリュームを構成します。

**nodeC** の `/etc/vfstab` ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは1行で記述してください。

```
/dev/vx/dsk/devgrp/vol01 /dev/vx/rdisk/devgrp/vol01 \  
/global/mountpoint ufs 3 no global,logging
```

- 4 **nodeC**で、**Oracle Solaris Cluster HA for NFS** データサービスが使用するファイルのシステム情報向けのボリュームを作成します。

```
nodeC# vxassist -g devgrp make vol05 120m disk1
```

ボリューム 5 vol05 には Oracle Solaris Cluster HA for NFS データサービスが使用するファイルシステム情報が含まれています。

- 5 **nodeC**で、デバイスグループと **Oracle Solaris Cluster** ソフトウェアを再同期化します。

```
nodeC# cldevicegroup sync devgrp
```

- 6 **nodeC**で、**vol05** 用のファイルシステムを作成します。

```
nodeC# newfs /dev/vx/rdisk/devgrp/vol05
```

- 7 **nodeC**で、**vol05** 用のマウントポイントを作成します。

次の例では、マウントポイント /global/etc を作成しています。

```
nodeC# mkdir /global/etc
```

- 8 **nodeC**で、**vol05** がマウントポイントで自動的にマウントされるよう構成します。

**nodeC** の /etc/vfstab ファイルに以下のテキストを追加するか、既存のテキストと置き換えます。テキストは 1 行で記述してください。

```
/dev/vx/dsk/devgrp/vol05 /dev/vx/rdisk/devgrp/vol05 \  
/global/etc ufs 3 yes global,logging
```

- 9 **nodeC** に **vol05** をマウントします。

```
nodeC# mount /global/etc
```

- 10 **vol05** がリモートシステムからアクセスできるようにします。

- a. **nodeC** に /global/etc/SUNW.nfs というディレクトリを作成します。

```
nodeC# mkdir -p /global/etc/SUNW.nfs
```

- b. **nodeC** に /global/etc/SUNW.nfs/dfstab.nfs-rs ファイルを作成します。

```
nodeC# touch /global/etc/SUNW.nfs/dfstab.nfs-rs
```

- c. **nodeC** の /global/etc/SUNW.nfs/dfstab.nfs-rs ファイルに次の行を追加します。

```
share -F nfs -o rw -d "HA NFS" /global/mountpoint
```

次の手順 [394 ページ](#)の「主クラスタで複製リソースグループを作成する」に進みます。

## ▼ 主クラスタで複製リソースグループを作成する

始める前に

手順 [393 ページ](#)の「二次クラスタのファイルシステムを NFS アプリケーション向けに構成する」を完了します。

- 1 **nodeA** にスーパーユーザーになるか、RBACの承認  
**solaris.cluster.modify**、**solaris.cluster.admin**、および**solaris.cluster.read**を提供する役割になりますしてアクセスします。

- 2 **SUNW.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、devgrp はデバイスグループの名前を指定します。

- 4 複製リソースグループに **SUNW.HASStoragePlus** リソースを追加します。

```
nodeA# clresource create -g devgrp-stor-rg -t SUNW.HASStoragePlus \
-p GlobalDevicePaths=devgrp \
-p AffinityOn=True \
devgrp-stor
```

-g リソースを追加するリソースグループを指定します。

-p GlobalDevicePaths= Sun StorageTek Availability Suite ソフトウェアが依存する拡張プロパティを指定します。

-p AffinityOn=True **SUNW.HASStoragePlus** リソースが、-x GlobalDevicePaths=で定義されたグローバルデバイスおよびクラスタファイルシステムに対して、アフィニティスイッチオーバーを実行することを指定します。したがって、複製リソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

これらの拡張プロパティについての詳細は、[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
```

主クラスタ上の複製リソースグループの論理ホスト名は lhost-reprg-prim です。

- 6 リソースを有効にし、リソースグループを管理し、リソースグループをオンラインにします。

```
nodeA# clresourcegroup online -e -M -n nodeA devgrp-stor-rg
```

-e 関連付けられたリソースを有効にします。

-M リソースグループを管理状態にします。

-n リソースグループをオンラインにするノードを指定します。

- 7 リソースグループがオンラインであることを確認します。

```
nodeA# clresourcegroup status devgrp-stor-rg
```

リソースグループの状態フィールドを調べ、複製リソースグループが nodeA でオンラインとなっていることを確認します。

次の手順 [396 ページの「二次クラスタで複製リソースグループを作成する」](#)に進みます。

## ▼ 二次クラスタで複製リソースグループを作成する

始める前に 手順[394 ページの「主クラスタで複製リソースグループを作成する」](#)を完了します。

- 1 nodeC にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割になりますしてアクセスします。

- 2 `SUNW.HAStoragePlus` というリソースタイプを登録します。

```
nodeC# clresourcetype register SUNW.HAStoragePlus
```

- 3 デバイスグループの複製リソースグループを作成します。

```
nodeC# clresourcegroup create -n nodeC devgrp-stor-rg
```

create リソースグループを作成します。

-n リソースグループのノードリストを指定します。

devgrp デバイスグループの名前。

devgrp-stor-rg 複製リソースグループの名前。

- 4 複製リソースグループに `SUNW.HAStoragePlus` リソースを追加します。

```
nodeC# clresource create \  
-t SUNW.HAStoragePlus \  
-p GlobalDevicePaths=devgrp \  
-p AffinityOn=True \  
devgrp-stor
```

create リソースを作成します。

-t リソースタイプを指定します。

-p `GlobalDevicePaths=` Sun StorageTek Availability Suite ソフトウェアが依存する拡張プロパティを指定します。

-p `AffinityOn=True` `SUNW.HAStoragePlus` リソースが、`-x GlobalDevicePaths=` で定義されたグローバルデバイスおよびクラスタファイルシステムに対して、アフィニティスイッチオーバーを実

行することを指定します。したがって、複製リソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

`devgrp-stor` 複製リソースグループの `HAStoragePlus` リソース

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 5 論理ホスト名リソースを複製リソースグループに追加します。

```
nodeC# clreslogicalhostname create -g devgrp-stor-rg lhost-reprg-sec
```

主クラスタ上の複製リソースグループの論理ホスト名は `lhost-reprg-sec` です。

- 6 リソースを有効にし、リソースグループを管理し、リソースグループをオンラインにします。

```
nodeC# clresourcegroup online -e -M -n nodeC devgrp-stor-rg
```

`onLine` オンラインにします。

`-e` 関連付けられたリソースを有効にします。

`-M` リソースグループを管理状態にします。

`-n` リソースグループをオンラインにするノードを指定します。

- 7 リソースグループがオンラインであることを確認します。

```
nodeC# clresourcegroup status devgrp-stor-rg
```

リソースグループの状態フィールドを調べ、複製リソースグループが `nodeC` でオンラインとなっていることを確認します。

次の手順 [397](#) ページの「主クラスタで NFS アプリケーションリソースグループを作成する」に進みます。

## ▼ 主クラスタで NFS アプリケーションリソースグループを作成する

この手順では、アプリケーションリソースグループを NFS に対して作成する方法を説明します。この手順はこのアプリケーションに固有で、別の種類のアプリケーションには使用できません。

始める前に 手順 [396](#) ページの「二次クラスタで複製リソースグループを作成する」を完了します。

- 1 `nodeA` にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.modify`、`solaris.cluster.admin`、および `solaris.cluster.read` を提供する役割になりますしてアクセスします。

- 2 **SUNW.nfs** をリソースタイプとして登録します。

