CXFS™ Software Installation and Administration Guide（日本語版）

007–4016–008JP

制作スタッフ

著作 Lori Johnson

編集 Rick Thompson

イラスト Chris Wengelski

製作 Glen Traefald

協力エンジニア François Barbou des Places、Ken Beck、Felix Blyakher、Laurie Costello、Dave Ellis、Brian Gaffey、Dean Jansa、Erik Jacobson、Chris Kirby、Ted Kline、Dan Knappe、Kent Koeninger、Linda Lait、Bob LaPreze、Steve Lord、Troy McCorkell、Lanet Merrill、Terry Merth、Nate Pearlstein、John Relph、Elaine Robinson、Dean Roehrich、Andy Tran、Wesley Smith、Paddy Sreenivasan、Connie Waring、Geoffrey Wehrman

COPYRIGHT

© 1999, 2001 Silicon Graphics, Inc. All rights reserved. このマニュアルの別の箇所に示されているとおり、提供されている部分の著作権はサード・パーティが保持している場合があります。この電子ドキュメントの内容の一部または全部について、Silicon Graphics, Inc. から事前に文書による許諾を得ずに、いかなる方法でも複製または頒布したり、派生的な文書を作成することはできません。

LIMITED RIGHTS LEGEND

The electronic (software) version of this document was developed at private expense; if acquired under an agreement with the USA government or any contractor thereto, it is acquired as "commercial computer software" subject to the provisions of its applicable license agreement, as specified in (a) 48 CFR 12.212 of the FAR; or, if acquired for Department of Defense units, (b) 48 CFR 227-7202 of the DoD FAR Supplement; or sections succeeding thereto. Contractor/manufacturer is Silicon Graphics, Inc., 1600 Amphitheatre Pkwy 2E, Mountain View, CA 94043-1351, USA.

商標および帰属

Silicon Graphics、O2、IRIS、および IRIX は登録商標です。SGI ロゴ、CXFS、IRIS FailSafe、IRIX Interactive Desktop、および Origin は Silicon Graphics, Inc. の登録商標です。

Adaptec は Adaptec, Inc. の商標です。BROCADE は Brocade Communication Systems, Inc. の商標です。CIPRICO は Ciprico Inc. の商標です。CLARiiON は Data General Corporation の商標です。Emulex は Emulex Corporation の商標です。FLEXlm は GLOBEtrotter, Inc. の商標です。Legato NetWorker は Legato Systems, Inc. の商標です。NFS および Sun は Sun Microsystems, Inc. の商標です。Netscape は Netscape Communications Corporation の商標です。Prisa は Prisa Networks, Inc. の商標です。QLogic は QLogic Corporation の商標です。UNIX は、X/Open Company Limited を通じて米国およびその他の国々で独占的にライセンス供与されている登録商標です。VERITAS は VERITAS Software Corporation の商標です。Windows は Microsoft Corporation の商標です。

カバー・デザイン Sarah Bolles、Sarah Bolles Design、Dany Galgani、および SGI Technical Publications

このガイドでの新機能

この更新版には、以下の新しい情報が含まれています。

- 実作業用の環境（つまり、メタデータ・サーバの再設定と回復が必要な環境）では、少なくとも **3** つの重み付けされたノードのクラスタが推奨されています。

  2 つの重み付けされたノードのクラスタを実作業で使用する場合は、以下のいずれかを行う必要があります。

  – データの破壊を避けるためにリセット回線を使用し、必ず 1 つのノードだけがエラー状態で実行されるようにます（リセット回線は、すべての CXFS クラスタに対して推奨されており、IRIS FailSafe と共に使用する場合は必須です）。

  – 1 つのノードの重みを 1 に設定し、その他は 0 に設定します。

  – タイブレーカー・ノードを設定します。

  ただし、2 つの重み付けされたノードのクラスタには問題があります。32 ページの「重み付けされた 2 ノード・クラスタでの回復と再設定の問題」および17 ページの「ハードウェア・リセット」を参照してください。

- 新しい `cluster_status` コマンド。`cad` デーモンによって収集されたステータス情報を表示するために `curses`(3X) インタフェースを提供します（この情報は `cxdetail` コマンドでも表示されます）。

- クラスタ・ノードでは、同バージョンまたは直前のバージョンの IRIX オペレーティング・システム (OS: Operating System) を実行できます（6.5.12f の場合は 6.5.11f および 6.5.12f など）。

- マウント操作の前後に独自のカスタム・スクリプトを実行する機能。144 ページの「マウント操作前後のスクリプトの実行」を参照してください。

- パーティション ID 情報。84 ページの「GUI を使ったノードの定義」および109 ページの「CLI を使ったノードの定義」を参照してください。

- 以下の点に関する明確化

  – ホスト名解決ルール。クラスタを設定する前に、これらのルールを理解して、ファイルを適切に設定することが非常に重要です。52 ページの「ホスト名解決: `/etc/sys_id`、`/etc/hosts`、`/etc/nsswitch.conf` の設定」を参照してください。

  – CXFS メンバーシップと `fs2d` メンバーシップの違い。7 ページの「用語」を参照してください。

- IRIS FailSafe と CXFS の同時実行環境でサポートされているノードの設定。27 ページの「IRIS FailSafe 2.1 との同時実行」を参照してください。
- CXFS を使用した、書込まれていないエクステントの追跡 (`unwritten=1|0`)。186 ページの「書込まれていないエクステントの追跡」を参照してください。
- `/etc/exports` ファイルへの CXFS マウント・ポイントの組込み。58 ページの「(オプション) `/etc/exports` の設定」を参照してください。
- サポートされているノードの数: 16 CXFS ノード、および同時実行での最高 8 つまでの IRIS FailSafe ノード。26 ページの「ハードウェアおよびソフトウェアの必要条件とサポート」を参照してください。
- 同時実行クラスタにおける情報の流れ。図1-10を参照してください。

- 最初の日本語版(上記は英語版前版からの改訂履歴です)。

改訂情報

バージョン	説明
001	1999 年 9 月 IRIX 6.5.6 リリースで CXFS 1.1 製品をサポート
002	1999 年 10 月 IRIX 6.5.6 リリースで CXFS 1.1 製品をサポート
003	1999 年 12 月 IRIX 6.5.7 リリースで CXFS 製品をサポート
004	2000 年 3 月 IRIX 6.5.8 リリースで CXFS 製品をサポート
005	2000 年 6 月 IRIX 6.5.9 リリースで CXFS 製品をサポート
006	2000 年 9 月 IRIX 6.5.10 リリースで CXFS 製品をサポート
007	2001 年 1 月 IRIX 6.5.11 リリースで CXFS 製品をサポート
008	2001 年 3 月 IRIX 6.5.12 リリースで CXFS 製品をサポート

目次

このガイドについて xxi
関連ドキュメント xxi
出版物の入手方法 xxii
表記規則 xxiii
ご意見とお問い合わせ先 xxiii

1. CXFS とは 1
XFS と CXFS の比較 1
サポートされている XFS の機能 2
サポートされていない XFS の機能 3
CXFS を使用すべきケース 3
XFS を使用すべきケース 4
ネットワーク・ファイルシステムと CXFS ファイルシステムの比較 4
ネットワーク・ファイルシステム 4
CXFS ファイルシステム 5
機能 5
制約 6
クラスタ環境 6
用語 7
ノード 7
プール 7
クラスタ 7
メンバーシップ 7
上限量 8
プライベート・ネットワーク 8
メタデータ 9
メタデータ・サーバとメタデータ・クライアント 9
再設定 9
回復 9
ハードウェア・コンポーネント 11

メンバーシップ上限量 13
CXFS メンバーシップ上限量 13
上限量の計算とノードの重み 13
上限量の変更の例 14
ネットワークの分割の例 15
fs2d メンバーシップ上限量 17
ハードウェア・リセット 17
メタデータ・クライアント/サーバ・モデル 18
システム表示 19
デーモン 20
通信パス 22
読取りおよび書込み時のメタデータの流れ 24
ハードウェアおよびソフトウェアの必要条件とサポート 26
IRIS FailSafe 2.1 との同時実行 27
重み付けされた 2 ノード・クラスタでの回復と再設定の問題 32
CXFS クラスタ・マネージャのツール 33
GUI の概要 33
CXFS マネージャ 34
CXFS クラスタ表示 36
GUI の起動 39
CXFS クラスタ表示の起動 40
CXFS マネージャの起動 41
クラスタ・アイテムの詳細の表示 42
タスクの実行 42
タスクセットの使用 43
CLI の概要 43
ヘルプの利用 44
アクションの完了とキャンセル 44
スクリプト・ファイルの使用 44

CLI 内部からのシェルの呼出し 46

2. CXFS ソフトウェアの初期インストールとシステムの準備 47

必要なソフトウェアのインストール 47

システム・ファイルの設定 52

ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定 52

/etc/services の設定 54

/etc/config/cad.options の設定 54

/etc/config/fs2d.options の設定 55

（オプション）/.rhosts の設定 57

（オプション）/etc/exports の設定 58

（オプション）自動再起動用の設定 58

ネットワーク・インタフェースの設定 58

シリアル・ポートの設定 61

システムの再起動 62

テスト 62

プライベート・ネットワーク・インタフェースのテスト 62

シリアル接続のテスト 63

3. クラスタの初期設定 65

準備手順 65

ライセンスの確認 66

クラスタ・デーモンの開始 66

クラスタ・デーモンが実行されていることの確認 66

ノードのホスト名の判断 67

chkconfig フラグがオンであることの確認 68

GUI を使った設定 68

GUI の起動 69

GUI を使った新規クラスタの設定 71

GUI を使った新規ファイルシステムの設定 72

CLI を使った設定 73

4. GUI タスク・リファレンス 81

初期設定ガイド・タスクセット 82

GUI を使った CXFS の既存の FailSafe クラスタの設定 83

GUI を使ったノード・タスク 84

GUI を使ったノードの定義 84

GUI を使ったクラスタ内のノードの追加と削除 90

GUI を使ったノードのリセット 90

GUI を使ったノードの変更 91

GUI を使った FailSafe ノードの CXFS への切替え 93

GUI を使ったノードの削除 94

GUI を使ったノードの表示 94

GUI を使ったクラスタ・タスク 95

GUI を使ったクラスタの定義 95

GUI を使ったクラスタ定義の変更 96

GUI を使った FailSafe クラスタの CXFS への切替え 96

GUI を使ったクラスタの削除 97

GUI を使ったクラスタの表示 97

GUI を使った CXFS サービス・タスク 98

GUI を使った CXFS サービスの開始 98

GUI を使った CXFS サービス（通常 CXFS シャットダウン）の停止 98

GUI を使ったタイブレーカー・ノードの設定 99

GUI を使ったログの設定 99

GUI を使ったログ・グループ定義の表示 100

GUI を使ったログ・グループの設定 100

GUI を使ったファイルシステム・タスク 101

GUI を使ったファイルシステムの定義 101

GUI を使ったファイルシステムのマウント 102

GUI を使ったファイルシステムのアンマウント 103

GUI を使ったファイルシステムの変更 103
GUI を使ったファイルシステムの削除 104
GUI を使った診断タスクとエラー回復 104
GUI を使った接続性のテスト 104
GUI を使ったローカル・ノードのメンバーシップ取消し 105
GUI を使ったローカル・ノードのメンバーシップ許可 105

5. CLI タスク・リファレンス 107

CLI を使ったデフォルトの設定 108
CLI を使ったノード・タスク 109
CLI を使ったノードの定義 109
CLI を使ったノードの変更 113
CLI を使ったノードの重みの変更例 114
パーティションの設定例 115
CLI を使ったノードのリセット 116
CLI を使ったノードの CXFS または FailSafe への切替え 116
CLI を使ったノードの削除 117
CLI を使ったノードの表示 120
CLI を使ったクラスタ・タスクとファイルシステム・タスク 121
CLI を使ったクラスタの定義 121
CLI を使ったクラスタの変更およびファイルシステムの定義とマウント 124
CLI を使ったクラスタの CXFS または FailSafe への切替え 127
CLI を使ったファイルシステムのアンマウント 127
CLI を使った定義されたファイルシステムの再マウント 128
CLI を使ったファイルシステムの削除 130
CLI を使ったクラスタの削除 131
CLI を使ったクラスタの表示 133
CLI を使った CXFS サービス・タスク 135
CLI を使った CXFS サービスの開始 135
CLI を使った CXFS サービスの停止 135

CLI を使ったタイブレーカー・ノードの設定 136
CLI を使ったログの設定 137
CLI を使ったログ・グループ定義の表示 137
CLI を使ったログ・グループの設定 137
CLI を使ったログ・グループの変更 138
CLI を使った診断とエラー・タスク 139
CLI を使ったネットワークの接続性のテスト 139
CLI を使ったシリアル接続のテスト 140
CLI を使ったローカル・ノードのメンバーシップ取消し 140
CLI を使ったローカル・ノードのメンバーシップ許可 140
スクリプトの例 141

6. 管理とメインテナンス 143

マウント操作前後のスクリプトの実行 144
例: `cxfs-pre-mount` 145
例: `cxfs-post-mount` 145
例: `cxfs-pre-umount` 145
例: `cxfs-post-umount` 146
`fsr`(1m) の使用 146
`find`(1) と `crontab`(1) の使用 146
階層ストレージ管理 (HSM: Hierarchical Storage Management) システムの使用 147
ファイルシステムのメタデータ・サーバの発見 148
`cluster_status` を使用したメタデータ・サーバの発見 148
`clconf_info` を使用したメタデータ・サーバの発見 148
GUI を使用したメタデータ・サーバの発見 149
メタデータ・サーバの回復 151
クラスタ設定データベースと CXFS のシャットダウン 151
クラスタ設定データベースのシャットダウン 152
ノードのステータスとクラスタ設定データベースのシャットダウン 152
ノードがメタデータ・サーバの場合 153

クラスタ設定データベースの再起動 . . . 153
通常 CXFS シャットダウン . . . 153
ノードのステータスと通常 CXFS シャットダウン . . . 154
通常 CXFS シャットダウンを実行できない場合 . . . 154
通常 CXFS シャットダウン後のクラスタへの再設定 . . . 154
強制 CXFS シャットダウン . . . 154
ノードのステータスと強制 CXFS シャットダウン . . . 155
強制 CXFS シャットダウン後のクラスタへの再設定 . . . 155
リセット機能と強制 CXFS シャットダウン . . . 156
再起動時のクラスタ設置データベースの再起動の回避 . . . 156
ログ・ファイル管理 . . . 156
大容量のログ・ファイルのローテーション . . . 157
すべてのログ・ファイルのローテーション . . . 157
ボリューム管理 . . . 158
ディスク管理 . . . 158
バックアップ . . . 159
NFS . . . 159
Quota . . . 159
RealTime . . . 159
SAMBA . . . 159
ファイルシステムのメインテナンス . . . 160
ファイルシステムのマウント . . . 160
ファイルシステムのアンマウント . . . 161
ファイルシステムの拡張 . . . 161
ダンプと復元 . . . 162
クラスタ設定データベースのバックアップと復元 . . . 162
chkconfig フラグ . . . 162

7. クラスタとノードのステータスのモニタ . . . 165

ログ・ファイルのモニタ . . . 165

システムのステータスのモニタ 166
cluster_status を使ったシステムのステータスのモニタ 166
GUI を使ったシステムのステータスのモニタ 167
ノードのステータス 168
cluster_status を使ったノードのステータスのモニタ 169
GUI を使ったノードのステータスのモニタ 169
CLI を使ったノードのステータスの照会 170
CLI を使ったシステム・コントローラへの ping の実行 171
CLI を使ったリセット・シリアル回線のモニタ 171
クラスタのステータス 171
cluster_status を使ったクラスタのステータスの照会 172
GUI を使ったクラスタのステータスの照会 172
CLI を使ったクラスタのステータスの照会 172

8. トラブルシューティング 173

トラブルシューティング方法 173
ツールについての理解 173
物理ストレージ・ツール 174
クラスタ設定ツール 174
クラスタ制御ツール 175
ネットワーク・ツール 176
クラスタ/ノードのステータス・ツール 176
パフォーマンス・モニタ・ツール 177
カーネルのステータス・ツール 178
ログ・ファイル 179
クラスタのステータスの識別 179
問題の検出 181
問題の回避 183
適切な開始 183
設定の統一性 183

残っているクラスタの削除 183
強制シャットダウン後の CXFS の再開 184
fs2d メンバーシップ上限量の安定性 184
GUI の使用 185
ログ・ファイル名とサイズ 185
Netscape と Brocade スイッチの GUI 185
書込まれていないエクステントの追跡と排他書込みトークンに関するパフォーマンスの問題 186
書込まれていないエクステントの追跡 186
排他書込みトークン 186
crontab ファイルによる過剰なファイルシステム・アクティビティの回避 187
システム容量の効果的な使用 188
ノード ID またはクラスタ ID を変更する前の再起動 188
使用されていないノードの削除 188
一般的な問題 188
ファイルシステムにアクセスできない 188
GUI が実行されない 189
ログ・ファイルがディスク容量を消費しすぎる 189
ノードを定義できない 189
システムがハングする 189
ノードは検出されるが、メンバーシップに設定されない 190
セル ID カウントと“ Membership Delivered” メッセージ 190
ログインできない 190
エラー・メッセージの理解 191
通常メッセージ 191
clconfd デーモンが異常終了する場合 193
論理スワップ・スペース不足 193
クラスタ名 ID 欠如エラー 194
CXFS メンバーシップの取消し 194
ライセンス・エラー 195

IP アドレス・エラー 195
SYSLOG エラー 196
cli エラー 197
clconfd エラー 198
crsd エラー 200
cmond エラー 201
fs2d エラー 201
一般メッセージ 203
ログ・ファイル・エラー 203
cad メッセージ 204
cli メッセージ 205
crsd エラー 206
fs2d エラー 207
修正処置 208
CXFS サービスの再開 208
クラスタ設定データベースのクリア 208
再起動 209
2 ノード・クラスタの回復 210
クラスタに再設定しない再起動 211
クラスタ・インフラストラクチャ・デーモンの停止と再開 212
クラスタ設定データベースの再作成 212

付録A. 初期設定チェックリスト 213

用語集 215

索引 221

図一覧

図1-1 再設定と回復 10
図1-2 リセットを使用した 4 ノード・クラスタの例 11
図1-3 リセットを使用した 2 ノード・クラスタの例 12
図1-4 ネットワークの分割による上限量とアクティブ・メタデータ・サーバの変化 16
図1-5 クラスタ設定データベースに対する読取り/書込みアクション 22
図1-6 クラスタ内にないノードの通信パス 23
図1-7 書込み時のメタデータの流れ 24
図1-8 クライアント A での書込み後のクライアント B での読取り時のメタデータの流れ . . . 25
図1-9 クライアント A での読取り後のクライアント B での読取り時のメタデータの流れ . . . 26
図1-10 同時実行時のクラスタ設定データベースに対する読取り/書込みアクション . . . 31
図1-11 「CXFS マネージャの概要」ウィンドウ 35
図1-12 「CXFS クラスタ表示(CXFS Cluster View)」ウィンドウ 37
図1-13 クラスタ表示の詳細 38
図3-1 CXFS マネージャ 70
図4-1 ノード定義の例 88
図4-2 システム・コントローラ設定の例 89
図6-1 メタデータ・サーバが表示されているウィンドウ 150
図7-1 ノード・タイプが表示されたノードの詳細の例 170

表一覧

表1-1 CXFS のデーモンとスレッド 20
表8-1 SYSLOG エラー・メッセージの形式 196
表8-2 ログ・エラー・メッセージの形式 204

このガイドについて

このガイドは、ストレージ・エリア・ネットワーク (SAN: Storage Area Network) で実行される CXFS について説明しています。ここでは、『IRIX Admin: Disks and Filesystems』で説明されている XFS ファイルシステムをすでに十分に理解しており、『XVM Volume Manager Administrator's Guide』を利用できることを想定しています。

CXFS クラスタのインストールと設定を行う前に、このマニュアル全体、特に 173 ページの 第8章「トラブルシューティング」をよくお読みください。

関連ドキュメント

以下のドキュメントには、補足情報が記載されています。

- 『EL Serial Port Server Installation Guide』(Digi Corporation によって提供されています)
- 『EL Serial Port Server Installation Guide Errata』
- 『FDDIXPress Administration Guide』
- 『IRISconsole Administrator's Guide』
- 『IRIS FailSafe Version 2 Administrator's Guide』
- 『IRIX 6.5 Installation Instructions』
- 『IRIX Admin: Disks and Filesystems』
- 『NIS Administrator's Guide』
- 『Performance Co-Pilot User's and Administrator's Guide』
- 『Performance Co-Pilot Programmer's Guide』
- 『Personal System Administration Guide』
- 『SGI TP 9400 RAID Owner's Guide』
- 『SGI TP9400 RAID Administration Guide』
- 『SGI Total Performance 9100 Storage System Owner's Guide』
- 『XVM Volume Manager Administrator's Guide』

CXFS では以下のマン・ページが提供されています。

- cdbBackup(1m)
- cdbRestore(1m)
- cmond(1m)
- cluster_mgr(1m)
- cluster_status(1m)
- crsd(1m)
- haStatus(1m)
- ha_cilog(1m)
- ha_cmsd(1m)
- ha_exec2(1m)
- ha_gcd(1m)
- ha_ifd(1m)
- ha_ifdadmin(1m)
- ha_macconfig(1m)
- ha_srmd(1m)

出版物の入手方法

SGI のドキュメントは、下記の SGI Technical Publications Library サイトで入手可能です。

http://techpubs.sgi.com

表記規則

このドキュメントでは、以下の表記規則が使用されています。

表記規則	意味
`command`	この固定スペース・フォントは、コマンド、ファイル、ルーチン、パス名、シグナル、メッセージ、プログラミング言語の構造などのリテラル項目を示します。
変数	イタリック体は、変数のエントリ、および定義される語や概念を示します。
`user input`	この太字体の固定スペース・フォントは、対話型セッションでユーザが入力するリテラル項目を示します。出力は、太字体ではない固定スペース・フォントで示されます。
[ ]	コマンドやディレクティブ行のオプション部分は、角括弧で囲みます。

ご意見とお問い合わせ先

このマニュアルの技術的正確性、内容、または構成についてご意見等ございましたら、弊社までお問い合わせください。コメントいただくマニュアルのタイトルとドキュメント番号も必ず一緒にお聞かせいただくようお願いいたします。オンラインの場合、ドキュメント番号はマニュアルの最初の部分に記載されています。印刷マニュアルの場合は、各ページの下部にドキュメント番号が記載されています。

ご連絡の際は、以下のいずれかの方法をご利用いただけます。

- 電子メールの場合は、

  `techpubs@sgi.com`

  までお送りください。

- 次の Technical Publications Library の World Wide Web ページからの場合は、「Feedback」オプションをクリックしてください。

  `http://techpubs.sgi.com`

- お客様相談窓口までご連絡いただき、SGI の問題追跡システムへの入力をお申付けください。

- 郵送の場合は、次の住所までお送りください。
  Technical Publications
  SGI
  1600 Amphitheatre Pkwy., M/S 535
  Mountain View, California 94043–1351, USA
- FAX の場合は、+1 650 932 0801 の “Technical Publications” 宛にお送りください。

いただいたコメントには迅速確実に対応いたします。

CXFS とは

CXFS を使用することで、コンピュータのグループは、高いパフォーマンスを保ったまま、一貫性のある方法で大量のデータを共有できます。CXFS は、ファイバ・チャネルなどのストレージ・エリア・ネットワーク (SAN: Storage Area Network) ディスクで動作します。CXFS と IRIS FailSafe は、同じインフラストラクチャを使用します。

この章では、以下の内容について説明します。

- 「XFS と CXFS の比較」
- 4 ページの「ネットワーク・ファイルシステムと CXFS ファイルシステムの比較」
- 6 ページの「クラスタ環境」
- 24 ページの「読取りおよび書込み時のメタデータの流れ」
- 26 ページの「ハードウェアおよびソフトウェアの必要条件とサポート」
- 27 ページの「IRIS FailSafe 2.1 との同時実行」
- 32 ページの「重み付けされた 2 ノード・クラスタでの回復と再設定の問題」
- 33 ページの「CXFS クラスタ・マネージャのツール」

メモ: CXFS クラスタのインストールと設定を行う前に、このマニュアル全体、特に 173 ページの 第8章「トラブルシューティング」をよくお読みください。

XFS と CXFS の比較

CXFS は、高いパフォーマンスを持つ SGI の XFS ファイルシステムに基づいており、XFS と同じファイルシステム構造を使用します。CXFS ファイルシステムは、標準的な XFS ファイルシステムの作成に使用するコマンドと同じ `mkfs`(1m) コマンドを使用して最初に作成します。

XFS ファイルシステムと CXFS ファイルシステムの最も大きな違いは、ファイルシステムのマウントと管理の方法です。

- XFS の場合
  - ファイルシステムは、mount(1m) コマンドで直接マウントするか、/etc/fstab のエントリにより起動時にシステムにマウントさせるか、またはファイルシステム・マネージャを利用してマウントします。
  - ファイルシステムは 1 つのシステム上に 1 つだけ存在します。
  - /etc/fstab ファイルには、ファイルシステムに関する静的な情報が含まれます。
- CXFS の場合
  - ファイルシステムは、CXFS コマンド・ライン・インタフェース (CLI: Command Line Interface) またはグラフィカル・ユーザ・インタフェース (GUI: Graphical User Interface) を使用してマウントします。CXFS ファイルシステムは、CXFS 管理ソフトウェアによってクラスタ全体にマウントされます。
  - ファイルシステムは、クラスタ内のすべてのシステム（ノード）に存在します。メタデータ（ファイルの名前、サイズ、場所、アクセス権など、ファイルを記述する情報）の更新の調整は、クラスタ内のすべてのノードに代わってメタデータ・サーバが行います。
  - 情報は /etc/fstab ファイルには格納されません（CXFS ファイルシステムは、実際には /etc/mtab ファイルに現れます）。

サポートされている XFS の機能

XFS の機能のうち CXFS にもあるものは以下のとおりです。

- 信頼性が高く高速（1 秒以下の精度）なログベースのファイルシステムの回復機能
- 64 ビットのスケーラビリティによる 1 ファイルあたり最大 900 万テラバイト（9 エクサバイト）のファイル・サイズ上限
- 高速性（高帯域幅、高トランザクション・レート、および高速なメタデータ操作）
- リアルタイム・ファイルシステム
- メタデータ領域の動的な割当て
- Quota。Quota のマウント・オプションは、ファイルシステムのすべてのマウントで同じでなければなりません。Quota は、クラスタ内のどこからでも通常の XFS ファイルシステムと同じように管理できます。
- Filesystem reorganizer（断片化解消プログラム）。このプログラムは、CXFS メタデータ・サーバから実行する必要があります。fsr_xfs(1M) のマン・ページを参照してください。

CXFS は、これらの基本的な XFS の機能を保ったまま、I/O をディスクとホストの間で直接分散します。効率的な XFS I/O パスでは非同期バッファ技術が採用されており、できるかぎり書込みを遅らせて不必要な物理 I/O を避けます。これにより、ファイルシステムは、データ領域を効率的、かつ多くの場合連続的に割当てることができます。一般的には、データは大容量の連続チャンクに割当てられるので、持続的に高い帯域幅が得られます。

XFS のディレクトリ構造は B-ツリーに基づいているので、ディレクトリ内のファイルの数が数万～数十万に増加しても高速な応答時間を維持できます。

XFS の機能についての詳細は、『IRIX Admin: Disks and Filesystems』を参照してください。

サポートされていない XFS の機能

CXFS でサポートされていない XFS の機能は以下のとおりです。

- I/O レート保証
- ファイルへのスワップ
- マウント・オプションのネスト(CXFS ファイルシステム上にファイルシステムをマウントすることはできません)

CXFS を使用すべきケース

CXFS は、以下の条件下で最高のパフォーマンスを発揮します。

- データ I/O 操作が 16 Kバイトを超える場合
- 1 つのプロセスだけが開いているファイルに対して読取りや書込みを行う場合
- ファイルを開いているすべてのプロセスが同じホスト上に存在するときに、そのファイルに対して読取りや書込みを行う場合
- 複数のホスト上の複数のプロセスが同じファイルを読取る場合
- 複数のホスト上の複数のプロセスのために、ダイレクト・アクセス I/O を使用して、ファイルに対して読取りや書込みを行う場合
- 大容量のファイルや大規模なファイル・アクセスを使用している場合

ほとんどのファイルシステム負荷では、上記のシナリオは大容量のファイル・アクセスを意味します。そのため、CXFS は高速なローカル・ファイル・パフォーマンスを実現できます。また、CXFS は、メタデータ I/O の量よりもデータ I/O の量が多い場合にも便利です。CXFS ではデータがネットワークを経由しないため、NFS よりも高速です。

XFS を使用すべきケース

操作によっては、ローカル XFS ファイルシステムよりも CXFS の方が低速になる場合があります。たとえば、メタデータ・サーバを経由したメタデータ操作は、ローカル・ファイルシステム上での操作よりも完了に時間がかかることがあります。以下に、メタデータのトランザクションの例を挙げます。

- ファイルの開閉
- ファイル・サイズの変更（一般的にはファイル・サイズの増加）
- ファイルの作成と削除
- ディレクトリの検索

また、バッファ I/O を使用して同じファイルへの読書きを行っている複数のホスト上の複数のプロセスも、ローカル・ファイルシステムの使用時より低速になる可能性があります。パフォーマンスにこのような差が生じるのは、分散されているファイル・バッファの間でコヒーレンシを維持するためです。バッファに格納されている共有ファイルに書き込みを行うと、ほかのホストでバッファされている（そのファイルに関する）データが無効になってしまいます。

パフォーマンスの高い分散型の共有読取りや書込みが頻繁に必要なアプリケーションでは、バッファ I/O の代わりにダイレクト・アクセス I/O を使用して高速なローカル・ファイル・パフォーマンスを実現できます。たとえば、分散型の並列データベースでは、ダイレクト・アクセス I/O が使用される傾向があります。

ネットワーク・ファイルシステムと CXFS ファイルシステムの比較

ネットワーク・ファイルシステムと CXFS ファイルシステムで実行される機能の多くは同じですが、ここでは、パフォーマンスと機能の面での重要な違いについて説明します。

ネットワーク・ファイルシステム

ローカル・エリア・ネットワーク (LAN: Local Area Network) 上でリモート・ファイルにアクセスすると、ローカル・ファイルにアクセスした場合よりも大幅に低速になる場合があります。トランザクション・レート（1 秒あたりの I/O）および帯域幅（1 秒あたりの M バイト）を大きく低下させる遅れの原因は、ネットワーク・ハードウェアとソフトウェアです。LAN ベースのネットワーク・ファイルシステムのクライアント/サーバ・アーキテクチャでは、このような遅れを回避するのは困難です。これらの遅れは、LAN の帯域幅とレーテンシ上の制限、およびデータ・サーバを経由する共有パスが原因で発生します。

ほとんどの既存の共有ファイルシステムの速度は、LAN の帯域幅の制限を受けます。10～1000 ビット Ethernet の上限は、それぞれ 1～100 M バイト/秒です。これは、ローカル・ディスクまたは共有ディスクに接続した複数のディスク・チャンネル全体で得られる帯域幅よりも数段遅いことになります。帯域幅のレートは、何階層ものネットワーク・プロトコルとサーバ・ソフトウェアによっても制限されます。

複数のクライアントが、中央のファイル・サーバを経由する必要があるデータを待っている場合は、共有ファイル・サーバがパフォーマンス上のボトルネックになることがあります。たとえば、ネットワーク・ファイルシステム (NFS: Network File System) および Samba サーバは、サーバに接続されているディスクからデータを読取って UDP/IP または TCP/IP パケットにコピーしてから、LAN を通じてクライアント・システムにデータを送信します。このサーバに多くのクライアントが同時にアクセスすると、サーバの応答性が低下します。

CXFS ファイルシステム

CXFS はクラスタ XFS ファイルシステムで、ネットワーク・ファイルシステムと同様の論理的なファイル共有が可能ですが、パフォーマンスと機能の面で大きな利点があります。CXFS は SAN 上で動作しますが、クラスタ内の各コンピュータ・システムは、共有されているディスクのセットに対して直接の高速なデータ・チャンネルを利用できます。

機能

CXFS には、以下のような独自の機能があります。

- ディスク・アクセスのためのピアツーディスク・モデル。共有ファイルは、クラスタ内のすべてのコンピュータ・システムでローカル・ファイルとして扱われます。各システムは、このディスクに対してローカル・ディスクと同等の速度で読取りと書込みを行うことができます。データは、ディスクから、I/O を要求したシステムに直接渡され、データ・サーバや LAN を経由することはありません。データ・パスの点では、各システムは SAN 上のピアであり、どのシステムでも、共有ディスクに高速に直接アクセスできるデータ・パスを利用できます。

  したがって、SAN にディスク・チャンネルやストレージを追加することで、帯域幅を拡張できます。大規模な IRIX システムでは、G バイトにまで帯域幅を拡張でき、さらに毎秒数 10 G バイトにも拡張可能です。これは、1～100 M バイト/秒の LAN 上を流れるネットワーク・ファイルシステムと比較するとはるかに優れています。

  このピアツーディスクのデータ・パスにより、大部分の LAN ベースの共有ファイルシステムに見られるファイル・サーバのデータ・パスのボトルネックも解消されます。

- 各ホスト・システムは、ローカルに接続されているディスクと同じように共有ディスクをバッファできます。これらの分散バッファのコヒーレンシは CXFS によって維持されるので、XFS ファイルシステムの先進的なバッファ技術をそのまま利用できます。

- ファイルシステムのフラットで単一のシステム表示。これは、ファイルシステムを共有しているすべてのホストで同一で、特定のホストに依存することはありません。パス名は通常の POSIX パス名 (`/u/`*`username`*`/`*`directory`*) です。

  メタデータ・サーバをあるノードから別のノードに移動したり、メタデータ・サーバの名前を変更したり、メタデータ・サーバの追加や交換を行っても、パスは変わりません。このため、管理者にとってもユー

ザにとってもストレージの管理が簡単になります。1 つの対称マルチプロセッサ (SMP: Symmetric Multiprocessor) ホスト上の複数のプロセスでも、複数のホストに分散されているプロセスでも、ファイルシステムは同じように表示され、各ホストのパフォーマンスも同等です。

パス名にファイル・サーバの名前が含まれる一般的なネットワーク・ファイルシステムとはこの点が異なります。この違いは、指定されているサーバを通じてディスクにアクセスする LAN ファイルシステムに余分な階層があるのと比べて、ダイレクトツーディスク I/O を使用する SAN のアーキテクチャが単純であることによるものです。

- POSIX、System V、および BSD インタフェースを含む完全な IRIX ファイルシステム・インタフェース。ここには、強制および任意レコード・ロックなどのファイルシステム・セマンティックスが含まれており、特殊なレコード・ロック・ライブラリは必要ありません。

制約

CXFS には、以下のような制約があります。

- 一部のファイルシステム・セマンティックスは適切ではないため、共有ファイルシステムではサポートされません。たとえば、root ファイルシステムは共有ファイルシステムとして適切ではありません。Root ファイルシステムは特定のホストに属しており、各特定のホストの特性に合わせてシステム・ファイルが設定されています。
- 名前付きパイプを使用するすべてのプロセスが同じノードに存在する必要があります。

クラスタ環境

この節では、以下の内容について説明します。

- 7 ページの「用語」
- 11 ページの「ハードウェア・コンポーネント」
- 13 ページの「メンバーシップ上限量」
- 17 ページの「ハードウェア・リセット」
- 18 ページの「メタデータ・クライアント/サーバ・モデル」
- 19 ページの「システム表示」
- 20 ページの「デーモン」
- 22 ページの「通信パス」

用語

この節では、CXFS の理解に必要な用語を定義します。215 ページの Glossary も参照してください。

ノード

ノードとは個々のコンピュータです。すべてのノードで、同バージョンの IRIX オペレーティング・システム (OS: Operating System) が実行されている必要があります（6.5.12f の場合は 6.5.11f と 6.5.12f など）。ノードは、ストレージ・システムとクラスタ内のノードを接続する SAN に接続されています。ノードが属することのできるクラスタは 1 つだけです。

この意味でのノードという用語は、Origin システムのノードとは異なる意味を持ちます。

プール

プールは、ネットワークによって相互に結合され、クラスタ設定データベースでノードとして定義されているノードのセット全体です。通常、これらのノードは互いに近い位置にあり、同じ目的に使用されます。プール内の各ノードには、複製されたクラスタ設定データベースが格納されています。

クラスタに追加できるノードはすべてプールの一部ですが、プール内のすべてのノードをクラスタの一部にする必要はありません。プールは 1 つだけです。ほかのプールを存在させることはできますが、各プールはお互いから分離した状態になり、ノードやクラスタ定義は共有されません。

クラスタ

クラスタは、クラスタとして定義されているプール内のノードのセットです。クラスタは単純な名前で識別されます。この名前は、プール内で一意でなければなりません。たとえば、クラスタとノードで同じ名前を使用することはできません。

クラスタ内のすべてのノードはプールにも存在しますが、プール内のすべてのノードがクラスタに存在するとはかぎりません。つまり、クラスタは、プール内のノードのサブセットで構成されている場合があります。各プールのクラスタは 1 つだけです。

メンバーシップ

メンバーシップには、以下のタイプがあります。

- CXFS メンバーシップ。これは、カーネル空間メンバーシップとも呼ばれ、クラスタでアクティブにファイルシステムを共有している、クラスタ内の CXFS ノードのグループです。クラスタで定義されているノードのサブセットの場合があります。ノードは起動処理中に CXFS メンバーシップを要求し、要求が受付けられると、クラスタのファイルシステムを共有できます。

- `fs2d` メンバーシップ。これは、ユーザ空間メンバーシップまたは fs2d メンバーシップとも呼ばれ、`fs2d` にアクセス可能で、クラスタ設定データベースの更新を受信できるプール内のノードのグループです。プール内で定義されたノードのサブセットである場合があります。

CXFS メンバーシップおよび `fs2d` メンバーシップは、*FailSafe* メンバーシップとは異なります。FailSafe についての詳細は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

上限量

上限量とは、クラスタを形成するために必要なノードの数のことで、メンバーシップに応じて以下のように変わります。

- CXFS メンバーシップの場合
  - 初期メンバーシップを形成するには、クラスタでノードの過半数 **(>50%)** が重み付けされている必要があります。
  - 既存のメンバーシップを維持するには、クラスタでノードの半数 **(50%)** が重み付けされている必要があります。
- `fs2d` メンバーシップの場合、クラスタの形成と維持には、プール内のノードの **50%** が必要です。

詳細については、13 ページの「メンバーシップ上限量」を参照してください。

プライベート・ネットワーク

プライベート・ネットワークは、クラスタ通信専用のネットワークで、管理者はアクセスできますが、ユーザはアクセスできません。

- クラスタ・ソフトウェアは、プライベート・ネットワークを使用して、クラスタ・データベースを設定を機能させるために必要なハートビート・メッセージとコントロール・メッセージを送信します。ハートビート・メッセージの受信に遅れがあった場合、クラスタ・ソフトウェアは、ノードが応答していないと判断して CXFS メンバーシップを取消し、設定によってはノードをリセットまたは切断することがあります。
- ネットワーク装置を再起動すると、クラスタ内のノードの通信が切断される場合があります。ノード間の通信が切断されると、そのクラスタはパフォーマンス低下状態になります。プライベート・ネットワークを使用するとネットワークのトラフィックが抑えられるため、不要なリセットや切断を回避するために役立ちます。

したがって、パブリック・ネットワークのパフォーマンスやセキュリティ特性が原因でクラスタに問題が発生する可能性があるという理由や、CXFS ハートビートがタイミングに大きく依存する（わずかな変動でも問題が発生する可能性があります）という理由から、プライベート・ネットワークは必須です。また、パブリック・ネットワークのメッセージング・プロトコルではパケットの覗き見やなりすましを防止できないので、アクセスが制限されたネットワークの方が、ユーザ・アクセスのあるネットワークよりも安全です。さらに、すべてのノードを同じローカル・ネットワーク・セグメント上に配置することが推奨されています。

メモ:プライベート・ネットワークにネットワーク上の問題がある場合は、CXFS を使用する前に解決してください。

ネットワーク・セグメントとパーティション設定の詳細については、15 ページの「ネットワークの分割の例」を参照してください。

メタデータ

メタデータは、ファイルの名前、サイズ、場所、アクセス権など、ファイルを記述する情報です。通常、メタデータは、XFS 内にある 1 ファイルあたり 512 バイト程度の小さなファイルです。メタデータは、ファイルの内容であるデータとは異なります。データのサイズは、数 Mバイトまたは数 Gバイトになる場合があります。

メタデータ・サーバとメタデータ・クライアント

メタデータ・サーバは、クラスタ内のその他のノードに代わって、ファイルシステムのためにメタデータを調整します。その他のノードは、メタデータ・クライアントと呼びます。データは、メタデータと異なり、ホストとディスクの間を直接移動します(ピアツーディスク)。

メモ:メタデータ・サーバおよびメタデータ・クライアントを、ネットワーク・ファイスシステムで使用される従来のデータ・パス型のクライアント/サーバ・モデルと混同しないように注意してください。プライベート Ethernet ネットワーク上でメタデータ・サーバを通じて送信されるのはメタデータ情報だけです。データは、ファイバ・チャネル接続を通じてメタデータ・クライアントのディスクに対して直接送受信されます。

各ファイルシステムでアクティブなメタデータ・サーバは 1 つだけです。

再設定

再設定とは、管理操作のために、あるノードから別のノードにメタデータ・サーバを移動するプロセスのことです。最初のノード上のその他のサービスが中断されることはありません。

回復

回復とは、あるノード上のサービスが中断した場合に、最初のノードから別のノードにメタデータ・サーバが移動されるプロセスのことです。

図1-1 に、再設定と回復の違いを示します。

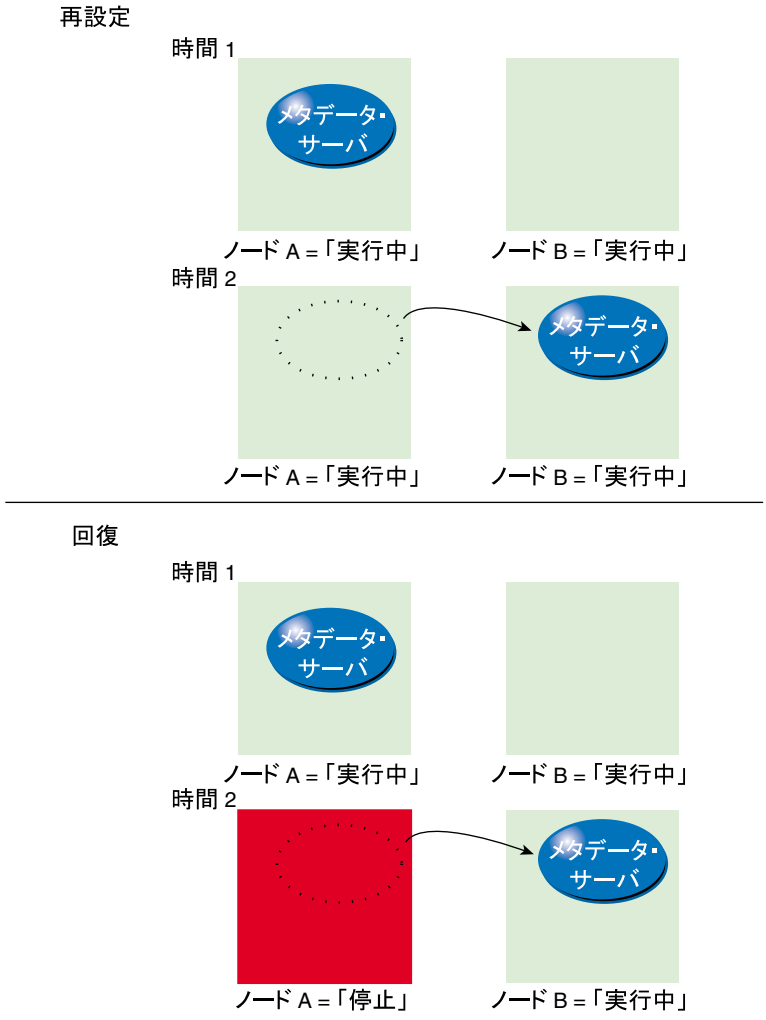
再設定
時間 1
メタデータ・サーバ
ノード A =「実行中」
ノード B =「実行中」
時間 2
メタデータ・サーバ
ノード A =「実行中」
ノード B =「実行中」
回復
時間 1
メタデータ・サーバ
ノード A =「実行中」
ノード B =「実行中」
時間 2
メタデータ・サーバ
ノード A =「停止」
ノード B =「実行中」

図1-1 再設定と回復

ハードウェア・コンポーネント

図1-2 に、リセット機能と Ethernet シリアル・ポート・マルチプレクサを使用した 4 ノード・クラスタにおける CXFS のハードウェア・コンポーネントの例を示します。パフォーマンスおよび制御の点から、ハブではなくスイッチが推奨されます（この図にはユーザ・ネットワークは示されていません）。

メモ: データの整合性を保証するため、特に重み付けされた 2 ノード・クラスタでは、リセット機能の使用が強く推奨されています。IRIS FailSafe ではリセット機能は必須です。

同様の設定では、Ethernet マルチプレクサの代わりに SCSI マルチプレクサを使用できます。リセット接続の接続設定は、IRIS FailSafe と同じです。詳細については、日本 SGI の営業担当員、もしくは営業フリーダイヤルまたは管理サービスまでお問い合わせください。

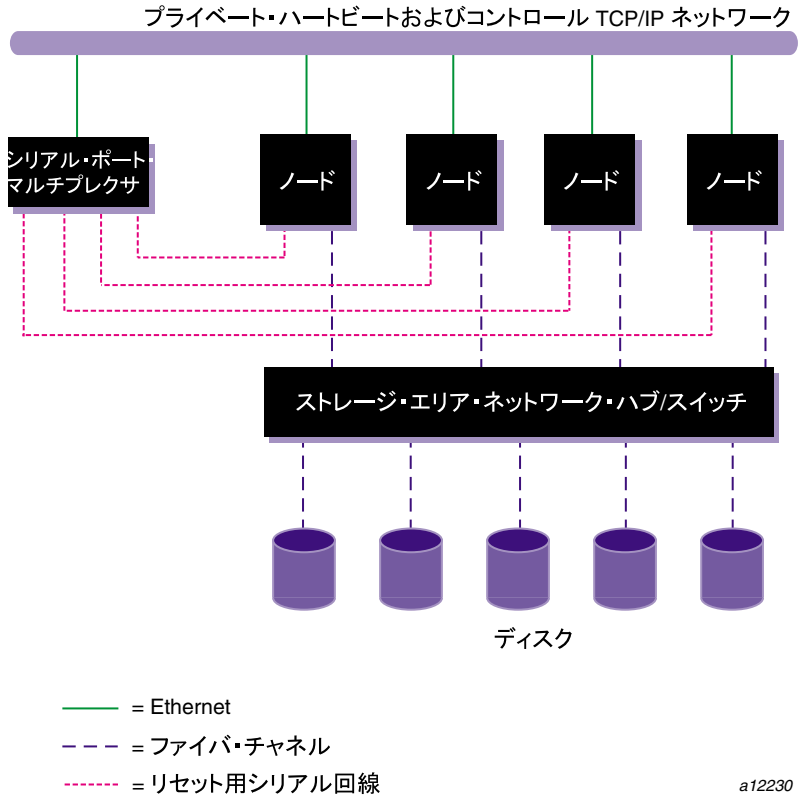

図1-2 リセットを使用した 4 ノード・クラスタの例

図1-3 に、リセット用の直接シリアル回線を使用する、重み付けされた 2 ノード・クラスタを示します（この図にはユーザ・ネットワークは示されていません）。

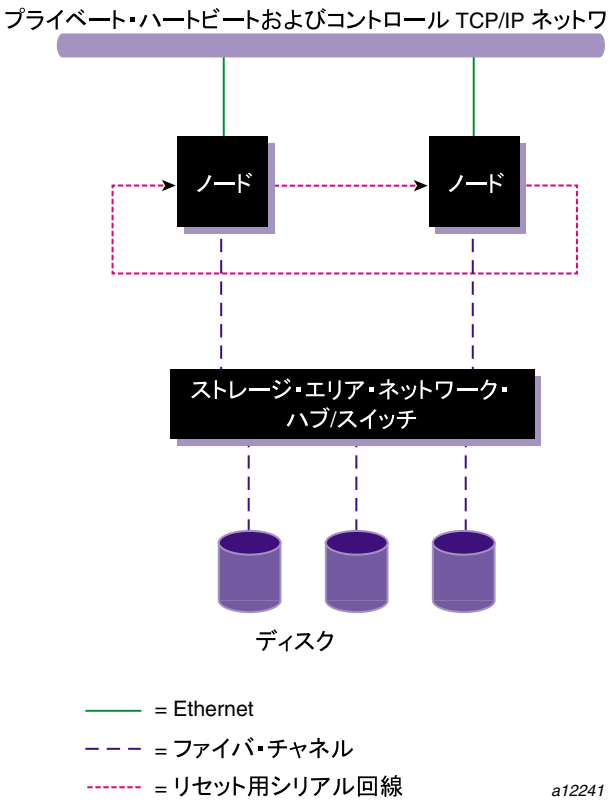

図1-3 リセットを使用した 2 ノード・クラスタの例

CXFS では、各ノードが SAN 上の論理ボリュームに直接アクセスできるので、パフォーマンスが向上します。各ノードは、NFS を使用したソリューションよりも密接に論理ボリュームに結合されています。データの整合性を保証するため、各ファイルシステムにメタデータ・サーバがあります。18 ページの「メタデータ・クライアント/サーバ・モデル」を参照してください。CXFS ソフトウェアにより、メタデータ・サーバが CXFS メンバーシップとアクセス上の問題を判断できるインフラストラクチャが提供されます。

メンバーシップ上限量

以下のように、メンバーシップの各タイプには別々の上限量のメカニズムがあります。

- 「CXFS メンバーシップ上限量」
- 17 ページの「fs2d メンバーシップ上限量」

CXFS メンバーシップ上限量

設計上、アクティブなメタデータ・サーバは 1 つのファイルシステムにつき 1 つだけです。ただし、クライアントとメタデータ・サーバの間でメタデータ情報を転送するために使用されるハートビート/コントロール・ネットワークに問題があると、問題が発生する可能性があります。ネットワーク異常などの何らかの理由でハートビート/コントロール・ネットワークが半分に分割されると、ネットワークが 2 つの小さなネットワーク(セグメント)になってしまうことがあります。このような場合は、CXFS メンバーシップ上限量によって、一方のメタデータ・サーバだけが SAN 上の CXFS ファイルシステムのメタデータの部分に書込みを行うようになります。

上限量の計算とノードの重み

CXFS メンバーシップ上限量は、CXFS メンバーシップに参加するノードの重みの合計を、クラスタ内で定義されているすべてのノードの合計の重みと比較することによって計算されます。

ノードのデフォルトの重みは 1 です。すべてのノードの重みが 1 の場合、初期 CXFS メンバーシップ上限量の計算結果は、原則的にノードの数の過半数 (>50%) になります。

クラスタが、1 つまたは複数の CXFS メタデータ・サーバと CXFS クライアントだけの複数のノードで構成されている場合は、クライアントの状態とは無関係に、メタデータ・サーバとなる可能性のあるノードだけで上限量が構成されるようにノードの重みを設定できます。この場合、メタデータ・サーバの重みは 1 に、クライアントの重みは 0 に定義する必要があります。重みが 0 のクライアントは、単体では CXFS メンバーシップを形成できません。この場合、CXFS メンバーシップには、メタデータ・サーバとなる可能性のあるノードの上限量が必ず必要です。

メモ: 少なくとも 1 つのノードのメンバーシップの重みが 0 より大きくなければなりません。また、メタデータ・サーバとなる可能性のあるノードのメンバーシップの重みも、0 より大きくなければなりません。

0 または 1 以外のメンバーシップの重みの値は推奨されません。

初期 CXFS メンバーシップ上限量では、クラスタを形成するには、重み付けされたノードの過半数 (>50%) が使用可能でなければなりません。すべてのノードの重みが 1 の大規模なクラスタの場合、クラスタを形成するには、大部分のノードの重みが 0 の場合よりもさらに多くのノードが使用可能でなければなりません。既存の CXFS メンバーシップ上限量を維持するには、クラスタで定義されている重み付けされたノードの半数 (50%) が必要です。

ハードウェア・リセットを使用しない場合は、タイブレーカー・ノードを設定して、ネットワークが分割されるのを回避してください。

メモ: データの整合性を保証するため、特に重み付けされた 2 ノード・クラスタでは、リセット機能の使用を強くお薦めします。IRIS FailSafe ではリセット機能は必須です。

タイブレーカー・ノードは、CXFS で特に識別されるノードで、クラスタ内のちょうど半数のノードが動作していて相互に通信できる場合に、クラスタの FailSafe クラスタ・メンバーシップを計算する処理で使用されます。デフォルトのタイブレーカー・ノードはありません。

ただし、重み付けされた 2 ノード・クラスタ内のタイブレーカー・ノードに異常が発生したり、管理者が CXFS サービスを停止した場合は、その他のノードによって強制シャットダウンが実行され、CXFS ファイルシステムはすべてアンマウントされます。

タイブレーカー・ノードがメタデータ・サーバとして動作できるよう、タイブレーカー・ノードの重みは 0 より大きくなければなりません。

CXFS ハートビート/コントロールに使用されているネットワークが半分に分割された場合は、メタデータ・サーバの上限量を持っているネットワークの部分と、タイブレーカー・ノードだけがクラスタに残ります。上限量の一部ではないハートビート/コントロール・ネットワークの部分にあるノードは、クラスタからシャットダウンされます。したがって、CXFS ハートビート/コントロール・ネットワークが半分に分割された場合は、SAN を通じて同時に同じ CXFS メタデータにアクセスするハートビート/コントロール・ネットワークのどちらの部分にも、アクティブなメタデータ・サーバがなくなります。

CXFS メンバーシップは、通常のメッセージとハートビートでそれ自体をモニタします。異常が発生した場合、違反ノードは CXFS メンバーシップから削除されるかまたはリセットされ、以降はこれらのノードが共有リソースにアクセスすることはできなくなります。CXFS メンバーシップから削除されたことを通信異常の発生によって検出したノードは、共有ディスクへのアクセスを強制的に終了し、最初にリセットされなかった場合は CXFS サービスを停止します。

CXFS メンバーシップで可能なノードの数を変更して、新しいノードを組込めるようにクラスタを拡大したり、CXFS メンバーシップから除外されたノードを削除することができます。現在停止中で今後も使用不能なノードを削除すると、ノードの重みが 1 の場合は、上限量の形成に必要な重みが減ります。

上限量の変更の例

以下に、CXFS メンバーシップ上限量を変更する場合の例を示します。

1 つのプライベート・ネットワーク上に 6 ノード（A、B、C、D、E、および F）のプールがあり、各ノードは重みが 1 で以下の状態だとします。

- C はタイブレーカー・ノード
- A、B、C、D、および E は CXFS クラスタのメンバー

この条件の場合、初期 CXFS メンバーシップ上限量に必要なノードの最低数は 3 つのノード (>50%) です。

B がシャットダウンしたり、CXFS メンバーシップから外れると、クラスタ内の残りのノードの数は A、C、D、E の 4 つになります。この状態では、CXFS メンバーシップ上限量を維持するための要件 (50%) がまだ満たされているので、このクラスタは引続き使用可能です。

ネットワークの分割の例

前の例の続きとして、図1-4 に、ルータがダウンしてハートビート/コントロール・ネットワークが実質的に 2 つに分割された状態を示します。ネットワーク・セグメント 2 上のノード（ノード D および E）にはタイブレーカー・ノードが含まれていないので上限量がありません。したがって、これらのノードは切断されます。ネットワーク・セグメント 1 では、その他のアクティブでない 2 つのメタデータ・サーバの一方がアクティブになり、クラスタにはネットワーク・セグメント 1 上のシステムだけが含まれることになります。ルータの修復後も、ネットワーク・セグメント 2 上にあったノードは、これらのノード上でクラスタ・サービスを再開するまでは切断されたままです。

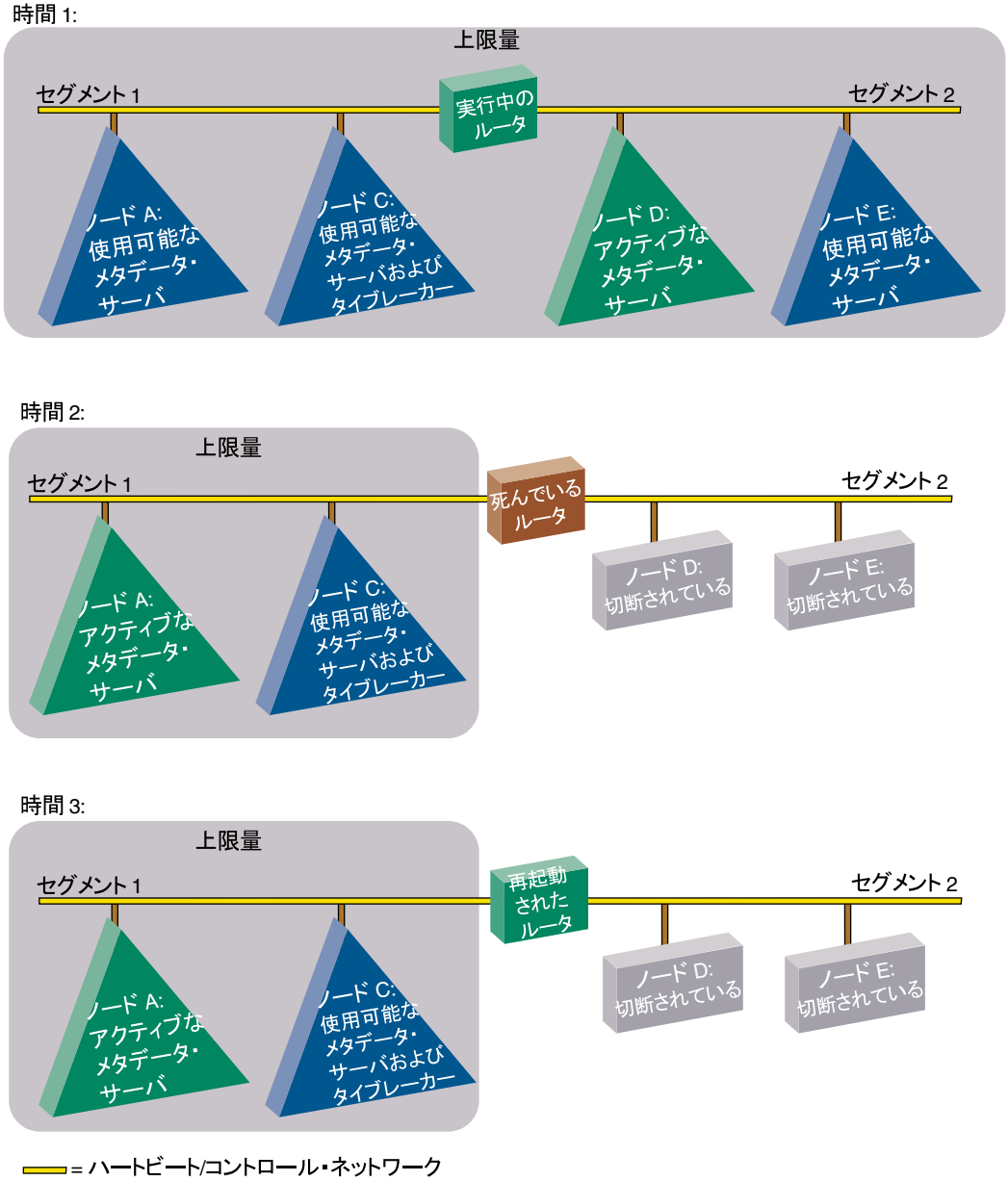

図1-4 ネットワークの分割による上限量とアクティブ・メタデータ・サーバの変化

fs2d メンバーシップ上限量

fs2d メンバーシップ上限量では、プール内のノードの半数 (50%) が fs2d で使用可能な場合に(したがって、クラスタ設定データベースの更新を受信できる場合に)初期クラスタを形成できます。fs2d メンバーシップ上限量を計算するときは、重み付けは使用されません。

メモ:これは CXFS メンバーシップとは異なります。CXFS メンバーシップの場合、初期クラスタを開始するには、クラスタ内の重み付けされたノードの過半数 (>50%) が使用可能である必要があります。CXFS メンバーシップは、重み付けされたノードの 50% だけで維持できます。

ハードウェア・リセット

クラスタとの接続が切断されたことを検出したノードは、強制的に共有ディスクへのアクセスを中止します。リセット・ハードウェアがない場合、以下のことを仮定すると、これだけでは共有ディスクのデータの整合性を十分に保証できない可能性があります。

- ノードは通信が切断されたことを検出できます。つまり、このノードは、エラーのために切断を検出できない状態ではありません。
- ノードは、意図したとおりにタイムリーに切断を検出します。つまり、このノードのエラーによって検出が遅れることはありません。

ただし、めったに発生しない特定のエラー状況でデータの整合性を保証するため、特に重み付けされた 2 ノード・クラスタの場合は、リモート・リセットに必要なハードウェアを使用することが推奨されています。IRIS FailSafe ではリセット機能は必須です。

最悪の状況は、ノードが通信の切断を検出しないにもかかわらず共有ディスクへのアクセスを引続き許可しており、データが破壊されてしまうような状況です。たとえば、ソフトウェアまたはハードウェアの異常が原因で、クラスタ内のあるノードがクラスタのその他のノードと通信できなくなっており、クラスタからアクティブなメンバーとして認識されていないにもかかわらず、引続き共有ディスクにアクセスできるような場合です。

この場合、リセット・ハードウェアがあれば、エラーが検出された瞬間、かつこのノードがクラスタから除外された状態を回復する前に、その他のいずれかのノードが異常ノードによるディスクへのアクセスを中止し、異常ノードがこれ以上のアクティビティを行わないようにすることができます。

既存の CXFS メンバーシップが 2 つの半数(それぞれが総重みの半分を持ちます)に分割されるような本当のネットワーク分割が発生した場合は、以下のような状況が発生します。

- タイブレーカー・ノードとリセットが設定されている場合は、タイブレーカー・ノードのある方の半数がもう一方の半数をリセットします。タイブレーカー・ノードのない方は、CXFS サービスを強制的にシャットダウンします。

- タイブレーカー・ノードがなくてもリセットが設定されている場合は、各半数がヒューリスティック遅延を使用して他方の半数のリセットを試みます。一方の半数はリセットに成功して続行され、他方はリセット・レースに負けて再起動されます。
- タイブレーカー・ノードがなく、リセットも設定されていない場合は、一方がレースに勝って他方をリセットすることを想定して、両方の半数がリセットを遅らせます。この場合は、どちらの半数もリセットされないので両方が動作し続け、データが破壊される可能性があります。

  このような状況を避けるため、少なくともタイブレーカー・ノードまたはリセット機能のいずれかを設定してください。ただし、重み付けされた 2 ノード・クラスタ内のタイブレーカー・ノードに異常が発生したり、管理者が CXFS サービスを停止した場合は、その他のノードによって強制シャットダウンが実行され、CXFS ファイルシステムはすべてアンマウントされます。

リセットに失敗した方の半数が CXFS メンバーシップを形成しようとしてネットワーク分割が解消されない場合は、その重みが半分になって、初期 CXFS メンバーシップを形成できなくなり、1 つのクラスタに 2 つのCXFS メンバーシップができるのを防止します。

リモート・リセット接続には以下の方法があります。

- 2 つのノードのクラスタは、シリアル回線で直接接続できます。
- 3 つ以上のノードのクラスタは、Ethernet マルチプレクサまたは SCSI マルチプレクサで接続する必要があります。各ノードは、ノードのリセット機能を持つノードであるオーナー・ホストを持つように定義されます。

詳細については、日本 SGI の営業担当員、もしくは営業フリーダイヤルまたは管理サービスまでお問い合わせください。

メタデータ・クライアント/サーバ・モデル

メタデータ・サーバは、以下のようなクラスタ調整機能を実行します。

- メタデータのログへの記録
- ファイルのロック
- バッファのコヒーレンシ

メタデータに対する CXFS の要求は、すべて TCP/IP ネットワーク上でメタデータ・サーバを通じてルーティングされ、メタデータの変更はすべてメタデータ・サーバに送信されます。メタデータ・サーバは、XFS の先進的なジャーナル機能を使用して、メタデータの変更をログに記録します。一般的にメタデータのサイズは小さいので、メタデータのトラフィックに対しては Fast Ethernet LAN の帯域幅で十分です。

ディスクとの直接のデータ操作と比較すると、CXFS メタデータ・サーバへの操作の頻度は通常は高くありません。たとえば、ファイルを開くと、メタデータ・サーバにファイルの情報が要求されます。ファイルが開い

たら、通常、プロセスは何度でも読取りや書込みを行うことができ、追加のメタデータ要求は必要ありません。メタデータ操作は、ファイル・サイズなどファイルのメタデータ属性が変更されたときに実行されます。

この場合、以下の規則が適用されます。

- 重みが指定されているかぎり、クラスタ内のどのノードもメタデータ・サーバとして動作できます。
- 複数のファイルシステムのメタデータ・サーバとして同時に動作できるのは、クラスタ内の 1 つのノードだけです。
- それぞれが別のファイルシステムのセットを持つ複数のノードをメタデータ・サーバとして動作させることができます。ただし、特定のファイルシステムは、1 つのノードの 1 つアクティブなメタデータ・サーバを使用します。
- 複数のノードを特定のファイルシステムのメタデータ・サーバとして使用可能なノードに設定できますが、アクティブなメタデータ・サーバになるのは、そのファイルシステムをマウントするノードのうちの最初のノードだけです。特定のファイルシステムのメタデータ・サーバとして使用可能なノードのリストは順序付きですが、ネットワーク・レーテンシやその他の予測不能な遅れがあるため、どのノードがアクティブなメタデータ・サーバになるかは予想できません。
- アクティブなクライアントがいる場合に、特定のファイルシステムのメタデータ・サーバとして使用可能な最後のノードが停止すると、すべてのクライアントがそのファイルシステムを使用できなくなります（メタデータ・サーバとして使用可能なほかのノードがリストにある場合は、回復が実行されます。詳細については、151 ページの「メタデータ・サーバの回復」を参照してください）。
- CXFS ファイルシステムをエクスポートしてその他の NFS クライアントで使用している場合、最適なパフォーマンスを得るには、アクティブなメタデータ・サーバからファイルシステムをエクスポートする必要があります。CXFS ファイルシステムの NFS エクスポートについての詳細は、144 ページの「マウント操作前後のスクリプトの実行」を参照してください。

詳細については、24 ページの「読取りおよび書込み時のメタデータの流れ」を参照してください。

システム表示

CXFS は、複数のファイルシステムの単一のシステム表示を備えています。SAN の各ホストはすべて共有ディスクに直接アクセスでき、どのホストでもファイルへのパス名は同じです。CXFS では、ディスク・チャネルとストレージを追加して、ホストからディスクへの直接の帯域幅を増やすことで、必要に応じて共有ファイルシステムのパフォーマンスを拡張できます。CXFS 共有ファイルのパフォーマンスは、LAN の速度や、データが中央のファイル・サーバを通じて渡されることによるボトルネックのために制限されることはありません。CXFS 共有ファイルでは、ローカル・ディスク・アクセスと同等の速度に、クラスタの柔軟性、スケーラビリティ、および信頼性が統合されています。

デーモン

表1-1 に、CXFS のデーモンとスレッドを示します。

メモ: CXFS は、バッファされている書込みをディスクに書込むために xfsd カーネル・スレッドを XFS と共有します。

同時実行 (CXFS and FailSafe) クラスタを使用している場合、FailSafe デーモンについての詳細は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

表1-1 CXFS のデーモンとスレッド

層	サブシステム	プロセス	説明
クラスタ・サービス (CXFS)	cluster_services	clconfd	CXFS クラスタ設定デーモン。CDB データベースからクラスタ設定を読取り、それに応じてローカル・カーネルの CXFS メンバーシップ・サービスを管理します。
クラスタ・ソフトウェア・インフラストラクチャ(クラスタ管理プロセス)	cluster_admin	cad	クラスタ管理デーモン。管理サービスを提供します。
	cluster_control	crsd	ノード・コントロール・デーモン。その他のノードとのシリアル接続をモニタします。その他のノードをリセットする機能があります。
		cmond	その他のすべてのデーモンを管理するデーモン。このプロセスは、クラスタ内のすべてのノードでほかのプロセスを開始し、異常時にはこれらのプロセスを再開します。
		fs2d	設定データベースを管理し、プール内のすべてのノードで各コピーの同期を保ちます。
カーネル・スレッド	sthreads	cmsd	CXFS メンバーシップとハートビートを管理します。

層	サブシステム	プロセス	説明
		Recovery	ノードの回復プロトコルを管理します。
		corpseleader	ノード間の回復を調整します。
		dcshake	CXFS クライアント上の動作していない CXFS v ノードをパージします。
		cxfsd	クライアントからサーバへのエクステントとサイズの更新の送信を管理します。CXFS クライアント上で動作するこのデーモンは、クライアント上の変更された i ノードを取得して、サイズと、書込まれていないエクステントの変更内容をサーバに戻します。
	xthreads	mesgtcprcv	開いているメッセージ・チャネルごとに 1 つのメッセージを読取ります。
		mesgtcpaccept	新しい接続を受付けます。
		mesgtcpdiscovery	ほかのノードのモニタと発見を行います。
		mesgtcpmulticast	ハートビートを提供します。

通信パス

以下の図に、FailSafe の通信パスを示します。

（以下の図には、cmond クラスタ・マネージャ・デーモンは表されていません。）

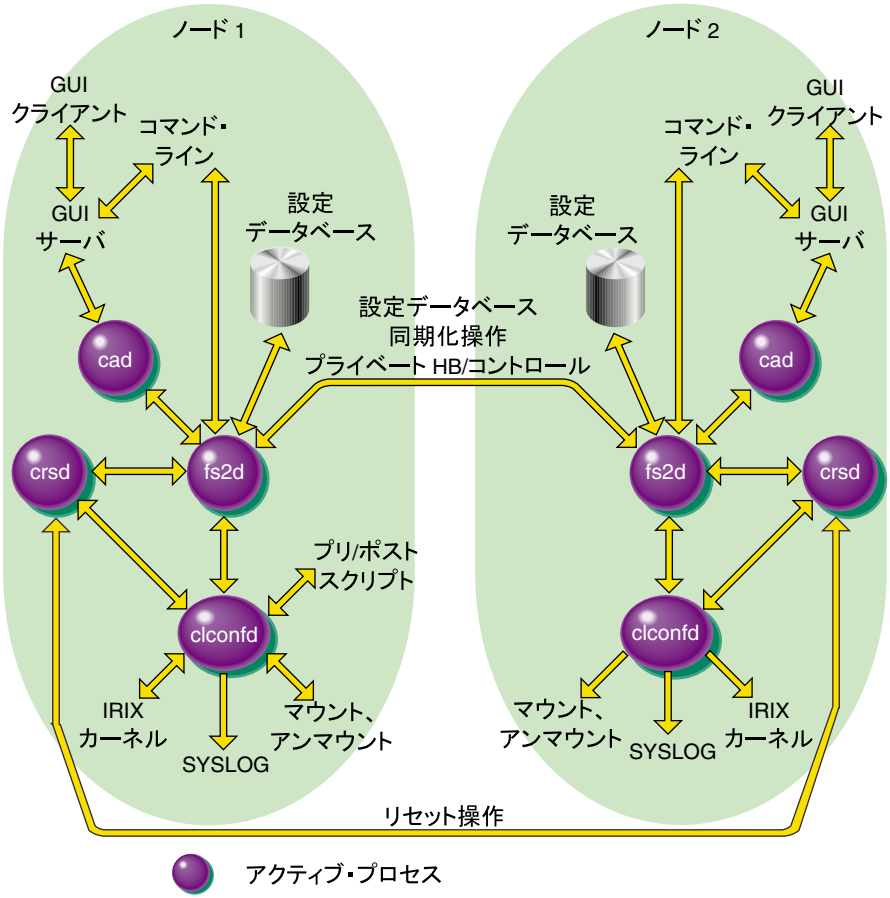

図1-5 クラスタ設定データベースに対する読取り/書込みアクション

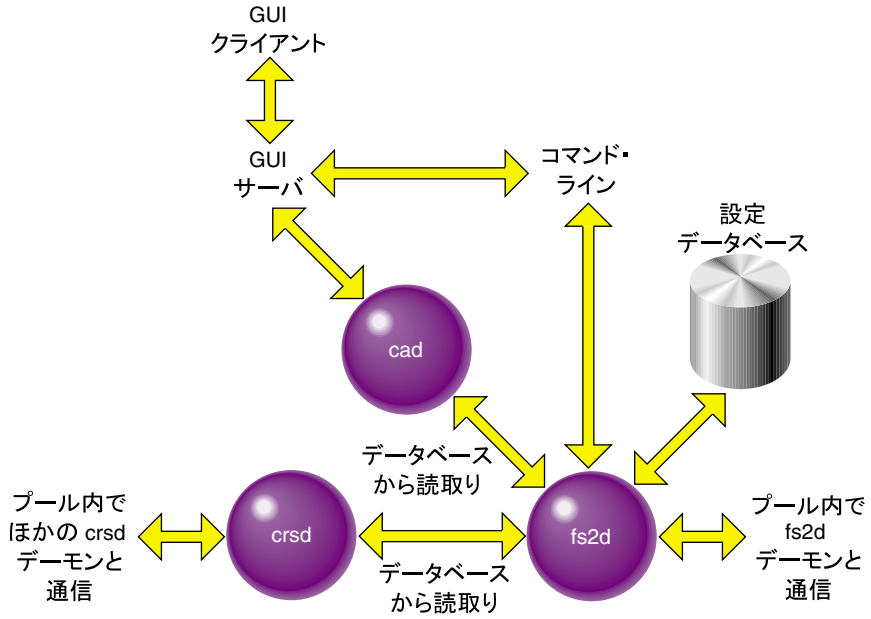

図1-6 クラスタ内にないノードの通信パス

読取りおよび書込み時のメタデータの流れ

以下の図に、メタデータの流れの例を示します。

メモ: 同時に複数の読取りトークンが存在する場合があります。

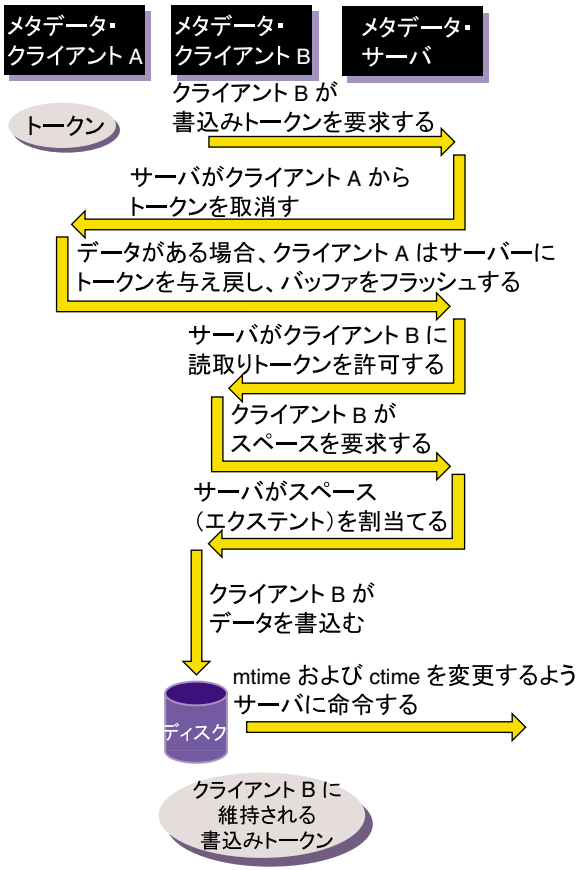

図1-7 書込み時のメタデータの流れ

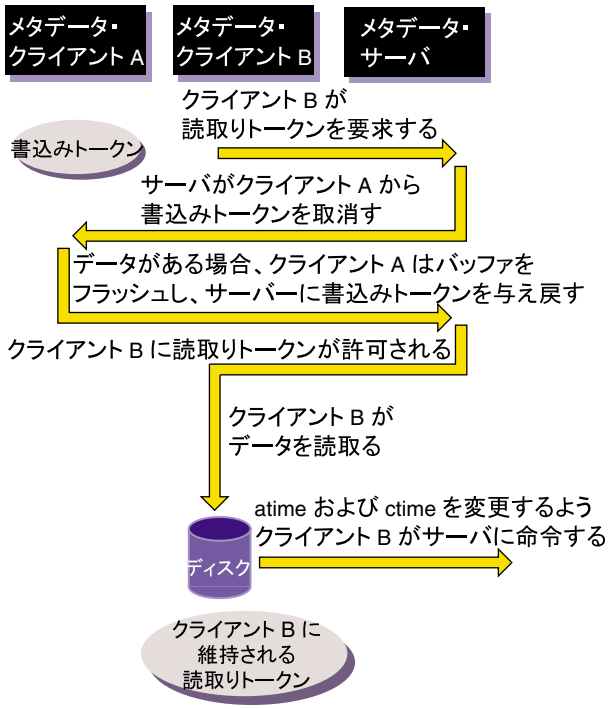

図1-8 クライアント A での書込み後のクライアント B での読取り時のメタデータの流れ

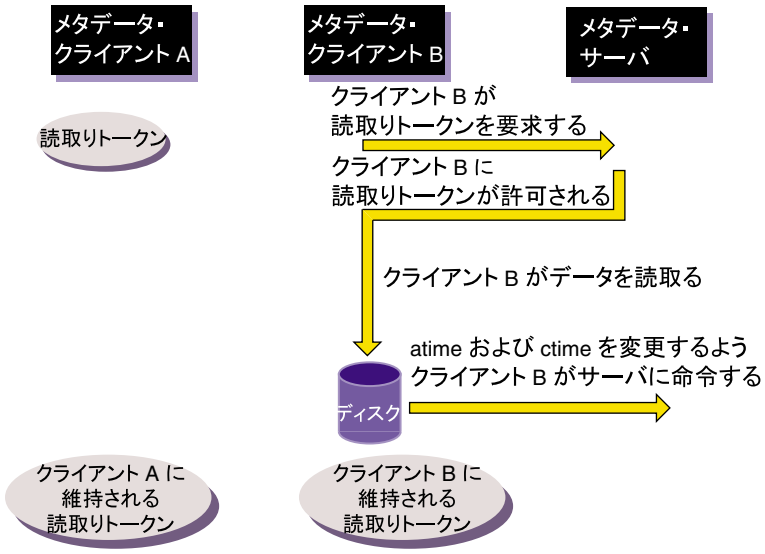

図1-9 クライアント A での読取り後のクライアント B での読取り時のメタデータの流れ

ハードウェアおよびソフトウェアの必要条件とサポート

CXFS では以下が必要です。

- CXFS だけのクラスタでは、最大 16 個のノードがサポートされています。CXFS のアーキテクチャは、最大 64 個のノードをサポートできるように設計されています。64 個のノードは、今後のリリースでサポートされる予定です。

  CXFS と IRIS FailSafe 2.1 の両方が実行されている(同時実行)クラスタでは最大 16 個のノードがサポートされており、これらのノードの 8 個までを FailSafe に使用できます。設定についての詳細は、27 ページの「IRIS FailSafe 2.1 との同時実行」を参照してください。

- プールとクラスタはそれぞれ 1 つだけです。
- すべてのノードで、同バージョンまたは直前のバージョンの IRIX OS が実行されている必要があります(6.5.12f の場合は 6.5.11f と 6.5.12f など)。
- CXFS 用の FLEXlm ライセンス・キーが必要です。
- 各 CXFS メタデータ・サーバとクライアントに接続されたプライベート TCP/IP ネットワークが必要です。
- Origin 3000、Origin 2000、Origin 200、Onyx2、または Onyx3 プラットフォームが必要です。

- Silicon Graphics O2 ワークステーション以外のサポートされている SAN ハードウェア設定。これには以下のプラットフォームが含まれます。
  - Origin 2000
  - Origin 200
  - Octane
- XVM ボリューム・マネージャ。IRIX リリースの一部として提供されています。XVM は、CXFS と共に使用する場合のみサポートされます。

CXFS は以下と共に動作します。

- Trusted IRIX/CMW (Trix)。CXFS は、単一のノードの Trix 環境でテストされています。ただし、Trix を分散環境に拡張するための作業は行われていません。したがって、Trix と CXFS を同時に実行したい場合は、クラスタのすべてのノードのユーザ ID、アクセス・コントロール・リスト (ACL: Access Control List)、および機能が同じになるようにシステムを設定する必要があります。
- IRIS FailSafe。27 ページの「IRIS FailSafe 2.1 との同時実行」および『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。
- IRISconsole。『IRISconsole Administrator's Guide』を参照してください。
- Ethernet マルチプレクサまたは SCSI マルチプレクサ (3 つ以上のノードのクラスタでリセット機能用に使用します)

IRIS FailSafe 2.1 との同時実行

CXFS と IRIS FailSafe 2.1 を同じシステムにインストールして実行できます。これを同時実行と呼びます。これにより、アプリケーション・レベルの高可用性とクラスタ・ファイルシステムを利用できるようになります。