```
nodeA# clresourcetype register SUNW.nfs
```

- 3 **SUNW.HASStoragePlus** をリソースタイプとして登録していない場合は、登録します。

```
nodeA# clresourcetype register SUNW.HASStoragePlus
```

- 4 デバイスグループ **devgrp** のアプリケーションリソースグループを作成します。

```
nodeA# clresourcegroup create \  
-p Pathprefix=/global/etc \  
-p Auto_start_on_new_cluster=False \  
-p RG_dependencies=devgrp-stor-rg \  
nfs-rg
```

Pathprefix=/global/etc

グループのリソースが管理ファイルを書き込むディレクトリを指定します。

Auto\_start\_on\_new\_cluster=False

アプリケーションリソースグループが自動的に起動しないように指定します。

RG\_dependencies=devgrp-stor-rg

アプリケーションリソースグループが依存するリソースグループを指定します。この例では、アプリケーションリソースグループは複製リソースグループ devgrp-stor-rg に依存しています。

アプリケーションリソースグループが新しい主ノードにスイッチオーバーすると、複製リソースグループが自動的にスイッチオーバーします。ただし、複製リソースグループが新しい主ノードにスイッチオーバーした場合は、アプリケーションリソースグループを手動でスイッチオーバーする必要があります。

nfs-rg

アプリケーションリソースグループの名前。

- 5 アプリケーションリソースグループに **SUNW.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/

ファイルシステムのマウントポイントがグローバルであることを指定します。

-p AffinityOn=True

アプリケーションリソースが -p GlobalDevicePaths= で定義されたグローバルデバイスとクラスタファイルシステム向けにアフィニティスイッチオーバーを実行するように指定します。したがって、アプリケーションリソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

nfs-dg-rs

NFSアプリケーション向けの HAStoragePlus リソースの名前。

これらの拡張プロパティについての詳細は、[SUNW.HAStoragePlus\(5\)](#) のマニュアルページを参照してください。

## 6 論理ホスト名リソースをアプリケーションリソースグループに追加します。

```
nodeA# clreslogicalhostname create -g nfs-rg \  
lhost-nfsrg-prim
```

主クラスタ上のアプリケーションリソースグループの論理ホスト名は lhost-nfsrg-prim です。

## 7 リソースを有効にし、アプリケーションリソースグループを管理し、アプリケーションリソースグループをオンラインにします。

### a. NFSアプリケーション向けの HAStoragePlus リソースを有効にします。

```
nodeA# clresource enable nfs-rs
```

### b. nodeA でアプリケーションリソースグループをオンラインにします。

```
nodeA# clresourcegroup online -e -M -n nodeA nfs-rg
```

online    リソースグループをオンラインにします。

-e        関連付けられたリソースを有効にします。

-M        リソースグループを管理状態にします。

-n        リソースグループをオンラインにするノードを指定します。

nfs-rg    リソースグループの名前。

## 8 アプリケーションリソースグループがオンラインであることを確認します。

```
nodeA# clresourcegroup status
```

アプリケーションリソースグループの状態フィールドを調べ、複製リソースグループが nodeA と nodeB でオンラインとなっているかどうかを調べます。

次の手順 [400 ページの「二次クラスタで NFS アプリケーションリソースグループを作成する」](#)に進みます。

## ▼ 二次クラスタで NFS アプリケーションリソースグループを作成する

始める前に 手順397 ページの「主クラスタで NFS アプリケーションリソースグループを作成する」を完了します。

- 1 **nodeC** にスーパーユーザーになるか、RBAC の承認 **solaris.cluster.modify**、**solaris.cluster.admin**、および **solaris.cluster.read** を提供する役割になりますしてアクセスします。

- 2 **SUNW.nfs** をリソースタイプとして登録します。

```
nodeC# clresourcetype register SUNW.nfs
```

- 3 **SUNW.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_dependencies=devgrp-stor-rg \  
nfs-rg
```

```
create
```

リソースグループを作成します。

```
-p
```

リソースグループのプロパティを指定します。

```
Pathprefix=/global/etc
```

グループのリソースが管理ファイルを書き込むディレクトリを指定します。

```
Auto_start_on_new_cluster=False
```

アプリケーションリソースグループが自動的に起動しないように指定します。

```
RG_dependencies=devgrp-stor-rg
```

アプリケーションリソースグループが依存するリソースグループを指定します。この例では、アプリケーションリソースグループは複製リソースグループに依存しています。

アプリケーションリソースグループが新しい主ノードにスイッチオーバーすると、複製リソースグループが自動的にスイッチオーバーします。ただし、複製リソースグループが新しい主ノードにスイッチオーバーした場合は、アプリケーションリソースグループを手動でスイッチオーバーする必要があります。

```
nfs-rg
```

アプリケーションリソースグループの名前。



- 5 アプリケーションリソースグループに **SUNW.HASStoragePlus** リソースを追加します。

```
nodeC# clresource create -g nfs-rg \  
-t SUNW.HASStoragePlus \  
-p FileSystemMountPoints=/global/mountpoint \  
-p AffinityOn=True \  
nfs-dg-rs
```

create

リソースを作成します。

-g

リソースを追加するリソースグループを指定します。

-t SUNW.HASStoragePlus

リソースのタイプに **SUNW.HASStoragePlus** を指定します。

-p

リソースのプロパティを指定します。

FileSystemMountPoints=/global/

ファイルシステムのマウントポイントがグローバルであることを指定します。

AffinityOn=True

アプリケーションリソースが `-x GlobalDevicePaths=` で定義されたグローバルデバイスとクラスタファイルシステム向けにアフィニティスイッチオーバーを実行するように指定します。したがって、アプリケーションリソースグループがフェイルオーバーまたはスイッチオーバーすると、関連デバイスグループがスイッチオーバーします。

nfs-dg-rs

NFS アプリケーション向けの **HASStoragePlus** 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.HASStoragePlus\(5\)](#) のマニュアルページを参照してください。

- 6 論理ホスト名リソースをアプリケーションリソースグループに追加します。

```
nodeC# clreslogicalhostname create -g nfs-rg \  
lhost-nfsrg-sec
```

二次クラスタ上のアプリケーションリソースグループの論理ホスト名は `lhost-nfsrg-sec` です。

- 7 **NFS** リソースをアプリケーションリソースグループに追加します。

```
nodeC# clresource create -g nfs-rg \  
-t SUNW.nfs -p Resource_dependencies=nfs-dg-rs nfs-rg
```

- 8 アプリケーションリソースグループが **nodeC** でオンラインになっていないことを確認します。

```
nodeC# clresource disable -n nodeC nfs-rs  
nodeC# clresource disable -n nodeC nfs-dg-rs
```

```
nodeC# clresource disable -n nodeC lhost-nfsrg-sec
nodeC# clresourcegroup online -n "" nfs-rg
```

`Auto_start_on_new_cluster=False` によって、リソースグループは再起動後もオフラインのままになります。

- 9 グローバルボリュームが主クラスタにマウントされている場合は、二次クラスタのグローバルボリュームのマウントを解除します。

```
nodeC# umount /global/mountpoint
```

ボリュームが二次クラスタにマウントされていると、同期が失敗します。

次の手順 402 ページの「データ複製の有効化例」に進みます。

## データ複製の有効化例

この節では、構成例のデータ複製をどのように有効にするかを説明します。この節では、Sun StorageTek Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、Sun StorageTek Availability のマニュアルを参照してください。