メモ: IRIS FailSafe では、CXFS ファイルシステムは高可用性であると想定されています。これは、CXFS ファイルシステムをクラスタの別のノードで使用できるようにする場合に、FailSafe フェイルオーバーが必要ないためです。したがって、FailSafe が CXFS ファイルシステムまたは XVM ボリュームの開始、停止、またはモニタを直接行うことはありません。また、CXFS ファイルシステムを FailSafe リソース・グループに追加しないでください。

高可用性アプリケーションが CXFS ファイルシステムと IP アドレスを使用する場合は、該当するアプリケーションと IP アドレスをリソース・グループに追加してください。アプリケーションが依存する CXFS ファイルシステムは、CXFS GUI または `cmgr`(1m) ツールを使用して、フェイルオーバー・ドメインのすべてのノードにマウントする必要があります。

リソース・グループとフェイルオーバー・ドメインについての詳細は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

ファイルシステムのメタデータ・サーバとして動作しているノードがダウンした場合は、使用可能なメタデータ・サーバのリストに含まれる別のノードが新しいメタデータ・サーバとして選択されます。151 ページの「メタデータ・サーバの回復」を参照してください。

以下の点に注意してください。

- CXFS と FailSafe を実行している場合も、プール、クラスタ、およびクラスタ設定はそれぞれ 1 つだけです。
- クラスタは、以下の 3 つのいずれかのタイプに設定できます。
  - `FailSafe`。この場合、すべてのノードのタイプも `FailSafe` になります。
  - `CXFS`。この場合、すべてのノードのタイプが `CXFS` になります。
  - `CXFS and FailSafe`(同時実行)。この場合、すべてのノードのタイプは `CXFS` または `CXFS and FailSafe` になり、アプリケーション・レベルの高可用性を得るために FailSafe と CXFS が使用されます。

    メモ: `FailSafe` タイプのノードだけで同時実行クラスタを設定することも可能ですが、この設定はサポートされていません。`CXFS and FailSafe`(同時実行)クラスタのすべてのノードのタイプは、`CXFS` または `CXFS and FailSafe` でなければなりません。
- 実作業用のクラスタは、重み付けされた 3 つ以上のノードと 16 個以下のノードで設定することが推奨されています (リセット・ケーブルが接続されていて重み付けされた 2 ノード・クラスタはサポートされていますが、この設定には特有の問題があります。32 ページの「重み付けされた 2 ノード・クラスタでの回復と再設定の問題」を参照してください)。クラスタのすべてのノードは、CXFS ノードでなければなりません。最大 8 つのノードでは、IRIS FailSafe も実行できます。

- 使用可能なすべてのメタデータ・サーバ・ノードは、以下のいずれかのタイプでなければなりません。
  - `CXFS`
  - `CXFS and FailSafe`
- `cmgr`(1m) (`cluster_mgr`) コマンドは 1 つですが、CXFS と FailSafe にはそれぞれ別のグラフィカル・ユーザ・インタフェースがあります。CXFS 設定の管理は CXFS GUI で行い、FailSafe 設定の管理は FailSafe GUI で行ってください。`cmgr` ではどちらも管理できます。
- CXFS GUI または CLI を使用して、既存の `FailSafe` クラスタとノードのタイプを `CXFS` または `CXFS and FailSafe` に変換できます。FailSafe GUI を使用すると、並行して操作を実行できます。変換後のノードは、アプリケーション・レベルの高可用性を提供するために FailSafe で使用したり、クラスタ・ファイルシステムを提供するために CXFS で使用できます。83 ページの「GUI を使った CXFS の既存の FailSafe クラスタの設定」を参照してください。

  ただし、以下の点に注意してください。

  - 対応する高可用性 (HA) サービスまたは CXFS サービスがアクティブな場合は、ノードのタイプを変更することはできません。まず、ノードのサービスを停止してください。
  - クラスタは、そのノードに対してオンにするすべての機能（FailSafe または CXFS、あるいはその両方）をサポートしていなければなりません。つまり、クラスタのタイプが `CXFS` の場合、すでにクラスタの一部であるノードのタイプを `FailSafe` に変更することはできません。ただし、ノードがクラスタのすべての機能をサポートする必要はありません。つまり、`CXFS and FailSafe` クラスタで `CXFS` ノードを使用することができます。

  116 ページの「CLI を使ったノードの CXFS または FailSafe への切替え」および127 ページの「CLI を使ったクラスタの CXFS または FailSafe への切替え」を参照してください。
- FailSafe では少なくとも 2 つのネットワーク・インタフェースが必要ですが、CXFS では、ハートビート・メッセージとコントロール・メッセージの両方に 1 つのインタフェースだけを使用します（CXFS GUI では、ネットワークに対してハートビートまたはコントロールの一方だけを選択できるように見えますが、このように選択しないでください）。

  同じノードで FailSafe と CXFS を使用する場合、CXFS に対しては優先度 1 のネットワークだけが使用されるので、このネットワークは、ハートビート・メッセージとコントロール・メッセージの両方を使用できるように設定する必要があります。

メモ: CXFS は 2 番目のネットワークにフェイルオーバーされません。ノードが `CXFS and FailSafe` の場合、優先度 1 のネットワークに異常が発生すると、CXFS には異常が発生しますが、FailSafe サービスは 2 番目のネットワークに移動できます。

優先度 1 のネットワークがなくなったために CXFS がノードをリセットすると、そのノードは FailSafe によって FailSafe メンバーシップから削除されます。これにより、リソース・グループはクラスタ内の別の FailSafe ノードにフェイルオーバーされます。

- FailSafe ノードが 2 つだけの場合は、タイブレーカー・ノードを使用しないでください。
- メタデータ・サーバのリストは、フェイルオーバー・ドメインのリストに完全に一致する必要があります（名前と名前の順序）。

図1-10 に、同時実行クラスタにおける情報の流れを示します。

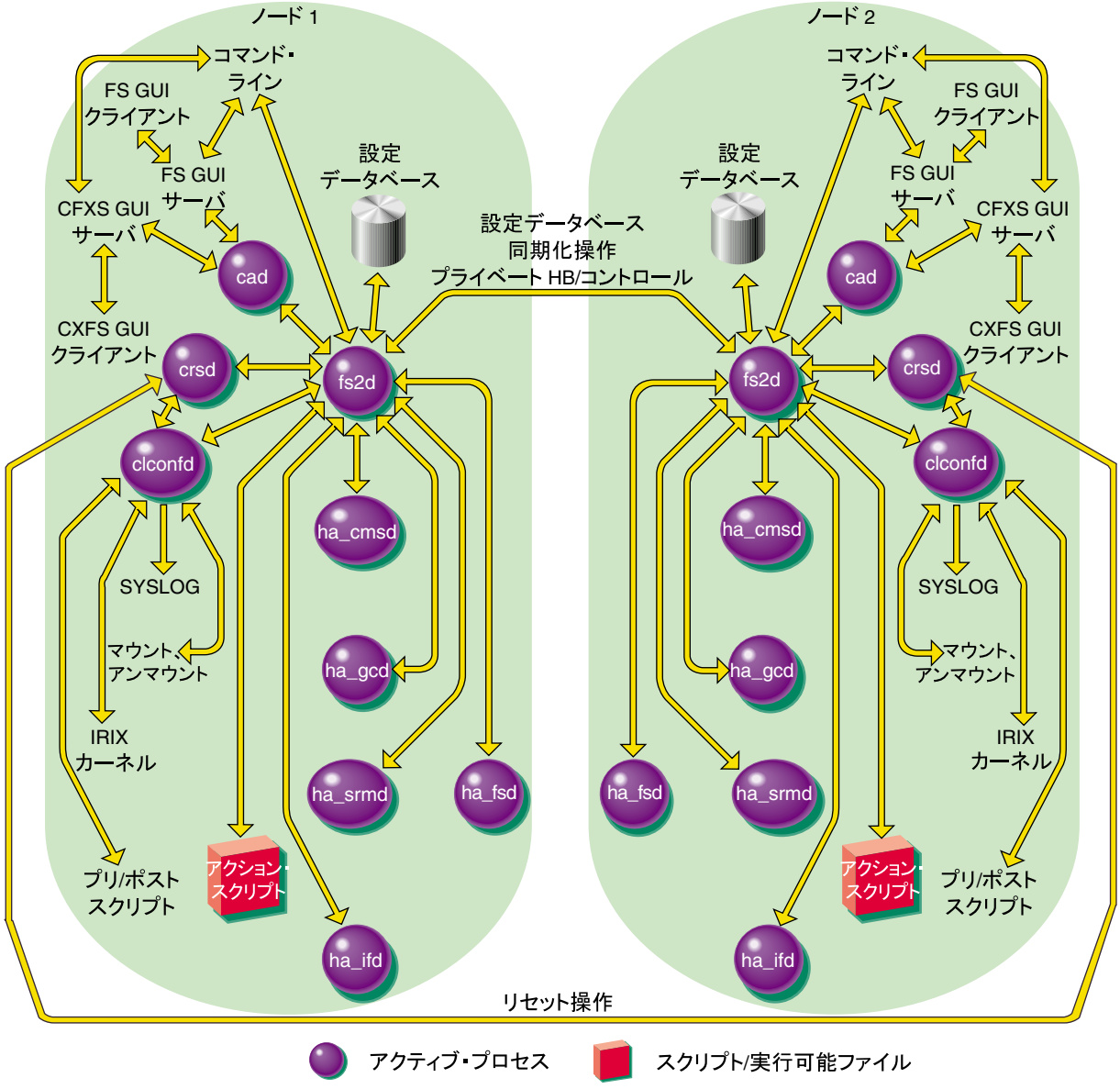

図1-10 同時実行時のクラスタ設定データベースに対する読取り / 書込みアクション

重み付けされた 2 ノード・クラスタでの回復と再設定の問題

実作業用の環境（つまり、メタデータ・サーバの再設定と回復が必要な環境）では、少なくとも 3 つの重み付けされたノードのクラスタが推奨されています。

2 つの重み付けされたノードのクラスタを実作業で使用する場合は、以下のいずれかを行う必要があります。

- リセット回線を使用してデータを確実に保護し、1 つのノードだけがエラー状態で実行されるよう保証します（リセット回線は、すべての CXFS クラスタに対して推奨されており、IRIS FailSafe と共に使用する場合は必須です）。
- 1 つのノードの重みを 1 に設定し、その他は 0 に設定します。
- タイブレーカー・ノードを設定します。

ただし、これらの方法を使用しても、重み付けされた 2 ノード・クラスタには回復と再設定の点で特有の問題があります。

リセット回線を持つ重み付けされた 2 ノード・クラスタがあり、一方のノードに問題が発生したとします。この場合、以下のような状況が発生します。

- 両方のノードの重みが 1 で、タイブレーカー・ノードがない場合

  クライアント・ノードはサーバのパニックまたはリセットを切抜けます。また、プライベート・ネットワーク上のメタデータ・サーバとの接続の切断を切抜ける場合もあります。接続が切断された場合は、クライアントとメタデータ・サーバの両方が他方のノードのリセットを試み、最初にリセットに成功したノードが残ります。

- 両方のノードの重みが 1 で、メタデータ・サーバがタイブレーカー・ノードの場合

  既存の CXFS メンバーシップ上限量は重みの 50% があれば維持できるので、クライアント・ノードはメタデータ・サーバの異常を切抜けます。

- 両方のノードの重みが 1 で、クライアントがタイブレーカー・ノードの場合

  サーバのパニックまたはリセット、およびプライベート・ネットワーク上のメタデータ・サーバとの接続の切断のどちらについても、クライアントは切抜けます。このクライアントがリストの 2 番目のメタデータ・サーバの場合は、アクティブなメタデータ・サーバになります。クライアントがリストにない場合は、ファイルシステムがアンマウントされます。

  メモ：通常は、プライマリ・メタデータ・サーバをタイブレーカー・ノードにします。

詳細については、11 ページの「ハードウェア・コンポーネント」および13 ページの「上限量の計算とノードの重み」を参照してください。

CXFS クラスタ・マネージャのツール

CXFS の設定タスクは、CXFS クラスタ・マネージャ・グラフィカル・ユーザ・インタフェース (GUI: Graphical User Interface) またはコマンド・ライン・インタフェース (CLI: Command Line Interface) のいずれかを使用して実行できます。クラスタの設定とモニタを行う際にこれらのツールが使用する下層のソフトウェアは同じですが、GUI は、実作業用システムでは特に重要な以下の機能を備えています。

- 青いテキストをクリックすると、その概念や入力フィールドに関する詳細を参照できます。「ヘルプ (Help)」ボタンをクリックしてオンライン・ヘルプを利用することもできます。
- クラスタの状態が視覚的に表示されるため、ステータスやフェイルオーバーをすばやく認識できます。
- 状態が動的に更新されるので、継続的にシステムをモニタできます。
- クラスタ設定情報に変更を加える前に、構文が正しいかどうかに関してすべての入力値がチェックされます。どのタスクでも、「OK」をクリックするまでクラスタ設定は更新されません。
- タスクおよびタスクセットにより、設定および管理操作を順番に進めることができ、タスクを実行しながらクラスタ設定に実際に変更を加えることができます。
- これらのグラフィカルなツールは、任意の IRIX マシンか、または Windows や Linux コンピュータとラップトップを含む Java が有効な Web ブラウザがインストールされた任意のコンピュータからリモートで安全に実行できます。

CLI の機能は GUI よりも限られています。CLI では、IRIX システム上だけでクラスタ・システムの設定と管理を行うことができます。また、ヘルプや書式付き出力は最低限しか用意されておらず、クエリーを実行した場合を除き、動的にステータスを参照することはできません。ただし、経験が豊富な管理者の方にとっては、基本的な設定タスクや実作業環境で独立した単一のタスクを実行する場合、またはスクリプトを実行して特定のクラスタ管理タスクを自動化する場合は、CLI の方が便利かもしれません。

GUI の概要

GUI を使用すると、単一のポイントからクラスタ全体を管理できます。CXFS デーモンが開始されている場合、クラスタの正確なステータスを参照するには、すべての CXFS デーモンが実行されているノードに接続するようにしてください。クラスタ内で CXFS デーモンが開始されていない場合は、プール内の任意のノードに接続できます。

すべてのタスクを実行するために必要な特権を持つことができるよう、GUI には `root` としてログインする必要があります。ただし、IRIX Interactive Desktop System Administration (`sysadmdesktop`) 製品の一部である特権マネージャを使用すると、システム管理者が一部またはすべての特権をその他の任意のユーザに許可できます。詳細については、『Personal System Administration Guide』を参照してください。

GUI は、CXFS マネージャおよび CXFS クラスタ表示で構成されます。

CXFS マネージャ

CXFS マネージャは、CXFS クラスタの設定と管理を行いやすくするタスクとタスクセットへのアクセスを提供します。

- タスクを使用して、CXFS クラスタの個々のコンポーネントを設定およびモニタできます。
- 設定ガイド・タスクセットは、より大きな目的を達成するために一まとめにされたタスクのグループで構成されます。たとえば、「新規クラスタの設定(Set Up a New Cluster)」を使用すると、新規クラスタの作成手順を順番に実行でき、タイトルをクリックするだけで必要な個々のタスクを起動できます。

CXFS マネージャには、以下のカテゴリがあります。

- 「概要(Overview)」。使用可能なカテゴリを説明します。
- 「設定ガイド(Guided Configuration)」。クラスタの設定、ファイルシステムの定義、および既存のクラスタの変更を行うためのタスクセットが含まれます。IRIS FailSafe クラスタが CXFS クラスタにもなるように変換するタスクセットも含まれます。
- 「ノードとクラスタ(Nodes & Cluster)」。ノードとクラスタの定義と管理を行うためのタスクが含まれており、CXFS サービスの開始（および再起動時にこれらのサービスが自動的に再開されるような設定）、または CXFS クラスタ・サービスの停止（および自動再開機能のオフ）を行うことができます。
- 「ファイルシステム(Filesystems)」。クラスタのファイルシステムを定義および管理するためのタスクが含まれます。
- 「診断とエラー回復(Diagnostics & Error Recovery)」。設定上の問題がないかどうかクラスタとノードをテストしたり、CXFS メンバーシップ・エラーから回復できます。
- 「タスクの検索(Find a Task)」。キーワードを使用して特定のタスクを検索できます。

図1-11 に、「CXFS マネージャの概要(CXFS Manager Overview)」ウィンドウを示します。

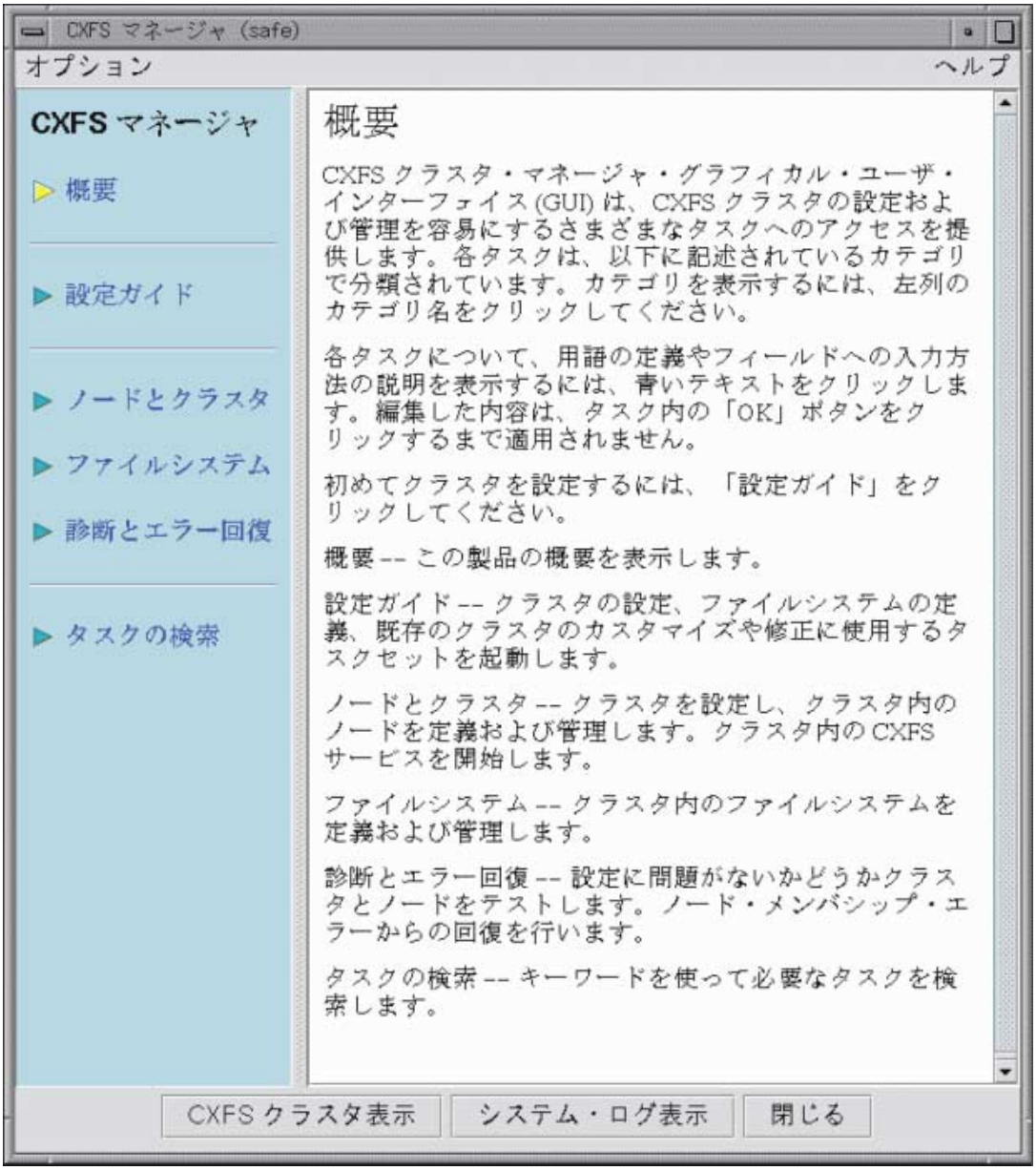

図1-11「CXFS マネージャの概要」ウィンドウ

CXFS クラスタ表示

CXFS クラスタ表示には、以下の機能が用意されています。

- タスク完了時の結果の表示。たとえば、「ノードの定義(Define a Node)」タスクが完了すると、クラスタ表示に新しいノードが表示されます。

注意: 次の作業に進む前に、タスクが完了するまで待つようにしてください。次の作業に移るのが早すぎると、エラーが発生します。設定データベースに変更を加えると、その都度クラスタ全体のステータス情報が送信されます。したがって、設定が大きくなるほど送信に時間がかかります。

- クラスタ、ノード、およびファイルシステムの関係の表示
- クラスタとノード・タイプを含むすべてのアイテムの設定とステータスの詳細へのアクセス
  - CXFS
  - FailSafe
  - CXFS and FailSafe
- アイコンの色によるクラスタの動作状態の表示
- 「CXFS マネージャ(CXFS Manager)」ウィンドウおよび /var/adm/SYSLOG ファイルへのアクセス
- ヘルプ情報へのアクセス

CXFS クラスタ表示からクラスタ、ノード、またはファイルシステム名をクリックすると、重要な情報を表示できます。

図1-12 に、「CXFS クラスタ表示(CXFS Cluster View)」ウィンドウの例を示します。

図1-12「CXFS クラスタ表示(CXFS Cluster View)」ウィンドウ

図1-13 に、以下のメニュー・アイテムを選択したときの例を示します。

「オプション(Options)」
>「詳細を表示(Show Details)」

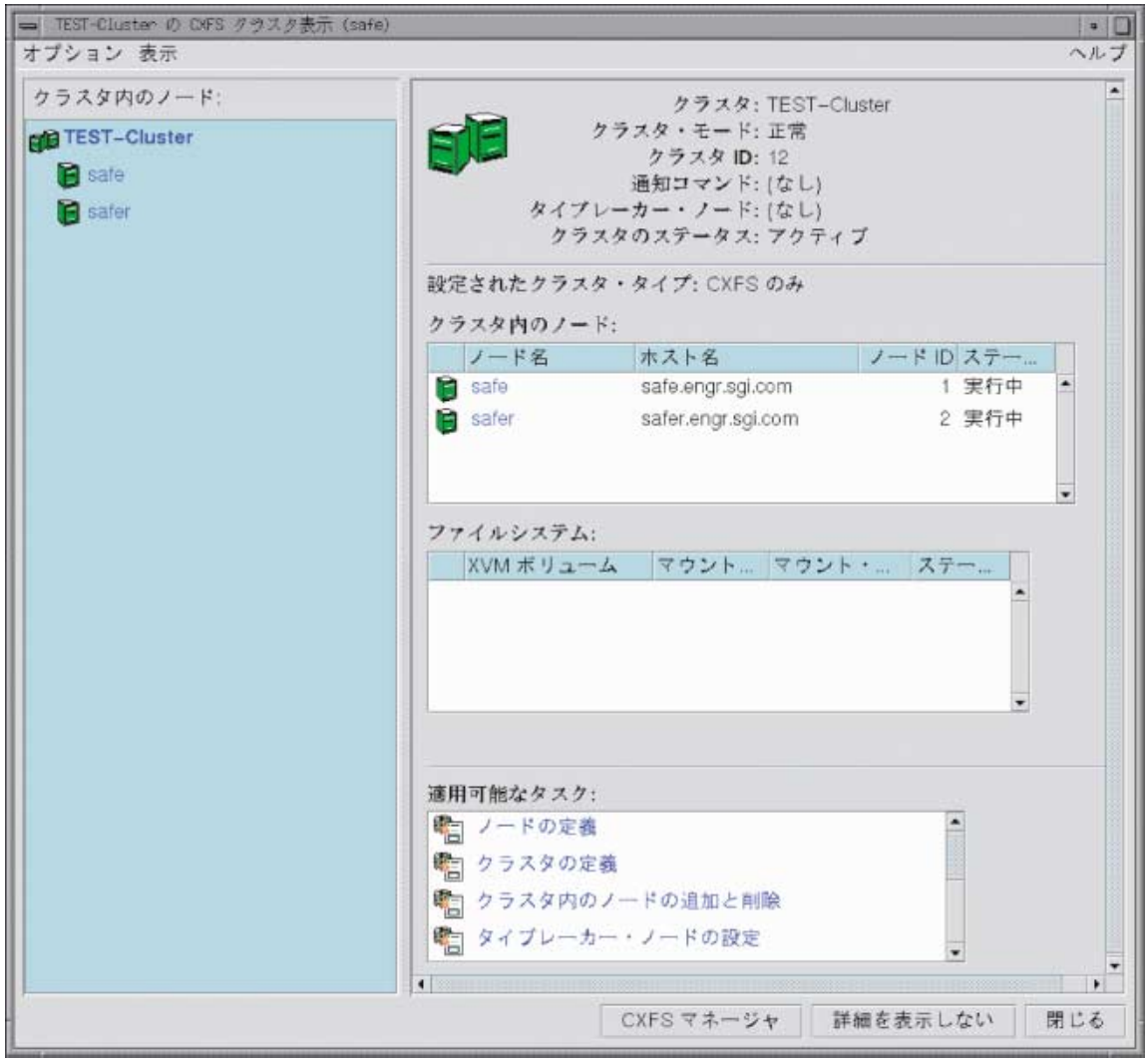

図1-13 クラスタ表示の詳細

GUI の起動

メモ: クラスタの表示は、どのノードから GUI を実行するかによって変わります。変更を加えたときは、次の変更を行う前に、「CXFS クラスタ表示(CXFS Cluster View)」ウィンドウに変更内容が表示されるまで待ってください。CXFS クラスタ表示に変更内容が表示されるまでは、その変更がクラスタ全体に伝わったことは保証されていません。設定データベースに変更を加えると、その都度クラスタ全体のステータス情報が送信されます。したがって、設定が大きくなるほど送信に時間がかかります。

変更は、特定の時点で実行されている GUI の 1 つのインスタンスだけから行ってください。2 つ目のインスタンスから変更を行うと、最初のインスタンスで加えた変更が上書きされる場合があります。

GUI は、CXFS クラスタ表示またはCXFS マネージャのいずれかを起動することで起動できます。GUI のその他の部分は、GUI のウィンドウの下部にある各ボタンをクリックして起動する必要があります。「FailSafe マネージャ(CXFS Manager)」ウィンドウと「CXFS クラスタ表示(CXFS Cluster View)」ウィンドウの両方を同時に開く場合は、この方法が推奨されています。これは、2 番目のウィンドウを開くときに、新しい Java プロセスを起動するのではなく既存の Java プロセスを再利用するためです。この方法を使用すると、クライアントのメモリ使用量が抑えられます。

ただし、常にCXFS クラスタ表示を使用して実行状況を確認し、変更を加えるのが早すぎることがないようにしてください。ノードを定義した後は、別のノードを定義する前に、定義したノードがCXFS クラスタ表示に表示されるまで待つ必要があります。クラスタを定義した後は、ノードを追加する前に、定義したクラスタが表示されるまで待ってください。

CXFS クラスタ表示の起動

CXFS クラスタ表示を起動するには、以下のいずれかの方法を使用します。

- Toolchest から以下を選択します。

  「CXFS」
  >「CXFS クラスタ表示(CXFS Cluster View)」

  メモ: Toolchest に CXFS のエントリを表示するには、CXFS のインストール後に Toolchest を再起動する必要があります。Toolchest を再起動するには、以下のコマンドを入力します。

  ```
  # killall toolchest
  # /usr/bin/X11/toolchest &
  ```

  これを反映させるには、クライアント・システムに `sysadm_cxfs.sw.desktop` がインストールされていなければなりません。

  ホストにログインするよう求めるダイアログ・ボックスが表示されます。

- 以下のコマンド・ラインを入力します。

  ```
  # /usr/sbin/cxdetail
  ```

  このコマンド・ラインにより、CXFS クラスタ表示が呼出されます。クラスタが定義されていない場合は、CXFS マネージャも起動されます。

  メモ: クラスタを定義する前に `cxdetail` を呼出すと、エラー・メッセージが表示されます。クラスタの定義中の場合は、このエラー・メッセージは無視してかまいません。

CXFS マネージャの起動

CXFS マネージャを起動するには、以下のいずれかの方法を使用します。

- Toolchest から以下を選択します。

  「CXFS」
  　>「CXFS マネージャ(CXFS Manager)」

  メモ: Toolchest に CXFS のエントリを表示するには、CXFS のインストール後に Toolchest を再起動する必要があります。Toolchest を再起動するには、以下のコマンドを入力します。

  ```
  # killall toolchest
  # /usr/bin/X11/toolchest &
  ```

  これを反映させるには、クライアント・システムに `sysadm_cxfs.sw.desktop` がインストールされていなければなりません。

- 以下のコマンド・ラインを入力します。

  ```
  # /usr/sbin/cxtask
  ```

- 以下の URL を使用します。

  http://*server*/CXFSManager

  (*server* には、管理するノードの名前を指定します。)

  <Enter> キーを押します。表示される Web ページで、「ディスク・ボリューム」アイコンをクリックします (この方法は、サーバで Web サーバが実行されており、そのサーバに `sysadm_cxfs.sw.web` がインストールされている場合にのみ使用できます)。

  IRIX 以外の OS から GUI を管理する場合は、この方法で CXFS マネージャを起動できます。IRIX システムで GUI を実行している場合は、Toolchest のメニュー・アイテムまたは `/usr/sbin/cxtask` コマンドを使用する方法が推奨されています。

  この方法で CXFS マネージャを起動できるのは、クライアント・マシンで以下の作業が完了している場合だけです。

  – Java Plug-in のインストール

  – すべての Java プロセスの終了

  – ブラウザの再起動

  – Java の有効化

この方法を使用する場合も、サーバに Web サブシステムがインストールされている必要があります。

数分たっても「ディスク・ボリューム」アイコンが表示されない場合は、「プラグインなし(non plug-in)」リンクをクリックできますが、ブラウザ固有の Java で実行することにより、操作上の不具合が発生する可能性があります。

クラスタ・アイテムの詳細の表示

クラスタ・アイテムの詳細を表示するには、以下の手順に従います。

1. 「CXFS クラスタ表示(CXFS Cluster View)」ウィンドウを開きます。
2. アイテムの名前またはアイコンをクリックします。

設定およびステータスの詳細が別のウィンドウに表示されます。同じウィンドウに詳細を表示するには、以下を選択します。

「オプション(Options)」
>「詳細を表示(Show Details)」

ウィンドウの右側にステータスの詳細が表示されます。

タスクの実行

個々のタスクを実行するには、以下の手順に従います。

1. 「CXFS マネージャ(CXFS Manager)」ウィンドウの左側のカラムでカテゴリの名前をクリックします。

   右側のカラムに個々のタスクとタスクセットのトピックのリストが表示されます。

2. 右側のカラムのタスクのタイトルをクリックします。

   「タスク(Task)」ウィンドウが表示されます。

   メモ: 青いテキストをクリックすると、その概念や入力フィールドに関する詳細を参照できます。

   ---

3. 該当するフィールドに情報を入力して「OK」をクリックし、タスクを完了します。タスクの中には複数のページで構成されるものもあります。このような場合は、「次へ(Next)」をクリックして次のウィンドウに移動し、そのウィンドウで情報を入力してから「OK」をクリックします。

   メモ: どのタスクでも、「OK」をクリックするまでクラスタ設定は更新されません。

   タスクが正常に完了したことを確認するダイアログ・ボックスが表示されます。

4. 必要に応じて、タスクの起動を続行します。

タスクセットの使用

GUI にはタスクセットも用意されており、複数の異なるタスクが必要な目的を達成するために必要な手順を順番に実行できます。CXFS タスクセットにアクセスするには、以下の手順に従います。

1. 「CXFS マネージャ(CXFS Manager)」ウィンドウの左側のカラムで「設定ガイド(Guided Configuration)」をクリックします。

   右側のカラムにタスクセットのリストが表示されます。

2. 右側のカラムでタスクセットをクリックします。

   目的を達成するために必要な一連のタスクのリストがウィンドウに表示されます。

3. 表示される手順に従って、タスクをクリックして起動します。

   タスクをクリックすると、そのタスクのウィンドウが表示されます。リストに表示されるすべてのタスクを完了したら、ウィンドウの左上角をダブルクリックするか、またはウィンドウに「閉じる(Close)」ボタンがある場合はそのボタンをクリックすると、タスクセットのウィンドウを閉じることができます。

CLI の概要

CLI を使用するには、root としてログインする必要があります。続いて以下を入力します。

```
# /usr/cluster/bin/cluster_mgr
```

または

```
# /usr/cluster/bin/cmgr
```

メモ: -p オプションを使用して、ほかのプロンプトを表示することもできます。ただし、ファイルシステムを定義するときは、必ずこのオプションを使用してください。

このコマンドを入力すると、以下のメッセージとクラスタ・マネージャ CLI のコマンド・プロンプト (cmgr>) が表示されます。

```
Welcome to SGI Cluster Manager Command-Line Interface

cmgr>
```

使用可能な cmgr コマンドについての詳細は、cmgr(1m) のマン・ページを参照してください。

ヘルプの利用

コマンド・プロンプトが表示されたら、以下のサブコマンドを入力できます。? または help と入力すると、いつでも CLI のヘルプを表示できます。

アクションの完了とキャンセル

クラスタのコンポーネントを作成または変更する場合は、以下のいずれかのコマンドを入力できます。

- cancel。現在のモードを中止して、変更を破棄します。
- done。現在の定義または変更を確定して、cmgr プロンプトに戻ります。

スクリプト・ファイルの使用

-f オプションを使用して入力ファイルを指定することで、cmgr コマンドを連続して実行できます。

```
cmgr -f input_file
```

または、ファイルの 1 行目に以下を含めておき、そのファイルをスクリプトとして実行することができます。

```
#!/usr/cluster/bin/cmgr -f
```

ファイルの各行は、有効な cmgr コマンド・ライン、コメント行(# で始まります)、または空白行でなければなりません。

メモ: 複数レベルのコマンドを完了するために done コマンド・ラインを含める必要があります。また、ファイルの最後には quit コマンド・ラインを含める必要があります。

入力ファイルのいずれかの行が失敗すると、cmgr は終了します。以下のように、-i オプションを使用すると、行の失敗を無視するように選択して処理を続行できます。

```
cmgr -if input_file
```

または、以下のようにスクリプトの 1 行目にこのオプションを含めることもできます。

```
#!/usr/cluster/bin/cmgr -if
```

メモ: cmgr コマンド・ラインをスクリプトの 1 行目として使用するときに -i を含める場合は、必ずまったく同じ構文(-if)を使用してください。

たとえば、ファイル /tmp/showme に以下のコマンドが含まれるとします。

```
cxfs6# more /tmp/showme
show clusters
show nodes in cluster TEST
quit
```

この場合は、以下のコマンドを実行でき、以下の出力が得られます。

```
cxfs6# /usr/cluster/bin/cmgr -if /tmp/showme

1 Cluster(s) defined
        TEST



Cluster TEST has following 3 machine(s)
        cxfs6
        cxfs7
        cxfs8
```

または、スクリプトの 1 行目に cmgr コマンド・ラインを含めて、そのスクリプトに実行アクセス権を指定し、showme 自体を実行できます。

```
cxfs6# more /tmp/showme
#!/usr/cluster/bin/cmgr -if
#
show clusters
show nodes in cluster TEST
quit

cxfs6# /tmp/showme

1 Cluster(s) defined
        TEST



Cluster TEST has following 3 machine(s)
        cxfs6
        cxfs7
        cxfs8
```

完全なクラスタを定義する場合についての例は、141 ページの「スクリプトの例」を参照してください。

CLI 内部からのシェルの呼出し

CLI 内部からシェルを呼出すには、以下を入力します。

```
cmgr> sh
cxfs6#
```

シェルを終了して cmgr> プロンプトに戻るには、以下を入力します。

```
cxfs6# exit
cmgr>
```

CXFS ソフトウェアの初期インストールとシステムの準備

注意: CXFS は複雑な製品です。CXFS が最適にインストールおよび設定されるよう、必ず日本 SGI から初期セットアップ・サービスをご購入ください。この章は、カスタマが直接使用することを目的としておらず、リファレンスとして提供されています。

この章は、以下の手順で構成されています。

1. 47 ページの「必要なソフトウェアのインストール」
2. 52 ページの「システム・ファイルの設定」
3. 58 ページの「(オプション) 自動再起動用の設定」
4. 58 ページの「ネットワーク・インタフェースの設定」
5. 61 ページの「シリアル・ポートの設定」
6. 62 ページの「システムの再起動」

これらの手順を完了したら、65 ページの 第3章「クラスタの初期設定」を参照してください。

メモ: CXFS クラスタのインストールと設定を行う前に、このマニュアル全体、特に 173 ページの 第8章「トラブルシューティング」をよくお読みください。IRIS FailSafe との同時実行を使用する場合は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

必要なソフトウェアのインストール

CXFS base CD のインストールには、約 30.3 Mバイトのスペースが必要です。必要なソフトウェアをインストールするには、以下の手順に従います。

1. プール内の各ノードで、『IRIX 6.5 Installation Instructions』に従って IRIX 6.5.12 にアップグレードします。

特定のノードがアップグレードされたことを確認するには、以下のコマンドを使用して、現在インストールされているシステムを表示します。

```
# uname -aR
```

2. (シリアル・ポート・サーバのあるサイトの場合)プール内の各ノードに、オペレーティング・システムに適したバージョンのシリアル・ポート・サーバ・ドライバをインストールします。シリアル・ポート・サーバに付属の CD を使用してください。インストールの後、システムを再起動してください。

   詳細については、シリアル・ポート・サーバに付属の以下のドキュメントを参照してください。

   - 『EL Serial Port Server Installation Guide』(Digi Corporation によって提供されています)
   - 『EL Serial Port Server Installation Guide Errata』

3. プール内の各ノードで、以下の作業を行います。

   a. CXFS license key をインストールします。ライセンス・キーが必要な製品を注文した場合、キーはインストールの注意事項と共に受注部門から自動的に電子メールで送信されます。この情報がない場合は、日本 SGI または最寄のサポート・プロバイダにお問い合せください。

      ライセンスが適切にインストールされている場合は、cxfslicense コマンドから以下の出力が表示されます。

      ```
      # /usr/cluster/bin/cxfslicense -d CXFS license granted.
      ```

      CXFS ライセンスが適切にインストールされていない場合は、以下のエラーが表示されます。

      ```
      CXFS not properly licensed for this host.  Run
                      '/usr/cluster/bin/cxfslicense -d'
             for detailed failure information.
      ```

      システムの CPU の数を増やすと、新しいライセンスが必要になることがあります。ただし、既存の O3000 システムを再パーティション設定した場合は、新しいライセンスは必要ありません。

      ソフトウェア・ライセンスのインストールについての詳細は、『IRIX 6.5 Installation Instructions』ブックレットを参照してください。

   b. CD-ROM #2 を CD-ROM ドライブに挿入します。

   c. 以下を入力して、挿入されている CD-ROM を読取るように Inst に指示します。

      ```
      Inst> from /CDROM/dist
      ```

注意: inst -r オプションを使用して別の root にインストールしないでください。終了操作 (exitops) の中には、別の root への相対パス名を使用しないものがあります。このため、-r オプションを使用すると、メインの root ファイルシステムと別の root ファイルシステムの両方で問題が発生する可能性があります。exitops についての詳細は、inst(1m) のマン・ページを参照してください。

d. <ENTER> キーを押して、CD-ROM を読取ります。

```
Install software from : [/CDROM/dist] <ENTER>
```

e. CXFS ソフトウェアをインストールします。

```
Inst> install cluster*
Inst> install cxfs*
Inst> install sysadm_base*
Inst> install sysadm_cxfs*
```

以下のサブシステムがインストールされます。

```
cluster_admin.man.man
cluster_admin.sw.base
cluster_control.man.man
cluster_control.sw.base
cluster_control.sw.cli
cluster_services.man.man
cluster_services.sw.base
cluster_services.sw.cli
cxfs.books.CXFS_AG
cxfs.sw.cxfs
cxfs.sw.xvm_cell
sysadm_base.man.priv
sysadm_base.man.relnotes
sysadm_base.man.server
sysadm_base.sw.client
sysadm_base.sw.dso
sysadm_base.sw.priv
sysadm_base.sw.server
sysadm_cxfs.man.pages
sysadm_cxfs.man.relnotes
sysadm_cxfs.sw.client
```

```
sysadm_cxfs.sw.desktop
sysadm_cxfs.sw.server
sysadm_cxfs.sw.web
```

sysadm_base がインストールされると、tcpmux サービスが有効になります。

メモ: root 以外のログインから CXFS クラスタ・マネージャ・グラフィカル・ユーザ・インタフェース (GUI: Graphical User Interface) を実行したい場合は、sysadmdesktop もインストールします。これにより、CXFS コマンドの実行に必要な特権を含むユーザ特権を付与できるコマンドが提供されます。sysadmdesktop をインストールする場合は、以下のサブシステムをインストールします。

```
sysadmdesktop.man.base
sysadmdesktop.man.relnotes
sysadmdesktop.sw.base
sysadmdesktop.sw.data
sysadmdesktop.sw.sysadm
```