ここでは、次の手順について説明します。

- 402 ページの「主クラスタで複製を有効にする」
- 404 ページの「二次クラスタで複製を有効にする」

### ▼ 主クラスタで複製を有効にする

- 1 `nodeA` にスーパーユーザーになるか、RBAC の承認 `solaris.cluster.read` を提供する役割割になりますしてアクセスします。

- 2 すべてのトランザクションをフラッシュします。

```
nodeA# lockfs -a -f
```

- 3 論理ホスト名 `lhost-reprg-prim` と `lhost-reprg-sec` がオンラインであることを確認します。

```
nodeA# clresourcegroup status
nodeC# clresourcegroup status
```

リソースグループの状態フィールドを調べます。

- 4 主クラスタから二次クラスタへのリモートミラー複製を有効にします。

この手順によって、主クラスタのマスターボリュームから二次クラスタのマスターボリュームへの複製が有効になります。さらに、`vol04` のリモートミラービットマップへの複製も有効になります。

- 主クラスタと二次クラスタが同期されていない場合は、次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -e lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- 主クラスタと二次クラスタが同期されている場合は、次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -E lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

## 5 自動同期機能を有効にします。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -a on lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

この手順で自動同期が有効になります。自動同期のアクティブ状態が on に設定されている場合、システムが再起動されたり障害が発生すると、ボリュームセットは再度同期化されます。

## 6 クラスタがロギングモードであることを確認します。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01
autosync: off, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: logging
```

ロギングモードでは、状態は logging で、自動同期のアクティブ状態は off です。ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。

## 7 ポイントインタイムスナップショットを有効にします。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/iiadm -e ind \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
nodeA# /usr/sbin/iiadm -w \
/dev/vx/rdisk/devgrp/vol02
```

この手順によって、主クラスタのマスターボリュームが同じクラスタのシャドウボリュームにコピーされるようになります。マスターボリューム、シャドウボリューム、およびポイントインタイムビットマップボリュームは同じデバイスグループに存在する必要があります。この例では、マスターボリュームは vol01、シャドウボリュームは vol02、ポイントインタイムビットマップボリュームは vol03 になります。

- 8 ポイントインタイムスナップショットをリモートミラーセットに設定します。  
次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -I a \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol02 \  
/dev/vx/rdisk/devgrp/vol03
```

この手順によって、ポイントインタイムスナップショットがリモートミラーボリュームセットに関連付けられます。Sun StorageTek Availability Suite ソフトウェアは、リモートミラー複製の前にポイントインタイムスナップショットを必ず取りま

次の手順 [404 ページの「二次クラスタで複製を有効にする」](#)に進みます。

## ▼ 二次クラスタで複製を有効にする

始める前に [手順402 ページの「主クラスタで複製を有効にする」](#)を完了します。

- 1 スーパーユーザーとして **nodeC** にアクセスします。
- 2 すべてのトランザクションをフラッシュします。

```
nodeC# lockfs -a -f
```

- 3 主クラスタから二次クラスタへのリモートミラー複製を有効にします。  
次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -n -e lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync
```

主クラスタが二次クラスタの存在を認識し、同期を開始します。クラスタの状態については、Sun 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。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/iiadm -e ind \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol02 \  

```

```
/dev/vx/rdisk/devgrp/vol03
nodeC# /usr/sbin/iiadm -w \
/dev/vx/rdisk/devgrp/vol02
```

- 5 ポイントインタイムスナップショットをリモートミラーセットに設定します。  
次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -I a \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol02 \
/dev/vx/rdisk/devgrp/vol03
```

次の手順 405 ページの「データ複製の実行例」に進みます。

## データ複製の実行例

この節では、構成例のデータ複製をどのように実行するかを説明します。この節では、Sun StorageTek Availability Suite ソフトウェアコマンドの `sndradm` と `iiadm` を使用します。これらのコマンドの詳細は、Sun StorageTek Availability Suite のマニュアル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。

- 405 ページの「リモートミラー複製を実行する」
- 406 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」
- 407 ページの「複製が正しく構成されていることを確認する」

### ▼ リモートミラー複製を実行する

この手順では、主ディスクのマスターボリュームが二次ディスクのマスターボリュームに複製されます。マスターボリュームは `vol01` で、リモートミラービットマップボリュームは `vol04` です。

- 1 スーパーユーザーとして `nodeA` にアクセスします。
- 2 クラスタがロギングモードであることを確認します。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01
autosync: off, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: logging
```

ロギングモードでは、状態は `logging` で、自動同期のアクティブ状態は `off` です。ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。

- 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
```

- nodeCで手順1から手順3を繰り返します。

- 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。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -m lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- 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。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -w lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync
```

- 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。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->  
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01  
autosync: on, max q writes:4194304, max q fbas:16384, mode:sync,ctag:  
devgrp, state: replicating
```

複製モードでは、状態は `replicating` で、自動同期のアクティブ状態は `on` です。主ボリュームに書き込みが行われると、Sun StorageTek Availability Suite ソフトウェアが二次ボリュームを更新します。

次の手順 [406](#) ページの「ポイントインタイムスナップショットを実行する」に進みます。

## ▼ ポイントインタイムスナップショットを実行する

この手順では、ポイントインタイムスナップショットを使用して、主クラスタのシャドウボリュームを主クラスタのマスターボリュームに同期させます。マスターボリュームは `vol01`、ビットマップボリュームは `vol04`、シャドウボリュームは `vol02` です。

始める前に 手順[405](#) ページの「リモートミラー複製を実行する」を完了します。

- 1 **nodeA** にスーパーユーザーになるか、RBACの承認 `solaris.cluster.modify` および `solaris.cluster.admin` を提供する役割になりますしてアクセスします。

- 2 **nodeA** で実行されているリソースを無効にします。

```
nodeA# clresource disable -n nodeA nfs-rs
```

- 3 主クラスタをロギングモードに変更します。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync
```

ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。複製は行われません。

- 4 主クラスタのシャドウボリュームを主クラスタのマスターボリュームに同期化させます。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/iiadm -u s /dev/vx/rdisk/devgrp/vol02  
nodeA# /usr/sbin/iiadm -w /dev/vx/rdisk/devgrp/vol02
```

- 5 二次クラスタのシャドウボリュームを二次クラスタのマスターボリュームに同期化させます。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/iiadm -u s /dev/vx/rdisk/devgrp/vol02  
nodeC# /usr/sbin/iiadm -w /dev/vx/rdisk/devgrp/vol02
```

- 6 **nodeA** でアプリケーションを再起動します。

```
nodeA# clresource enable -n nodeA nfs-rs
```

- 7 二次ボリュームを主ボリュームと再同期化させます。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync
```

次の手順 407 ページの「複製が正しく構成されていることを確認する」に進みます。

## ▼ 複製が正しく構成されていることを確認する

始める前に 手順406 ページの「ポイントインタイムスナップショットを実行する」を完了します。

- 1 **nodeA** および **nodeC** にスーパーユーザーになるか、**RBAC**の承認 **solaris.cluster.admin** を提供する役割になりますを使用してアクセスします。
- 2 主クラスタが複製モードで、自動同期機能がオンになっていることを確認します。  
次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:  

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。  

```
/dev/vx/rdisk/devgrp/vol01 ->
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01
autosync: on, max q writes:4194304, max q fbas:16384, mode:sync,ctag:
devgrp, state: replicating
```

複製モードでは、状態は **replicating** で、自動同期のアクティブ状態は **on** です。主ボリュームに書き込みが行われると、Sun StorageTek Availability Suite ソフトウェアが二次ボリュームを更新します。
- 3 主クラスタが複製モードでない場合は、複製モードにします。  
次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:  

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```
- 4 クライアントマシンにディレクトリを作成します。
  - a. クライアントマシンにスーパーユーザーとしてログインします。  
次のようなプロンプトが表示されます。  

```
client-machine#
```
  - b. クライアントマシンにディレクトリを作成します。  

```
client-machine# mkdir /dir
```
- 5 ディレクトリを主クラスタのアプリケーションにマウントし、マウントしたディレクトリを表示します。
  - a. ディレクトリを主クラスタのアプリケーションにマウントします。  

```
client-machine# mount -o rw lhost-nfsrg-prim:/global/mountpoint /dir
```
  - b. マウントしたディレクトリを表示します。  

```
client-machine# ls /dir
```



- 6 ディレクトリを二次クラスタのアプリケーションにマウントし、マウントしたディレクトリを表示します。
- 主クラスタのアプリケーションからディレクトリのマウントを解除します。  

```
client-machine# umount /dir
```
  - 主クラスタのアプリケーションリソースグループをオフラインにします。  

```
nodeA# clresource disable -n nodeA nfs-rs
nodeA# clresource disable -n nodeA nfs-dg-rs
nodeA# clresource disable -n nodeA lhost-nfsrg-prim
nodeA# clresourcegroup online -n "" nfs-rg
```
  - 主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。Sun StorageTek Availability Suite ソフトウェアの場合:  

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

 ディスクのデータボリュームに書き込みが行われると、同じディスクのビットマップファイルが更新されます。複製は行われません。
  - 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
```
  - 二次クラスタのアプリケーションリソースグループをオンラインにします。  

```
nodeC# clresourcegroup online -n nodeC nfs-rg
```
  - 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#
```
  - 手順 4** で作成したディレクトリを二次クラスタのアプリケーションにマウントします。  

```
client-machine# mount -o rw lhost-nfsrg-sec:/global/mountpoint /dir
```
  - マウントしたディレクトリを表示します。  

```
client-machine# ls /dir
```
- 7 **手順 5** で表示されたディレクトリが**手順 6** で表示されたディレクトリと同じであることを確認します。

- 8 主クラスタのアプリケーションをマウントされたディレクトリに戻します。
- a. 二次クラスタのアプリケーションリソースグループをオフラインにします。

```
nodeC# clresource disable -n nodeC nfs-rs
nodeC# clresource disable -n nodeC nfs-dg-rs
nodeC# clresource disable -n nodeC lhost-nfsrg-sec
nodeC# clresourcegroup online -n "" nfs-rg
```

- b. グローバルボリュームを二次クラスタからマウント解除します。

```
nodeC# umount /global/mountpoint
```

- c. 主クラスタのアプリケーションリソースグループをオンラインにします。

```
nodeA# clresourcegroup online -n nodeA nfs-rg
```

- d. 主クラスタを複製モードに変更します。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -u lhost-reprg-prim \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \
/dev/vx/rdisk/devgrp/vol01 \
/dev/vx/rdisk/devgrp/vol04 ip sync
```

主ボリュームに書き込みが行われると、Sun StorageTek Availability Suite ソフトウェアが二次ボリュームを更新します。

参照 [410 ページの「フェイルオーバーの管理方法の例」](#)

## フェイルオーバーの管理方法の例

ここでは、フェイルオーバーの開始方法と、アプリケーションがどのように二次クラスタに転送されるかを説明します。フェイルオーバーの後、DNS エントリを更新します。詳細については、[384 ページの「フェイルオーバーの管理のガイドライン」](#)を参照してください。

ここでは、次の手順について説明します。

- [410 ページの「スイッチオーバーを呼び出す」](#)
- [412 ページの「DNS エントリを更新する」](#)

### ▼ スイッチオーバーを呼び出す

- 1 **nodeA** および **nodeC** にスーパーユーザーになるか、**RBAC** の承認 **solaris.cluster.admin** を提供する役割になりますを使用してアクセスします。

2 主クラスタをロギングモードに変更します。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -n -l lhost-reprg-prim \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 lhost-reprg-sec \  
/dev/vx/rdisk/devgrp/vol01 \  
/dev/vx/rdisk/devgrp/vol04 ip sync
```

ディスクのデータボリュームに書き込みが行われると、同じデバイスグループのビットマップボリュームが更新されます。複製は行われません。

3 主クラスタと二次クラスタがロギングモードで、自動同期がオフであることを確認します。

a. nodeA で、モードと設定を確認します。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeA# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 ->  
lhost-reprg-sec:/dev/vx/rdisk/devgrp/vol01  
autosync:off, max q writes:4194304,max q fbas:16384,mode:sync,ctag:  
devgrp, state: logging
```

b. nodeC で、モードと設定を確認します。

次のコマンドを実行します。Sun StorageTek Availability Suite ソフトウェアの場合:

```
nodeC# /usr/sbin/sndradm -P
```

次のような出力が表示されます。

```
/dev/vx/rdisk/devgrp/vol01 <-  
lhost-reprg-prim:/dev/vx/rdisk/devgrp/vol01  
autosync:off, max q writes:4194304,max q fbas:16384,mode:sync,ctag:  
devgrp, state: logging
```

nodeA と nodeC の状態は logging で、非同期のアクティブ状態は off でなければなりません。

4 二次クラスタで主クラスタからのテイクオーバーの準備ができていることを確認します。

```
nodeC# fsck -y /dev/vx/rdisk/devgrp/vol01
```

5 二次クラスタにスイッチオーバーします。

```
nodeC# clresourcegroup switch -n nodeC nfs-rg
```

次の手順 [412 ページの「DNS エントリを更新する」](#)に進みます。

## ▼ DNS エントリを更新する

DNS がクライアントをクラスタにどのようにマッピングするかについては、[図 A-6](#) を参照してください。

始める前に 手順410 ページの「スイッチオーバーを呼び出す」を完了します。

**1 nsupdate コマンドを開始します。**

詳細は、[nsupdate\(1M\)](#) のマニュアルページを参照してください。

**2 両方のクラスタについて、アプリケーションリソースグループの論理ホスト名とクラスタ IP アドレス間の現在の DNS マッピングを削除します。**

```
> update delete lhost-nfsrg-prim A
> update delete lhost-nfsrg-sec A
> update delete ipaddress1rev.in-addr.arpa ttl PTR lhost-nfsrg-prim
> update delete ipaddress2rev.in-addr.arpa ttl PTR lhost-nfsrg-sec
```

*ipaddress1rev* 主クラスタの IP アドレス (逆順) です。

*ipaddress2rev* 二次クラスタの IP アドレス (逆順) です。

*ttl* 秒単位の有効時間です。一般的な値は 3600 になります。

**3 両方のクラスタについて、アプリケーションリソースグループの論理ホスト名とクラスタ IP アドレス間の、新しい DNS マッピングを作成します。**

主論理ホスト名を二次クラスタの IP アドレスにマッピングし、二次論理ホスト名を主クラスタの IP アドレスにマッピングします。