---

4. GUI の Web ベースのバージョンを使用する場合、任意のプラットフォームで動作する Java が有効な Web ブラウザを使用して管理操作を実行するには、接続先のプール・ノードに sysadm_base.sw.client、sysadm_cxfs.sw.client、および sysadm_cxfs.sw.web サブシステムがインストールされている必要があります。これらのサブシステムは、手順 3e でインストールしたデフォルト・ソフトウェアの一部です。

   また、以下のいずれかがインストールされている必要もあります。

   - nss_fasttrack.sw.server（6.5.5 applications CD-ROM から）
   - nss_enterprise.sw.server（Netscape CD-ROM から）

   これらのサブシステムの 1 つがまだインストールされていない場合は、適切な CD-ROM をロードして、該当するサブシステムをインストールしてください。

5. CXFS の管理を実行するワークステーション（管理ワークステーションと呼び、クラスタ内またはクラスタ外のノードを使用できます）で IRIX デスクトップから GUI クライアントを実行する場合は、以下のサブシステムをインストールします。

```
Inst> keep *
Inst> install java_eoe.sw (SGI version 3.1.1)
Inst> install java_eoe.sw32
Inst> install sysadm_base.man
Inst> install sysadm_base.sw.client
Inst> install sysadm_cxfs.man
Inst> install sysadm_cxfs.sw.client
```

```
Inst> install sysadm_cxfs.sw.desktop
Inst> go
```

注意: GUI は Java 1.1.8 でのみ動作します。これは、IRIX 6.5.*x* リリースに付属する Java のバージョンです。

SGI の Web サイトには Java 2 も掲載されていますが、このバージョンの Java は CXFS GUI では使用できません。1.1.8 以外のバージョンの Java を使用すると、GUI が失敗します。

6. 管理ワークステーションが、Java をサポートする Web ブラウザから GUI を起動する IRIX マシンの場合は(この設定は推奨されていません)、CXFS CD から `java_plugin` をインストールします。

メモ: Web ブラウザから GUI を起動する方法は、IRIX では推奨されていません。上記の手順 5 のように、IRIX デスクトップから GUI クライアントを実行する方法が推奨されています。

`java_plugin` のすべてのサブシステムをインストールしようとすると、サブシステム(`java_plugin.sw.swing101`、`java_plugin.sw.swing102`、および `java_plugin.sw.swing103`)に互換性のないことが `inst` によって報告されます。これらの 3 つのサブシステムは GUI では使用されないので、インストールしないでください。

Java Plug-in がインストールされていない場合にブラウザから GUI を実行すると、ブラウザは以下の URL にリダイレクトされます。

`http://java.sun.com/products/plugin/1.1/plugin-install.html`

Java Plug-in をインストールしたら、すべてのブラウザ・ウィンドウを閉じ、ブラウザを再起動してください。

IRIX 以外のワークステーションに対しては、上記の URL から Java Plug-in をダウンロードします。Java Plug-in がインストールされていない場合にブラウザから GUI を実行すると、ブラウザはこのサイトにリダイレクトされます。

7. `inst` を終了します。

```
Inst> quit
```

requickstarting プロセスは、完了までに数分かかる場合があります。

ソフトウェアをインストールして `inst` インタフェースを終了すると、変更内容を適用するためにシステムを再起動するようプロンプトが表示されます。ただし、再起動は62 ページの「システムの再起動」で説明する手順で行います。

システム・ファイルの設定

CXFS ソフトウェアをインストールするときは、システム・ファイルに関するいくつかの考慮事項があります。ネットワーク設定は非常に重要です。クラスタの各ノードは、その他のネットワーク・ルーティングを経由せずに、論理名と IP アドレスの両方で同じクラスタ内のすべてノードと相互に通信できなければなりません。つまり、適切な名前解決が重要です。静的ルーティングが推奨されています。

この節では、必要な必須の変更とオプションの変更について説明します。

ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定

注意: CXFS クラスタを設定する前に、以下の規則を理解しておくことが非常に重要です。

CXFS クラスタには、以下のホスト名解決規則と推奨事項が適用されます。

- ホスト名は、アンダースコア (_) で始めたり、スペースを含めることはできません。
- /etc/sys_id ファイルの値は、/etc/hosts ファイルのノードのプライマリ・ホスト名 (/etc/hosts のノードの IP アドレスの直後のフィールド) に一致しなければなりません。このフィールドは、ホスト名または完全修飾ドメイン名で指定できます。

  以下の場合は、/etc/sys_id の名前を使用する必要があります。

  – CXFS GUI に初めてログインする場合の GUI のログイン・ウィンドウの「サーバ(Server)」フィールド

  – プールの最初のノードを定義する場合の「ホスト名(Hostname)」フィールド

  /etc/hosts ファイルの形式は以下のとおりです。*primary_hostname* には、単純なホスト名または完全修飾ドメイン名を指定できます。

  IP_address primary_hostname aliases

  2 つのネットワークを用意することが推奨されています。CXFS が使用するのはハートビート/コントロール用のプライマリ・ネットワークだけですが、2 つのネットワークを用意しておくと、必要な場合にリセット・デーモンが 2 番目のネットワークにフェイルオーバーできます。リセット・デーモンは、リセット要求を異常ノードの適切なデーモンに転送し、ネットワークを通じてリセット・ハードウェアにアクセスできる場合は、そのハードウェアにアクセスします。

  たとえば、/etc/hosts に以下が含まれているとします。

```
# The public interface:
128.2.3.4  color-green.sgi.com color-green green
```

```
# The private interface:
192.0.1.1  color-green-private.sgi.com  color-green-private green-private
```

/etc/sys_id ファイルは、ホスト名 color-green または完全修飾ドメイン名 color-green.sgi.com で指定できます。エイリアス green を含めることはできません。

この場合、ログイン画面の「サーバ」フィールド、および「新規ノードの定義(Define a new node)」ウィンドウの「ホスト名」フィールドには、ホスト名 color-green または完全修飾ドメイン名 color-green.sgi.com を入力します。

メモ: /etc/sys_id の値を使用する必要があるのは、初期設定中に最初のノードにログインするときだけです。以降のノードには、green などのエイリアスを使用できます。

- nsd(1m) ネーム・サービス・デーモンを使用する場合は、NIS (Network Information Service) またはドメイン・ネーム・サービス (DNS: Domain Name Service) よりも先にローカル・ファイルがアクセスされるようシステムを設定する必要があります。つまり、/etc/nsswitch.conf の hosts の行の最初に files がリストされていなければなりません。例は、次のとおりです。

  ```
  hosts:      files nis dns
  ```

  (nis と dns の順序は CXFS では重要ではありません。files を最初にする必要があります。)

  /etc/config/netif.options ファイルでは、インタフェースの 1 つが /etc/sys_id ($HOSTNAME) の値と等しくなければなりません。

  USN (Unified Name Service) およびネーム・サービス・デーモンについての詳細については、nsd(1m) のマン・ページを参照してください。

- /etc/nsswitch.conf または /etc/hosts ファイルを変更した場合は、nsadmin restart コマンドを使用して nsd を再開する必要があります。これにより、キャッシュもフラッシュされます。

  これらのファイルの変更後に nsd(1m) を再開しなければならない理由は、nsd ネーム・サービス・デーモンは実際に /etc/hosts の内容を読取って、その内容をより高速に検索できる形式でメモリ・キャッシュに配置するためです。したがって、変更を反映させて新しい /etc/hosts の情報を RAM キャッシュに配置するには、nsd を再開しなければなりません。/etc/nsswitch.conf を変更した場合は、管理対象のファイルのタイプ(hosts や passwd など)、情報を得るために呼出すサービス、およびこれらのサービスの呼出し順序を判断するために、nsd がこのファイルを再度読取る必要があります。

  クラスタ・サービスがアクティブなときは、クラスタで実行中のノードの IP アドレスや、クラスタの最初のノードの IP アドレスを変更することはできません。

- /etc/hosts ファイルで完全修飾ドメイン名を使用する場合は、すべて完全修飾ドメイン名を使用してください。特定のノードの /etc/sys_id で完全修飾ドメイン名を使用した場合、クラスタ内の

ノードの /etc/hosts ファイルにそのホストの IP/ホスト名情報を定義するときは、すべてのノードでそのノードの完全修飾ドメイン名を使用する必要があります。

通常、完全修飾ドメイン名を使用するかどうかは、クライアント(NFS など)でのクライアント・サーバ・プログラムの名前解決方法や、デフォルトの解決方法などに応じて決定します。

- 最初のノード(クラスタ設定データベースを作成するノード)を定義した後で、/etc/nsswitch.conf ファイルのホスト名解決方法を変更した場合は、データベースを再度作成する必要があります。212 ページの「クラスタ設定データベースの再作成」を参照してください。
- IRIS FailSafe との同時実行を使用する場合は、/etc/sys_id の値を IP アドレス・エイリアスに関連付ける /etc/hosts のエントリは追加しないでください。この場合は、プライマリ・アドレスを使用しなければなりません。

/etc/services の設定

プール内の各ノードに cluster_admin 製品をインストールする前に、各ノードの /etc/services ファイルを編集して、sgi-cad および sgi-crsd のエントリを含めます。これらのプロセスに割当てられたポート番号は、プール内のすべてのノードで同じである必要があります。

以下に、sgi-cad および sgi-crsd の /etc/services エントリの例を示します。

```
sgi-crsd        7500/udp            # Cluster reset services daemon
sgi-cad         9000/tcp            # Cluster Admin daemon
```

/etc/config/cad.options の設定

各ノードの /etc/config/cad.options ファイルには、cad プロセスの開始時にクラスタ管理デーモンによって読取られるパラメータのリストが含まれています。cad は、GUI にクラスタ情報を提供します。

cad.options ファイルでは、以下のオプションを設定できます。

--append_log	cad ログ情報を cad ログ・ファイルに上書きするのではなく、追加します。
--log_file *filename*	cad ログ・ファイル名。これは、-lf *filename* として指定することもできます。
-vvvv	詳細レベル。v という文字の数は、ログのレベルを示します。-v に設定すると、最小限のメッセージだけがログに記録されます。-vvvv に設定すると、最大限のメッセージがログに記録されます。

デフォルトのファイルでは、以下のオプションが指定されています。

```
-lf /var/cluster/ha/log/cad_log --append_log
```

以下に、中間レベルの詳細度を使用した /etc/config/cad.options ファイルの例を示します。

```
-vv -lf /var/cluster/ha/log/cad_nodename --append_log
```

デフォルトのログ・ファイルは /var/cluster/ha/log/cad_log です。ログ・ファイルがすでに存在する場合は、エラーおよび警告メッセージはそのログ・ファイルに追加されます。

CLI または GUI を使用して /etc/config/cad.options ファイルの内容を変更することはできません。

メモ: 初期設定時以外に cad.options ファイルを変更した場合、変更を反映させるには、cad プロセスを再開する必要があります。これは、ノードを再起動するか、または次のコマンドを入力することで実行できます。

```
# /etc/init.d/cluster restart
```

実行中のクラスタでこのコマンドを実行した場合、クラスタは動作し続けますが、GUI と cad(1m) デーモンの接続は切断され、最接続するよう GUI によってプロンプトが表示されます。

ライセンスについての詳細は、47 ページの「必要なソフトウェアのインストール」を参照してください。

/etc/config/fs2d.options の設定

各ノードの /etc/config/fs2d.options ファイルには、cad プロセスの開始時にこのデーモンによって読取られるパラメータのリストが含まれています。fs2d デーモンは、プール内のすべてのノードへのクラスタ設定データベース (CDB) の配布を管理します。

fs2d.options ファイルでは、以下のオプションを設定できます。

-logevents *event name*	選択されたイベントをログに記録します。使用されるイベントの名前には、all、internal、args、attach、chandle、node、tree、lock、datacon、trap、notify、access、 storage などがあります。デフォルトは all です。
-logdest *log destination*	ログの記録先を設定します。使用されるログの記録先には、all、stdout、stderr、syslog、logfile などがあります。複数の記録先が指定されている場合、ログ・メッセージはそれらのすべてに書込まれます。logfile を指定した場合は、-logfile オプションも指定しないと効果がありません。デフォルトは -logdest stderr ですが、fs2d をデーモンとして実行する場合、ログは無効になります。これは、fs2d がデーモンとして実行されているときは、stdout および stderr が閉じられるためです。デフォルトは logfile です。
-logfile *filename*	ログ・ファイル名を設定します。デフォルトは /var/cluster/ha/log/fs2d_log です。

`-logfilemax` *maximum size*	ログ・ファイルの最大サイズ(バイト単位)を設定します。ファイルが最大サイズを超過した場合は、既存の `filename.old` が削除された後、現在のファイルの名前が `filename.old` に変更され、新しいファイルが作成されます。単一のメッセージが複数のファイルに分割されることはありません。`-logfile` が設定されている場合、デフォルトは 10000000 です。
`-loglevel` *loglevel*	ログ・レベルを設定します。使用されるログ・レベルには、`always`、`critical`、`error`、`warning`、`info`、`moreinfo`、`freq`、`morefreq`、`trace`、`busy` などがあります。デフォルトは `info` です。
`-trace` *trace_class*	選択されたイベントをトレースします。使用されるトレース・クラスには、`all`、`rpcs`、`updates`、`transactions`、`monitor` などがあります。このオプションを指定した場合は、`-tracefile` または `-tracelog`、あるいはその両方も指定してください。イベントの 1 つまたは複数のクラスのトレースを要求しても、`-tracefile` または `-tracelog`、あるいはその両方を指定しないかぎり、トレースは実行されません。デフォルトは `transactions` です。
`-tracefile` *filename*	トレース・ファイル名を設定します。デフォルトはありません。
`-tracefilemax` *maximum_size*	トレース・ファイルの最大サイズ(バイト単位)を設定します。ファイルが最大サイズを超過した場合は、既存の `filename.old` が削除された後、現在のファイルの名前が `filename.old` に変更され、新しいファイルが作成されます。
`-[no]tracelog`	-tracelog の場合はログの記録先をトレースし、-notracelog の場合はログの記録先をトレースしません。このオプションを設定すると、トレース・メッセージはログの記録先に送信されます。トレース・ファイルがある場合は、このファイルにもトレース・メッセージが書込まれます。デフォルトは `-tracelog` です。
`-[no]parent_timer`	-parent_timer の場合は親が存在すると終了し、-noparent_timer の場合は終了しません。デフォルトは `-noparent_timer` です。
`-[no]daemonize`	-daemonize の場合はデーモンとして実行し、-nodaemonize の場合はデーモンとして実行しません。デフォルトは `-daemonize` です。
`-l`	デーモンとして実行しません。
`-h`	使用法メッセージを出力します。
`-o help`	使用法メッセージを出力します。

これらのオプションに対してデフォルト値を使用すると、レベル `info` 以下のすべてのログ・メッセージと、トランザクション・イベントのすべてのトレース・メッセージがファイル `/var/cluster/ha/log/fs2d_log` に記録されるよう、システムが設定されます。ファイル・サイズが 10 Mバイトに達した場合、このファイルは拡張子が `.old` で同じ名前のファイルに移動され、ログは同じ名前の新しいファイルにロールオーバーされます。単一のメッセージが複数のファイルに分割されることはありません。

以下に、すべての fs2d ログ情報を /var/adm/SYSLOG に送信し、すべての fs2d トレース情報を /var/cluster/ha/log/fs2d_ops1 に送信する /etc/config/fs2d.options ファイルの例を示します。すべてのログ・イベントと、rpcs、updates、および transactions というトレース・イベントがログに記録されています。トレース・ファイル /var/cluster/ha/log/fs2d_ops1 のサイズが 100000000 を超過した場合は、このファイルの名前は /var/cluster/ha/log/fs2d_ops1.old に変更され、新しいファイル /var/cluster/ha/log/fs2d_ops1 が作成されます。単一のメッセージが複数のファイルに分割されることはありません（以下の例は改行して読みやすくしてあります）。

```
-logevents all -loglevel trace -logdest syslog
-trace rpcs -trace updates -trace transactions
-tracefile /var/cluster/ha/log/fs2d_ops1 -tracefilemax 100000000
```

以下に、すべてのログとトレース・メッセージを 1 つのファイル /var/cluster/ha/log/fs2d_chaos6 に送信する /etc/config/fs2d.options ファイルの例を示します。このファイルでは、最大サイズが 100000000 に指定されています。-tracelog は、トレース結果をログ・ファイルに送信します（以下の例は改行して読みやすくしてあります）。

```
-logevents all -loglevel trace -trace rpcs -trace updates
-trace transactions -tracelog
-logfile /var/cluster/ha/log/fs2d_chaos6
-logfilemax 100000000 -logdest logfile.
```

メモ: 初期設定時以外に fs2d.options ファイルを変更した場合、変更を反映させるには、fs2d プロセスを再開する必要があります。これは、ノードを再起動するか、または次のコマンドを入力することで実行できます。

```
# /etc/init.d/cluster restart
```

実行中のクラスタでこのコマンドを実行した場合、クラスタは動作し続けますが、GUI と cad(1m) デーモンの接続は切断され、最接続するよう GUI によってプロンプトが表示されます。

（オプション）/.rhosts の設定

CXFS に付属する接続性診断テストを使用する場合、rcp(1) や rsh(1) などのリモート・コマンドを実行するには、各ノードの /.rhosts ファイルでクラスタ内のすべてのホストが相互にアクセスできるようにします。たとえば、cxfs0、cxfs1、および cxfs2 の 3 つのノードで構成されるクラスタがあるとします。各ノードの /.rhosts ファイルは、以下のようになります（ノード名はプロンプトで示されています）。

```
cxfs0# cat /.rhosts
cxfs1 root
cxfs1-priv root
cxfs2 root
```

```
cxfs2-priv root

cxfs1# cat /.rhosts
cxfs0 root
cxfs0-priv root
cxfs2 root
cxfs2-priv root

cxfs2# cat /.rhosts
cxfs0 root
cxfs0-priv root
cxfs1 root
cxfs1-priv root
```

（オプション）/etc/exports の設定

各ノードの /etc/exports ファイルは、NFS クライアントにエクスポートされているファイルシステムを記述します。CXFS マウント・ポイントが exports ファイルに含まれている場合、CXFS マウント後にファイルシステムが再エクスポートされないかぎり、空のマウント・ポイントがエクスポートされます。

（オプション）自動再起動用の設定

ノードがリセットされたりノードの電源がオンになったときに自動的にノードを再起動したい場合は、以下のように各ノードの起動パラメータの AutoLoad 変数を yes に設定する必要があります。

```
# nvram AutoLoad yes
```

この設定が推奨されていますが、CXFS では必須ではありません。

この変数の設定は、以下のコマンドを使用してチェックできます。

```
# nvram AutoLoad
```

ネットワーク・インタフェースの設定

ネットワークを設定するときは、以下の点に注意してください。

- クラスタのすべてのノードは、ルーティングが指定されていなくても、IP アドレスと論理名を使用して直接通信できなければなりません。

- プライベート・ネットワークは、ハートビートおよびコントロール・ネットワーク専用にします。このネットワークではその他の負荷はサポートされません。
- ハートビートおよびコントロール・ネットワークがすべてのノードに接続されており、すべてのノードがそのネットワークに対して同じサブネットを使用するよう設定されている必要があります。

ネットワーク・インタフェースを設定するには、以下の手順に従います。

1. ネーム・サービスを使用可能にします。ローカル名前解決を使用する必要があります。DNS または NIS を使用している場合も、ノードのすべての IP アドレスとホスト名をすべてのノードの /etc/hosts に追加してください。例は、次のとおりです。

   ```
   190.0.2.1 server1-company.com server1
   190.0.2.3 stocks
   190.0.3.1 priv-server1
   190.0.2.2 server2-company.com server2
   190.0.2.4 bonds
   190.0.3.2 priv-server2
   ```

   続いて、これらのすべての IP アドレスをクラスタのその他のノードの /etc/hosts に追加する必要があります。

   hosts(4)、named(1M), dns(7P)、および nis(7P) のマン・ページを参照してください。『IRIX Admin Networking and Mail』および『NIS Administrator's Guide』も参照してください。

   メモ: NIS や DNS をクラスタ・ノードの IP アドレス・ルックアップ専用に使用すると、NIS や DNS サービスの信頼性が低下した場合に可用性が低下します。

   ---

   52 ページの「ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定」を参照してください。

2. 1 つのノードで、ノードのインタフェースとそれらの IP アドレスを /etc/config/netif.options に追加します。

   例:

   ```
   if1name=ec0
   if1addr=$HOSTNAME
   ```

   $HOSTNAME は、/etc/hosts で設定されている IP アドレスのエイリアスです。52 ページの「ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定」を参照してください。

   インタフェースがほかにもある場合は、インタフェースの名前と IP アドレスは、以下のような行に表示されます。

```
if2name=
if2addr=
```

この例では、コントロール・ネットワーク名および IP アドレスは以下のとおりです。

```
if3name=ec3
if3addr=priv-$HOSTNAME
```

この例のコントロール・ネットワーク IP アドレスである `priv-$HOSTNAME` は、`/etc/hosts` で設定されている IP アドレスのエイリアスです。

3. ノードに 9 つ以上のインタフェースがある場合は、`if_num` の値をインタフェースの数に変更します。インタフェースが 8 個未満の場合、行は以下のようになります。

   ```
   if_num=8
   ```

4. その他のノードに対して手順 1〜3 を繰返します。

5. ルートがコントロール・ネットワークを通じてアドバタイズされるように、各ノードの `/etc/config/routed.options` ファイルを編集します。オプションのリストについては、`routed`(1M) のマン・ページを参照してください。

   例:

   ```
   -q -h -Prdisc_interval=45
   ```

   これらのオプションは、以下を実行します。

   - ルーティングをオフにします。
   - ホストまたはポイントツーポイント・ルートがアドバタイズされないようにします(同じ方向のネットワーク・ルートがある場合)。
   - Router Discovery Advertisement が送信される公称周期を 45 秒(およびそれらの寿命を 135 秒)に設定します。

6. 各ノードで電子メール・エイリアスを設定します。この電子メール・エイリアスは、CXFS が CXFS クラスタ外のユーザやクラスタ内のその他のノードのユーザにクラスタが移行したことを通知する電子メールを送信するためのもです。たとえば、`cxfs1` および `cxfs2` という 2 つのノードがある場合は、`cxfs1` で、`/etc/aliases` ファイルに以下の行を追加します。

   ```
   cxfs_admin:operations@console.xyz.com,admin_user@cxfs2.xyz.com
   ```

   `cxfs2` で、`/usr/lib/aliases` ファイルに以下の行を追加します。

   ```
   cxfs_admin:operations@console.xyz.com,admin_user@cxfs1.xyz.com
   ```

選択したエイリアス(この場合は cxfs_admin)が、システムの設定時にメールの送信先アドレスに使用する値になります。この例では、operations がクラスタ外のユーザで、admin_user が各ノードのユーザです。

メモ: /usr/lib/aliases の編集が終わったら、newaliases(1M) コマンドを実行してください。

---

7. FDDI が使用されている場合は、FDDIXpress リリース・ノートおよび『FDDIXPress Administration Guide』の説明に従って、新しい FDDI ステーションの設定と確認を完了します。

シリアル・ポートの設定

ノードが別のマシンをリセットするように設定されている場合は、リセット・シリアル・ケーブルが接続されている tty ポートに対して getty プロセスをオフにする必要があります。この作業は、リセットを受信するノードではなく、リセットを実行するノードで実行してください。. これを行うには、各ノードで以下の手順を実行します。

1. リセット・シリアル回線に使用するポートを決定します。
2. ファイル /etc/inittab を開いて編集します。
3. 手順 1 のポート番号について、右側のコメントを参照してそのポートの行を検索します。
4. この行の 3 つ目のフィールドを off に変更します。例は、次のとおりです。

```
t2:23:off:/sbin/getty -N ttyd2 co_9600          # port 2
```

5. ファイルを保存します。
6. 以下のコマンドを入力して、変更を反映します。

```
# killall getty
# init q
```

メモ: IRISconsole システムで実行されるリセット・デーモンを使用してマルチノード・クラスタを設定する場合は、CXFS システムが実行しているリセット・デーモンと競合する可能性があるため、IRISconsole にはリセット・ポートを設定しないでください。

システムの再起動

各ノードで以下のコマンドを実行して、ノードを再起動します。

```
# reboot
```

シャットダウン・プロセスによって autoconfig(1M) が実行され、変更が反映されたカーネルが生成されます。

テスト

この節では、以下の内容について説明します。

- 「プライベート・ネットワーク・インタフェースのテスト」
- 63 ページの「シリアル接続のテスト」

プライベート・ネットワーク・インタフェースのテスト

プール内の各ノードの各プライベート・ネットワークに対して、以下のコマンドを入力します。*nodeIPaddress* は、ノードの IP アドレスです。

```
# /usr/etc/ping -c 3 nodeIPaddress
```

以下のような標準の ping(1) 出力が表示されます。

```
PING IPaddress (190.x.x.x: 56 data bytes
64 bytes from 190.x.x.x: icmp_seq=0 tt1=254 time=3 ms
64 bytes from 190.x.x.x: icmp_seq=1 tt1=254 time=2 ms
64 bytes from 190.x.x.x: icmp_seq=2 tt1=254 time=2 ms
```

ping が失敗する場合は、以下の手順に従います。

1. ifconfig を使用して、ネットワーク・インタフェースが UP に設定されていることを確認します。例は、次のとおりです。

```
# /usr/etc/ifconfig ec3
ec3: flags=c63<UP,BROADCAST,NOTRAILERS,RUNNING,FILTMULTI,MULTICAST>
inet 190.x.x.x netmask 0xffffff00 broadcast 190.x.x.x
```

   出力の最初の行にある UP は、インタフェースが設定されていることを示します。

2. ケーブルが正しく接続されていることを確認します。

各ノードでこの手順を繰返します。

シリアル接続のテスト

シリアル接続をテストするには、以下の手順に従います。

1. ノードとシリアル・マルチプレクサの電源がオンになっていることを確認します。
2. プール内のノードの 1 つで cmgr(1m) コマンドを開始します。

   ```
   # cmgr
   ```

3. 各ノードで HA サービスを停止します。

   ```
   stop cx_services for cluster clustername
   ```

   例:

   ```
   cmgr> stop cx_services for cluster cxfs6-8
   ```

   ノードが正常にスタンバイ状態に移行して CXFS プロセスが終了するまで待ちます。このプロセスには数分かかる場合があります。

4. 以下のいずれかを入力して、シリアル接続をテストします。

   - クラスタ全体をテストするには、以下を入力します。

     ```
     test serial in cluster clustername
     ```

     例:

```
cmgr> test serial in cluster cxfs6-8
Status: Testing serial lines ...
Status: Checking serial lines using crsd (cluster reset services) from node cxfs8
Success: Serial ping command OK.

Status: Checking serial lines using crsd (cluster reset services) from node cxfs6
Success: Serial ping command OK.

Status: Checking serial lines using crsd (cluster reset services) from node cxfs7
Success: Serial ping command OK.

Notice: overall exit status:success, tests failed:0, total tests executed:1
```

- 個々のノードをテストするには、以下を入力します。

  test serial in cluster *clustername* node *machinename*

  例:

```
cmgr> test serial in cluster cxfs6-8 node cxfs7
Status: Testing serial lines ...
Status: Checking serial lines using crsd (cluster reset services) from node cxfs6
Success: Serial ping command OK.

Notice: overall exit status:success, tests failed:0, total tests executed:1
```

- ping だけを使用して個々のノードをテストするには、以下を入力します。

  admin ping node *nodename*

  例:

```
cmgr> admin ping node cxfs7

ping operation successful
```

5. コマンドが失敗する場合は、すべてのケーブルが正しく接続されていることを確認してから、再度コマンドを実行します。
6. クラスタのほかのノードでこの手順を繰返します。

クラスタの初期設定

この章では、グラフィカル・ユーザ・インタフェース (GUI: Graphical User Interface) またはコマンド・ライン・インタフェース (CLI: Command Line Interface) を使用してクラスタを初期設定するために必要な手順の概要を説明します。213 ページの付録A「初期設定チェックリスト」で提供されているワークシートを使用できます。既存の IRIS FailSafe クラスタから切替える場合は、83 ページの「GUI を使った CXFS の既存の FailSafe クラスタの設定」を参照してください。

メモ: 初期インストールの場合は、GUI の設定ガイド・タスクセットを使用することが強く推奨されています。68 ページの「GUI を使った設定」を参照してください。また、クラスタを設定する前に、173 ページの 第8章「トラブルシューティング」を含むこのマニュアル全体をよくお読みください。

CXFS クラスタの設定と管理は複雑なタスクになる場合があります。問題の回避、診断、および解決についてのアドバイスは、173 ページの 第8章「トラブルシューティング」を参照してください。

この章は、47 ページの 第2章「CXFS ソフトウェアの初期インストールとシステムの準備」の手順が実行済みであることを想定しています。また、タスク・リファレンスおよび『XVM Volume Manager Administrator's Guide』の詳しい説明を参照しています。

メモ: CXFS では、各ノードにライセンスがインストールされていなければなりません。システムの CPU の数を増やすと、新しいライセンスが必要になることがあります。47 ページの「必要なソフトウェアのインストール」を参照してください。

準備手順

以下の手順を完了して、初期クラスタの設定準備を整えます。

- 66 ページの「ライセンスの確認」
- 66 ページの「クラスタ・デーモンが実行されていることの確認」
- 67 ページの「ノードのホスト名の判断」
- 68 ページの「`chkconfig` フラグがオンであることの確認」
- 66 ページの「クラスタ・デーモンの開始」

設定作業中は、ログ・ファイルにさまざまな情報専用メッセージが示されます。191 ページの「通常メッセージ」を参照してください。

ライセンスの確認

cxfslicense コマンドを使用して、CXFS ライセンスがあることを確認します。

```
cxfslicense -d
```

例:

```
# /usr/cluster/bin/cxfslicense -d
CXFS license granted.
```

適切にインストールされたライセンスがある場合は、各ノードの /var/flexlm/license.dat ファイルに FEATURE CXFS という行が示されます。このファイルは各ノードに固有なので、ノード間でコピーすることはできません。

ソフトウェア・ライセンスのインストールについての詳細は、『IRIX 6.5 Installation Instructions』ブックレットを参照してください。

クラスタ・デーモンの開始

以下のコマンドを入力してクラスタ・デーモンを開始します。

```
# /etc/init.d/cluster start
```

クラスタ・デーモンが実行されていることの確認

ソフトウェアをはじめてインストールするときは、以下のデーモンが実行されている必要があります。

- fs2d
- cmond
- cad
- crsd

CXFS サービスを開始すると、clconfd デーモンも開始されます。

実行されているデーモンを判断するには、以下のコマンドを入力します。

```
ps -ef | grep cluster
```

以下に、5 つのデーモンがすべて実行されている場合の出力の例を示します。ここでは、見やすいようにデーモン名を強調表示してあります。

```
cxfs6 # ps -ef | grep cluster
 root  342    1  0  Mar 01 ?  37:01 /usr/lib32/cluster/cbe/fs2d /var/cluster/cdb/cdb.db #
 root  346    1  0  Mar 01 ?   0:24 /usr/cluster/bin/cmond -L info -f /var/cluster/ha/log/cmond_log
 root  348  346  0  Mar 01 ?  36:58 /usr/cluster/bin/cad -l -lf /var/cluster/ha/log/cad_log --append_log
 root  352  346  0  Mar 01 ?  19:41 /usr/cluster/bin/crsd -l
 root  356  346  0  Mar 01 ?  61:12 /usr/cluster/bin/clconfd -l
 root 17873 17859  0 12:19:53 pts/0 0:00 grep cluster
```

デーモンを再開する必要がある場合は、以下のコマンドを入力します。

```
# /etc/init.d/cluster start
```

詳細については、212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」および20 ページの「デーモン」を参照してください。

ノードのホスト名の判断

クラスタを初期設定する場合、GUI にログインするときとプールの最初のノードを定義するときは、IP アドレスまたは /etc/sys_id の値(/etc/hosts のノードの IP アドレスのプライマリ名に一致していなければなりません)を使用する必要があります。hostname(1) コマンドを実行すると、/etc/sys_id の値が表示されます。例は、次のとおりです。

```
# hostname
cxfs
```

また、nsd(1m) を使用する場合は、NIS (Network Information Service) またはドメイン・ネーム・サービス (DNS: Domain Name Service) よりも先にローカル・ファイルがアクセスされるようシステムを設定する必要があります。52 ページの「ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定」を参照してください。

注意 : これらのファイルを適切に設定すること、およびプールの最初のノードに対してはプライマリ名を入力することが非常に重要です。以降のノードの定義にはエイリアスを使用できます。47 ページの「必要なソフトウェアのインストール」を参照してください。

chkconfig フラグがオンであることの確認

以下のフラグが on に設定されていることが chkconfig(1m) の出力に表示されることを確認します。

```
# chkconfig
        Flag                State
        ====                =====
        cluster             on
        cxfs_cluster        on
```

このように表示されない場合は、on に設定して再起動します。

例:

```
# init 1
# /etc/chkconfig cluster on
# /etc/chkconfig cxfs_cluster on
# init 2
```

GUI を使った設定

GUI を使ってクラスタを初期設定するには、以下を実行します。

- 65 ページの「準備手順」
- 69 ページの「GUI の起動」
- 71 ページの「GUI を使った新規クラスタの設定」
- 72 ページの「GUI を使った新規ファイルシステムの設定」

メモ: クラスタの表示は、どのノードから GUI を実行するかによって変わります。変更を加えたときは、次の変更を行う前に、「CXFS クラスタ表示(CXFS Cluster View)」ウィンドウに変更内容が表示されるまで待ってください。CXFS クラスタ表示に変更内容が表示されるまでは、その変更がクラスタ全体に伝わったことは保証されていません。変更は、特定の時点の GUI の 1 つのインスタンスだけから行ってください。2 つ目のインスタンスから変更を行うと、最初のインスタンスで加えた変更が上書きされる場合があります。

GUI の起動

以下のコマンドを入力して、CXFS マネージャおよび CXFS クラスタ表示を起動します。

```
# /usr/sbin/cxdetail
```

接続先のノードの名前と root パスワードを入力します。

注意: プールの最初のノードでは、「サーバ(Server)」フィールドで /etc/sys_id の値を使用することが非常に重要です。52 ページの「ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定」を参照してください。

クラスタをまだ定義してないので、クラスタが使用できないことが通知されますが、これはこの時点では正常なメッセージです。

クラスタ表示のメニューから、クラスタ内の要素を選択してチェックできます。CXFS マネージャを呼出すには、以下のメニュー・アイテムを選択します。

「オプション(Options)」
>「CXFS マネージャ(CXFS Manager)」

図3-1 に、CXFS マネージャの初期ウィンドウを示します。

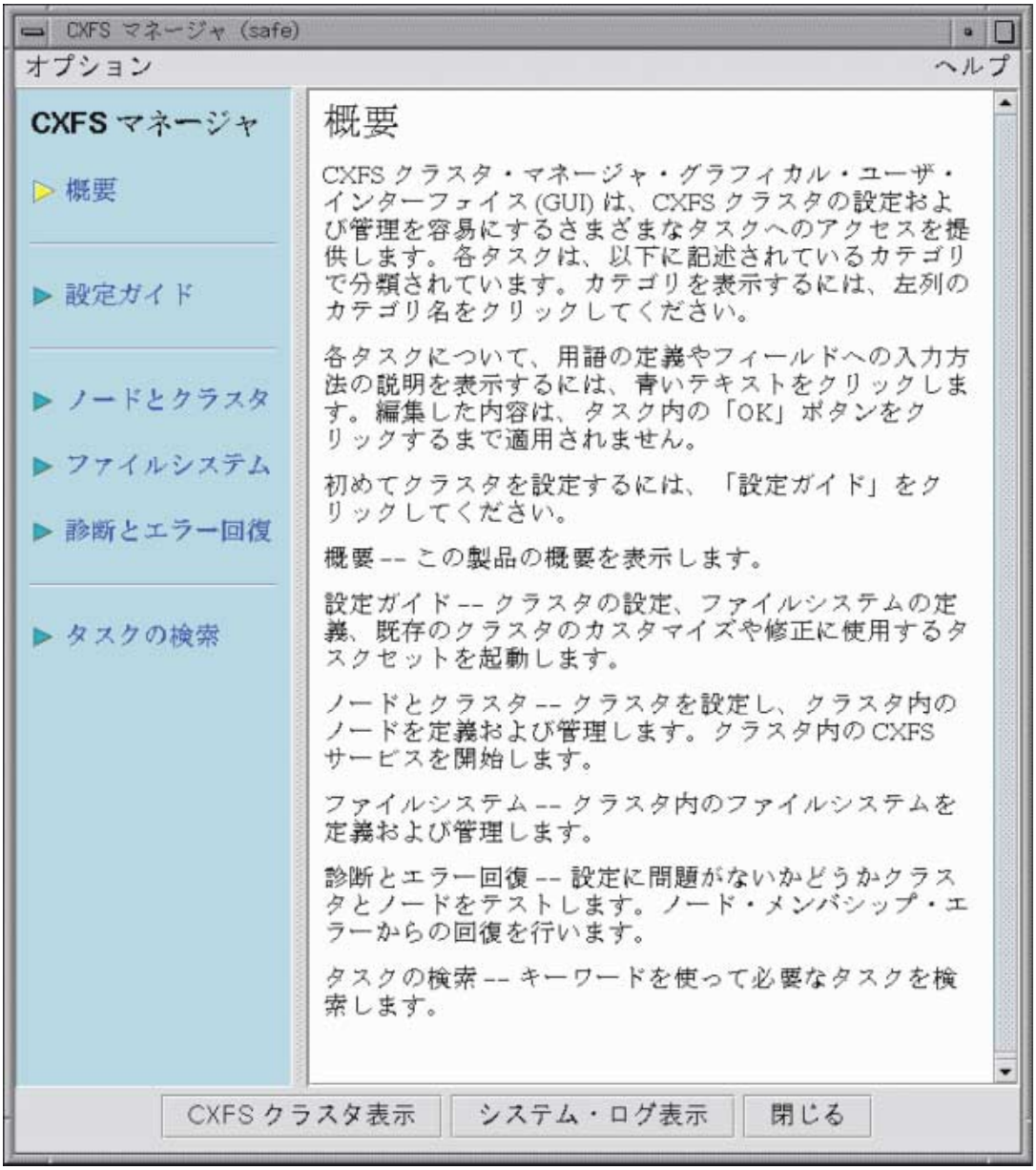

図3-1 CXFS マネージャ

GUI の起動方法はほかにもあります。詳細については、33 ページの「CXFS クラスタ・マネージャのツール」を参照してください。

GUI を使った新規クラスタの設定

メモ: CXFS タスク内で青いテキストをクリックすると、その概念や入力フィールドに関する詳細を参照できます。どのタスクでも、「OK」をクリックするまでクラスタ設定は更新されません。

「設定ガイド(Guided Configuration)」の「新規クラスタの設定(Set Up a New Cluster)」タスクセットを使用すると、新しいクラスタを作成するために必要な手順を実行できます。このタスクセットには、ほかの箇所で詳しく説明されているタスクが含まれます。

GUI を使用すると、クラスタとそのコンポーネントを CXFS クラスタ表示で簡単に表示できます。CXFS クラスタ表示を使用して実行状況を確認し、ノードを追加するのが早すぎることがないようにしてください。40 ページの「CXFS クラスタ表示の起動」を参照してください。

以下を実行します。

1. 「ノードの定義(Define a Node)」をクリックして、接続先のノードを定義します。最初のノードに対しては、`/etc/sys_id` に示されているホスト名を使用しなければなりません。67 ページの「ノードのホスト名の判断」を参照してください。84 ページの「GUI を使ったノードの定義」を参照してください。

2. (オプション)クラスタ表示に最初のノードのアイコンが表示されたら、「ノードの定義」をクリックして、クラスタのその他のノードを定義します。ホスト名/IP アドレスのペアおよびネットワークの優先度は、クラスタ内の各ノードに対して同じでなければなりません。84 ページの「GUI を使ったノードの定義」を参照してください。

   メモ: CXFS クラスタ表示にこのノードが表示されるまで、別のノードは定義しないでください。ノードを追加するのが早すぎると(データベースにノードを含めることができる前にノードを追加すると)、エラーが発生します。

   大規模なクラスタの場合は、重み付けされた最初の 3 つのノードを定義してから次の手順に進むことが推奨されています。つまり、正常な小さいクラスタを作成してから残りのノードを追加します。

3. 「クラスタの定義(Define a Cluster)」をクリックして、クラスタ定義を作成します。95 ページの「GUI を使ったクラスタの定義」を参照してください。

4. クラスタ表示にクラスタのアイコンが表示されたら、「クラスタ内のノードの追加と削除(Add/Remove Nodes in Cluster)」をクリックして、新しいクラスタにノードを追加します。90 ページの「GUI を使ったクラスタ内のノードの追加と削除」を参照してください。