```
> update add lhost-nfsrg-prim ttl A ipaddress2fwd
> update add lhost-nfsrg-sec ttl A ipaddress1fwd
> update add ipaddress2rev.in-addr.arpa ttl PTR lhost-nfsrg-prim
> update add ipaddress1rev.in-addr.arpa ttl PTR lhost-nfsrg-sec
```

*ipaddress2fwd* 二次クラスタの IP アドレス (正順) です。

*ipaddress1fwd* 主クラスタの IP アドレス (正順) です。

# 索引

---

## A

autoboot プロパティ, 249  
Availability Suite, データ複製に使用, 375

## B

boot コマンド, 67-69

## C

cconsole コマンド, 22, 26  
ccp コマンド, 21, 26  
claccess コマンド, 21  
cldevicegroup コマンド, 21  
cldevice コマンド, 21  
clinterconnect コマンド, 21  
clnasdevice コマンド, 21  
clnode check コマンド, 21  
clnode コマンド, 292  
clquorum コマンド, 21  
clreslogicalhostname コマンド, 21  
clresourcegroup コマンド, 21, 292-294  
clresourcetype コマンド, 21  
clresource コマンド, 21  
clresource コマンド, リソースとリソースグループの削除, 295  
clressharedaddress コマンド, 21  
clsetup ユーティリティー, 20, 21, 28  
clsnmp host コマンド, 21  
clsnmpmib コマンド, 21

clsnmpuser コマンド, 21  
cltelemattribute コマンド, 21  
cluster check コマンド, 21  
    vfstab ファイル確認, 178  
cluster shutdown コマンド, 63-73  
clzonecluster  
    boot, 67-69  
    説明, 28  
    停止, 63-73  
clzonecluster コマンド, 21  
CPU シェア  
    グローバルクラスタの投票ノード, 309  
    グローバルクラスタの非投票ノード, 311  
    グローバルクラスタの非投票ノード、専用のプロセッサセット, 314  
    構成, 307  
    制御, 307  
crlogin コマンド, 26  
cssh コマンド, 26  
ctelnet コマンド, 26

## D

DID 情報, 手動更新, 186-187  
DID 情報の手動更新, 186-187  
DR, 「動的再構成」を参照

## E

EMC SRDF  
    DID デバイスを構成する, 114-116

**EMC SRDF (続き)**

- キャンパスクラスタの主ルームの完全なフェイルオーバー後の復元, 122-124
- 構成例, 117-124
- 構成を確認する, 116-117
- 制限, 94
- 適応型コピー, 93
- ドミノモード, 93
- 複製グループを構成する, 113-114
- ベストプラクティス, 95
- 要件, 94
- を管理する, 112-124
- /etc/inet/hosts ファイル, 排他的 IP ゾーン上の構成, 251
- /etc/nsswitch.conf ファイル, 非大域ゾーンの変更, 250
- /etc/vfstab ファイル, 50
- 設定の確認, 178
- マウントポイントの追加, 177

**F**

- failback プロパティ, 161
- fence\_level, 「複製時」を参照
- Oracle Solaris Cluster ソフトウェアへのパッチの適用, 319-321

**G**

- GUI 管理ツール, 20, 365-373
  - Oracle Solaris Cluster Manager, 365
  - Sun Management Center, 366

**H**

- Hitachi TrueCopy
  - DID デバイスの構成, 104-106
  - 管理, 101-112
  - 構成の確認, 106-107
  - 構成例, 107-112
  - 制限, 94
  - データモードまたはステータスモード, 93

**Hitachi TrueCopy (続き)**

- 複製グループの構成, 102-104
- ベストプラクティス, 95
- 要件, 94
- hosts ファイル, 排他的 IP ゾーン上の構成, 251

**I****IPMP**

- 管理, 241
- ステータス, 35
- 排他的 IP ゾーン上のグループ構成, 251
- IP アドレス, 排他的 IP ゾーンのネームサービスへの追加, 251

**K**

- /kernel/drv/,md.conf ファイル, 135

**L**

- lofi ファイル, アインストール, 284

**M**

- md.tab ファイル, 23
- MIB
  - SNMP イベントの有効化と無効化, 287
  - SNMP イベントプロトコルの変更, 288

**N**

- NAS, 「ネットワーク接続ストレージ定足数デバイス」を参照
- NetApp, 「ネットワーク接続ストレージ定足数デバイス」を参照
- netcon コマンド, 22
- Network Appliance, 「ネットワーク接続ストレージ定足数デバイス」を参照

nsswitch.conf ファイル, 非大域ゾーンの  
変更, 250  
ntp.conf.cluster ファイル, 272  
numsecondaries プロパティ, 163

## O

OpenBoot PROM (OBP), 269  
Oracle Solaris Cluster Manager, 20, 365  
RBAC の役割, 設定, 367  
起動, 370  
サーバーアドレスの変更, 368  
Oracle Solaris OS  
CPU 制御, 307  
svcadm コマンド, 270  
グローバルクラスタ定義, 17  
グローバルクラスタの管理作業, 18  
ゾーンクラスタ定義, 17  
ノードの起動に関する特別な指示, 78-80  
ノードの再起動に関する特別な指示, 80-84  
ホストベースの複製, 91  
Oracle Solaris Cluster 定足数サーバー, 定足数デ  
バイスとしてサポート, 194  
Oracle Solaris ゾーン  
autoboot プロパティ, 249  
nsswitch.conf ファイルの変更, 250  
共有 IP ゾーン, 250  
排他的 IP ゾーン  
hosts ファイルの構成, 251  
IPMP グループの構成, 251  
Oracle Solaris ソフトウェア, SMF, 249

## R

raw ディスクデバイス, 命名規則, 177  
raw ディスクデバイスグループ, 追加, 137-138  
RBAC, 55-62  
Oracle Solaris Cluster Manager, 367  
グローバルクラスタの投票ノードの場合, 56  
権限プロファイル (説明), 56-57  
作業  
カスタム役割を追加, 60  
使用, 55

RBAC, 作業 (続き)  
設定, 55  
役割の追加, 57  
ユーザーの変更, 61  
非投票ノードの場合, 56  
Role-Based Access Control, 「RBAC」を参照

## S

SATA, 196  
SATA ストレージ, 定足数デバイスとしてサ  
ポート, 194  
secure shell, 26, 27  
showrev -p コマンド, 29  
SMF, オンラインサービスの確認, 249  
SNMP  
イベント MIB の有効化と無効化, 287  
プロトコルの変更, 288  
ホストの無効化, 289  
ホストの有効化, 288  
ユーザーの削除, 291  
ユーザーの追加, 290  
SNMP イベント MIB の有効化と無効化, 287  
Solaris OS  
「Oracle Solaris OS」も参照  
Solaris ボリュームマネージャー, raw ディスクデ  
バイス名, 177  
SRDF  
「EMCSRDF」を参照  
ssh, 27  
Sun Management Center  
インストール, 22  
概要, 366  
説明, 20  
Sun NAS, 定足数デバイスとしてサポート, 194  
Sun StorageTek Availability Suite, データ複製に使  
用, 375  
Sun ZFS Storage Appliance, 定足数デバイスとして追  
加, 197  
SunMC, 「Sun Management Center」を参照  
Sun NAS 定足数デバイス, 追加, 197  
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, 194  
System Service Processor (SSP), 22

**T**

## TrueCopy

「Hitachi TrueCopy」を参照