   「次へ(Next)」をクリックして、タスクの 2 番目の画面に移動します。

5. (オプション)「接続性のテスト(Test Connectivity)」をクリックして、ノードが物理的に接続されていることを確認します。104 ページの「GUI を使った接続性のテスト」を参照してください。
6. 「CXFS サービスの開始(Start CXFS Services)」をクリックします。98 ページの「GUI を使った CXFS サービスの開始」を参照してください。
7. 「閉じる(Close)」をクリックします。「閉じる」をクリックすると、タスクセットは終了します。これはタスクを元に戻すものではありません。

GUI を使った新規ファイルシステムの設定

メモ: CXFS タスク内で青いテキストをクリックすると、その概念や入力フィールドに関する詳細を参照できます。どのタスクでも、「OK」をクリックするまでクラスタ設定は更新されません。

「新規ファイルシステムの設定(Set Up a New Filesystem)」タスクセットを使用すると、新しいファイルシステムを作成してクラスタ内のすべてのノードにマウントするために必要な手順を実行できます。このタスクセットには、ほかの箇所で詳しく説明されているタスクが含まれます。

以下を実行します。

1. CXFS サービスがまだ開始されていない場合は、「CXFS サービスの開始(Start CXFS Services)」をクリックします（現在のステータスはタスク・リンクの下に表示されています）。98 ページの「GUI を使った CXFS サービスの開始」を参照してください。
2. クラスタ内のいずれかのノードのシェル・ウィンドウを開き、`fx`(1M) コマンドを使用して、ボリュームで使用するディスク・ドライブのボリューム・ヘッダを作成します（この手順に対しては GUI は使用しません）。詳細については、`fx`(1M) のマン・ページと、『IRIX Admin: Disks and Filesystems』のディスク管理手順の実行に関する章を参照してください。
3. シェル・ウィンドウで XVM 論理ボリュームを作成します。`xvm`(1m) `probe` コマンドを使用して、ラベルの付いていないボリュームのリストを表示できます。詳細については、『XVM Volume Manager Administrator's Guide』を参照してください。
4. シェル・ウィンドウで、`mkfs`(1M) コマンドを使用して CXFS ファイルシステムを作成します。『IRIX Admin: Disks and Filesystems』を参照してください。
5. GUI に戻り、「ファイルシステムの定義(Define a Filesystem)」をクリックします。101 ページの「GUI を使ったファイルシステムの定義」を参照してください。
6. 「ファイルシステムのマウント(Mount a Filesystem)」をクリックします。102 ページの「GUI を使ったファイルシステムのマウント」を参照してください。

各ファイルシステムに対して上記の手順を繰返します。

CLI を使った設定

メモ: 初期インストールの場合は、GUI の設定ガイド・タスクセットを使用することが強く推奨されています。68 ページの「GUI を使った設定」を参照してください。mount(1M) コマンドについての詳細は、マン・ページを参照してください。

CLI (cmgr(1m)) を使ってクラスタを初期設定するには、以下を実行します。

1. 65 ページの「準備手順」
2. クラスタの一部にできるノードを定義します。ホスト名 /IP アドレスのペアおよびネットワークの優先度は、クラスタ内の各ノードに対して同じでなければなりません。109 ページの「CLI を使ったノードの定義」を参照してください。

   大規模なクラスタの場合は、最初の 3 つのノードを定義してから次の手順に進むことが推奨されています。つまり、正常な小さいクラスタを作成してから残りのノードを追加します。以下に、3 つのノードを定義する順序の例を示します。

   メモ: プロンプトに対してデフォルト値を使用するには、<Enter> キーを押します。このマニュアルの例では、<Enter> キーは示されていません。

   cxfs6 という名前の最初のノードを定義するには、以下を入力します。

```
cxfs6 # /usr/cluster/bin/cmgr -p
Welcome to SGI Cluster Manager Command-Line Interface

cmgr> define node cxfs6
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ?
Is this a FailSafe node <true|false> ? false
Is this a CXFS node <true|false> ? true
Node ID[optional]?
Reset type <powerCycle>  ? (powerCycle)
Do you wish to define system controller info[y/n]:n
Number of Network Interfaces ? (1)
NIC 1 - IP Address ? cxfs6
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ?  true
NIC 1 - (use network for control messages) <true|false> ? true
NIC 1 - Priority <1,2,...> ? 1
Node Weight ? (1)
```

```
Successfully defined node cxfs6
```

cxfs7 という名前の 2 番目のノードを定義するには、以下を入力します。

```
cmgr> define node cxfs7
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ?
Is this a FailSafe node <true|false> ? false
Is this a CXFS node <true|false> ? true
Node ID ?
Reset type <powerCycle>  ? (powerCycle)
Do you wish to define system controller info[y/n]:n
Number of Network Interfaces ? (1)
NIC 1 - IP Address ? cxfs7
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ? true
NIC 1 - (use network for control messages) <true|false> ? true
NIC 1 - Priority <1,2,...> ? 1
Node Weight ? (1)

Successfully defined node cxfs7
```

cxfs8 という名前の 3 番目のノードを定義するには、以下を入力します。

```
cmgr> define node cxfs8
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ?
Is this a FailSafe node <true|false> ? false
Is this a CXFS node <true|false> ? true
Node ID ?
Reset type <powerCycle>  ? (powerCycle)
Do you wish to define system controller info[y/n]:n
Number of Network Interfaces ? (1)
NIC 1 - IP Address ? cxfs8
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ? true
NIC 1 - (use network for control messages) <true|false> ? true
NIC 1 - Priority <1,2,...> ? 1
Node Weight ? (1)

Successfully defined node cxfs8
```

これでプールに 3 つのノードが定義されました。これを確認するには、以下のコマンドを入力します。

```
cmgr> show nodes in pool

3 Machine(s) defined
        cxfs6
        cxfs7
        cxfs8
```

ノード cxfs6 の内容を表示するには、以下のコマンドを入力します。

```
cmgr> show node cxfs6

Logical Machine Name: cxfs6
Hostname: cxfs6.americas.sgi.com
Node Is FailSafe: false
Node Is CXFS: true
Nodeid: 13203
Reset type: powerCycle
ControlNet Ipaddr: cxfs6
ControlNet HB: true
ControlNet Control: true
ControlNet Priority: 1
Node Weight: 1
```

3. クラスタを定義してノードを追加します。121 ページの「CLI を使ったクラスタの定義」を参照してください。

   たとえば、TEST という名前のクラスタを定義して、定義済みのノードを追加するには、以下を入力します。

```
cmgr> define cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> false ?
Is this a CXFS cluster  <true|false> true ?
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster mode <normal|experimental>[optional]
Cluster ID ? 20
Number of Cluster FileSystems ? (0)

No nodes in cluster TEST
```

```
Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST ? add node cxfs6
TEST ? add node cxfs7
TEST ? add node cxfs8
TEST ? done
Successfully defined cluster TEST
```

詳細については、121 ページの「CLI を使ったクラスタの定義」を参照してください。

クラスタとその内容を確認するには、以下を入力します。

```
cmgr> show clusters

1 Cluster(s) defined
        TEST

cmgr> show cluster TEST
Cluster Name: TEST
Cluster Is FailSafe: false
Cluster Is CXFS: true
Cluster ID: 20
Cluster CX mode: normal

Cluster TEST has following 3 machine(s)
        cxfs6
        cxfs7
        cxfs8
```

スクリプトを使用したこの手順の例は、141 ページの「スクリプトの例」を参照してください。

4. 以下を入力して、クラスタの CXFS サービスを開始します。

   ```
   start cx_services for cluster clustername
   ```

   例:

   ```
   cmgr> start cx_services for cluster TEST

   CXFS services have been activated in cluster TEST
   ```

   このアクションにより CXFS サービスが開始され、システムの再起動時に常に CXFS サービスが自動的に再開されるように設定されます。

メモ: GUI または CLI を使用して CXFS サービスを停止すると、自動再開機能はオフになります。自動再開機能をオンに戻すには、CXFS サービスを再度開始する必要があります。

クラスタが動作していることを確認するには、以下の cmgr(1m) コマンドを使用できます。

```
show status of cluster clustername
```

例:

```
cmgr> show status of cluster cxfs6-8

Cluster (cxfs6-8) is not configured for FailSafe


CXFS cluster state is ACTIVE.
```

また、clconf_info(1m) コマンドを使用することもできます。例は、次のとおりです。

```
cxfs6 # /usr/cluster/bin/clconf_info
Membership since Thu Mar  1 08:15:39 2001
Node          NodeId     Status    Age   Incarnation     CellId
cxfs6              6         UP      0             0          2
cxfs7              7         UP      0             0          1
cxfs8              8         UP      0             0          0
1 CXFS FileSystems
/dev/cxvm/concat0 on /concat0  enabled  server=(cxfs7)  2 client(s)=(cxfs8,cxfs6)
```

詳細については、133 ページの「CLI を使ったクラスタの表示」を参照してください。

5. クラスタ内のいずれかのノードのシェル・ウィンドウを開き、fx(1M) コマンドを使用して、ディスク・ドライブのボリューム・ヘッダを作成します。詳細については、『IRIX Admin: Disks and Filesystems』を参照してください。
6. シェル・ウィンドウで XVM 論理ボリュームを作成します。詳細については、『XVM Volume Manager Administrator's Guide』を参照してください。
7. シェル・ウィンドウで、mkfs(1M) コマンドを使用して ファイルシステムを作成します。詳細については、『XVM Volume Manager Administrator's Guide』を参照してください。
8. cmgr(1M) の modify cluster サブコマンドを使用して、クラスタにファイルシステムを追加します。124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」を参照してください。

以下に、test1 ファイルシステムのメタデータ・サーバとして使用可能な 2 つのノードの例を示します。クラスタの開始時に cxfs6（ランク 0 の優先サーバ）が動作していなかったり、異常が発生したり、クラスタから削除された場合は、cxfs8（ランク 1）が使用されます。

メモ: 特定のファイルシステムのメタデータ・サーバのリストは順序付きですが、ネットワーク・レーテンシやその他の予測不能な遅れがあるため、起動サイクル中にどのノードが実際にサーバになるかは予想できません。

以下を実行します。デフォルト設定を使用するには、<Enter> キーを押します。

```
cmgr> modify cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> ? (false)
Is this a CXFS cluster <true|false> ? (true)
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster mode <normal|experimental>[optional] ? (normal)
Cluster ID ? (20)
Number of Cluster FileSystems ? (0) 2
FILESYSTEM 0 - Device Name ? /dev/cxvm/dks2d72s0
FILESYSTEM 0 - Mount Point ? /mnts/fs1
FILESYSTEM 0 - Mount Options[optional] ?
FILESYSTEM 0 - Status <enabled|disabled> ? enabled
FILESYSTEM 0 - Forced unmount ? <true|false> ? false
Number of Servers ? (0) 2
FILESYSTEM 0 SERVER 0 - Server Node ? cxfs6
FILESYSTEM 0 SERVER 0 - Server Rank ? 0
FILESYSTEM 0 SERVER 1 - Server Node ? cxfs7
FILESYSTEM 0 SERVER 1 - Server Rank ? 1
FILESYSTEM 1 - Device Name ? /dev/cxvm/dks2d73s0
FILESYSTEM 1 - Mount Point ? /mnts/fs2
FILESYSTEM 1 - Mount Options[optional] ?
FILESYSTEM 1 - Status <enabled|disabled> ? enabled
FILESYSTEM 1 - Forced unmount ? <true|false> ? false
Number of Servers ? (0) 1
FILESYSTEM 1 SERVER 0 - Server Node ? cxfs8
FILESYSTEM 1 SERVER 0 - Server Rank ? 0

Current nodes in cluster TEST:
Node - 1: cxfs6
```

```
Node - 2: cxfs7
Node - 3: cxfs8

Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST ? done
Successfully modified cluster TEST
```

クラスタ TEST の変更内容を表示するには、次を入力します。

```
cmgr> show cluster TEST
Cluster Name: TEST
Cluster Is FailSafe: false
Cluster Is CXFS: true
Cluster ID: 20
Cluster CX mode: normal
FileSystem Device Name: /dev/cxvm/dks2d72s0
FileSystem Mount Point: /mnts/fs1
FileSystem Mount Options:
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs7
        FileSystem Server Rank: 1
        FileSystem Server Node: cxfs6
        FileSystem Server Rank: 0
FileSystem Device Name: /dev/cxvm/dks2d73s0
FileSystem Mount Point: /mnts/fs2
FileSystem Mount Options: enabled
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs8
        FileSystem Server Rank: 0


Cluster TEST has following 3 machine(s)
        cxfs6
        cxfs7
        cxfs8
```

9. cmgr を終了するには、以下のコマンドを入力します。

```
cmgr> quit
```

GUI タスク・リファレンス

この章では、以下のタスクを実行するために CXFS マネージャ・グラフィカル・ユーザ・インタフェース (GUI: Graphical User Interface) を使用する方法について説明します。

- 初期設定タスクセット
  - 71 ページの「GUI を使った新規クラスタの設定」
  - 72 ページの「GUI を使った新規ファイルシステムの設定」
  - 83 ページの「GUI を使った CXFS の既存の FailSafe クラスタの設定」
- ノード・タスク
  - 84 ページの「GUI を使ったノードの定義」
  - 90 ページの「GUI を使ったクラスタ内のノードの追加と削除」
  - 90 ページの「GUI を使ったノードのリセット 」
  - 91 ページの「GUI を使ったノードの変更」
  - 93 ページの「GUI を使った FailSafe ノードの CXFS への切替え」
  - 94 ページの「GUI を使ったノードの削除」
  - 94 ページの「GUI を使ったノードの表示」
- クラスタ・タスク
  - 95 ページの「GUI を使ったクラスタの定義」
  - 96 ページの「GUI を使ったクラスタ定義の変更」
  - 96 ページの「GUI を使った FailSafe クラスタの CXFS への切替え」
  - 97 ページの「GUI を使ったクラスタの削除」
  - 97 ページの「GUI を使ったクラスタの表示」
- CXFS サービス・タスク
  - 98 ページの「GUI を使った CXFS サービスの開始」
  - 98 ページの「GUI を使った CXFS サービス（通常 CXFS シャットダウン）の停止」

  - 99 ページの「GUI を使ったタイブレーカー・ノードの設定」
  - 99 ページの「GUI を使ったログの設定」
- ファイルシステム・タスク
  - 101 ページの「GUI を使ったファイルシステムの定義」
  - 102 ページの「GUI を使ったファイルシステムのマウント」
  - 103 ページの「GUI を使ったファイルシステムのアンマウント」
  - 103 ページの「GUI を使ったファイルシステムの変更」
  - 104 ページの「GUI を使ったファイルシステムの削除」
- 診断およびエラー回復タスク
  - 104 ページの「GUI を使った接続性のテスト」
  - 105 ページの「GUI を使ったローカル・ノードのメンバーシップ取消し」
  - 105 ページの「GUI を使ったローカル・ノードのメンバーシップ許可」

メモ: CXFS では、各ノードにライセンスがインストールされていなければなりません。ライセンスを正しくインストールせずにソフトウェアをインストールすると、エラーが発生し、CXFS GUI を使用することはできません。ライセンスについての詳細は、47 ページの「必要なソフトウェアのインストール」を参照してください。

初期設定ガイド・タスクセット

初期設定タスクセットには、以下の 3 つがあります。

- 71 ページの「GUI を使った新規クラスタの設定」
- 72 ページの「GUI を使った新規ファイルシステムの設定」
- 83 ページの「GUI を使った CXFS の既存の FailSafe クラスタの設定」

その他のタスクセットには、以下を実行するために必要な手順の概要が示されています。

- クラスタの変更
- クラスタ・ノードの修正またはアップグレード
- ステータスのチェック

これらのタスクセットにアクセスするには、CXFS マネージャの左側にある「設定ガイド(Guided Configuration)」をクリックします。

GUI を使った CXFS の既存の FailSafe クラスタの設定

メモ: CXFS タスク内で青いテキストをクリックすると、その概念や入力フィールドに関する詳細を参照できます。どのタスクでも、「OK」をクリックするまでクラスタ設定は更新されません。

「CXFS の既存の FailSafe クラスタの設定(Set Up an Existing FailSafe Cluster for CXFS)」タスクセットを使用すると、既存の FailSafe ノードおよびクラスタを CXFS に切替えるために必要な手順を実行できます。このタスクセットには、ほかの箇所で詳しく説明されているタスクが含まれます。FailSafe もインストールされている場合、このタスクセットは CXFS GUI に表示されます。

IRIS FailSafe および CXFS のデータベースは 1 つだけです。特定のノードが両方の製品に適用される場合は、変更内容が両方の製品に対して適切であることを確認してください。

以下を実行します。

1. 「CXFS クラスタの FailSafe への切り替え(Convert a CXFS Cluster to FailSafe)」をクリックします。96 ページの「GUI を使った FailSafe クラスタの CXFS への切替え」を参照してください。これにより、クラスタ・タイプは `CXFS and FailSafe` に変わります。
2. FailSafe GUI を使用して、切替える HA サービスをノードで停止します。『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。
3. CXFS GUI を使用して、2 番目のハートビートおよびコントロール・ネットワークをノード定義に追加します。91 ページの「GUI を使ったノードの変更」を参照してください。
4. 「CXFS ノードの FailSafe への切り替え(Convert a CXFS Node to FailSafe)」をクリックして、ローカル・ノード(接続しているノード)に切替えます。切替えたノードのタイプは、`CXFS and FailSafe` または `CXFS` になります。93 ページの「GUI を使った FailSafe ノードの CXFS への切替え」を参照してください。
5. 「CXFS ノードの FailSafe への切り替え(Convert a CXFS Node to FailSafe)」をクリックして、別のノードに切替えます。切替えたいノードに対して、この手順を繰返します。
6. 「CXFS サービスの開始(Start CXFS Services)」をクリックして、CXFS サービスを開始します。

GUI を使ったノード・タスク

この節では、GUI を使用してノードを定義、変更、削除、表示、およびリセットする方法について説明します。

メモ:「新規クラスタの設定(Set Up a New Cluster)」設定ガイド・タスクセットを使用すると、クラスタおよびノードを設定するために必要なタスクを実行できます。71 ページの「GUI を使った新規クラスタの設定」を参照してください。

GUI を使ったノードの定義

クラスタ・ノードは、単一の IRIX イメージです。通常、クラスタ・ノードは個々のコンピュータです。ノードという用語は、このガイド内では略して使用されることがあり、この意味でのノードという用語は、Origin システムのノードとは異なる意味を持ちます。プールは、クラスタリングに利用できるノードのセット全体です。クラスタ設定データベースは、プール内のすべてのノードに配布されます。

定義する最初のノードは、クラスタ管理を実行するためにログインしたノードでなければなりません。最初のノードを定義するには、`/etc/sys_id` に表示されているホスト名を使用しなければなりません。

メモ:CXFS タスク内で青いテキストをクリックすると、その概念や入力フィールドに関する詳細を参照できます。どのタスクでも、「OK」をクリックするまでクラスタ設定は更新されません。

ノードを定義するには、以下を実行します。

1. 以下を入力します。

   - 「ホスト名(Hostname)」: `mynode.company.com`(すべてのノードで解決される場合は、`mynode` に省略できます)など、定義するノードのホスト名。プール内の最初のノードを定義するには、`/etc/sys_id` に表示されているホスト名を使用しなければなりません。`/etc/sys_id` ファイルに表示されているとおりにホスト名を表示するには、`hostname`(1) コマンドを使用します。52 ページの「ホスト名解決: `/etc/sys_id`、`/etc/hosts`、`/etc/nsswitch.conf` の設定」を参照してください。

   - 「論理名(Logical Name)」: ホスト名の簡略名(`lilly` など)、またはまったく別の名前(`nodeA` など)。論理名は、アンダースコア (_) で始めたり、空白として使用される文字を含めたり、255 文字より長くすることはできません。

     メモ:ノードの名前を変更する場合は、そのノードを削除した後、新しいノードを定義しなければなりません。

- 「メンバーシップ重み(Membership Weight)」: 0 または 1 で、CXFS メンバーシップ定義でのノードの重みを指定します。CXFS メンバーシップを決定するときにノードが考慮に入れられないようにする場合は、0 を使用します。デフォルトは 1 です。少なくとも 1 つのノードの重みに 1 を指定しなければなりません。0 または 1 以外の値は推奨されません。

  CXFS メンバーシップ上限量を形成するためには、重み付けされたノードの **50%** 以上が初期クラスタに必要です。クラスタを維持するには、重み付けされたノードの 50% が実行中でなければなりません。詳細については、13 ページの「上限量の計算とノードの重み」を参照してください。

- 「受信クラスタ・メッセージ用ネットワーク(Networks for Incoming Cluster Messages)」: 以下を実行します。

  – 「ネットワーク(Network)」: プライベート・ネットワークの IP アドレスまたはホスト名を入力します。ホスト名は、`/etc/hosts` で解決されなければなりません。ネットワークの優先度は、クラスタ内の各ノードに対して同じでなければなりません。ホスト名の使用についての詳細は、52 ページの「ホスト名解決: `/etc/sys_id`、`/etc/hosts`、`/etc/nsswitch.conf` の設定」を参照してください。プライベート・ネットワークが必要である理由については、8 ページの「プライベート・ネットワーク」を参照してください。

  メモ: GUI では複数のネットワークを入力できますが、CXFS が使用するのは優先度が 1 のネットワークだけで、このネットワークはフェイルオーバーされません。ただし、優先度が 1 のネットワークに異常が発生した場合でもクラスタ・インフラストラクチャがリセット用として使用できる追加のネットワークを定義できます。

  – 「受信メッセージ(Messages to Accept)」: 「ハートビートおよびコントロール(Heartbeat and Control)」を選択します。

  メモ: GUI では「ハートビート(Heartbeat)」または「コントロール(Control)」だけを指定できますが、これらの設定は CXFS では使用できないので選択しないでください。CXFS は、ハートビートとコントロールの両方に対して単一のネットワークを使用します。

  ネットワークを一時的に定義して、ネットワークでメッセージを受信しないようにする場合は、「なし(None)」設定を使用できます。詳細については、6 ページの「クラスタ環境」を参照してください。

  – ネットワークをリストに追加するには、「追加(Add)」をクリックします。

  後でこのネットワークを変更する場合は、リストでそのネットワークをクリックして選択し、「変更(Modify)」をクリックします。

  リストからネットワークを削除する場合は、リストでそのネットワークをクリックして選択し、「削除(Delete)」をクリックします。

- 「ノード ID(Node ID)」: プール内のノード全体で一意な、1～32767 の範囲の整数(オプション)。数字を指定しないと、CXFS によって自動的に ID が計算されます。デフォルトの ID は、マシンのシリアル・ナンバーなどのマシン固有の情報を基にした 5 桁の数字で、連続する数字ではありません。ノードを定義したら、ノード ID 番号は変更しないでください(デフォルトのタイブレーカーがなくなります。詳細については、13 ページの「メンバーシップ上限量」を参照してください)。

  システム・コントローラ・ポートの設定画面に移動するには、「次へ(Next)」をクリックします。

- 「パーティション ID(Partition ID)」: パーティションが設定された Origin 3000 システムのパーティションを一意に定義します(オプション)。システムにパーティションが設定されていない場合、このフィールドは空のままにしてください。

メモ: この値を決定するには、`mkpart`(1m) コマンドを使用してください。

  - `-n` オプションは、パーティション ID をリストします(システムにパーティションが設定されていない場合は 0 になります)。
  - `-l` オプションは、さまざまなパーティションのブリックをリストします(GUI では *rack#.slot#* 形式を使用してください)。

    例は、次のとおりです(読みやすくするために出力を切詰めてあります)。

```
# mkpart -n
Partition id = 1
# mkpart -l
partition: 3 = brick: 003c10 003c13 003c16 003c21 003c24 003c29 ...
partition: 1 = brick: 001c10 001c13 001c16 001c21 001c24 001c29 ...
```

    「パーティション ID(Partition ID)」フィールドには、以下のいずれかを入力できます。

```
1
001.10
```

  次の画面に移動するには、「次へ」をクリックします。

- ノードをリセットするためにシステム・コントローラ・ポートを使用するかどうかを選択できます。ノードをリセットするために CXFS でシステム・コントローラを使用できるようにする場合は、「リセット・パラメータの設定(Set Reset Parameters)」チェックボックスをオンにし、次の情報を指定します。

  - このノードの場合:
    - 「ポート・タイプ(Port Type)」: プルダウン・リストから、「Challenge」、「MSC」(モジュール・システム・コントローラ)、または「MMSC」を選択します。

ノードのポート・タイプがわからない場合は、「ポート・タイプ(Port Type)」という青いテキストをクリックして、ポート・タイプに基づいたマシンのリストを参照してください。

- 「ポート・パスワード(Port Password)」: システム・コントローラ・ポートのパスワードで、ノードの `root` パスワードや PROM パスワードではありません。マシンによっては、システム管理者がこのパスワードを設定していない可能性があります。システム・コントローラ・ポート・パスワードを設定または変更する場合は、ご使用のノードのハードウェア・マニュアルを参照してください。
- 「一時的に無効なポート(Temporarily Disable Port)」: リセット情報を指定して、現時点ではリセット機能を使用できないようにする場合は、このボックスをオンにします。このボックスがオンになっていると、CXFS はノードをリセットできません。

– オーナー(リセット・コマンドを送信するノード)の場合:

- 「論理名(Logical Name)」: リモート・リセット・コマンドを送信するノードの名前。シリアル・ケーブルは、システム・コントローラ・ポートを通じて、定義されているノードとオーナー・ノードに物理的に接続されていなければなりません。ランタイム時に、ノードは CXFS プールで定義される必要があります。

  プルダウン・リストから論理名を選択したり、まだ定義されていないノードの論理名を入力できます。ただし、CXFS のノードは、ノードの接続性診断タスクを実行する前に定義しなければなりません。

- 「TTY デバイス(TTY Device)」: システム・コントローラが接続されているオーナー・ノードの端末ポート (TTY) の名前(`/dev/ttyd2` など)。オーナー・ノードからノードをリモートで制御できるよう、ケーブルのもう一方はこのノードのシステム・コントローラ・ポートに接続します。

リセット機能をまったく使用しない場合は、「リセット・パラメータの設定(Set Reset Parameters」ボックスをクリックして、選択を解除します(オフにします)。

2. 「OK」をクリックします。

図4-1 および 図4-2 に、画面の例を示します。

メモ: ネットワーク・インタフェース・フィールドへの入力として、ホスト名または IP 番号を使用できます。ただし、ホスト名を使用するにはノードに DNS が必要となるため、実際の IP 番号を使用することをお薦めします。

---

safe の ノードの定義 (ステップ 1/2)
新規ノードの定義
このタスクを使用して、クラスタに入れることができる新しいノードを作成できます。ほかのタスクと同様、「OK」をクリックするまで、変更は反映されません。 オプションの設定はイタリック体で表されています (日本語画面では、プロパティ名の後にアスタリスク (*) が付いています)。
ノードが定義されていない場合は、ログインするノードを最初に定義する必要があります。この場合、ホスト名はあらかじめ指定されます。
メモ : 青色のテキストをクリックすると、その用語の定義、またはそのフィールドに入力する方法に関する説明を表示できます。
ホスト名 : safe.engr.sgi.com
論理名 : safe
メンバーシップ重み : 1
受信クラスタ・メッセージ用ネットワーク :
ネットワーク :
受信メッセージ :
ハートビートおよびコントロール
追加
優先度 ネットワーク 受信メッセージ
1. 192.0.2.127 ハートビートおよびコントロール
変更
削除
ノード ID*: 1
パーティション ID*:
前へ
OK
キャンセル
ヘルプ
次へ

図4-1 ノード定義の例

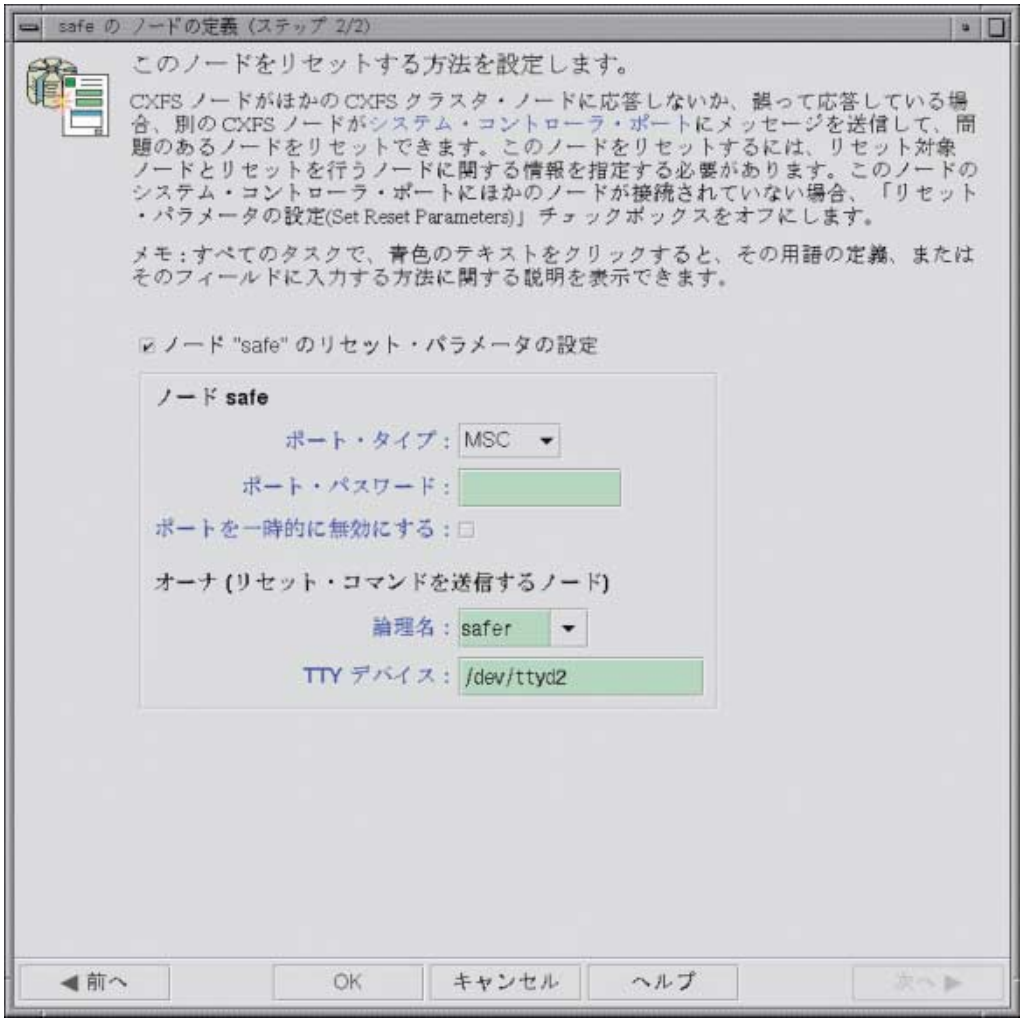

図4-2 システム・コントローラ設定の例

メモ：最初のノードのアイコンがCXFS クラスタ表示で表示されるまで、2 番目のノードは追加しないでください。設定データベースに変更を加えると、その都度クラスタ全体のステータス情報が送信されます。したがって、設定が大きくなるほど送信に時間がかかります。詳細については、40 ページの「CXFS クラスタ表示の起動」を参照してください。

GUI を使ったクラスタ内のノードの追加と削除

プールにノードを追加して、クラスタを定義したら、クラスタ内に含めるノードを指定できます。

メモ: クラスタ・アイコンがCXFS クラスタ表示に表示されるまで、ノードを追加したり削除しないでください。詳細については、40 ページの「CXFS クラスタ表示の起動」を参照してください。

以下を実行します。

1. 目的のノードを追加または削除します。
   - ノードを追加するには、その論理名を「利用可能なノード(Available Nodes)」プルダウン・メニューから選択し、「追加(Add)」をクリックします。ノード名は、「クラスタに追加するノード(Nodes to Go into Cluster)」リストに表示されます。
   - ノードを削除するには、その論理名を「クラスタに追加するノード(Nodes to Go into Cluster)」画面でクリックします(論理名が強調表示されます)。その後、「削除(Delete)」をクリックします。
2. 「OK」をクリックします。

GUI を使ったノードのリセット

GUI を使用してクラスタ内のノードをリセットできます。これにより、指定したノードのシステム・コントローラ・ポートにリセット・コマンドが送信されます。ノードがリセットされると、クラスタ内のほかのノードで変更が検出され、アクティブなクラスタからこのノードが削除されます。ノードは、再起動するとCXFS メンバーシップに再設定されます。

ノードをリセットするには、以下を実行します。

1. 「リセットするノード(Node to Reset)」: リセットするノードをプルダウン・リストから選択します。
2. 「OK」をクリックします。

GUI を使ったノードの変更

ノードを定義したら、そのノードを変更できます。ノードの名前を変更する場合は、そのノードを削除した後、新しいノードを定義しなければなりません。

ノードを変更するには、以下を実行します。

1. 「論理名(Logical Name)」: ノードの論理名をプルダウン・リストから選択します。論理名を選択すると、このノードの情報がさまざまなフィールドに入力されます。
2. 該当するフィールドの情報を、以下のように変更します。
   - 「メンバーシップ重み(Membership Weight)」: CXFS メンバーシップ重みを変更する場合は、テキスト入力領域をクリックし、<Backspace> キーを使用して現在の値を削除します。次に、正しい値を追加します。0 または 1 の値のみを使用してください。詳細については、13 ページの「上限量の計算とノードの重み」を参照してください。

     メモ: CXFS では、ノードが現在クラスタでアクティブな場合、CXFS メンバーシップ重みまたはネットワークの変更は直ちに反映されません。変更は、次回ノードをクラスタに設定する際(つまり、再起動の後、または CXFS サービスを停止して再開した後)に考慮されます。

   - 「受信クラスタ・メッセージ用ネットワーク(Networks for Incoming Cluster Messages)」: GUI では複数のネットワークを入力できますが、CXFS が使用するのは優先度が 1 のネットワークだけで、このネットワークはフェイルオーバーされません。ただし、優先度が 1 のネットワークに異常が発生した場合でもクラスタ・インフラストラクチャがリセット用として使用できる追加のネットワークを定義できます。ネットワークの優先度は、クラスタ内の各ノードに対して同じでなければなりません。
     - 「ネットワーク(Network)」: 受信クラスタ・メッセージ用ネットワークを追加する場合は、IP アドレスまたはホスト名を「ネットワーク」テキスト・フィールドに入力し、「追加(Add)」をクリックします。
     - すでにリストに存在しているネットワークを変更する場合は、リストでそのネットワークをクリックして選択します。その後、「変更(Modify)」をクリックします。これにより、ネットワークは、リストからテキスト入力領域に移動されます。これで、ネットワークを変更できます。ネットワークをリストに追加し直すには、「追加」をクリックします。
     - ネットワークを削除する場合は、優先度リストでそのネットワークをクリックして選択します。その後、「削除(Delete)」をクリックします。
     - ネットワークの優先度を変更する場合は、優先度リストでそのネットワークをクリックして選択します。その後、上下の矢印をクリックして、リスト内でネットワークを別の位置に移動します。

メモ: GUI では「ハートビート(Heartbeat)」または「コントロール(Control)」だけを指定できますが、これらの設定は CXFS では使用できないので選択しないでください。CXFS は、ハートビートとコントロールの両方に対して単一のネットワークを使用します。

ネットワークを一時的に定義して、ネットワークでメッセージを受信しないようにする場合は、「なし(None)」設定を使用できます。詳細については、6 ページの「クラスタ環境」を参照してください。

– 「パーティション ID(Partition ID)」: パーティションが設定された Origin 3000 システムのパーティションを一意に定義します(オプション)。システムにパーティションが設定されていない場合、このフィールドは空のままにしてください。

メモ: この値を決定するには、`mkpart`(1m) コマンドを使用してください。

- `-n` オプションは、パーティション ID をリストします(システムにパーティションが設定されていない場合は 0 になります)。
- `-l` オプションは、さまざまなパーティションのブリックをリストします(GUI では *rack#.slot#* 形式を使用してください)。

  例は、次のとおりです(読みやすくするために出力を切詰めてあります)。

  ```
  # mkpart -n
  Partition id = 1
  # mkpart -l
  partition: 3 = brick: 003c10 003c13 003c16 003c21 003c24 003c29 ...
  partition: 1 = brick: 001c10 001c13 001c16 001c21 001c24 001c29 ...
  ```

  「パーティション ID(Partition ID)」フィールドには、以下のいずれかを入力できます。

  ```
  1
  001.10
  ```

次の画面に移動するには、「次へ(Next)」をクリックします。

- 次のページに移動するには、「次へ」をクリックします。必要に応じて、値を変更します。

3. 「OK」をクリックします。

GUI を使った FailSafe ノードの CXFS への切替え

FailSafe もインストールされている場合、このタスクは CXFS GUI に表示されます。ノードを定義したら、そのノードを変更できます。

既存の IRIS FailSafe ノード(タイプ `FailSafe`)を `CXFS and FailSafe` または `CXFS` のいずれかのタイプに切替えるには、以下を実行します。

1. FailSafe GUI を使用して、切替える HA サービスをノードで停止します。『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。
2. CXFS GUI を使用して、2 番目のハートビートおよびコントロール・ネットワークをノード定義に追加します。91 ページの「GUI を使ったノードの変更」を参照してください。
3. ノードを切替えます。
   - 「論理名(Logical Name)」: ノードの論理名をプルダウン・リストから選択します。
   - 「メンバーシップ重み(Membership Weight)」: CXFS メンバーシップ重みを変更する場合は、テキスト入力領域をクリックし、<Backspace> キーを使用して現在の値を削除します。次に、正しい値を追加します。0 または 1 の値のみを使用してください。詳細については、13 ページの「上限量の計算とノードの重み」を参照してください。
   - 「FailSafe の設定を維持する(Keep FailSafe Settings)」:
     - タイプ `CXFS and FailSafe` に切替えるには、チェックボックスをオンにします。
     - タイプ `CXFS` に切替えるには、チェックボックスをオフのままにします。
   - 「OK」をクリックします。

メモ: ノードの名前を変更する場合は、そのノードを削除した後、新しいノードを定義しなければなりません。

その他のパラメータを変更するには、91 ページの「GUI を使ったノードの変更」を参照してください。変更内容が FailSafe と CXFS の両方に対して適切であることを確認してください。

FailSafe に適用されるよう CXFS ノードを切替えるには、`cmgr`(1m) コマンドまたは FailSafe GUI を使用します。FailSafe GUI についての詳細は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

GUI を使ったノードの削除

プールからノードを削除する前に、クラスタからそのノードを削除してください。詳細については、96 ページの「GUI を使ったクラスタ定義の変更」を参照してください。

ノードを削除するには、以下を実行します。

1. 「削除するノード(Node to Delete)」: 削除するノードの論理名をプルダウン・リストから選択します。
2. 「OK」をクリックします。

GUI を使ったノードの表示

ノードを定義したら、以下を表示できます。

- ノードの属性
- 特定のクラスタのメンバーであるノード
- 定義されているノード

GUI を使用した場合は、クラスタの定義済みのノードとそれらのノードの属性をCXFS クラスタ表示で簡単にグラフィック表示できます。40 ページの「CXFS クラスタ表示の起動」を参照してください。

プール内で定義されているすべてのノードを表示するには、次の「クラスタ表示(Cluster View)」メニュー・アイテムを使用します。

「表示(View)」
>「プール内のノード(Nodes in Pool)」

クラスタに属するノードをすべて表示するには、次のメニュー・アイテムを使用します。

「表示(View)」
>「クラスタ内のノード(Nodes in Cluster)」

ファイルシステムを表示するには、次のメニュー・アイテムを使用します。

「表示(View)」
>「ファイルシステム(Filesystems)」

詳細なステータスおよび設定情報を表示するには、任意の名前またはアイコンをクリックします。

GUI を使ったクラスタ・タスク

この節では、クラスタを定義、変更、削除、および表示するために個々の GUI タスクを実行する方法について説明します。

メモ:「新規クラスタの設定(Set Up a New Cluster)」設定ガイド・タスクセットを使用すると、クラスタおよびノードを設定するために必要なタスクを実行できます。71 ページの「GUI を使った新規クラスタの設定」を参照してください。

GUI を使ったクラスタの定義

クラスタは、プライベート・ネットワークによって互いに連結されたノードの集合です。クラスタは、単純な名前で識別されます。特定のノードは、1 つだけのクラスタのメンバーにできます。

クラスタを定義するには、以下を実行します。

1. 次の情報を入力します。

   - 「クラスタ名(Cluster Name)」: クラスタの論理名。名前は 255 文字以下にしてください。
   - 「クラスタ ID(Cluster ID)」: ネットワーク内で一意な 1～128 の範囲の数字。クラスタ ID は、ネットワーク上に存在する可能性のあるほかのクラスタからクラスタ情報を受付けないようにするために、IRIX カーネルによって使用されます。カーネルは通信にデータベースを使用しないため、クラスタ通信を確認するためにクラスタ ID が必要です。カーネルのこの情報は、いったん初期化したら変更できません。このため、クラスタを定義したらクラスタ ID は変更しないでください。
   - 「クラスタ・モード(Cluster Mode)」: クラスタは、通常、デフォルトの通常(`Normal`)モードに設定します。

     実験中(`Experimental`)にモードを設定すると、ノード異常を発生させることなくクラスタをデバッグできるよう、CXFS の CXFS メンバーシップ・コードでハートビートがオフになります。これは、短時間だけネットワークを切断したい場合(クラスタ・ネットワーク・アクティビティがほかになく、ハートビートがなくても異常が検出されるという条件で)や、CXFS ノードでカーネル・デバッガを入力したい場合などに便利です。実験中(`Experimental`)モードは、デバッグする場合にのみ使用することをお薦めします。

   - 「管理者に通知(Notify Administrator)」:

     – 「メールで(By e-mail)」: これを選択した場合、電子メール・プログラム(デフォルトでは `/usr/sbin/Mail`)と、識別された管理者の電子メール・アドレスを指定する必要があります。複数のアドレスを指定するには、各アドレスをカンマで区切ります。CXFS では、ノードまたはクラスタのステータスが変更されるたびに、電子メールがそれらのアドレスに送信されます。アドレスを指定しなかった場合、通知は送信されません。

- 「コマンドで(By other command)」: これを選択した場合、ノードまたはクラスタのステータスが変更されるたびに実行されるコマンドを指定する必要があります。
- 「なし(Never)」: これを選択した場合、通知が送信されないことが指定されます。

2. 「OK」をクリックします。

GUI を使ったクラスタ定義の変更

クラスタの状態の変更をクラスタ管理者に通知する方法を変更するには、以下を実行します。

1. 「クラスタ名(Cluster Name)」: プルダウン・リストから選択します。
2. 「クラスタ・モード(Cluster Mode)」: クラスタは、通常、デフォルトの通常(`Normal`)モードに設定します。実験中(`Experimental`)モードについての詳細は、95 ページの「GUI を使ったクラスタの定義」を参照してください。
3. 「管理者に通知(Notify Administrator)」: 希望の通知方法を選択します。詳細については、95 ページの「GUI を使ったクラスタの定義」を参照してください。
4. 「OK」をクリックします。

クラスタを構成しているノードを変更するには、90 ページの「GUI を使ったクラスタ内のノードの追加と削除」を参照してください。

メモ: クラスタの名前を変更する場合は、そのクラスタを削除した後、新しいクラスタを定義しなければなりません。

CXFS サービスがノードで開始されている場合は、サービスを再起動するか、クラスタの名前を変更するときにクラスタ ID 番号を流用してください。

GUI を使った FailSafe クラスタの CXFS への切替え

FailSafe もインストールされている場合、このタスクは CXFS GUI に表示されます。

既存の IRIS FailSafe クラスタ(つまり、タイプ `FailSafe`)からの情報を切替えて、CXFS(つまり、タイプ `CXFS and FailSafe` またはタイプ `CXFS`)に適用されるクラスタを作成するには、以下を実行します。

1. 「クラスタ名(Cluster Name)」: プルダウン・リストから選択します。
2. 「クラスタ ID(Cluster ID)」: ネットワーク内で一意な 1〜128 の範囲の数字。クラスタ ID は、ネットワーク上に存在する可能性のあるほかのクラスタからクラスタ情報を受付けないようにするために、IRIX カーネルによって使用されます。カーネルは通信にデータベースを使用しないため、クラスタ

通信を確認するためにクラスタ ID が必要です。カーネルのこの情報は、いったん初期化したら変更できません。このため、クラスタを定義したらクラスタ ID は変更しないでください。

3. 「OK」をクリックします。

クラスタは、IRIS FailSafe と CXFS の両方に適用されるようになります。クラスタを構成しているノードを変更するには、90 ページの「GUI を使ったクラスタ内のノードの追加と削除」を参照してください。

メモ: クラスタの名前を変更する場合は、そのクラスタを削除した後、新しいクラスタを定義しなければなりません。

GUI を使ったクラスタの削除

ノードが含まれているクラスタは削除できません。したがって、最初に、クラスタに含まれているノードをクラスタから移動してください。詳細については、96 ページの「GUI を使ったクラスタ定義の変更」を参照してください。

クラスタを削除するには、以下を実行します。

1. 「削除するクラスタ(Cluster to Delete)」: クラスタ名をプルダウン・リストから選択します。
2. 「OK」をクリックします。

GUI を使ったクラスタの表示

GUI を使用すると、クラスタとそのコンポーネントを CXFS クラスタ表示で簡単に表示できます。40 ページの「CXFS クラスタ表示の起動」を参照してください。

「表示(View)」メニューから、クラスタ内の要素を選択してチェックできます。クラスタの詳細を表示するには、クラスタ名またはアイコンをクリックします。ステータスおよび設定の情報が、新しいウィンドウに表示されます。この情報を同じウィンドウで表示するには、まず、次のメニュー・アイテムを選択してください。

「オプション(Options)」
> 「詳細を表示(Show Details)」

ウィンドウの右側にステータスの詳細が表示されます。これらを非表示にするには、同じメニュー・アイテムを使用して、「詳細を表示(Show Details)」設定を切替えます。また、`cluster_status` コマンドを使用することもできます。165 ページの 第7章「クラスタとノードのステータスのモニタ」を参照してください。

GUI を使った CXFS サービス・タスク

以下のタスクでは、CXFS クラスタ・サービスを開始および停止する方法と、ログを設定する方法について説明します。

GUI を使った CXFS サービスの開始

CXFS サービスを開始し、システムが再起動されるたびに CXFS サービスが自動的に再開されるよう設定するには、以下を実行します。

1. 「アクティブにするノード(Node(s) to Activate)」:「すべてのノード(All Nodes)」、または、CXFS サービスを開始する個々のノードを選択します。
2. 「OK」をクリックします。

GUI を使った CXFS サービス(通常 CXFS シャットダウン)の停止

ノードで CXFS サービスが停止されると、ファイルシステムは、自動的にそのノードからアンマウントされます。

CXFS サービスを一時的に停止する(つまり、再起動するとサービスが再開されるよう設定されている場合に再開できるようにする)には、GUI 外のシェル・ウィンドウで次のコマンド・ラインを使用します。

```
# /etc/init.d/cxfs stop
```

以下の手順を実行することによって、指定したノードまたはクラスタで CXFS を停止し、再起動によって CXFS サービスが再開されないようにすることができます。

メモ:この方法を使用して CXFS サービスを停止した場合、ノードが再起動されてもサービスは再開されません。

---

1. 「停止するノード(Node(s) to Deactivate)」:「すべてのノード(All Nodes)」、または CXFS サービスを停止する個々のノードを選択します。

   ノードで CXFS サービスを停止すると、そのノードはファイルシステムにアクセスできなくなります。そのノードがファイルシステムのメタデータ・サーバとして動作している場合、使用可能なメタデータ・サーバのリストに含まれる別のノードが選択されます。このプロセス中は、ファイルシステムのクライアントで遅れが発生します。

2. 「強制(Force)」: エラーが発生しても(通常はエラーが発生して停止操作を防止します)、CXFS サービスを強制的に停止する場合は、「強制」チェックボックスをオンにします。

3. 「OK」をクリックします。プロセスを完了するには、数分かかります。

ノードで CXFS サービスを停止すると、そのノードは、クラスタのアクティブなメンバーではなくなります。CXFS サービスは、システムが再起動されても再開されません。

注意: CXFS サービスは、`shutdown`(1M) コマンドまたは `reboot`(1M) コマンドを使用する前に停止してください。CXFS サービスがアクティブなときに `shutdown` または `reboot` を実行すると、クラスタのほかのノードによってノード異常と解釈され、そのノードに対して強制的に回復が実行されます。

GUI を使ったタイブレーカー・ノードの設定

タイブレーカー・ノードは、(CXFS メンバーシップ重みが等しい) ノードのちょうど半数が実行中で互いに通信可能である場合に、CXFS メンバーシップ上限量が維持されるかどうかを決定します。タイブレーカー・ノードは、必ず重み付けされている必要があります。デフォルトのタイブレーカー・ノードはありません。詳細については、13 ページの「メンバーシップ上限量」を参照してください。

注意: 重み付けされた 2 ノード・クラスタ内のタイブレーカー・ノードに異常が発生したり、管理者が CXFS サービスを停止した場合は、その他のノードによって強制シャットダウンが実行され、CXFS ファイルシステムはすべてアンマウントされます。

データの整合性を保証するため、SGI では、すべての CXFS システム (特に、重み付けされた 2 ノード・クラスタ) に対してリセット・ハードウェアを使用することをお薦めしています。IRIS FailSafe ではリセットは必須です。

現在のタイブレーカー・ノードは、クラスタ表示の詳細に表示されます。

タイブレーカー・ノードを設定するには、以下を実行してください。

1. 「タイブレーカー・ノード(Tie-Breaker Node)」: 希望のノードをリストから選択します。現在タイブレーカーがある場合は、デフォルトでそのノードが選択されます。

   タイブレーカー・ノードの設定を解除するには、「なし(None)」を選択します。

2. 「OK」をクリックします。

GUI を使ったログの設定

CXFS では、各 CXFS デーモンのログを維持します。CXFS は、通常の処理と重大なエラーの両方を、`/var/adm/SYSLOG` のほかに各ログ・グループの個々のログにも記録します。維持するログのレベルに基づいて、ログをカスタマイズできます。

注意: ログ・ファイルの名前は変更しないでください。名前を変更すると、エラーが発生します。

ログ・グループを定義するときは、次の情報を指定します。

- 「ログ・グループ(Log Group)」: 同じログ設定を使用する 1 つまたは複数の CXFS プロセスのセット。通常、ログ・グループは、`crsd` などの 1 つの CXFS デーモンに対応します。
- 「ログ・レベル(Log Level)」: 関連付けられているログ・グループのログ・ファイルに CXFS が書込むログ・メッセージの量を制御する数字。
- 「ログ・ファイル(Log File)」: メッセージのログを記録するファイル。

165 ページの「ログ・ファイルのモニタ」も参照してください。

GUI を使ったログ・グループ定義の表示

ログ・グループ定義を表示するには、以下を実行します。

1. 「ログ・グループ(Log Group)」: 表示するログ・グループをメニューから選択します。

   そのログ・グループの現在のログ・レベルとログ・ファイルがタスク・ウィンドウに表示されます。必要に応じて、このウィンドウでこれらの設定を変更できます。

2. 「OK」をクリックします。

GUI を使ったログ・グループの設定

ログ・グループを設定するには、「ログの設定(Set Log Configuration)」タスク内の以下を実行します。

1. 適切な情報を入力します。

   - 「ログ・グループ(Log Group)」: ログ・グループをプルダウン・リストから選択します。ログ・グループは、同じログ設定に基づいて同じログ・ファイルにログを記録するプロセスのセットです。各 CXFS デーモンは、ログ・グループを 1 つ作成します。設定は、`cli` ログ・グループと `crsd` ログ・グループについてはプール内のすべてのノードに適用され、`clconfd` ログ・グループと `diags` ログ・グループについてはクラスタ内のすべてのノードに適用されます。
   - 「ログ・レベル(Log Level)」: ログ・レベルを選択します。これは、ログの量を指定します。

注意: 「デフォルト(Default)」ログ・レベルは非常に冗長であるため、これを使用すると、ディスク容量の問題が発生する場合があります。この場合は、低いログ・レベルを選択できます。156 ページの「ログ・ファイル管理」、54 ページの「`/etc/config/cad.options` の設定」、および55 ページの「`/etc/config/fs2d.options` の設定」も参照してください。

値は、以下のとおりです。

- 「オフ(Off)」: ログには記録されません。
- 「最小(Minimal)」: 重大なエラーと通常の操作の通知をログに記録します(これらのメッセージのログは、/var/adm/SYSLOG ファイルにも記録されます)。
- 「情報(Info)」:「最小(Minimal)」通知と警告をログに記録します。
- 「デフォルト(Default)」: すべての「情報(Info)」メッセージとその他の通知をログに記録します。
- 「Debug 0」～「Debug 9」: 数字の大きさに応じて、ログに記録されるデバッグ情報(データ構造を含む)が増えます。

メモ: cmgr(1M) コマンドでは、これらのログ・レベルを示すために、数字のセットを使用します。137 ページの「CLI を使ったログ・グループの設定」を参照してください。

- 「ログ・ファイル(Log File): この値は変更しないでください。

2. 「OK」をクリックします。

GUI を使ったファイルシステム・タスク

以下のタスクでは、ファイルシステムを共有 XVM ボリュームとして設定できます。これらの共有ボリュームには、CXFS クラスタ内のすべてのノードから直接アクセスできます。各ボリュームは、それぞれのデバイス名で識別されます。各ボリュームには、クラスタ内の各ノードと同じマウント・ポイントが必要です。

メモ:「新規ファイルシステムの設定(Set Up a New Filesystem)」設定ガイド・タスクセットを使用すると、新しいファイルシステムを設定するために必要な手順を実行できます。72 ページの「GUI を使った新規ファイルシステムの設定」を参照してください。

GUI を使ったファイルシステムの定義

メモ:このタスクでは、ディスク・ドライブのボリューム・ヘッダ、XVM 論理ボリューム、およびファイルシステムが作成されていることを想定しています。詳細については、68 ページの「GUI を使った設定」を参照してください。

ファイルシステムのメタデータ・サーバを定義するには、以下を実行します。

1. 次の情報を入力します。

- 「XVM ボリューム名(XVM Volume Name)」: これは、CXFS クラスタ内のすべてのノード間で共有される XVM ボリュームのデバイス名です。名前は必ず /dev/cxvm/ で始まるので、この情報はデフォルトで提供されます。
- 「マウント・ポイント(Mount Point)」: この値は、指定した XVM ボリュームが接続されるディレクトリです。このディレクトリ名は、スラッシュ (/) で始まる必要があります。詳細については、mount(1M) のマン・ページを参照してください。
- (オプション)「マウント・オプション(Mount Options)」: これらのオプションは mount(1M) に渡され、指定した XVM ボリュームへのアクセスを制御するために使用されます。使用可能なオプションのリストについては、fstab(4) のマン・ページを参照してください。
- 「メタデータ・サーバ(Metadata Servers)」: メタデータ・サーバとして動作するノードのリスト。サーバのリストにノードを追加するには、その名前をプルダウン・ノード・リストから選択し、「追加(Add)」をクリックします。

  サーバのリストからノードを削除するには、その名前をクリックして選択し、「削除(Delete)」をクリックします。

メモ: サーバの順序は重要です。リストの最初のノードが優先メタデータ・サーバになります。論理名をクリックして選択した後、矢印ボタンをクリックして、使用される順にサーバを並べ替えます。

しかし、ネットワーク・レーテンシやその他の予測不能な遅れがあるため、起動サイクル中にどのサーバが実際にサーバになるかは予想できません。このため、リスト内で最初に使用可能なノードが、メタデータ・サーバとして使用されます。

---

2. 「OK」をクリックします。

GUI を使ったファイルシステムのマウント

既存のファイルシステムをクラスタ内のすべてのノードにマウントするには、以下を実行します。

1. 「マウントするファイルシステム(Filesystem to Mount)」: ファイルシステムをプルダウン・メニューから選択します。
2. 「OK」をクリックします。

メモ: CXFS サービスがアクティブではない場合、ファイルシステムのマウントは完全に成功しません。ファイルシステムはマウント準備済みとしてマークされ、「ファイルシステムのマウント(Mount a Filesystem)」タスクに警告メッセージが表示されます。ファイルシステムは、CXFS サービスを開始するまで、実際にマウントされません。詳細については、98 ページの「GUI を使った CXFS サービスの開始」を参照してください。

GUI を使ったファイルシステムのアンマウント

ファイルシステムをクラスタ内のすべてのノードからアンマウントするには、以下を実行します。

1. 「アンマウントするファイルシステム(Filesystem to Unmount)」: ファイルシステムをプルダウン・メニューから選択します。
2. 「強制(Force)」: `umount -k` オプションを使用して強制的にアンマウントするには、「強制」切替えボタンをクリックします。`-k` オプションは、該当するファイルシステムのファイルまたは現在のディレクトリを開いているプロセスを強制終了した後、アンマウントします。詳細については、`umount`(1M) のマン・ページを参照してください。
3. 「OK」をクリックします。

GUI を使ったファイルシステムの変更

既存のファイルシステムを変更するには、以下を実行します。

1. 「変更するファイルシステム(Filesystem to Modify)」: ファイルシステムをプルダウン・メニューから選択します。これにより、さまざまなフィールドに、選択したファイルシステムの情報が表示されます。
2. 「マウント・ポイント(Mount Point)」および「マウント・オプション(Mount Options)」: 選択したファイルシステムに対して表示されている情報を、必要に応じて変更します。テキストを消去するには、<Backspace> キーを使用してテキストを削除するか、テキストを選択して上書き入力します。
3. (オプション)「マウント・オプション(Mount Options)」: これらのオプションは `mount`(1M) に渡され、指定した XVM ボリュームへのアクセスを制御するために使用されます。使用可能なオプションのリストについては、`fstab`(4) のマン・ページを参照してください。
4. 「メタデータ・サーバ(Metadata Servers)」:
   - サーバのリストからノードを削除するには、その名前をクリックした後、「削除(Delete)」をクリックします。
   - サーバのリストに新しいノードを追加するには、そのノードをプルダウン・ノード・リストから選択し、「追加(Add)」をクリックします。
   - サーバの優先度を再度並べ替えるには、その名前をクリックして選択した後、必要に応じて矢印ボタンをクリックします。
5. 「OK」をクリックします。

GUI を使ったファイルシステムの削除

現在マウントされているファイルシステムは、削除できません。ファイルシステムをアンマウントするには、103 ページの「GUI を使ったファイルシステムのアンマウント」を参照してください。

アンマウントされたファイルシステムを完全に削除するには、以下を実行します。

1. 「削除するファイルシステム(Filesystem to Delete)」: ファイルシステムの名前をプルダウン・リストから選択します。
2. 「OK」をクリックします。

GUI を使った診断タスクとエラー回復

この節では、GUI を使用して接続性をテストする方法を説明します。ネットワークおよびクラスタ内のノードのシリアル接続は、要求された入力値を入力することによってテストできます。クラスタ内のノードをすべて同時にテストしたり、個々のノードを指定してテストできます。

GUI を使った接続性のテスト

接続性をテストするには、「CXFS マネージャ(CXFS Manager)」から以下を実行します。

メモ:「ノードの接続性テスト(Test Node Connectivity)」画面では、ホスト間の `rsh`(1) アクセスが必要です。`/.rhosts` ファイルには、接続性をテストするホストとローカル・ホストを含めなければなりません。

---

1. 適切なラジオ・ボタンをクリックして、ネットワーク別またはシリアル接続別のどちらでテストするかを選択します。
2. テストするノードをプルダウン・リストから選択し、「追加(Add)」をクリックして、テスト・リストにそのノードを追加します。

   テストするノードのリストからノードを削除するには、その論理名をクリックして選択し、「削除(Delete)」をクリックします。
3. テストを開始するには、「テストの開始(Start Tests)」をクリックします。テストを停止するには、「テストの停止(Stop Tests)」をクリックします。
4. 別のテストを実行するには、「出力をクリア(Clear Output)」をクリックしてステータス画面をクリアし、手順 3 から新たに実行します。
5. ウィンドウを閉じるには、「閉じる(Close)」をクリックします。

GUI を使ったローカル・ノードのメンバーシップ取消し

ローカル・ノードの CXFS メンバーシップは、強制 CXFS シャットダウンの実行が必要な場合（151 ページの「クラスタ設定データベースと CXFS のシャットダウン」を参照してください）など、エラーが発生したときにのみ取消すことをお薦めします。

ローカル・ノードの CXFS メンバーシップを取消すには、以下を実行します。

1. ポップアップ・ウィンドウに表示されるローカル・ノードの名前を確認します。
2. 「OK」をクリックします。

このタスクの結果は、クラスタのその他の部分でノード異常と解釈されます。このため、クラスタのその他の部分では、CXFS メンバーシップ上限量が失われたために異常が発生したり、異常ノードのリセットが決定される場合があります。リセットを回避するには、ノード定義を変更して、システム・コントローラ・ステータスを無効にできます。

GUI を使ったローカル・ノードのメンバーシップ許可

強制 CXFS シャットダウンを必要とする問題を解決したら、ローカル・ノード（GUI が接続しているノード）の CXFS メンバーシップを許可する必要があります。これによって、そのノードにクラスタ内のメンバーシップの再適用を許可します。強制 CXFS シャットダウンは、手動またはカーネルによって実行できます。詳細については、151 ページの「クラスタ設定データベースと CXFS のシャットダウン」を参照してください。

以下の状況では、ローカル・ノードの CXFS メンバーシップをアクティブに許可する必要があります。

- 「GUI を使ったローカル・ノードのメンバーシップ取消し」で説明されているように、手動取消しの後
- コンソールまたは /var/adm/SYSLOG のエラー・メッセージによって指示された場合
- カーネルによって取消しが実行された後。この状況は、/var/adm/SYSLOG 内の次のメッセージによって示されます。

```
Membership lost - withdrawing from cluster
```

ローカル・ノードの CXFS メンバーシップを許可するには、以下を実行します。

1. ポップアップ・ウィンドウに表示されるローカル・ノードの名前を確認します。
2. 「OK」をクリックします。

CLI タスク・リファレンス

メモ: CXFS では、各ノードにライセンスがインストールされていなければなりません。ライセンスを正しくインストールせずにソフトウェアをインストールすると、CXFS CLI は使用できません。ライセンスについての詳細は、47 ページの「必要なソフトウェアのインストール」を参照してください。

CXFS クラスタ・マネージャ・コマンド・ライン・インタフェース (CLI: Command Line Interface) は、`cmgr`(1M) コマンドから構成されます。CLI を呼出すには、以下のいずれかを入力します。

```
# /usr/cluster/bin/cmgr
# /usr/cluster/bin/cluster_mgr
```

-p オプションを使用して、ほかのプロンプトを表示することもできます。このオプションは、ファイルシステム定義に必要です。CLI の使用についての一般情報は、43 ページの「CLI の概要」を参照してください。

この章では、以下のタスクを実行するために `cmgr`(1m) CLI を使用する方法について説明します。

- 108 ページの「CLI を使ったデフォルトの設定」
- ノード・タスク
  - 109 ページの「CLI を使ったノードの定義」
  - 113 ページの「CLI を使ったノードの変更」
  - 116 ページの「CLI を使ったノードのリセット」
  - 116 ページの「CLI を使ったノードの CXFS または FailSafe への切替え」
  - 117 ページの「CLI を使ったノードの削除」
  - 120 ページの「CLI を使ったノードの表示」
- クラスタおよびファイルシステム・タスク
  - 121 ページの「CLI を使ったクラスタの定義」
  - 124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」
  - 127 ページの「CLI を使ったクラスタの CXFS または FailSafe への切替え」
  - 127 ページの「CLI を使ったファイルシステムのアンマウント」

  - 128 ページの「CLI を使った定義されたファイルシステムの再マウント」
  - 130 ページの「CLI を使ったファイルシステムの削除」
  - 131 ページの「CLI を使ったクラスタの削除」
  - 133 ページの「CLI を使ったクラスタの表示」
- CXFS サービス・タスク
  - 135 ページの「CLI を使った CXFS サービスの開始」
  - 135 ページの「CLI を使った CXFS サービスの停止」
  - 136 ページの「CLI を使ったタイブレーカー・ノードの設定」
  - 137 ページの「CLI を使ったログの設定」
- 診断タスク
  - 139 ページの「CLI を使ったネットワークの接続性のテスト」
  - 140 ページの「CLI を使ったローカル・ノードのメンバーシップ取消し」
  - 140 ページの「CLI を使ったローカル・ノードのメンバーシップ許可」

この章では、スクリプト内で `cmgr`(1m) コマンドを使用する例も示します。141 ページの「スクリプトの例」を参照してください。

クラスタを設定するために実行しなければならないタスクの概要については、81 ページの 第4章「GUI タスク・リファレンス」を参照してください。

タスクは、特定の階層を使用して実行する必要があります。たとえば、パーティション ID を変更するには、そのノード名をまず識別しなければなりません。

また、`cluster_status` ツールを使用して、curses モードでステータスを表示することもできます。165 ページの 第7章「クラスタとノードのステータスのモニタ」を参照してください。

CLI を使ったデフォルトの設定

デフォルトのクラスタおよびノードを設定して、CLI の現在のセッションの設定プロセスを簡略化できます。名前を明示的に指定しないと、デフォルトが使用されます。デフォルト値を指定するには、以下のコマンドを使用できます。

```
set cluster  clustername
set node hostname
```

現在のデフォルトを表示するには、次を使用します。

```
show set defaults
```

例:

```
cmgr> set cluster TEST
cmgr> set node cxfs6
cmgr> show set defaults
Default cluster set to: TEST

Default node set to: cxfs6
Default cdb set to: /var/cluster/cdb/cdb.db
Default resource_type is not set
Extra prompting is set off
```

CLI を使ったノード・タスク

この節では、CLI を使用してノードを定義、変更、削除、表示、およびリセットする方法について説明します。

メモ:設定データベースを変更するたびに、クラスタのステータス情報全体が送信されるため、設定が大きくなるほど、時間がかかるようになります。

CLI を使ったノードの定義

ノードを定義するには、以下のコマンドを使用します。

```
define node hostname
    set hostname to hostname
    set nodeid to ID
    set partition_id to partitionID
    set reset_type to reset_type(powerCycle)
    set sysctrl_type to model_type
    set sysctrl_password to password
    set sysctrl_status to enabled|disabled
    set sysctrl_owner to node_sending_reset_command
    set sysctrl_device to /dev/ttyd2
    set sysctrl_owner_type to tty_device
    set is_failsafe to true|false
```

```
set is_cxfs to true|false
set weight to 0|1
add nic IP_address_or_hostname_(if_DNS)
        set heartbeat to true|false
        set ctrl_msgs to true|false
        set priority to integer
remove nic IP_address_or_hostname_(if_DNS)
```

使用法に関する注記

- ネットワーク・インタフェースは 8 つまで指定できますが、優先後が最も高いネットワークのみが使用されます（フェイルオーバーはありません）。このネットワークはプライベートにする必要があります。8 ページの「プライベート・ネットワーク」を参照してください。
- ノードの重みは 0 または 1 にします。デフォルトは 1 です。少なくとも 1 つのノードの CXFS メンバーシップ重みに、0 より大きい値を設定する必要があります。
- ノード ID は、プール内のノード全体で一意な、1～32767 の範囲の整数です。数字を指定しないと、CXFS によって自動的に ID が計算されます。デフォルトの ID は、マシンのシリアル・ナンバーなどのマシン固有の情報を基にした 5 桁の数字で、連続する数字ではありません。ノードを定義した後にノード ID 番号を変更しないでください。
- パーティション ID は、パーティションが設定された Origin 3000 システムのパーティションを一意に定義します。

メモ: この値を決定するには、mkpart(1m) コマンドを使用してください。

– -n オプションは、パーティション ID をリストします(システムにパーティションが設定されていない場合は 0 になります)。

– -l オプションは、さまざまなパーティションのブリックをリストします(CLI では *rack#.slot#* 形式を使用してください)。

例は、次のとおりです(読みやすくするために出力を切詰めてあります):

```
# mkpart -n
Partition id = 1
# mkpart -l
partition: 3 = brick: 003c10 003c13 003c16 003c21 003c24 003c29 ...
partition: 1 = brick: 001c10 001c13 001c16 001c21 001c24 001c29 ...
```

「パーティション ID(Partition ID)」フィールドには、以下のいずれかを入力できます。

```
1
001.10
```

システムにパーティションが設定されていない場合、このフィールドは空のままにしてください。

パーティション ID の設定を解除するには、0 または none という値を使用します。

- ネットワークの優先度は、クラスタ内の各ノードに対して同じでなければなりません。
- プール内の最初のノードを定義する場合は、*hostname* としてそのノードの名前を /etc/sys_id で使用する必要があります。

詳細については、13 ページの「上限量の計算とノードの重み」および84 ページの「GUI を使ったノードの定義」を参照してください。

デフォルト情報を使用するには、プロンプト・モードで <Enter> キーを押します。例では、<Enter> キーは示されていません。一般情報については、84 ページの「GUI を使ったノードの定義」を参照してください。

```
cmgr> define node hostname
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ? hostname
Is this a FailSafe node <true|false> ? true|false
Is this a CXFS node <true|false> ? true
Node ID ? [optional] ID_number
```

```
Partition ID ? [optional] (0)ID_number
Do you wish to define system controller info[y/n]:y|n
Reset type <powerCycle> ? (powerCycle)
Do you wish to define system controller info[y/n]:y|n
Sysctrl Type <chalL|msc|mmsc>? (msc) model_type
Sysctrl Password[optional] ? ( )password
Sysctrl Status <enabled|disabled> ? enabled|disabled
Sysctrl Owner ? node_sending_reset_command
Sysctrl Device ? /dev/ttyd2
Sysctrl Owner Type <tty> ? (tty) tty_device
Number of Network Interfaces ? (1) 1|2|3|4|5|6|7|8
NIC 1 - IP Address ? IP_address_or_hostname_(if_DNS)
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ? true|false
NIC 1 - (use network for control messages) <true|false> ? true|false
NIC 1 - Priority <1,2,...> ? priority_number
Node Weight ? (1) 0|1
```

プロンプト表示のない例:

```
cxfs6 # /usr/cluster/bin/cmgr
Welcome to SGI Cluster Manager Command-Line Interface

cmgr> define node cxfs6
Enter commands, when finished enter either "done" or "cancel"

cxfs6 ? set is_failsafe to false
cxfs6 ? set is_cxfs to true
cxfs6 ? set weight to 1
cxfs6 ? add nic cxfs6
Enter network interface commands, when finished enter "done" or "cancel"

NIC - 1 ? set heartbeat to true
NIC - 1 ? set ctrl_msgs to true
NIC - 1 ? set priority to 1
NIC - 1 ? done
cxfs6 ? done
```

プロンプト表示のある例:

```
cxfs6 # /usr/cluster/bin/cmgr -p
Welcome to SGI Cluster Manager Command-Line Interface

cmgr> define node cxfs6
```

```
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ?
Is this a FailSafe node <true|false> ? false
Is this a CXFS node <true|false> ? true
Node ID[optional]?
Partition ID ? [optional] (0)
Reset type <powerCycle>  ? (powerCycle)
Do you wish to define system controller info[y/n]:n
Number of Network Interfaces ? (1)
NIC 1 - IP Address ? cxfs6
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ? true
NIC 1 - (use network for control messages) <true|false> ? true
NIC 1 - Priority <1,2,...> ? 1
Node Weight ? (1)

Successfully defined node cxfs6
```

CLI を使ったノードの変更

既存のノードを変更するには、以下のコマンドを入力します。

```
modify node hostname
    set hostname to hostname
    set nodeid to nodeID
    set partition_id to partitionID
    set reset_type to reset_type
    set sysctrl_type to chalL|msc|mmsc
    set sysctrl_password to password
    set sysctrl_status to enabled|disabled
    set sysctrl_owner to node_sending_reset_command
    set sysctrl_device to /dev/ttyd2
    set sysctrl_owner_type to tty_device
    set is_failsafe to true|false
    set is_cxfs to true|false
    set weight to 0|1
    add nic IP_address_or_hostname_(if_DNS)
            set heartbeat to true|false
            set ctrl_msgs to true|false
            set priority to integer
    remove nic IP_address_or_hostname_(if_DNS)
```

ノード定義時に指定した情報は、ノード ID を除いてすべて変更可能です。109 ページの「CLI を使ったノードの定義」を参照してください。

注意: ノードを定義した後にノード ID 番号を変更しないでください。

CLI を使ったノードの重みの変更例

0 というノードの重みを変更するには、以下のコマンドを入力します。現在の設定を維持するには、<Enter> キーを押します（例では、<Enter> キーは示されていません）。

```
cmgr> modify node cxfs6
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ? (cxfs6.americas.sgi.com)
Is this a FailSafe node <true|false> ? (false)
Is this a CXFS node <true|false> ? (true)
Node ID[optional] ? (13203)
Partition ID[optional] ? (0)
Reset type <powercycle> ? (powerCycle)
Do you wish to modify system controller info[y/n]:n
Number of Network Interfaces ? (1)
NIC 1 - IP Address ? (cxfs6)
NIC 1 - Heartbeat HB (use network for heartbeats)   <true|false> ? (true)
NIC 1 - (use network for control messages) <true|false> ? (true)
NIC 1 - Priority <1,2,...> ? (1)
Node Weight ? (1) 0

Successfully modified node cxfs6
```

以下に結果を示します。

```
cmgr> show node cxfs6
Logical Machine Name: cxfs6
Hostname: cxfs6.americas.sgi.com
Node Is FailSafe: false
Node Is CXFS: true
Nodeid: 13203
Reset type: powerCycle
ControlNet Ipaddr: cxfs6
ControlNet HB: true
ControlNet Control: true
ControlNet Priority: 1
```

```
Node Weight: 0
```

パーティションの設定例

以下に、Origin 3000 システムのパーティション設定の例を示します。

```
# cmgr
Welcome to SGI Cluster Manager Command-Line Interface

cmgr> modify node n_preston
Enter commands, when finished enter either "done" or "cancel"

n_preston ? set partition_id to 1
n_preston ? done

Successfully modified node n_preston
```

プロンプト表示ありでこの機能を実行するには、以下を入力します。

```
# cmgr -p
Welcome to SGI Cluster Manager Command-Line Interface

cmgr> modify node n_preston
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ? (preston.engr.sgi.com)
Is this a FailSafe node <true|false> ? (true)
Is this a CXFS node <true|false> ? (true)
Node ID[optional] ? (606)
Partition ID[optional] ? (0) 1
Reset type <powerCycle> ? (powerCycle)
Do you wish to modify system controller info[y/n]:n
Number of Network Interfaces ? (2)
NIC 1 - IP Address ? (preston)
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ? (true)
NIC 1 - (use network for control messages) <true|false> ? (true)
NIC 1 - Priority <1,2,...> ? (1)
NIC 2 - IP Address ? (192.168.168.1)
NIC 2 - Heartbeat HB (use network for heartbeats) <true|false> ? (true)
NIC 2 - (use network for control messages) <true|false> ? (true)
NIC 2 - Priority <1,2,...> ? (2)
Node Weight ? (1)
```

```
Successfully modified node n_preston

cmgr> show node n_preston
Logical Machine Name: n_preston
Hostname: preston.engr.sgi.com
Node Is FailSafe: true
Node Is CXFS: true
Nodeid: 606
Partition id: 1
Reset type: powerCycle
ControlNet Ipaddr: preston
ControlNet HB: true
ControlNet Control: true
ControlNet Priority: 1
ControlNet Ipaddr: 192.168.168.1
ControlNet HB: true
ControlNet Control: true
ControlNet Priority: 2
Node Weight: 1
```

パーティション ID の設定を解除するには、0 または none という値を使用します。

CLI を使ったノードのリセット

CXFS が実行されている場合、次のコマンドを使用してノードをリセットできます。

`admin reset node` *`hostname`*

このコマンドは、CXFS デーモンを使用して、指定したノードをリセットします。

CXFS デーモンが実行されていない場合でも、 `admin reset` コマンドの `standalone` オプションを使用すると、クラスタ内のノードをリセットできます。

`admin reset standalone node` *`hostname`*

このコマンドは、CXFS デーモンを経由しません。

CLI を使ったノードの CXFS または FailSafe への切替え

既存の FailSafe ノードを CXFS にも適用するよう切替えるには、`modify` コマンドを使用して設定を変更します。

メモ: 対応する高可用性 (HA) サービスまたは CXFS サービスがアクティブな場合は、ノードの FailSafe または CXFS をオフにできません。まず、ノードのサービスを停止してください。

例:

```
cmgr> modify node cxfs6
Enter commands, when finished enter either "done" or "cancel"

cxfs6 ? set is_FailSafe to true
cxfs6 ? done

Successfully modified node cxfs6
```

プロンプト表示のある例:

```
cmgr> modify node cxfs6
Enter commands, you may enter "done" or "cancel" at any time to exit

Hostname[optional] ? (cxfs6.americas.sgi.com)
Is this a FailSafe node <true|false> ? (false) true
Is this a CXFS node <true|false> ? (true)
Node ID[optional] ? (13203)
Partition ID[optional] ? (0)
Reset type <powerCycle> ? (powerCycle)
Do you wish to modify system controller info[y/n]:n
Number of Network Interfaces ? (1)
NIC 1 - IP Address ? (cxfs6)
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ? (true)
NIC 1 - (use network for control messages) <true|false> ? (true)
NIC 1 - Priority <1,2,...> ? (1)
Node Weight ? (0)

Successfully modified node cxfs6
```

CLI を使ったノードの削除

ノードを削除するには、次のコマンドを使用します。

```
delete node hostname
```

ノードを削除できるのは、現在そのノードがクラスタの一部でない場合だけです。クラスタに現在ノードが含まれている場合は、そのクラスタをまず変更して、ノードを削除しなければなりません。

たとえば、次のように設定されている TEST というクラスタがあるとします。

```
cmgr> show cluster TEST
Cluster Name: TEST
Cluster Is FailSafe: true
Cluster Is CXFS: true
Cluster ID: 20
Cluster HA mode: normal
Cluster CX mode: normal
FileSystem Device Name: /dev/cxvm/dks2d72s0
FileSystem Mount Point: /mnts/fs1
FileSystem Mount Options:
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs6
        FileSystem Server Rank: 0
        FileSystem Server Node: cxfs7
        FileSystem Server Rank: 1
FileSystem Device Name: /dev/cxvm/dks2d73s0
FileSystem Mount Point: /mnts/fs2
FileSystem Mount Options: enabled
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs8
        FileSystem Server Rank: 0


Cluster TEST has following 3 machine(s)
        cxfs6
        cxfs7
        cxfs8
```

cxfs8 ノードを削除するには、プロンプト・モードで以下を実行します（CXFS サービスがそのノードで停止されていると想定しています）。

```
cmgr> modify cluster TEST
Enter commands, when finished enter either "done" or "cancel"

Is this a FailSafe cluster <true|false> ? (false)
Is this a CXFS cluster <true|false> ? (true)
```

```
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster CXFS mode <normal|experimental>[optional] ? (normal)
Cluster ID ? (20)
Number of Cluster FileSystems ? (0)

Current nodes in cluster TEST:
Node - 1: cxfs6
Node - 2: cxfs7
Node - 3: cxfs8

Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST ? remove node cxfs8
TEST ? done
Successfully modified cluster TEST

cmgr> show cluster TEST
Cluster Name: TEST
Cluster Is FailSafe: false
Cluster Is CXFS: true
Cluster ID: 20
Cluster CX mode: normal


Cluster TEST has following 2 machine(s)
        cxfs6
        cxfs7
```

プールから cxfs8 を削除するには、次を入力します。

```
cmgr> delete node cxfs8

IMPORTANT: NODE cannot be deleted if it is a member of a cluster.
The LOCAL node can not be deleted if some other nodes are still defined.


Deleted machine (cxfs6).
```

CLI を使ったノードの表示

ノードを定義したら、次のコマンドを使用して、そのノードのパラメータを表示できます。

```
show node hostname
```

例は、次のとおりです。

```
cmgr> show node cxfs6
Logical Machine Name: cxfs6
Hostname: cxfs6.americas.sgi.com
Node Is FailSafe: false
Node Is CXFS: true
Nodeid: 13203
Reset type: powerCycle
ControlNet Ipaddr: cxfs6
ControlNet HB: true
ControlNet Control: true
ControlNet Priority: 1
Node Weight: 1
```

定義されているすべてのノードのリストは、次のコマンドを使用して表示できます。

```
show nodes in pool
```

例:

```
cmgr> show nodes in pool

3 Machine(s) defined
        cxfs8
        cxfs6
        cxfs7
```

指定したクラスタに対して定義されているすべてのノードのリストは、次のコマンドを使用して表示できます。

```
show nodes[in cluster clustername]
```

例:

```
cmgr> show nodes in cluster TEST



Cluster TEST has following 3 machine(s)
```

```
        cxfs6
        cxfs7
        cxfs8
```

デフォルトのクラスタが指定されている場合は、このコマンドを使用する際にクラスタを指定する必要はありません。例は、次のとおりです。

```
cmgr> set cluster TEST
cmgr> show nodes

Cluster TEST has following 3 machine(s)
        cxfs6
        cxfs7
        cxfs8
```

CLI を使ったクラスタ・タスクとファイルシステム・タスク

この節では、CLI を使用してクラスタを定義、変更、削除、および表示する方法について説明します。また、CXFS サービスを開始および停止する方法についても説明します。

CLI を使ったクラスタの定義

CLI を使用してクラスタを定義するときは、同じコマンドを使用して、クラスタの定義およびクラスタへのノードの追加を行います。一般情報については、95 ページの「GUI を使ったクラスタの定義」を参照してください。