**U**

/usr/cluster/bin/clresource, リソースグループ  
の削除, 295

/usr/cluster/bin/cluster check コマンド, vfstab  
ファイル確認, 178

**V**

/var/adm/messages ファイル, 87

## Veritas

オンラインバックアップ, 345

カプセル化されたルートファイルシステムの復  
元, 360

カプセル化されていないルートファイルシステ  
ムを復元, 358

管理, 99-100

## Veritas File System (VxFS)

管理, 179

クラスタファイルシステムのマウント, 179

Veritas Volume Manager (VxVM), raw ディスクデバ  
イス名, 177

## vfstab ファイル

設定の確認, 178

マウントポイントの追加, 177

VxFSによってサポートされる機能, 124

VxVM, 99-100

**Z**

## ZFS

デバイスグループ追加, 138

ファイルシステムの削除, 296-298

複製, 138

ルートファイルシステムの制限, 125

ZFS Storage Appliance, 「Sun ZFS Storage Appliance

定足数デバイス」を参照

**あ**

インストール, lofi デバイスファイル, 284

アクセス権、グローバルデバイスの, 98

アダプタ、トランスポート, 231

アフィニティスイッチオーバー、データ複製用  
の構成, 395

アフィニティフェイルオーバー、データ複製の  
拡張プロパティ, 380

アプリケーションリソースグループ

ガイドライン, 381

データ複製の構成, 397-399

アンインストール, Oracle Solaris Cluster ソフト  
ウェア, 282

**い**

移行, グローバル名前空間, 131

一覧表示

定足数構成, 215

デバイスグループ構成, 166

イベント MIB

SNMP の有効化と無効化, 287

SNMP プロトコルの変更, 288

**え**

エラーメッセージ

/var/adm/messages ファイル, 87

ノードの削除, 259-260

遠隔ログイン, 26

**か**

開始

グローバルクラスタノード, 73-87

ゾーンクラスタノード, 73-87

ノード, 73-87

概要, 定足数, 191-217

確認

SMF, 249

vfstab 設定, 178

クラスタインターコネクトの状態, 227



## 確認 (続き)

- グローバルマウントポイント, 50, 182
- データ複製構成, 407-410
- ファイルシステム名, 338
- 監視, ディスクパス, 183-185
- 監視解除, ディスクパス, 185
- 管理
  - Hitachi TrueCopy で複製されたデバイス, 101-112
  - IPMP, 225-243
  - クラスタインターコネクとパブリックネットワーク, 225-243
  - クラスタファイルシステム, 124
  - グラフィカルユーザーインタフェース (Graphical User Interface、GUI) ツールによるクラスタ, 365-373
  - グローバルクラスタ, 18
  - グローバルクラスタ設定, 261-305
  - グローバルクラスタの非投票ノード, 18
  - ストレージベースの複製されたデバイス, 101-124
  - ゾーンクラスタ, 18, 294
  - 定足数, 191-217
- 管理コンソール, 22
- 管理する, EMC SRDF 複製デバイス, 112-124
- 「管理役割を追加 (Add Administrative)」ウィザード, 説明, 57

## き

## 起動

- Oracle Solaris Cluster Manager, 370
- グローバルクラスタ, 63-87
- グローバルクラスタノード, 73-87
- ゾーンクラスタ, 63-87
- ゾーンクラスタノード, 73-87
- ノード, 73-87
- 非クラスタモード, 84
- 非大域ゾーン, 73
- 共通エージェントコンテナ
  - セキュリティー鍵の再生成, 369
  - ポート番号の変更, 367
- 共有 IP ゾーン, 「Oracle Solaris ゾーン」を参照

- 共有 SCSI ディスク, 定足数デバイスとしてサポート, 194
- 共有ディスクパス
  - 監視, 182-190
  - 自動再起動の無効化, 190
  - 自動再起動の有効化, 189-190
- 切り換え, デバイスグループの主ノード, 167-168

## く

## クラスタ

- クラスタ名の変更 (Veritas CVM の使用時), 264
- 再起動パッチの適用, 326
- 時刻の設定, 267
- 名前の変更, 262-263
- ノード認証, 265
- バックアップ, 23, 337-349
- ファイルの復元, 349
- クラスタインターコネク
  - 管理, 225-243
  - 状態の確認, 227
  - 動的再構成, 227
- クラスタ検査
  - コマンド
    - への変更, 45
- クラスタコンソールへの安全な接続, 27
- クラスタコントロールパネル (CCP), 22
- クラスタの時刻の設定, 267
- クラスタファイルシステム, 97-190
  - 管理, 124
  - グローバルクラスタ投票ノード, 124
  - グローバルクラスタの非投票ノード, 124
  - 削除, 180-182
  - 設定の確認, 178
  - 追加, 176-180
  - マウントオプション, 178
- クラスタファイルシステムのマウントオプション, 要件, 178
- グローバル
  - デバイス, 97-190
    - アクセス権の設定, 98
    - 動的再構成, 98-99
    - 名前空間, 97-100, 129
    - マウントポイント, 確認, 50, 182

## グローバルクラスタ

- 管理, 261-305
- 起動, 63-87
- 構成の検証, 45
- 構成の表示, 36
- コンポーネントの状態, 33
- 再起動, 69
- 定義, 18
- 停止, 63-87
  - ノードの削除, 253
- グローバルクラスタ投票ノード, クラスタファイルシステムの管理, 124
- グローバルクラスタノード
  - 起動, 73-87
  - 再起動, 80-84
  - シャットダウン, 73-87
- グローバルクラスタの投票ノード, CPUシェア, 309
- グローバルクラスタの非投票ノード
  - CPUシェア, 311, 314
  - 管理, 18
  - クラスタファイルシステムの管理, 124
  - パッチの適用, 324
- グローバルクラスタ非投票ノード
  - 停止と再起動, 73
- グローバル名前空間, 移行, 131
- グローバル名前空間の更新, 129
- 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, 231
- 権限プロファイル, RBAC, 56-57
- 検索
  - グローバルクラスタ用ノード ID, 265
  - ゾーンクラスタ用ノード ID, 265
- 検証
  - グローバルクラスタ構成, 45
  - ゾーンクラスタ構成, 45

## こ

- 構成
  - データ複製, 375-412
  - デバイスグループのマイナー番号, 147
  - 構成済みリソースを表示, 31
  - 構成例(構内クラスタ化)
    - 2か所、ストレージベースのデータ複製, 92-96
    - 2か所、ストレージベースの複製, 92-96
  - 構内クラスタ
    - ストレージベースのデータ複製, 92-96
    - ストレージベースのデータ複製を使用した復旧, 95
  - 公平配分スケジューラ, CPUシェアの構成, 308
- コマンド
  - boot, 67-69
  - cconsole, 22, 26
  - ccp, 21, 26
  - claccess, 21
  - cldevice, 21
  - cldevicegroup, 21
  - clinterconnect, 21
  - clnasdevice, 21
  - clnode check, 21
  - clquorum, 21
  - clreslogicalhostname, 21
  - clresource, 21
  - clresourcegroup, 21
  - clresourcetype, 21
  - clressharedaddress, 21
  - clsetup, 21
  - clsnmp host, 21
  - clsnmp mib, 21
  - clsnmp user, 21
  - cltelemetryattribute, 21
  - cluster check, 21, 24, 45, 50
  - cluster shutdown, 63-73
  - clzonecluster, 21, 63-73
  - clzonecluster boot, 67-69
  - clzonecluster verify, 45
  - crlogin, 26
  - cssh, 26
  - ctelnet, 26
  - netcon, 22
  - メタセット, 97-100

コマンド行管理ツール, 20

コンソール

安全な接続, 27

への接続, 26

## さ

再起動

グローバルクラスタ, 69

グローバルクラスタノード, 80-84

ゾーンクラスタ, 69

ゾーンクラスタノード, 80-84

最後の定足数デバイス, 削除, 207

再生成, セキュリティー鍵, 369

削除

SNMP ホスト, 289

SNMP ユーザー, 291

Solaris Volume Manager デバイスグループ, 140

クラスタファイルシステム, 180-182

グローバルクラスタ上の非投票ノード, 256

最後の定足数デバイス, 207

ストレージアレイ, 256

すべてのデバイスグループからノードを, 141

ゾーンクラスタから, 252

定足数デバイス, 193, 206

デバイスグループ, 155

デバイスグループからノードを, 157

デバイスグループから, ボリューム, 154-155

トランスポートケーブル, アダプタ, ス

イッチ, 231

ノード, 251, 253

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, 295

作成, 新しいディスクグループ, 139-140

サポートされている定足数デバイスタイプ, 194

サポートされる機能, VxFS によって, 124

## し

シャットダウン

グローバルクラスタノード, 73-87

ゾーンクラスタノード, 73-87

ノード, 73-87

修復, 定足数デバイス, 216

使用, 役割 (RBAC), 55

状態

グローバルクラスタコンポーネント, 33

ゾーンクラスタコンポーネント, 33

## す

スイッチ, トランスポート, 231

スイッチバック, データ複数での実行ガイドライン, 385

ストレージアレイ, 削除, 256

ストレージベースのデータ複製, 92-96

制限, 94

定義, 90

と定足数デバイス, 95

復旧, 95

ベストプラクティス, 95

要件, 94

ストレージベースの複製されたデバイス, 管理, 101-124

スナップショット

「ストレージベースの複製」を参照

ポイントインタイム, 377

## せ

セキュリティ鍵, 再生成, 369

設定, 役割 (RBAC), 55

専用のプロセッサセット, 構成, 314

## そ

ゾーンクラスタ

アプリケーション用に準備, 294

管理, 261-305

起動, 63-87

構成の検証, 45

構成の表示, 36

コンポーネントの状態, 33

再起動, 69

サポートされている直接マウント, 296-298

## ゾーンクラスタ (続き)

- ゾーンパスの移動, 294
- 定義, 18
- 停止, 63-87
- ファイルシステムの削除, 294
- 複製, 294
- ゾーンクラスタノード
  - IP アドレスおよび NIC の指定, 245-251
  - 起動, 73-87
  - 再起動, 80-84
  - シャットダウン, 73-87
- ゾーンパス, 移動, 294
- 属性, 「プロパティ」を参照

## た

- 耐障害性, 定義, 376
- タイムアウト, 定足数デバイスのデフォルト値の変更, 217

## ち

- 直接接続共有ディスク定足数デバイス, 追加, 196
- 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, 296-298

## つ

## 追加

- SNMP ホスト, 288
- SNMP ユーザー, 290
- Solaris Volume Manager デバイスグループ, 137
- Sun NAS 定足数デバイス, 197
- Sun ZFS Storage Appliance NAS 定足数デバイス, 197
- ZFS デバイスグループ, 138
- 新しいボリュームをデバイスグループに, 146
- カスタム役割 (RBAC), 60
- クラスタファイルシステム, 176-180
- グローバルクラスタにノードを, 246
- ゾーンクラスタにノードを, 246
- 直接接続共有ディスク定足数デバイス, 196

## 追加 (続き)

- 定足数サーバー定足数デバイス, 202
- 定足数デバイス, 195
- デバイスグループ, 135, 137-138
- デバイスグループにノードを, 156
- トランスポートケーブル, アダプタ, スイッチ, 228
- ネットワーク接続ストレージ定足数デバイス, 199
- ノード, 245-251
- 役割 (RBAC), 57

## て

## 停止

- グローバルクラスタ, 63-87
- グローバルクラスタノード, 73-87
- ゾーンクラスタ, 63-87
- ゾーンクラスタノード, 73-87
- ノード, 73-87
- 非大域ゾーン, 73
- ディスクグループ
  - 構成変更の登録, 151
  - 作成, 139-140
  - 登録, 148
  - 変更, 147
- ディスクのカプセル化, 144
- ディスクパス
  - 監視, 182-190
    - 障害のあるディスクパスを表示, 186
  - 監視解除, 185
- ディスクパスの
  - 状態エラーの解決, 186-187
  - モニタリング, 97-190
- 定足数
  - 概要, 191-217
  - 管理, 191-217
- 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, 202
  - 削除のトラブルシューティング, 207
  - 追加, 202

- 定足数デバイス
  - 交換, 209
  - 構成の一覧表示, 215
  - 最後の定足数デバイスの削除, 207
  - 削除, 193, 206
  - 修復, 216
  - 追加, 195
    - Sun NAS 定足数デバイス, 197
    - Sun ZFS Storage Appliance NAS 定足数デバイス, 197
    - 直接接続共有ディスク定足数デバイス, 196
    - 定足数サーバー定足数デバイス, 202
    - ネットワーク接続ストレージ定足数デバイス, 199
  - デバイスの動的再構成, 193
  - デフォルトのタイムアウトの変更, 217
  - とストレージベースの複製, 95
  - ノードリストの変更, 210
  - 保守状態、デバイスを, 212
  - 保守状態、デバイスを保守状態から戻す, 213
- 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, 194
- 定足数デバイスの交換, 209
- データ複製, 89-96
  - DNS エントリの更新, 412
  - ガイドライン
    - スイッチオーバーの管理, 384
    - フェイルオーバーの管理, 384
    - リソースグループの構成, 379
  - 概要, 376
  - 構成
    - NFS アプリケーション用ファイルシステム, 392-393
    - NFS アプリケーションリソースグループ, 397-399
    - アフィニティスイッチオーバー, 380, 395
    - デバイスグループ, 389
  - 構成の確認, 407-410
  - 構成例, 385
  - ストレージベースの, 90, 92-96
  - 定義, 90-91
  - 同期, 377
  - 必要なハードウェアとソフトウェア, 387
  - 非同期, 377
- データ複製 (続き)
  - フェイルオーバーの管理, 410-412
  - ポイントインタイムスナップショット, 377, 406-407
  - ホストベースの, 90
  - 有効化, 402-405
  - リソースグループ
    - アプリケーション, 381
    - 共有アドレス, 383
    - 構成, 380
    - 作成, 394-396
    - スケーラブルアプリケーション, 383
    - フェイルオーバーアプリケーション, 381-382
    - 命名規則, 380
    - リモートミラー, 376, 405-406
    - 例, 405-410
  - 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, 398, 401
    - 複製リソース, 395, 396
  - データ複製のためのスケーラブルアプリケーション, 383
  - データ複製のフェイルオーバーアプリケーションガイドライン
    - フェイルオーバーの管理, 384
    - リソースグループ, 381-382
    - 管理, 410-412
  - データ複製用の共有アドレスリソースグループ, 383
  - データ複製用のスイッチオーバー
    - アフィニティスイッチオーバー, 380
    - 実行, 410-412
  - データ複製用のフェイルオーバーアプリケーション, アフィニティスイッチオーバー, 380
  - 適用
    - ゾーンクラスタに非再起動パッチを, 321
    - パッチ, 321
    - 非再起動パッチ, 329
  - デバイス, グローバル, 97-190
  - デバイスグループ
    - raw ディスク
      - 追加, 137-138

## デバイスグループ (続き)

## SVM

追加, 135

新しいマイナー番号の割当て, 147

確認登録, 153

管理の概要, 127

構成の表示, 166

削除と登録解除, 140, 155

主所有者権, 161

追加, 137

データ複製の構成, 389

変更プロパティ, 161

保守状態, 168

デバイスグループの主所有者権, 161

デバイスグループの主ノードの切り替

え, 167-168

電源管理, 261

## と

同期データ複製, 93, 377