クラスタを定義するには、以下のコマンドを使用します。

```
define cluster clustername
    set is_failsafe to true|false
    set is_cxfs to true|false
    set clusterid to ID
    set notify_cmd to command
    set notify_addr to address
    set ha_mode to normal|experimental
    set cx_mode to normal|experimental
    add node node1name
    add node node2name
    ...
```

メモ: set ha_mode コマンドおよび set cx_mode コマンドは、常に normal に設定します。set cx_mode コマンドは CXFS にのみ適用し、set ha_mode コマンドは IRIS FailSafe にのみ適用します。

以下に、プロンプト表示ありのコマンドを示します。

```
cmgr> define cluster clustername
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> true|false ?
Is this a CXFS cluster  <true|false> true|false ?
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster CXFS mode <normal|experimental>[optional]use_default_of_normal
Cluster ID ? ID_number
Number of Cluster FileSystems ? (0) should_be_set_to_0_when_defining_a_cluster

No nodes in cluster clustername

Add nodes to or remove nodes from cluster clustername
Enter "done" when completed or "cancel" to abort

clustername ? add node node1name
clustername ? add node node2name
...
clustername ? done
Successfully defined cluster clustername
```

メモ: 通常、ファイルシステムは、クラスタを変更するとき（ボリューム・ヘッダの作成、XVM 論理ボリュームの作成、およびファイルシステムのマウントの手順を実行した後）に定義します。ファイルシステムを定義するには、プロンプト・モードを使用しなければなりません。タスクのこの順序についての詳細は、73 ページの「CLI を使った設定」を参照してください。

クラスタは、デフォルトの normal モードに設定します。experimental にモードを設定すると、ノード異常を発生させることなくクラスタをデバッグできるよう、CXFS メンバーシップ・コードでハートビートがオフになります。これは、短時間だけネットワークを切断したい場合（クラスタ・ネットワーク・アクティビティがほかになく、ハートビートがなくても異常が検出されるという条件で）に便利です。ただし、実作業用のクラスタでは experimental モードを使用しないでください。このモードは、日本 SGI のカスタマ・サポートから指示があった場合にのみ使用してください。SGI では、カスタマが experimental を使用することを推奨していません。

例:

```
cmgr> define cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> false ?
Is this a CXFS cluster  <true|false> true ?
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster CXFS mode <normal|experimental>[optional]
Cluster ID ? 20
Number of Cluster FileSystems ? (0)

No nodes in cluster TEST

Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST ? add node cxfs6
TEST ? add node cxfs7
TEST ? add node cxfs8
TEST ? done
Successfully defined cluster TEST
```

クラスタ ID は、ネットワーク内で一意な 1〜128 の範囲の数字です。クラスタ ID は、ネットワーク上に存在する可能性のあるほかのクラスタからクラスタ情報を受付けないようにするために、IRIX カーネルによって使用されます。カーネルは通信にデータベースを使用しないため、クラスタ通信を確認するためにクラスタ ID が必要です。カーネルのこの情報は、いったん初期化したら変更できません。このため、クラスタを定義したらクラスタ ID は変更しないでください。

プロンプト表示なしでこれを行うには、以下を入力します。

```
cmgr> define cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

cluster TEST? set is_cxfs to true
cluster TEST? set clusterid to 20
cluster TEST? add node cxfs6
cluster TEST? add node cxfs7
cluster TEST? add node cxfs8
cluster TEST? done
Successfully defined cluster TEST
```

CLI を使ったクラスタの変更およびファイルシステムの定義とマウント

クラスタを定義したら、その属性を変更できます。

注意: ファイルシステムをすでに定義している場合は、ファイルシステム・プロンプトの数に 0 を入力しないでください。これを行った場合、ファイルシステムに関する現在の情報がすべて失われます。

- 新しいファイルシステムを追加するには、現在定義されている数よりも 1 つ多いファイルシステム数を入力します(つまり、*n*+1)。以前のファイルシステムが表示されている場合にそのデフォルトを維持するには、<Enter> キーを押します。新しいファイルシステムが表示されている場合は、正しい情報を入力します。
- ファイルシステムを完全に削除するには、1 つ少ないファイルシステム数をプロンプトで入力します(つまり、*n*-1)。リスト内の最後のファイルシステムを削除すると、ファイルシステムが削除されます。ただし、最後ではないファイルシステムを削除する場合は、情報を再入力しなければなりません。
- ファイルシステムをアンマウントするには、127 ページの「CLI を使ったファイルシステムのアンマウント」を参照してください。
- ファイルシステムを再マウントするには、128 ページの「CLI を使った定義されたファイルシステムの再マウント」を参照してください。

コマンドは、以下のとおりです。

```
modify cluster clustername
    set is_failsafe to true|false
    set is_cxfs to true|false
    set clusterid to ID
    set notify_cnd to command
    set notify_addr to address
    set ha_mode to normal|experimental
    set cx_mode to normal|experimental
    add node node1name
    add node node2name
    ...
    remove node node1name
    remove node node2name...
```

以下に、2 つのファイルシステムを追加するためのクラスタ変更プロセスの例を示します。fs1 ファイルシステムに対して使用可能なメタデータ・サーバには、cxfs6 と cxfs7 の 2 つのタイプがあります。クラスタが開始されたときに cxfs6(ランクが 0 の優先サーバ)が実行中ではない場合は、cxfs7(ランク 1)が使用されます。

メモ: 特定のファイルシステムのメタデータ・サーバのリストは順序付きですが、ネットワーク・レーテンシやその他の予測不能な遅れがあるため、起動サイクル中にどのノードが実際にサーバになるかは予想できません。

以下を実行します。デフォルト設定を使用するには、<Enter> キーを押します。

```
cmgr> modify cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> ? (false)
Is this a CXFS cluster <true|false> ? (true)
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster mode <normal|experimental>[optional] ? (normal)
Cluster ID ? (20)
Number of Cluster FileSystems ? (0) 2
FILESYSTEM 0 - Device Name ? /dev/cxvm/dks2d72s0
FILESYSTEM 0 - Mount Point ? /mnts/fs1
FILESYSTEM 0 - Mount Options[optional] ?
FILESYSTEM 0 - Status <enabled|disabled> ? enabled
FILESYSTEM 0 - Forced unmount ? <true|false> ? false
Number of Servers ? (0) 2
FILESYSTEM 0 SERVER 0 - Server Node ? cxfs6
FILESYSTEM 0 SERVER 0 - Server Rank ? 0
FILESYSTEM 0 SERVER 1 - Server Node ? cxfs7
FILESYSTEM 0 SERVER 1 - Server Rank ? 1
FILESYSTEM 1 - Device Name ? /dev/cxvm/dks2d73s0
FILESYSTEM 1 - Mount Point ? /mnts/fs2
FILESYSTEM 1 - Mount Options[optional] ?
FILESYSTEM 1 - Status <enabled|disabled> ? enabled
FILESYSTEM 1 - Forced unmount ? <true|false> ? false
Number of Servers ? (0) 1
FILESYSTEM 1 SERVER 0 - Server Node ? cxfs8
FILESYSTEM 1 SERVER 0 - Server Rank ? 0

Current nodes in cluster TEST:
Node - 1: cxfs6
Node - 2: cxfs7
Node - 3: cxfs8
```

```
Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST ? done
Successfully modified cluster TEST
```

クラスタ TEST の変更内容を表示するには、次を入力します。

```
cmgr> show cluster TEST
Cluster Name: TEST
Cluster Is FailSafe: false
Cluster Is CXFS: true
Cluster ID: 20
Cluster CX mode: normal
FileSystem Device Name: /dev/cxvm/dks2d72s0
FileSystem Mount Point: /mnts/fs1
FileSystem Mount Options:
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs7
        FileSystem Server Rank: 1
        FileSystem Server Node: cxfs6
        FileSystem Server Rank: 0
FileSystem Device Name: /dev/cxvm/dks2d73s0
FileSystem Mount Point: /mnts/fs2
FileSystem Mount Options: enabled
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs8
        FileSystem Server Rank: 0


Cluster TEST has following 3 machine(s)
        cxfs6
        cxfs7
        cxfs8
```

ファイルシステムを定義、変更、マウント、およびアンマウントするには、プロンプト・モード (cmgr -p) を使用してください。

メモ: CXFS サービスがアクティブではない場合、ファイルシステムのマウントは完全に成功しません。ファイルシステムはマウント準備済みとしてマークされますが、CXFS サービスを開始するまでは実際にマウントされません。詳細については、135 ページの「CLI を使った CXFS サービスの開始」を参照してください。

CLI を使ったクラスタの CXFS または FailSafe への切替え

クラスタを切替えるには、modify コマンドに続けて以下のコマンドを使用します。

```
set is_failsafe to true|false
set is_cxfs to true|false
```

たとえば、FailSafe にも適用されるよう CXFS クラスタ TEST を切替えるには、次のコマンドを入力します。

```
cmgr> modify cluster TEST
Enter commands, when finished enter either "done" or "cancel"

TEST ?set is_failsafe to true
```

クラスタは、そのノードに関してオンにするすべての機能(FailSafe または CXFS、あるいはその両方)をサポートしていなければなりません。つまり、クラスタがタイプ CXFS の場合は、クラスタの一部であるノードをタイプ FailSafe または CXFS and FailSafe に変更することはできません。ただし、ノードがクラスタのすべての機能をサポートする必要はありません。つまり、タイプ CXFS and FailSafe のクラスタでタイプ CXFS のノードを使用できます。

CLI を使ったファイルシステムのアンマウント

ファイルシステム定義を削除することなくファイルシステムをアンマウントする(ファイルシステムを再マウントできるようにする)には、そのステータスを disabled に変更します。これは、プロンプト・モード (cmgr -p) で行う必要があります。

以下の例では、fs2 ファイルシステムをアンマウントします。デフォルト設定を使用する場合は <Enter> キーを押し、必要な引数を入力したらすぐに done と入力します。

```
cmgr> modify cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> ? (true)
Is this a CXFS cluster <true|false> ? (true)
Cluster Notify Cmd [optional] ?
```

```
Cluster Notify Address [optional] ?
Cluster HA mode [optional] ? (normal)
Cluster CXFS mode [optional] ? (normal)
Cluster ID ? (20)
Number of Cluster FileSystems ? (2)
FILESYSTEM 0 - Device Name ? (/dev/cxvm/dks2d72s0)
FILESYSTEM 0 - Mount Point ? (/mnts/fs1)
FILESYSTEM 0 - Mount Options[optional] ?
FILESYSTEM 0 - Status  ? (enabled)
FILESYSTEM 0 - Forced unmount ? <true|false> ? (false)
Number of Servers ? (2)
FILESYSTEM 0 SERVER 0 - Server Node ? (cxfs6)
FILESYSTEM 0 SERVER 0 - Server Rank ? (0)
FILESYSTEM 0 SERVER 1 - Server Node ? (cxfs7)
FILESYSTEM 0 SERVER 1 - Server Rank ? (1)
FILESYSTEM 1 - Device Name ? (/dev/cxvm/dks2d73s0)
FILESYSTEM 1 - Mount Point ? (/mnts/fs2)
FILESYSTEM 1 - Mount Options[optional] ? (enabled)
FILESYSTEM 1 - Status  ? (enabled) disabled
FILESYSTEM 1 - Forced unmount ? <true|false> ? (false)
Number of Servers ? (1)
FILESYSTEM 1 SERVER 0 - Server Node ? (cxfs8)
FILESYSTEM 1 SERVER 0 - Server Rank ? (0)

Current nodes in cluster TEST:
Node - 1: cxfs6
Node - 2: cxfs7
Node - 3: cxfs8

Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST ? done
Successfully modified cluster TEST
```

CLI を使った定義されたファイルシステムの再マウント

すでに定義されているファイルシステムを再マウントするには、そのステータスを enabled に変更します。以下の例では、fs2 ファイルシステムを再マウントします。デフォルト設定を使用する場合は <Enter>

キーを押し、必要な引数を入力したらすぐに done と入力します。プロンプト・モード (cmgr -p) を使用してください。

```
cmgr> modify cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> ? (true)
Is this a CXFS cluster <true|false> ? (true)
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster HA mode [optional] ? (normal)
Cluster CXFS mode [optional] ? (normal)
Cluster ID ? (20)
Number of Cluster FileSystems ? (2)
FILESYSTEM 0 - Device Name ? (/dev/cxvm/dks2d72s0)
FILESYSTEM 0 - Mount Point ? (/mnts/fs1)
FILESYSTEM 0 - Mount Options[optional] ?
FILESYSTEM 0 - Status  ? (enabled)
FILESYSTEM 0 - Forced unmount ? <true|false> ? (false)
Number of Servers ? (2)
FILESYSTEM 0 SERVER 0 - Server Node ? (cxfs7)
FILESYSTEM 0 SERVER 0 - Server Rank ? (1)
FILESYSTEM 0 SERVER 1 - Server Node ? (cxfs6)
FILESYSTEM 0 SERVER 1 - Server Rank ? (0)
FILESYSTEM 1 - Device Name ? (/dev/cxvm/dks2d73s0)
FILESYSTEM 1 - Mount Point ? (/mnts/fs2)
FILESYSTEM 1 - Mount Options[optional] ? (enabled)
FILESYSTEM 1 - Status  ? (disabled) enabled
FILESYSTEM 1 - Forced unmount ? <true|false> ? (false)
Number of Servers ? (1)
FILESYSTEM 1 SERVER 0 - Server Node ? (cxfs8)
FILESYSTEM 1 SERVER 0 - Server Rank ? (0)

Current nodes in cluster TEST:
Node - 1: cxfs6
Node - 2: cxfs7
Node - 3: cxfs8

Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort
```

CLI を使ったファイルシステムの削除

順序付けの番号が最も大きいファイルシステムを完全に削除するには、そのファイルシステムをまず無効にした後、1 つのファイルシステムだけを指定します。ファイルシステムを変更するには、プロンプト・モード (cmgr -p) を使用しなければなりません。例は、次のとおりです。

```
cmgr> modify cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> ? (true)
Is this a CXFS cluster <true|false> ? (true)
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster HA mode [optional] ? (normal)
Cluster CXFS mode [optional] ? (normal)
Cluster ID ? (20)
Number of Cluster FileSystems ? (2)
FILESYSTEM 0 - Device Name ? (/dev/cxvm/dks2d72s0)
FILESYSTEM 0 - Mount Point ? (/mnts/fs1)
FILESYSTEM 0 - Mount Options[optional] ?
FILESYSTEM 0 - Status  ? (enabled)
FILESYSTEM 0 - Forced unmount ? <true|false> ? (false)
Number of Servers ? (2)
FILESYSTEM 0 SERVER 0 - Server Node ? (cxfs6)
FILESYSTEM 0 SERVER 0 - Server Rank ? (0)
FILESYSTEM 0 SERVER 1 - Server Node ? (cxfs7)
FILESYSTEM 0 SERVER 1 - Server Rank ? (1)
FILESYSTEM 1 - Device Name ? (/dev/cxvm/dks2d73s0)
FILESYSTEM 1 - Mount Point ? (/mnts/fs2)
FILESYSTEM 1 - Mount Options[optional] ?
FILESYSTEM 1 - Status <enabled|disabled> ? (enabled) disabled
FILESYSTEM 1 - Forced unmount ? <true|false> ? (false)
Number of Servers ? (1)
FILESYSTEM 1 SERVER 0 - Server Node ? (cxfs8)
FILESYSTEM 1 SERVER 0 - Server Rank ? (0)

Current nodes in cluster TEST:
Node - 1: cxfs6
Node - 2: cxfs7
Node - 3: cxfs8

Add nodes to or remove nodes from cluster TEST
```

```
Enter "done" when completed or "cancel" to abort

TEST ? done
Successfully modified cluster TEST

cmgr> modify cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Is this a FailSafe cluster <true|false> ? (true)
Is this a CXFS cluster <true|false> ? (true)
Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster HA mode [optional] ? (normal)
Cluster CXFS mode [optional] ? (normal)
Cluster ID ? (20)
Number of Cluster FileSystems ? (2) 1
FILESYSTEM 0 - Device Name ? (/dev/cxvm/dks2d72s0)
FILESYSTEM 0 - Mount Point ? (/mnts/fs1)
FILESYSTEM 0 - Mount Options[optional] ?
FILESYSTEM 0 - Status  ? (enabled)
FILESYSTEM 0 - Forced unmount ? <true|false> ? (false)
Number of Servers ? (2)
FILESYSTEM 0 SERVER 0 - Server Node ? (cxfs6)
FILESYSTEM 0 SERVER 0 - Server Rank ? (0)
FILESYSTEM 0 SERVER 1 - Server Node ? (cxfs7)
FILESYSTEM 0 SERVER 1 - Server Rank ? (1)

Current nodes in cluster TEST:
Node - 1: cxfs6
Node - 2: cxfs7
Node - 3: cxfs8

Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST? done
Successfully modified cluster TEST
```

CLI を使ったクラスタの削除

クラスタを削除するには、次のコマンドを使用します。

```
delete cluster clustername
```

ただし、ノードが含まれているクラスタは削除できません。したがって、ノードで CXFS サービスをまず停止した後、クラスタを再定義して、ノードが含まれないようにする必要があります。

例:

プロンプト表示のない例:

```
cmgr> modify cluster TEST
Enter commands, when finished enter either "done" or "cancel"

TEST ? remove node cxfs6
TEST ? remove node cxfs7
TEST ? remove node cxfs8
TEST ? done
Successfully modified cluster TEST

cmgr> delete cluster TEST

cmgr> show clusters

cmgr>
```

プロンプトの使用:

```
cmgr> modify cluster TEST
Enter commands, you may enter "done" or "cancel" at any time to exit

Cluster Notify Cmd [optional] ?
Cluster Notify Address [optional] ?
Cluster mode <normal|experimental>[optional] ? (normal)
Cluster ID ? (55)
Number of Cluster FileSystems ? (0) 0

Current nodes in cluster TEST:
Node - 1: cxfs6
Node - 2: cxfs7
Node - 3: cxfs8

Add nodes to or remove nodes from cluster TEST
Enter "done" when completed or "cancel" to abort

TEST ? remove node cxfs6
```

```
TEST ? remove node cxfs7
TEST ? remove node cxfs8
TEST ? done
Successfully modified cluster TEST

cmgr> delete cluster TEST

cmgr> show clusters

cmgr>
```

CLI を使ったクラスタの表示

クラスタとその内容を表示するには、以下のコマンドを使用します。

```
show clusters
show cluster clustername
```

例:

```
cmgr> show clusters

1 Cluster(s) defined
        TEST


cmgr> show cluster TEST
Cluster Name: TEST
Cluster Is FailSafe: true
Cluster Is CXFS: true
Cluster ID: 20
Cluster HA mode: normal
Cluster CX mode: normal
FileSystem Device Name: /dev/cxvm/dks2d72s0
FileSystem Mount Point: /mnts/fs1
FileSystem Mount Options:
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs6
        FileSystem Server Rank: 0
        FileSystem Server Node: cxfs7
        FileSystem Server Rank: 1
FileSystem Device Name: /dev/cxvm/dks2d73s0
FileSystem Mount Point: /mnts/fs2
FileSystem Mount Options: enabled
FileSystem Status: enabled
FileSystem Force flag: false
        FileSystem Server Node: cxfs8
        FileSystem Server Rank: 0
```

CLI を使った CXFS サービス・タスク

以下のタスクでは、CXFS サービスを開始および停止する方法と、ログ・レベルを設定する方法について説明します。

CLI を使った CXFS サービスの開始

CXFS サービスを開始し、システムが再起動されるたびに CXFS サービスが自動的に再開されるよう設定するには、以下のいずれかのコマンドを使用します。

```
start cx_services [on node hostname ] for cluster clustername

start cx_services for cluster clustername
```

例:

```
cmgr> start cx_services for cluster TEST
```

CLI を使った CXFS サービスの停止

ノードで CXFS サービスが停止されると、ファイルシステムは、自動的にそのノードからアンマウントされます。

CXFS サービスを一時的に停止する(つまり、再起動するとサービスが再開されるよう設定されている場合に再開できるようにする)には、CLI 外のシェル・ウィンドウで次のコマンド・ラインを使用します。

```
# /etc/init.d/CXFS stop
```

指定したノードまたはクラスタで CXFS サービスを停止し、再起動によって CXFS サービスが再開されないようにするには、次のコマンドを使用します。

メモ:この方法を使用して CXFS サービスを停止した場合、ノードが再起動されてもサービスは再開されません。

```
stop cx_services [on node hostname]for cluster clustername[force]
```

例:

```
cmgr> stop cx_services on node cxfs6 for cluster TEST

CXFS services have been deactivated on node cxfs6 (cluster TEST)
```

```
cmgr> stop cx_services for cluster TEST
```

ノードで CXFS サービスを停止すると、そのノードは、クラスタのアクティブなメンバーではなくなります。

注意: CXFS サービスを停止すると、ノードは「非アクティブ(INACTIVE)」としてマークされ、再起動後にクラスタに再設定されなくなります。ノードがクラスタに再設定されるようにするには、CLI または GUI を使用して CXFS サービスを再開しなければなりません。

CLI を使ったタイブレーカー・ノードの設定

タイブレーカー・ノードは、(重みが等しい)ノードのちょうど半数が互いに通信可能である場合に、CXFS メンバーシップ上限量が維持されるかどうかを決定します。タイブレーカー・ノードは、必ず重み付けされている必要があります。デフォルトのタイブレーカー・ノードはありません。

注意: 重み付けされた 2 ノード・クラスタ内のタイブレーカー・ノードに異常が発生したり、管理者が CXFS サービスを停止した場合は、その他のノードによって強制シャットダウンが実行され、CXFS ファイルシステムはすべてアンマウントされます。データの整合性を保証するため、SGI では、すべての CXFS システム(特に、重み付けされた 2 ノード・クラスタ)に対してハードウェア・リセットを使用することをお薦めしています。IRIS FailSafe ではリセットは必須です。

タイブレーカー・ノードを設定するには、以下のように modify コマンドを使用します。

```
modify cx_parameters in cluster clustername
set tie_breaker to hostname
```

タイブレーカー・ノードの設定を解除するには、次のコマンドを使用します。

```
set tie_breaker to ""
```

例:

```
cmgr> modify cx_parameters in cluster TEST

(Enter "cancel" at any time to abort)

Tie Breaker Node ? (cxfs7) cxfs8
Successfully modified cx_parameters

cmgr> show cx_parameters in cluster TEST

_CX_TIE_BREAKER=cxfs8
```

例:

```
cmgr> modify cx_parameters in cluster TEST
Enter commands, when finished enter either "done" or "cancel"

TEST ? set tie_breaker to cxfs8
TEST ? done
Successfully modified cx_parameters
```

CLI を使ったログの設定

CXFS ログについての一般情報は、99 ページの「GUI を使ったログの設定」を参照してください。

CLI を使ったログ・グループ定義の表示

ログ・グループ定義を表示するには、次のコマンドを使用します。

```
show log_groups
```

このコマンドを実行すると、現在定義されているすべてのログ・グループと共に、ログ・グループ名、ログ・レベル、およびログ・ファイルが表示されます。

CLI を使ったログ・グループの設定

ログ・グループは、次のコマンドを使用して設定できます。

```
define log_group log_group on node hostname [in cluster clustername]
```

log_group 変数は、以下のいずれかになります。

```
clconfd
cli
crsd
diags
```

注意: ログ・ファイルの名前は変更しないでください。名前を変更すると、エラーが発生します。

log_level 変数には、以下のいずれかの値が含まれます。

- 0: ログには記録されません。
- 1: 重大なエラーと通常の操作の通知をログに記録します(これらのメッセージのログは、SYSLOG ファイルにも記録されます)。

- 2:「最小(Minimal)」通知と警告をログに記録します。
- 5〜7: 数字の大きさに応じて、ログに記録される詳細通知が増えます。
- 10〜19: 数字の大きさに応じて、ログに記録されるデバッグ情報(データ構造を含む)が増えます。

ログ・レベル 5 を使用してログ・グループ cli をノード cxfs6 で定義する場合:

```
cmgr> define log_group cli on node cxfs6 in cluster TEST

(Enter "cancel" at any time to abort)

Log Level ? (11) 5

CREATE LOG FILE OPTIONS

        1) Add Log File.
        2) Remove Log File.
        3) Show Current Log Files.
        4) Cancel. (Aborts command)
        5) Done. (Exits and runs command)

Enter option:5
Successfully defined log group cli
```

CLI を使ったログ・グループの変更

ログ・グループを変更するには、次のコマンドを使用します。

```
modify log_group log_group_name on node hostname [in cluster clustername]
```

ログ・グループの変更は、定義に使用するコマンドと同じコマンドを使用して行います。

たとえば、10 になるようログ・レベル cli を変更するには、以下を入力します。

```
cmgr> modify log_group cli on node cxfs6 in cluster TEST

(Enter "cancel" at any time to abort)

Log Level ? (2) 10

MODIFY LOG FILE OPTIONS

        1) Add Log File.
```

```
        2) Remove Log File.
        3) Show Current Log Files.
        4) Cancel. (Aborts command)
        5) Done. (Exits and runs command)

Enter option:5
Successfully modified log group cli
```

CLI を使った診断とエラー・タスク

この節では、接続性をテストする方法を示します。

CLI を使ったネットワークの接続性のテスト

クラスタでのネットワークの接続性をテストするには、CLI を使用できます。このテストでは、指定したノードが、ノードで設定されている各インタフェースを通じて互いに通信できるかどうかをチェックします。CXFS が実行されている場合、このテストは行われません。

クラスタ内のノードに対するネットワークの接続性をテストするには、次のコマンドを使用します。

```
test connectivity in cluster clustername [on node hostname1 node hostname2 ...]
```

例:

```
cmgr> test connectivity in cluster TEST on node cxfs7
Status: Testing connectivity...
Status: Checking that the control IP_addresses are on the same networks
Status: Pinging address cxfs7 interface ef0 from node cxfs7 [cxfs7]
Notice: overall exit status:success, tests failed:0, total tests executed:1
```

このテストでは、最初のエラーが発生するとエラー・メッセージが表示され、応答しなかったノードが示されます。このテストを実行した後でエラー・メッセージを受取った場合は、ifconfig コマンドを使用して、ネットワーク・インタフェースが設定されていることを確認します。例は、次のとおりです(改行して読みやすくしてあります)。

```
# /usr/etc/ifconfig ef0
ef0: flags=405c43 <UP,BROADCAST,RUNNING,FILTMULTI,MULTICAST,CKSUM,DRVRLOCK,IPALIAS>
inet 128.162.89.39 netmask 0xffff0000 broadcast 128.162.255.255
```

出力の最初の行にある UP は、インタフェースが設定されていることを示します。

ネットワーク・インタフェースが設定されている場合は、ネットワーク・ケーブルが正しく接続されていることを確認し、テストを再び実行します。

CLI を使ったシリアル接続のテスト

63 ページの「シリアル接続のテスト」を参照してください。

CLI を使ったローカル・ノードのメンバーシップ取消し

強制 CXFS シャットダウンの前などにローカル・ノードの CXFS メンバーシップを取消すには、そのローカル・ノードで次を入力します。

```
admin cxfs_stop
```

このコマンドは、クラスタのその他の部分でノード異常と解釈されます。このため、クラスタのその他の部分では、CXFS メンバーシップ上限量が失われたために異常が発生したり、異常ノードのリセットが決定される場合があります。リセットを回避するには、ノード定義を変更して、システム・コントローラ・ステータスを無効にできます。

CLI を使ったローカル・ノードのメンバーシップ許可

ローカル・ノードの CXFS メンバーシップを許可すると、そのノードに CXFS メンバーシップの再適用を許可します。以下の状況では、ローカル・ノードの CXFS メンバーシップをアクティブに許可する必要があります。

- 「CLI を使ったローカル・ノードのメンバーシップ取消し」で説明されているように、手動取消しの後
- コンソールまたは /var/adm/SYSLOG のエラー・メッセージによって指示された場合
- カーネルによって取消しが実行された後。この状況は、/var/adm/SYSLOG 内の次のメッセージによって示されます。

  ```
  Membership lost - withdrawing from cluster
  ```

ローカル・ノードの CXFS メンバーシップを許可するには、次のコマンドを使用します。

```
admin cxfs_start
```

151 ページの「クラスタ設定データベースと CXFS のシャットダウン」も参照してください。

スクリプトの例

次のスクリプトによって、タイプ CXFS and FailSafe の 3 ノード・クラスタが定義されます。ノードのタイプは、CXFS です。

メモ:この例では、CXFS で必要な 1 つのネットワーク・インタフェースのみを定義します。ここでは、簡潔にするためにホスト名を使用していますが、52 ページの「ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定」の規則に従わなければなりません。また、混乱を避けるために IP アドレスを代わりに使用することもできます。

この例では、システム・コントローラ、FailSafe リソース、およびフェイルオーバー・ポリシー定義は取上げていません。これらのトピックについての詳細は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

```
!/usr/cluster/bin/cmgr -if
#
#Script to define a three-node cluster


define node cxfs6
        set hostname to cxfs6
        set is_cxfs to true
        set weight to 1
        add nic cxfs6
                set heartbeat to true
                set ctrl_msgs to true
                set priority to 1
        done
done

define node cxfs7
        set hostname to cxfs7
        set is_cxfs to true
        set weight to 1
        add nic cxfs7
                set heartbeat to true
                set ctrl_msgs to true
                set priority to 1
        done
done
```

```
define node cxfs8
        set hostname to cxfs8
        set is_cxfs to true
        set weight to 1
        add nic cxfs8
                set heartbeat to true
                set ctrl_msgs to true
                set priority to 1
        done
done


define cluster TEST
        set is_cxfs to true
        set is_failsafe to true
        set clusterid to 20
        add node cxfs6
        add node cxfs7
        add node cxfs8
done
quit
```

このスクリプトを実行すると、次のような出力が表示されます。

```
Successfully defined node cxfs6


Successfully defined node cxfs7


Successfully defined node cxfs8

Successfully defined cluster TEST
```

管理とメインテナンス

オフライン管理タスクは、クラスタ内のすべてのノードから実行できます。ただし、ファイルシステムがマウントされている場合は、サーバから管理を実行しなければなりません。

CXFS と XFS において、以下は同じです。

- ディスクの概念
- ファイルシステムの概念
- ユーザ・インタフェース
- ファイルシステムの作成

これらのトピックについての詳細は、『IRIX Admin: Disks and Filesystems』を参照してください。

この章の残りの部分では、以下のトピックについて説明します。

- 144 ページの「マウント操作前後のスクリプトの実行」
- 146 ページの「`fsr`(1m) の使用」
- 146 ページの「`find`(1) と `crontab`(1) の使用」
- 147 ページの「階層ストレージ管理 (HSM: Hierarchical Storage Management) システムの使用」
- 148 ページの「ファイルシステムのメタデータ・サーバの発見」
- 151 ページの「メタデータ・サーバの回復」
- 151 ページの「クラスタ設定データベースと CXFS のシャットダウン」
- 156 ページの「再起動時のクラスタ設置データベースの再起動の回避」
- 158 ページの「ボリューム管理」
- 158 ページの「ディスク管理」
- 160 ページの「ファイルシステムのメインテナンス」
- 162 ページの「ダンプと復元」
- 162 ページの「クラスタ設定データベースのバックアップと復元」
- 162 ページの「`chkconfig` フラグ」

マウント操作前後のスクリプトの実行

clconfd デーモンは、以下のファイルが /var/cluster/clconfd-scripts ディレクトリに存在しているかどうかをチェックし、存在する場合は、指定した CXFS ファイルシステムのマウントまたはアンマウントの前後にこれらのファイルを実行します。

- cxfs-pre-mount
- cxfs-post-mount
- cxfs-pre-umount
- cxfs-post-umount

このような名前のスクリプトを作成し、それらのスクリプトを使用して、CXFS ファイルシステムが正常にマウントされた場合のファイルシステムの NFS エクスポートなど、必要なアクションに適用できます。実行可能ファイルは、スクリプトまたはバイナリの場合があります。ファイルの名前は、上記とまったく同じように指定しなければならず、root 実行アクセス権が必要です。

メモ: /etc/exports ファイルは、NFS クライアントにエクスポートされているファイルシステムを記述します。CXFS マウント・ポイントが exports ファイルに含まれている場合、cxfs-post-mount スクリプトを使用して CXFS マウント後にファイルシステムが再エクスポートされないかぎり、空のマウント・ポイントがエクスポートされます。

ファイルには以下の引数が渡されます。

- cxfs-pre-mount: ファイルシステム・デバイス名
- cxfs-post-mount: ファイルシステム・デバイス名および終了コード
- cxfs-pre-umount: ファイルシステム・デバイス名
- cxfs-post-umount: ファイルシステム・デバイス名および終了コード

ファイルシステム名はスクリプトに渡されるため、異なるファイルシステムに対して異なるアクションが取られるようスクリプトを作成できます。また、終了コードは post に渡されるので、操作の結果（成功または失敗）に基づいて異なるアクションを取るようスクリプトを作成することが可能です。

clconfd デーモンは、これらのスクリプトの終了コードをチェックします。異常が発生した場合は（ゼロ以外）、以下が行われます。

- cxfs-pre-mount および cxfs-pre-unmount の場合、対応するマウントまたはアンマウント操作は実行されません。

- cxfs-post-mount および cxfs-post-umount の場合、clconfd によって操作全体（-pre- スクリプトを含みます）が再試行されます。

これは、ファイルシステムをホストにマウントしない場合、そのファイルシステムに対して cxfs-pre-mount スクリプトは異常を返しますが、cxfs-post-mount スクリプトは成功を返すことを意味しています。

以下は、スクリプト例のセットです。

例: cxfs-pre-mount

```
#!/bin/sh
#/var/cluster/clconfd-scripts/cxfs-pre-mount

echo "Preparing to mount CXFS file system \"$1\"" >> /dev/console
MNTPNT=`mount | grep $1 | cut -f 3 -d" "`
if [ -n "${MNTPNT}" ] ; then
    /usr/etc/exportfs -u $MNTPNT
fi
```

例: cxfs-post-mount

```
#!/bin/sh
#/var/cluster/clconfd-scripts/cxfs-post-mount

if [ $2 -eq 0 ] ; then
    echo "Successfully mounted CXFS file system \"$1\"" >> /dev/console
    /usr/etc/exportfs -a
else
    echo "FAILED to mount CXFS file system \"$1\" (error=$2)" >> /dev/console
fi
exit $2
```

例: cxfs-pre-umount

```
#!/bin/sh
#/var/cluster/clconfd-scripts/cxfs-pre-umount

echo "Preparing to unmount CXFS file system \"$1\"" >> /dev/console
MNTPNT=`mount | grep $1 | cut -f 3 -d" "`
if [ -n "${MNTPNT}" ] ; then
```

```
    /usr/etc/exportfs -u $MNTPNT
fi
```

例: cxfs-post-umount

```
#!/bin/sh
#/var/cluster/clconfd-scripts/cxfs-post-umount

if [ $2 -eq 0 ] ; then
    echo "Successfully unmounted CXFS file system \"$1\"" >> /dev/console
else
    echo "FAILED to unmount CXFS file system \"$1\" (error=$2)" >> /dev/console
    /usr/etc/exportfs -a
fi
exit $2
```

fsr(1m) の使用

fsr(1m) コマンドは、操作対象のファイルシステムのメタデータ・サーバでのみ使用できます。bulkstat システム・コールは、CXFS クライアントに対しては無効になっています。fsr は、ファイルシステムの現在のメタデータ・サーバだけに対して手動で使用してください。

find(1) と crontab(1) の使用

find(1) コマンドは、crontab(1) ファイル内に含めないでください。ファイルシステムは各ノードに対してローカルであると認識されるので、各ノードで検索が行われます。したがって、find を使用すると、システムの速度が低下し、一時的にメタデータ・サーバのメモリが大量に消費されます。

階層ストレージ管理 (HSM: Hierarchical Storage Management) システムの使用

CXFS では、XSDM (X/Open Data Storage Management Specification) とも呼ばれる DMAPI (Data Management Application Programming Interface) を通して、階層ストレージ管理 (HSM: Hierarchical Storage Management) 製品の使用をサポートします。HSM 製品の例は、DMF (Data Migration Facility) です。

HSM アプリケーションでは、すべての DMAPI インタフェース・コールをメタデータ・サーバを通して行う必要があります。CXFS マウントされたファイルシステムとの DMAPI インタフェースは、CXFS クライアント・ノードでは提供されません。CXFS クライアントは、HSM アプリケーションとのすべての通信をメタデータ・サーバを通してルーティングします。この場合、一般的には、HSM アプリケーションが CXFS メタデータ・サーバで実行されている必要があります。

メモ: 現在の CXFS リリースは、DMF などの HSM で使用される DMAPI ファイルシステムの再設定をサポートしていません。この機能は、今後のリリースで提供されます。

HSM を CXFS で使用するには、以下を実行します。

- クラスタ内のすべてのノードで IRIX 6.5.11f 以降が実行されていることを確認します。クラスタ・ノードでは、同バージョンまたは直前のバージョンの IRIX オペレーティング・システム (OS: Operating System) を実行できます (6.5.12f の場合は 6.5.11f および 6.5.12f など)。
- クラスタ内の各ノードに、`eoe.sw.dmi` サブシステムをインストールします。
- 管理するファイルシステムをマウントするときに、`dmi` オプションを使用します。この手順についての詳細は、101 ページの「GUI を使ったファイルシステムの定義」または124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」を参照してください。

  メモ: クラスタ内のすべてのノードでは、IRIX 6.5.11f 以降が実行されていなければなりません。メタデータ・サーバの再設定および回復は、以前のリリースでは無効です。

- 管理する各ファイルシステムに対して、メタデータ・サーバで HSM アプリケーションを開始します。

ファイルシステムのメタデータ・サーバの発見

メタデータ・サーバは、CLI または GUI を使用して発見できます。

cluster_status を使用したメタデータ・サーバの発見

cluster_status コマンドを使用して、メタデータ・サーバを発見できます。例は、次のとおりです。

```
# /var/cluster/cmgr-scripts/cluster_status

+ Cluster=cxfs6-8  FailSafe=Not Configured CXFS=ACTIVE              15:15:33
   Nodes =   cxfs6    cxfs7    cxfs8
FailSafe =
    CXFS =      UP       UP       UP

CXFS              DevName           MountPoint           MetaServer     Status
        /dev/cxvm/concat0             /concat0                cxfs7         UP
```

詳細については、166 ページの「cluster_status を使ったシステムのステータスのモニタ」を参照してください。

clconf_info を使用したメタデータ・サーバの発見

clconf_info コマンドを使用して、特定のファイルシステムの現在のメタデータ・サーバを発見できます。たとえば、以下では、cxfs7 がメタデータ・サーバであることが示されています。

```
cxfs6 # clconf_info
Membership since Thu Mar  1 08:15:39 2001
Node          NodeId     Status    Age   Incarnation     CellId
cxfs6              6         UP      0             0          2
cxfs7              7         UP      0             0          1
cxfs8              8         UP      0             0          0
1 CXFS FileSystems
/dev/cxvm/concat0 on /concat0  enabled  server=(cxfs7)  2 client(s)=(cxfs8,cxfs6)
```

GUI を使用したメタデータ・サーバの発見

以下を実行します。

1. CXFS クラスタ表示を開きます。このタスクについての詳細は、40 ページの「CXFS クラスタ表示の起動」を参照してください。
2. メニュー・バーから以下のアイテムを選択します。

   「表示(View)」
   >「ファイルシステム(Filesystems)」

   「オプション(Options)」
   >「詳細を表示(Show Details)」
3. 表示するファイルシステムの名前をクリックします。メタデータ・サーバの名前が表示されます。