動的再構成, 98-99

クラスタインターコネク, 227

定数デバイス, 193

パブリックネットワークインタフェース, 242

## 登録

ディスクグループ構成の変更, 151

デバイスグループとしてのディスクグループ, 148

## 登録解除

Solaris Volume Manager デバイスグループ, 140

デバイスグループ, 155

ドメインネームシステム (DNS), データ複製での更新, 412

ドメインネームシステム (Domain Name System, DNS), 更新のガイドライン, 384

トランスポートアダプタの追加, 228, 231

トランスポートケーブル

追加, 228, 231

無効にする, 235

有効にする, 234

トランスポートスイッチの追加, 228, 231

## な

## 名前空間

移行, 131

グローバル, 97-100

## に

## 二次ノード

希望数の設定, 163

デフォルト数, 161

## ね

ネームサービス, 排他的 IP ゾーンの IP アドレスマッピングの追加, 251

ネットワークアプライアンス NAS, 定数デバイスとしてサポート, 194

ネットワーク接続ストレージ定数デバイス, 追加およびインストール, 199

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, 392-393

## の

## ノード

ID 検索, 265

起動, 73-87

グローバルクラスタからのノードの削除, 253

グローバルクラスタからの非投票ノードの削除, 256

グローバルクラスタに再起動パッチを適用, 321

## 削除

エラーメッセージ, 259-260

シャットダウン, 73-87

主, 161

ゾーンクラスタからの削除, 252

追加, 245-251

デバイスグループから削除, 141, 157

デバイスグループに追加, 156

二次, 161

## ノード (続き)

- 認証, 265
- 負荷制限の設定, 292-294
- プライマリ, 98-99
- への接続, 26

## は

排他的 IP ゾーン, 「Oracle Solaris ゾーン」を参照

## バックアップ

- クラスタ, 23, 337-349
- ファイルシステム, 338
- ボリュームをオンライン, 345
- ミラーのオンライン, 342
- ルートファイルシステム, 339

## パッチ

- クラスタおよびファームウェアへの適用, 326
- グローバルクラスタに再起動パッチを適用, 321
- グローバルクラスタに非再起動パッチを適用, 321
- グローバルクラスタの非投票ノードでの, 324
- 注意事項, 320
- 非再起動パッチの適用, 329

## パブリックネットワーク

- 管理, 225-243
- 動的再構成, 242

## ひ

非クラスタモードでの起動, 84

## ビットマップ

- ポイントインタイムスナップショット, 377
- リモートミラー複製, 376

非同期データ複製, 93, 377

## 表示

- グローバルクラスタ構成, 36
- 障害のあるディスクパス, 186
- ゾーンクラスタ構成, 36

## ふ

## ファイル

- /etc/vfstab, 50
- md.conf, 135
- md.tab, 23
- 対話形式で1つずつ復元, 350
- 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, 392-393
  - カプセル化されたルートの復元, 360
  - カプセル化されていないルートを復元, 358
  - ゾーンクラスタ内で削除, 294
  - 名前の確認, 338
  - バックアップ, 338
  - ルートの復元
    - 説明, 350
    - ボリュームから, 353
    - メタデバイスから, 353

## 負荷制限

- concentrate\_load プロパティ, 292
- preemption\_mode プロパティ, 292
- ノード上での設定, 292

負荷制限の設定, ノード上, 292-294

## 復元

- カプセル化されたルートファイルシステム, 360
- カプセル化されていないルートファイルシステム, 358
- クラスタファイル, 349
- 個々のファイルを対話形式で, 350
- ルートファイルシステム, 350
  - ボリュームから, 353
  - メタデバイスから, 353

## 複製

- 「データ複製」を参照
- Hitachi TrueCopy で複製されたデバイス, 102-104

複製, ストレージベースの, 92-96

復旧, ストレージベースのデータ複製を装備したクラスタ, 95

プライベートホスト名

- ゾーンへの割り当て, 250

## プロパティ

- failback, 161

- プロパティ (続き)  
  numsecondaries, 163  
  preferenced, 161  
プロファイル, RBAC 権限, 56-57
- へ
- ベストプラクティス  
  EMC SRDF, 95  
  Hitachi TrueCopy, 95  
  ストレージベースのデータ複製, 95
- 変更  
  numsecondaries プロパティ, 163  
  Oracle Solaris Cluster Manager のサーバーアドレス, 368  
  SNMP イベント MIB プロトコル, 288  
  クラスタ名, 262-263  
  クラスタ名 (Veritas CVM の使用時), 264  
  主ノード, 167-168  
  ディスクグループ, 147  
  定足数デバイスのノードリスト, 210  
  プロパティ, 161  
  ポート番号、共通エージェントコンテナの使用, 367  
  ユーザー (RBAC), 61  
  プライベートホスト名, 270
- ほ
- ポイントインタイムスナップショット  
  実行, 406-407  
  定義, 377
- ポート番号、共通エージェントコンテナの使用の変更, 367
- 保守状態  
  ノード, 278  
  定足数デバイス, 212  
  定足数デバイスを保守状態から戻す, 213
- 保守状態にする  
  定足数デバイス, 212  
  定足数デバイスを, 212
- ホスト  
  SNMP の追加と削除, 288, 289
- ホストベースのデータ複製  
  定義, 90  
  例, 375-412
- ボリューム  
  「ストレージベースの複製」を参照  
  オンラインでバックアップ, 345  
  デバイスグループからの削除, 154-155  
  デバイスグループへの追加, 146  
  ボリュームマネージャー、Veritas, 99-100
- ま
- マウントポイント  
  /etc/vfstab ファイルの修正, 177  
  グローバル, 50
- マルチユーザーサービス, 確認, 249
- 満杯の /var/adm/messages ファイルの修復, 87
- み
- ミラー、オンラインバックアップ, 342
- む
- 無効にする、トランスポートケーブルを, 235
- め
- 命名規則  
  raw ディスクデバイス, 177  
  複製リソースグループ, 380
- メタセットコマンド, 97-100
- も
- モニタリング、共有ディスクパス, 189-190



## や

## 役割

- カスタム役割を追加, 60
- 設定, 55
- 役割の追加, 57

## ゆ

有効にする、トランスポートケーブルを, 234

## ユーザー

- SNMP の削除, 291
  - SNMP の追加, 290
  - プロパティの変更, 61
- ユーザーアカウントツール, 説明, 61

## り

## リソース

- 構成済みタイプを表示, 31
- 削除, 295

## リソースグループ

- データ複製
  - 構成, 380
  - 構成のガイドライン, 379
  - フェイルオーバーでの役割, 380

リモート複製, 「ストレージベースの複製」を参照

リモートミラー化, 「ストレージベースの複製」を参照

## リモートミラー複製

- 実行, 405-406
- 定義, 376

リリース情報, 29

## 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, 296-298

## れ

## 例

- 機能検証検査の実行, 48-49
- クラスタファイルシステムの作成, 179
- 対話型検証検査の一覧表示, 48

## ろ

ローカルミラー化, 「ストレージベースの複製」を参照

ログイン, 遠隔, 26

論理ホスト名リソース, データ複製フェイルオーバーでの役割, 380

## グ

グローバルクラスタの非投票ノード

プライベートホスト名の追加, 273

グローバルクラスタ非投票ノード

プライベートホスト名, 削除, 275

プライベートホスト名の変更, 273

## ノ

## ノード

グローバルクラスタでの名前変更, 275

ゾーンクラスタでの名前変更, 275

保守状態にする, 278

## ノード名の変更

グローバルクラスタ, 275

ゾーンクラスタ, 275

## フ

## ファイル

ntp.conf.cluster, 272

プ

プライベートホスト名

グローバルクラスタの非投票ノード, 273

グローバルクラスタ非投票ノードの変更, 273

グローバルクラスタ非投票ノード上で削

除, 275

変更, 270