図6-1 に、`sleepy` が `/dev/cxvm/babbage_1` XVM ボリュームのメタデータ・サーバである例を示します。

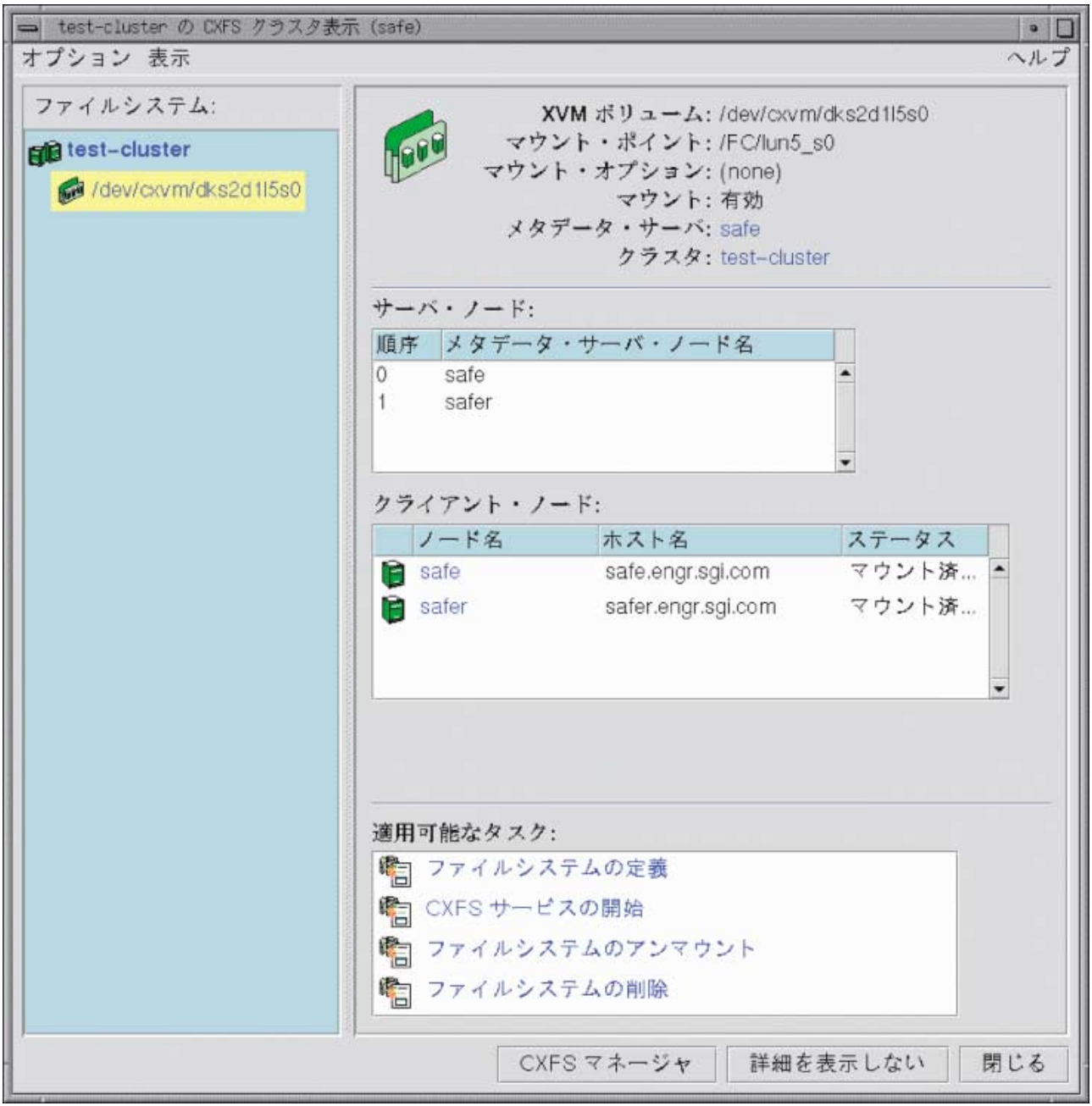

図6-1 メタデータ・サーバが表示されているウィンドウ

メタデータ・サーバの回復

ファイルシステムのメタデータ・サーバとして動作しているノードがダウンした場合は、使用可能なメタデータ・サーバのリストに含まれる別のノードが新しいメタデータ・サーバとして選択されます。これは、ファイルシステムを定義する際に、使用可能なサーバが少なくとも 2 つはリストに含まれていることが前提となります。詳細については、101 ページの「GUI を使ったファイルシステムの定義」または124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」を参照してください。選択されるサーバはファイルシステム・クライアントでなければなりません。その他のファイルシステム・クライアントでは、再設定プロセス中に遅れが発生します。各ファイルシステムの回復に要する時間は、有効な i ノードの数に依存します。全体の遅延時間は、各ファイルシステムの回復に要する時間の合計です。回復の際にファイルシステムがどの程度アクティブであるかに応じて、遅れの合計はファイルシステムあたり最高 7 分になる場合があります。

クライアントがダウンした場合、メタデータ・サーバは、そのクライアントをクリーンアップします。このプロセス中は、その他のクライアントで遅れが発生する場合があります。ダウンしたクライアントがトークンを保持していた場合、遅れはそのトークンによって異なります。クライアントがトークンを持っていなければ遅れはありません。別のクライアントの処理を進める上で取消さなければならないトークンをクライアントが保持している場合は、回復によって異常ノード・トークンが返されるまで、その他のクライアントは中断されます（たとえば、クライアントが書込みトークンを持っていて、別のクライアントが読取りを行いたい場合など）。遅れの実際の長さは、以下によって異なります。

- CXFS メンバーシップの状況
- サーバがダウンしているかどうか
- このファイルシステムの回復に関連する回復順序でのサーバの位置

ダウンしたクライアントは、すべてのメタデータ・サーバによってそのクライアントがクリーンアップされるまで、CXFS メンバーシップに再設定できません。

クラスタ設定データベースと CXFS のシャットダウン

この節では、以下の実行方法について説明します。

- 「クラスタ設定データベースのシャットダウン」
- 「通常 CXFS シャットダウン」
- 「強制 CXFS シャットダウン」

問題がある場合は、173 ページの 第8章「トラブルシューティング」を参照してください。状態についての詳細は、165 ページの 第7章「クラスタとノードのステータスのモニタ」を参照してください。

クラスタ設定データベースのシャットダウン

クラスタ設定データベースのシャットダウンを行うと、クラスタ設定データベースを管理する以下のユーザ空間デーモンが終了されます。

- cad
- clconfd
- cmond
- crsd
- fs2d

ノードでデータベースをシャットダウンしても、共有ファイルシステムへのアクセスは引続き使用可能でノードはクラスタのメンバーのままですが、データベースの更新は取得できません。マシンを再起動すると、すべてのサービスが再開されます。

クラスタ設定データベースのシャットダウンを実行するには、次を入力します。

```
# /etc/init.d/cluster stop
```

また、起動時にデーモンが再開されるのを無効にする場合は、次を入力します。

```
# /etc/chkconfig cluster off
```

ノードのステータスとクラスタ設定データベースのシャットダウン

クラスタ設定データベースのシャットダウンは、ノードでメインテナンス操作を実行した後にそのノードを再起動し、「アクティブ(ACTIVE)」ステータスに戻す場合に行うのが適切です。

クラスタ設定データベースのシャットダウンを実行した場合、ノードのステータスは DOWN になり、以下のような影響があります。

- DOWN ノードは引続きクラスタの一部と見なされますが、使用可能ではなくなります。
- DOWN ノードは、クラスタ設定データベースの更新を取得しなくなります。ただし、ノードを再起動すると、すべての更新が通知されます。

  設定データベースが更新されないと、CXFS のカーネル部分がアクティブな場合に問題が発生することがあります。つまり、ノードが CXFS へのアクセスを持ち続けると、そのノードのカーネル・レベルでは更新を参照できなくなり、残りのノードがこれらのカーネル・レベルの更新を伝えても応答しなくなります。このため、クラスタは設定の更新に基づいて動作できなくなります。

ノードがメタデータ・サーバの場合

ノードが CXFS ファイルシステムのメタデータ・サーバの場合、ノードを停止したままにするには、シャットダウンを完了する前にメタデータ・サーバを別のノードに移動します。

新しいメタデータ・サーバにファイルシステムを移動するには、以下を実行します。

1. メタデータ・サーバで `fuser`(1m) コマンドを実行し、ローカル・ユーザがいないことを確認します。
2. `umount`(1m) コマンドを使用して、メタデータ・サーバだけから CXFS ファイルシステムをアンマウントします(この場合以外は、`umount` を使用して CXFS ファイルシステムを手動でアンマウントしないでください)。

   ```
   # umount filesystem
   ```

   詳細については、103 ページの「GUI を使ったファイルシステムのアンマウント」または127 ページの「CLI を使ったファイルシステムのアンマウント」と、161 ページの「ファイルシステムのアンマウント」を参照してください。

ファイルシステムに定義されたメタデータ・サーバがほかにも存在すると想定して、再設定プロセスが自動的に開始されます。詳細については、101 ページの「GUI を使ったファイルシステムの定義」または124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」を参照してください。選択されるサーバはファイルシステム・クライアントでなければなりません。その他のファイルシステム・クライアントでは、再設定プロセス中に遅れが発生します。

クラスタ設定データベースの再起動

クラスタ設定データベースを再起動するには、次を入力します。

```
# /etc/init.d/cluster start
```

通常 CXFS シャットダウン

通常 CXFS シャットダウンは、ノードの CXFS サービスをすべて停止して、CXFS メンバーシップ上限量から削除する場合に実行します。通常 CXFS シャットダウンにより、以下が行われます。

- すべてのファイルシステムがアンマウントされる
- クラスタ内のこのノードの CXFS メンバーシップが終了される
- 「非アクティブ(`INACTIVE`)」としてノードにマークが付けられる

この結果、クラスタ・ディスクが使用可能になり、クラスタ設定データベースの更新はこのノードに伝えられなくなります。ノードを再起動しても、シャットダウン状態のままです。

通常 CXFS シャットダウンを実行するには、次の cmgr(1m) コマンドを入力します。

```
cmgr> stop cx_services on node nodename for cluster clustername
```

また、GUI を使用することもできます。98 ページの「GUI を使った CXFS サービス（通常 CXFS シャットダウン）の停止」を参照してください。

メモ: このアクションにより、1 つのノードの CXFS サービスが停止され、そのノードの停止後に新しい CXFS メンバーシップが形成されます。複数のノードでサービスを停止する場合は、このコマンドを複数回入力するか、GUI を使用してタスクを実行してください。

ノードのステータスと通常 CXFS シャットダウン

ノードで通常 CXFS シャットダウンを実行すると、その状態は「非アクティブ(INACTIVE)」になります。したがって、CXFS メンバーシップ上限量の計算に影響はありません。153 ページの「通常 CXFS シャットダウン」を参照してください。

通常 CXFS シャットダウンを実行できない場合

以下の状況では、通常 CXFS シャットダウンを実行できません。

- ローカル・ノード上。このノードは、クラスタ・マネージャが実行されているノード、または GUI が接続されているノードです。
- ノード上で CXFS サービスを停止することによって、CXFS メンバーシップ上限量が失われた場合。

これらの状況において CXFS シャットダウンを実行する場合は、強制 CXFS シャットダウンを実行しなければなりません。154 ページの「強制 CXFS シャットダウン」を参照してください。

通常 CXFS シャットダウン後のクラスタへの再設定

ノードは、再起動後にクラスタに再設定されません。ノードは、GUI（98 ページの「GUI を使った CXFS サービスの開始」を参照してください）または次のコマンドを使用して、CXFS サービスを明示的に再度アクティブにした場合のみ、クラスタに再設定されます。

```
cmgr> start cx_services on node nodename for cluster clustername
```

強制 CXFS シャットダウン

強制 CXFS シャットダウンは、クラスタが CXFS メンバーシップ上限量の条件を下回ることになってもノードをシャットダウンする場合に行うのが適切です。154 ページの「強制 CXFS シャットダウン」を参照してください。

注意: 強制 CXFS シャットダウンを実行して、クラスタが CXFS メンバーシップ上限量を下回った場合、クラスタに異常が発生する可能性があります。

メタデータ・サーバで強制シャットダウンを行うと、メンバーシップが直ちに失われます。この時点でセカンダリ・メタデータ・サーバによって引継ぎ(およびファイルシステムの回復)が行われなければなりません。さもないと、上限量が失われ、続いてすべてのノードで強制シャットダウンが行われます。

上限量を強制的に失わせる強制 CXFS シャットダウンを行った場合、クラスタの残りの部分は、同様にただちに強制シャットダウンを行わなければならないため、異常ノードをリセットしません。

強制 CXFS シャットダウンを実行するには、次の cmgr(1m) コマンドを入力します。

```
cmgr> admin cxfs_stop
```

このアクションは、CXFS メンバーシップ上限量が失われた後、カーネルによって自動的に実行することもできます。

ノードのステータスと強制 CXFS シャットダウン

強制 CXFS シャットダウンの後も、ノードは引続き設定されたクラスタの一部と見なされており、クラスタ設定データベースを伝えるとき(これらのサービスはまだ実行されています)や、fs2d メンバーシップ上限量を計算するとき(これにより、クラスタのその他の部分の上限量が失われ、その他のノードで強制シャットダウンが行われます)に考慮に入れられます。状態は、「非アクティブ(INACTIVE)」です。

データベースの整合性を保つには、このノードがアクセス可能なままで、クラスタ・インフラストラクチャ・デーモンを実行し続けることが重要です。特に、プール内の半数を超えるノードが停止していたり、インフラストラクチャ・デーモンを実行していない場合は、クラスタ設定データベースの更新が伝えられらなくなり、結果として整合性が失われます。安全性の点から、今後も使用不可能なノードはクラスタとプールから削除してください。以下を参照してください。

- 90 ページの「GUI を使ったクラスタ内のノードの追加と削除」または 124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」
- 94 ページの「GUI を使ったノードの削除」または 117 ページの「CLI を使ったノードの削除」

強制 CXFS シャットダウン後のクラスタへの再設定

強制 CXFS シャットダウンの後は、ノードが再起動されるか、次の cmgr(1m) コマンドを入力してノードに対して明示的に CXFS メンバーシップを許可するまで、ローカル・ノードは CXFS メンバーシップを再開しません。

```
cmgr> admin cxfs_start
```

この手順は、GUI を使用して実行することもできます。105 ページの「GUI を使ったローカル・ノードのメンバーシップ許可」を参照してください。

強制シャットダウンをノードで実行する場合、クラスタに戻される前にノードで CXFS を再開する必要があります。ノードがクラスタ内にあり、アクティブになっていることがクラスタ設定データベースに示されているときにこれを行うと、ノードは CXFS メンバーシップ・デーモンを再開します。したがって、この操作は、データベースをリセットした後や、CXFS サービスを停止した後に行うことができます。

例:

```
cmgr> admin cxfs_start
```

リセット機能と強制 CXFS シャットダウン

注意: リセット機能を使用してノードで強制シャットダウンを実行して、シャットダウンによってクラスタ上限量が失われなかった場合、ノードは適切なノードによってリセット(再起動)されます。

リセットについての詳細は、17 ページの「ハードウェア・リセット」を参照してください。

再起動時のクラスタ設置データベースの再起動の回避

chkconfig(1M) に対する cxfs_cluster フラグは、clconfd デーモンを制御します。このフラグがオフの場合、clconfd は次回再起動時に開始されず、カーネルはクラスタに設定されません。システムやハードウェアのアップグレードなどのメインテナンス作業を行う場合にクラスタからノードを一時的に削除する場合は、ノードを再起動する前にオフにしておくと便利です。

以下を実行します。

```
# /etc/chkconfig cxfs_cluster off
# reboot
```

ログ・ファイル管理

ディスクがいっぱいにならないよう、ログ・ファイルは、少なくとも毎週ローテーションさせます。

以下の節では、スクリプトの例を示します。このようなスクリプトを定期的に実行するには、root crontab にエントリを加えることができます。

ログ・レベルについての詳細は、100 ページの「GUI を使ったログ・グループの設定」を参照してください。

大容量のログ・ファイルのローテーション

大容量のファイルを新しい場所にコピーするには、次のようなスクリプトを使用できます。新しい場所にあるファイルは、このスクリプトが実行されるたびに上書きされます。

```
#!/bin/sh
# Argument is maximum size of a log file (in characters) - default: 500000

size=${1:-500000}
find /var/cluster/ha/log -type f ! -name '*.OLD' -size +${size}c -print | while read log_file; do
        cp ${log_file} ${log_file}.OLD
        echo '*** LOG FILE ROTATION ' `date` '***' > ${log_file}
done
```

54 ページの「 /etc/config/cad.options の設定」および 55 ページの「/etc/config/fs2d.options の設定」も参照してください。

すべてのログ・ファイルのローテーション

すべてのファイルを新しい場所にコピーするには、次のようなスクリプトを使用できます。

```
#!/bin/sh

DATE=`/sbin/date +'%U-%a'`
LOG_DIR="/var/cluster/ha/log"
HOST=`/usr/bsd/hostname -s`
LOG_FILES="cad_log cmond_log fs2d_log"
LOG_HFILES="cli clconfd crsd"

LOG_ARCH=$LOG_DIR"/Old-Log"

if [ ! -d $LOG_ARCH ] ; then
   mkdir $LOG_ARCH
fi

for file in $LOG_FILES
do

  rm -f ${LOG_ARCH}/${file}-${DATE}
  cp ${LOG_DIR}/${file} ${LOG_ARCH}/${file}-${DATE}
  echo "Log Rotation at `date`" > ${LOG_DIR}/${file}
done
```

```
for file in $LOG_HFILES
do

  rm -f ${LOG_ARCH}/${file}_${HOST}-${DATE}
  cp ${LOG_DIR}/${file}_${HOST} ${LOG_ARCH}/${file}_${HOST}-${DATE}
  echo "Log Rotation at `date`" > ${LOG_DIR}/${file}_${HOST}
done
```

ボリューム管理

CXFS では、XVM ボリューム・マネージャを使用します。XVM を使用することで、多数のディスクを高トランザクション・レート、高帯域幅、および高信頼性のファイルシステムに結合できます。CXFS では、以下を提供するために XVM を使用します。

- ディスクのストライプ
- ミラー化
- 連結
- 高度な回復機能

メモ: CXFS デーモンを開始する前に XVM コマンドを実行しようとすると、警告メッセージが表示され、XVM のローカル・ドメインに移動します。

XVM のローカル・ドメイン内にいる場合はファイルシステムを定義できますが、後から CXFS を開始してもファイルシステムは表示されません。CXFS を開始すると、XVM はクラスタ・ドメインに切替わり、ファイルシステムは認識されなくなります。これは、ファイルシステムをローカル・ドメイン内で定義したためです。クラスタ・ドメインでファイルシステムを使用するには、`give` コマンドを使用しなければなりません。したがって、ボリュームは、クラスタ・ドメイン内で直接定義することをお薦めします。

詳細については、『XVM Volume Manager Administrator's Guide』を参照してください。

ディスク管理

この節では、バックアップ、NFS、Quota、RealTime、および SAMBA について説明します。

バックアップ

CXFS では、VERITAS NetBackup や Legato NetWorker などの商用バックアップ・パッケージを使用して、ローカル・エリア・ネットワーク (LAN: Local Area Network) から独立したバックアップを作成できます。これらのパッケージを使用すると、バックアップ作業をバックアップ・サーバに統合できると同時に、低速の LAN ではなくストレージ・エリア・ネットワーク (SAN: Storage Area Network) を通じてデータを渡すことができます。

たとえば、バックアップ・サーバとして指定された SAN 上のホストでバックアップ・パッケージを実行できます。このサーバでは、SAN ディスクに接続されているテープ・ドライブとのチャネル接続を使用できます。また、CXFS を通じてファイルシステムを表示したり、バックアップ・サーバを通じてディスクからテープ・ドライブに直接データを転送するバックアップ・アプリケーションを実行できます。

これにより、非常に容量の大きいファイルシステムの場合でも、バックアップ帯域幅をストレージ・サイズに合わせて拡張できます。ディスク・チャネルの数、バックアップ・サーバのサイズ、およびテープ・チャネルの数は、バックアップ帯域幅の条件を満たすよう増やすことができます。

NFS

NFS サーバを CXFS 上に配置して、クラスタの一部ではないコンピュータ・システムでファイルシステムを共有できます。これは、クライアント・ノードまたはサーバ・ノードで実行できます。

Quota

XFS の Quota は、サポートされていますが、Quota のマウント・オプションは、ファイルシステムのすべてのマウントで同じでなければなりません。Quota は、クラスタ内のどこからでも XFS ファイルシステムと同じように管理できます。

RealTime

RealTime がサポートされています。CXFS ファイルシステムには、RealTime サブボリュームを含めることができます。これは、XFS の RealTime サブボリュームと同じように動作します。

SAMBA

CXFS 上でSAMBA を実行し、NT マシンをクライアントにしてファイルシステムにアクセスできるようにすることが可能です。

ファイルシステムのメインテナンス

通常、ファイルシステム情報は /etc/fstab に格納されますが、CXFS ファイルシステム情報はクラスタ全体に関連しているため、複製されたクラスタ設定データベースに格納されます。

管理者として、CXFS クラスタ・マネージャ・ツールを使用して CXFS ファイルシステム設定を行います。GUI についての詳細は101 ページの「GUI を使ったファイルシステム・タスク」、CLI についての詳細は121 ページの「CLI を使ったクラスタ・タスクとファイルシステム・タスク」をそれぞれ参照してください。

情報は、クラスタ全体に自動的に同じ方法で伝えられます。クラスタ設定デーモンは、情報が利用可能になるとすぐに、この情報に基づいて各ノードでファイルシステムをマウントします。

CXFS ファイルシステムは、クラスタ内のすべてのノードで自動的にマウントされます。クラスタがアクティブな場合は、新しい CXFS ファイルシステムを設定に追加できます。

クラスタ設定内の変更がクラスタ設定デーモンによって検出されると、設定されているすべてのファイルシステムで mount -a コマンドと同等のコマンドが実行されます。

注意: ファイルシステムがマウントされている間は、CXFS ファイルシステム定義を変更したり削除しないでください。CXFS ファイルシステム定義を変更または削除する場合は、まずファイルシステムをアンマウントしてから行い、変更後はファイルシステムを再びマウントしてください。

ファイルシステムのマウント

マウント情報は、GUI の「ファイルシステムのマウント(Mount a Filesystem)」タスク(「新規ファイルシステムの設定(Set Up a New Filesystem)」設定ガイド・タスクセットの一部です)、または CLI の cmgr(1M) のサブコマンド modify を使用して入力します。GUI を使用したマウントについての詳細は、72 ページの「GUI を使った新規ファイルシステムの設定」および101 ページの「GUI を使ったファイルシステムの定義」を参照してください。CLI を使用した新しいファイルシステムの定義およびマウントについての詳細は124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」、すでに定義されているが現在はアンマウントされているファイルシステムのマウントについての詳細は128 ページの「CLI を使った定義されたファイルシステムの再マウント」をそれぞれ参照してください。

正しく定義およびマウントされていれば、CXFS ファイルシステムは、複製されたデータベースに収集された情報に基づいて、ローカル・クラスタ設定デーモン clconfd によって各ノードで自動的にマウントされます。ファイルシステム設定がデータベースに入力されたら、ユーザによる操作は必要ありません。

注意: CXFS ファイルシステムをマウントするために、mount(1M) コマンドは使用しないでください。このコマンドを使用すると、ほかのマシンと異なる方法でファイルシステムが使用されるため、データが失われたり、破壊される可能性があります。

CXFS ファイルシステムは、クラスタ内のすべてのノードでマウントされているか、またはまったくマウントされていないかのいずれかでなければなりません。その他の場合は、ファイルシステムがノードごとに異なる方法でマウントされているため、データの損失や破壊につながる可能性があります。GUI および CLI では、クラスタ内のノードのサブセットにファイルシステムをマウントすることはできません。

CXFS を使用している場合、マウント・ポイントはネストできません。つまり、/usr や /usr/home のように、ファイルシステム内にファイルシステムを含めることはできません。

ファイルシステムのアンマウント

CXFS ファイルシステムをアンマウントするには、GUI の「ファイルシステムのアンマウント(Unmount a Filesystem)」タスク、または CLI の cmgr のサブコマンド modify を使用します。詳細については、103 ページの「GUI を使ったファイルシステムのアンマウント」または127 ページの「CLI を使ったファイルシステムのアンマウント」を参照してください。

これらのタスクによって、クラスタ内のすべてのノードからファイルシステムがアンマウントされます。このアクションはすべてのノードでアンマウントを実行しますが、ファイルシステムが使用中の場合は、いくつかのノードで操作が失敗することもあります。メタデータ・サーバでは、すべてのクライアントがファイルシステムを正常にアンマウントするまで、アンマウントは成功しません。すべてのノードは、成功するまでアンマウントを再試行しますが、すべてのノードでファイルシステムがアンマウントされたことを示す一元化されたレポートはありません。すべてのノードからファイルシステムがアンマウントされていることを確認するには、以下のいずれかを実行します。

- ファイルシステムがアンマウントされていることを示すメッセージを、メタデータ・サーバの SYSLOG ファイルをチェックして確認する
- メタデータ・サーバで GUI または CLI を実行し、サーバからファイルシステムを無効にした後、ファイルシステムが完全に無効になっていることが GUI で表示されるまで待つ（一部のクライアントにまだマウントされている場合はエラーになり、残っているクライアントが GUI に表示されます）

ファイルシステムの拡張

CXFS ファイルシステムを拡張するには、以下を実行します。

1. CXFS ファイルシステムをアンマウントします。詳細については、103 ページの「GUI を使ったファイルシステムのアンマウント」または127 ページの「CLI を使ったファイルシステムのアンマウント」を参照してください。
2. CXFS ファイルシステムを XFS ファイルシステムとしてマウントします。『IRIX Admin: Disks and Filesystems』を参照してください。
3. ファイルシステムを拡張するには、xfs_growfs(1M) コマンドを使用します。

4. umount(1M) コマンドを使用して、XFS ファイルシステムをアンマウントします。
5. CXFS ファイルシステムをマウントします。102 ページの「GUI を使ったファイルシステムのマウント」または128 ページの「CLI を使った定義されたファイルシステムの再マウント」を参照してください。

ダンプと復元

ダンプおよび復元の手順は、サーバ・ノードから実行しなければなりません。xfsdump(1M) コマンドおよび xfsrestore(1M) コマンドでは、CXFS サーバ・ノードだけで機能する特殊なシステム・コールが利用されます。したがって、これらのコマンドはサーバからのみ使用してください。

ダンプ・プロセス中は、ファイルシステムのアクティブなクライアントがいてもかまいません。

クラスタ設定データベースのバックアップと復元

データベースを保存しておき、ディスクがクラッシュした場合や不正な変更が行われた場合などに保存時の状態に復元できるようにしたいときは、データベースのバックアップを実行することをお薦めします。

クラスタ・データベースのバックアップを実行するには、cdbBackup(1M) コマンドを使用します。

復元を実行するには、クラスタ・サービスを停止し、古いデータベースを削除して、cdbRestore(1M) コマンドを使用します。詳細については、マン・ページを参照してください。

chkconfig フラグ

メモ: これらのフラグは、通常は管理者が chkconfig コマンドを使用して操作することはなく、GUI または cmgr によって設定または設定の解除が行われます。

CXFS には、chkconfig(1M) コマンドに対して以下のフラグがあります。

- cluster。ほかのクラスタ管理デーモン(主に複製されたクラスタ設定データベース)を制御します。これがオフの場合、データベース・デーモンは次回再起動されるときに開始されず、ほかのノードでクラスタ設定を変更してもデータベースのローカル・コピーは更新されません。これにより、後で問題が発生することがあります。特に、ノードの過半数でデータベース・デーモンが実行されていない場合は、問題が発生する可能性が高くなります。データベース・デーモンが実行されていない場合、クラスタ設定データベースにローカルでアクセスすることはできず、ノードはクラスタに設定されません。
- cxfs_cluster。clconfd デーモン、およびシステム・シャットダウン中に cxfs_shutdown コマンドが使用されるかどうかを制御します。

cxfs_shutdown コマンドは、再起動の前にクラスタからの取消しを試みます。取消しが試みられない場合、再起動は失敗と見なされ、その他のノードでは回復が必要となります。

メモ: これらのフラグでは、クラスタではなく、プロセスのみを制御します。クラスタを制御するプロセスを停止しても、クラスタは停止されません。プロセスを開始すると、CXFS サービスがデータベースでアクティブとマークされている場合のみ、クラスタが開始されます。

クラスタとノードのステータスのモニタ

システムのステータスは、以下の方法で表示できます。

- /var/cluster/ha/logs のログ・ファイルをモニタする
- GUI または tail(1) コマンドを使用してシステム・ログを表示し、SYSLOG ファイルの最後を表示する
- GUI の CXFS クラスタ表示または cluster_status コマンドを使用して、クラスタの状態を継続的に監視する
- GUI または CLI のいずれかを使用して、個々のノードまたはクラスタのステータスを照会する
- ls(1) コマンドを使用して、ファイルシステムを手動でテストする
- PCP (Performance Co-Pilot) を使用してシステムをモニタする。PCP を使用すると、すべてのノードに対する読取り/書込みの分散負荷をモニタし、警告およびログを実行できます。『Performance Co-Pilot User's and Administrator's Guide』、『Performance Co-Pilot Programmer's Guide』、および pmie(1) と pmieconf(1) のマン・ページを参照してください。

メモ: PCP パッケージの XVM 統計はデフォルトではインストールされないため、手動でインストールする必要があります。

以下の節では、これらのタスクの一部を実行するための手順について説明します。

ログ・ファイルのモニタ

問題に対してリストされる以下のログ・ファイルをモニタする必要があります。

- /var/adm/SYSLOG (システム・ログ。クラスタが形成されていることを示す Membership delivered メッセージを検索)
- /var/cluster/ha/log/cad_log (GUI および clconfd からのイベント)
- /var/cluster/ha/log/clconfd_*hostname* (カーネルのステータス)
- /var/cluster/ha/log/cli_*hostname* (コマンド・ライン・インタフェース・ログ)
- /var/cluster/ha/log/cmond_log (その他のデーモンのモニタ)
- /var/cluster/ha/log/crsd_*hostname* (デーモン・ログのリセット)

- /var/cluster/ha/log/diags_*hostname*(シリアルおよびネットワークの接続性のテストなどの診断ツールの出力)
- /var/cluster/ha/log/fs2d_log(fs2d メンバーシップのステータス)
- /var/sysadm/salog(システム管理ログ)

ディスクがログ・メッセージでいっぱいになる場合は、156 ページの「ログ・ファイル管理」を参照してください。

注意: ログ・ファイルの名前は変更しないでください。名前を変更すると、エラーが発生します。

システムのステータスのモニタ

システムのステータスは、GUI または cluster_status コマンドを使用してモニタできます。

cluster_status を使ったシステムのステータスのモニタ

curses インタフェースを使用してクラスタをモニタするには、cluster_status コマンドを使用できます。たとえば、以下の例に、1 つのファイルシステムがマウントされており、ヘルプ・テキストが表示されている 3 ノード・クラスタを示します。

```
# /var/cluster/cmgr-scripts/cluster_status
+ Cluster=cxfs6-8  FailSafe=Not Configured CXFS=ACTIVE                15:15:33
   Nodes =   cxfs6    cxfs7    cxfs8
FailSafe =
    CXFS =      UP       UP       UP

CXFS              DevName           MountPoint           MetaServer     Status
        /dev/cxvm/concat0             /concat0                cxfs7         UP

                       +--------+ cluster_status Help +---------+
                       |  on s  - Toggle Sound on event         |
                       |  on r  - Toggle Resource Group View    |
                       |  on c  - Toggle CXFS View              |
                       |     h  - Toggle help screen            |
                       |     i  - View Resource Group detail    |
                       |     q  - Quit cluster_status           |
                       +--- Press 'h' to remove help window --+

------------------------------------------------------------------
cmd('h' for help) >
```

上記の例では、ノードまたはクラスタのステータスが変わると、サウンドがアクティブになります。r コマンドおよび i コマンドは CXFS とは関係がなく、IRIS FailSafe だけで使用できます。s 設定は、-m(ミュート)オプションと共に cluster_status を呼出すことによってオーバーライドできます。

GUI を使ったシステムのステータスのモニタ

クラスタの状態を継続的に監視する簡単な方法は、CXFS クラスタ表示を使用することです。クラスタ表示を開始するには、40 ページの「CXFS クラスタ表示の起動」を参照してください。

コンポーネントの状態のカラーの凡例は、以下のとおりです。

グレー	正常だが、オンラインまたはアクティブになっていない
緑	正常で、アクティブまたはオンラインになっている
点滅する赤	コンポーネントに関する問題がある

ノードのステータス

ノードのステータスを照会するには、そのノードの論理名を指定します。ノードのステータスは、以下のいずれかになります。

- 「実行中(UP)」: CXFS サービスが開始されており、ノードが CXFS メンバーシップの一部であることを意味します。詳細については、13 ページの「メンバーシップ上限量」を参照してください。
- 「停止(DOWN)」: CXFS サービスが開始されており、ノードがクラスタの一部として定義されているにもかかわらず、このノードは現在の CXFS メンバーシップ内に含まれていないことを意味します。
- 「非アクティブ(INACTIVE)」: 「CXFS サービスの開始(Start CXFS Services)」タスクが実行されていないことを意味します。
- 「不明(UNKNOWN)」: クエリーを実行しているノードで CXFS サービスが実行されていないために状態を判別できないことを意味します。

状態の情報は、CXFS サービスが開始されている場合にのみ実行されるデーモンによって交換されます。ほかのノードのステータスを報告するには、指定したノードで CXFS サービスが実行されている必要があります。

たとえば、`node2` の状態を表示するには、`node1` で CXFS サービスが開始されていなければなりません。`node1` で CXFS サービスが開始されていれば、クラスタ内のその他のすべてのノードの状態は正確に報告されます。ただし、`node1` の CXFS サービスが開始されていない場合は、以下の状態が報告されます。

- このノード自体の状態としては「非アクティブ(INACTIVE)」が報告されます。これは、「CXFS サービスの開始(Start CXFS Services)」タスクが実行されていないと判断できるためです。
- その他のすべてのノードの状態としては「不明(UNKNOWN)」が報告されます。これは、その他のノードと情報を交換するために必要なデーモンが実行されていないので状態を判断できないためです。

cluster_status を使ったノードのステータスのモニタ

クラスタ内のノードのステータスをモニタするには、cluster_status コマンドを使用できます。たとえば、次の出力は、CXFS クラスタ内の 3 つのノードがすべて実行中であることを示しています。

```
+ Cluster=cxfs6-8  FailSafe=Not Configured CXFS=ACTIVE              15:15:33
   Nodes =   cxfs6    cxfs7    cxfs8
FailSafe =
    CXFS =      UP       UP       UP
```

c コマンドを off に切替えた場合、CXFS 行はなくなります。

GUI を使ったノードのステータスのモニタ

ノードのステータスをモニタするには、CXFS クラスタ表示を使用できます。クラスタ表示を起動するには、40 ページの「CXFS クラスタ表示の起動」を参照してください。

デフォルトのクラスタのノードの動作状態をモニタするには、「表示(View)」メニュー ->「クラスタ内のノード(Nodes in Cluster)」を選択します。

CXFS または FailSafe、あるいはその両方にノードが適用されるかどうかを判断するには、画面でそのノード名をダブルクリックします。図7-1に、タイプ CXFS and FailSafe のノードの例を示します。

図7-1 ノード・タイプが表示されたノードの詳細の例

CLI を使ったノードのステータスの照会

ノードのステータスを照会するには、次の CLI コマンドを使用します。

```
cmgr> show status of node node_name
```

CLI を使ったシステム・コントローラへの ping の実行

CXFS が実行されている場合、次のコマンドを使用して、ノードのシステム・コントローラが応答しているかどうかを判断できます。

```
cmgr> admin ping node node_name
```

このコマンドは、CXFS デーモンを使用して、システム・コントローラが応答しているかどうかをテストします。

CXFS デーモンが実行されていない場合でも、admin ping コマンドの standalone オプションを使用すると、クラスタ内のノードのリセット接続性を確認できます。

```
cmgr> admin ping standalone node  node_name
```

このコマンドは、ping コマンドを直接呼出して、指定したノードでシステム・コントローラが応答しているかどうかをテストします。

CLI を使ったリセット・シリアル回線のモニタ

ノードでシステム・コントローラに対して ping を実行するには、以下のように CLI を使用できます（改行して読みやすくしてあります）。

```
cmgr> admin ping dev_name device_name of dev_type device_type
with sysctrl_type system_controller_type
```

クラスタのステータス

クラスタのステータスを照会するには、そのクラスタの名前を指定します。クラスタのステータスは、以下のいずれかになります。

- 「アクティブ(ACTIVE)」: クラスタが実行中であることを意味します。
- 「非アクティブ(INACTIVE)」:「CXFS サービスの開始(Start CXFS Services)」タスクが実行されていないことを意味します。
- 「エラー(ERROR)」: 一部のノードが「停止(DOWN)」状態であることを意味します（つまり、クラスタは実行されているべきなのに、実行されていません）。
- 「不明(UNKNOWN)」: クエリーを実行しているノードで CXFS サービスが実行されていないために状態を判別できないことを意味します。詳細については、168 ページの「ノードのステータス」を参照してください。

cluster_status を使ったクラスタのステータスの照会

次の出力では、CXFS クラスタ内が実行中であり、cxfs7 が /dev/cxvm/concat0 XVM ボリュームのメタ・サーバであることを示しています。

```
+ Cluster=cxfs6-8  FailSafe=Not Configured CXFS=ACTIVE               15:18:28
   Nodes =   cxfs6    cxfs7    cxfs8
FailSafe =
    CXFS =      UP       UP       UP

CXFS              DevName           MountPoint           MetaServer     Status
        /dev/cxvm/concat0             /concat0                cxfs7          UP
```

GUI を使ったクラスタのステータスの照会

クラスタのステータスをモニタするには、CXFS クラスタ表示を使用できます。クラスタ表示を開始するには、40 ページの「CXFS クラスタ表示の起動」を参照してください。

CLI を使ったクラスタのステータスの照会

ノードおよびクラスタのステータスを照会するには、次のコマンドを使用します。

```
cmgr> show status of cluster cluster_name
```

クラスタが実行中の場合、/usr/cluster/bin/clconf_info を使用して詳細情報を表示できます。

例:

```
cxfs6 # clconf_info
Membership since Thu Mar  1 08:15:39 2001
Node         NodeId     Status    Age   Incarnation     CellId
cxfs6             6         UP      0             0          2
cxfs7             7         UP      0             0          1
cxfs8             8         UP      0             0          0
1 CXFS FileSystems
/dev/cxvm/concat0 on /concat0  enabled  server=(cxfs7)  2 client(s)=(cxfs8,cxfs6)
```

トラブルシューティング

CXFS クラスタの設定と管理は複雑なタスクになる場合があります。一般に、ほとんどの問題は、ノードを再起動することによって解決できますが、この章のトピックでは、再起動せずに解決する方法を説明します。

- 「トラブルシューティング方法」
- 183 ページの「問題の回避」
- 188 ページの「一般的な問題」
- 191 ページの「エラー・メッセージの理解」
- 208 ページの「修正処置」

トラブルシューティング方法

CXFS 問題をトラブルシューティングするには、以下を実行します。

- 173 ページの「ツールについての理解」
- 179 ページの「クラスタのステータスの識別」
- 181 ページの「問題の検出」

ツールについての理解

この節では、CXFS のトラブルシューティングに必要なツールの概要について説明します。

注意: ここでリストされているコマンドの多くはこのマニュアルの対象外であり、簡単に参照できるように紹介されているだけです。これらのコマンドを使用する前に、参照されているその他のガイドやマン・ページで詳細を参照してください。

- 174 ページの「物理ストレージ・ツール」
- 174 ページの「クラスタ設定ツール」
- 175 ページの「クラスタ制御ツール」
- 176 ページの「ネットワーク・ツール」

- 176 ページの「クラスタ / ノードのステータス・ツール」
- 177 ページの「パフォーマンス・モニタ・ツール」
- 178 ページの「カーネルのステータス・ツール」
- 179 ページの「ログ・ファイル」

物理ストレージ・ツール

以下の物理ストレージ・ツールについて理解します。

- ハードウェア・インベントリを表示するには、hinv(1m) コマンドを使用します。

  # **/sbin/hinv**

  出力が予想したものとは異なる場合、scsiha(1m) コマンドを使用してデバイスの検査を行い、SCSI バス・リセットを実行します。

  # **/usr/sbin/scsiha -pr *bus_number***

- I/O デバイスを設定するには、ioconfig(1m) コマンドを使用します。

  # **/sbin/ioconfig -f /hw**

- 物理ボリュームを表示するには、xvm(1m) コマンドを使用します。

  # **/sbin/xvm show -v phys/**

『XVM Volume Manager Administrator's Guide』を参照してください。

クラスタ設定ツール

以下のクラスタ設定ツールについて理解します。

- XVM ボリュームを設定するには、xvm(1m) コマンドを使用します。

  # **/sbin/xvm**

  『XVM Volume Manager Administrator's Guide』を参照してください。

- CXFS ノードおよび CXFS クラスタを設定するには、GUI または CLI のいずれかを使用します。
  - クラスタ表示が開かれている GUI:

    ```
    # /usr/sbin/cxdetail
    ```

    33 ページの「GUI の概要」および81 ページの 第4章「GUI タスク・リファレンス」を参照してください。
  - プロンプト表示ありの cmgr(1m) コマンド・ライン

    ```
    # /var/cluster/bin/cmgr -p
    ```

    43 ページの「CLI の概要」および107 ページの 第5章「CLI タスク・リファレンス」を参照してください。
- データベースを再度初期化するには、cdbreinit コマンドを使用します。

  ```
  # /usr/cluster/bin/cdbreinit
  ```

  212 ページの「クラスタ設定データベースの再作成」を参照してください。

クラスタ制御ツール

以下のクラスタ制御ツールについて理解します。

- クラスタ・サービス・デーモンを開始および停止するには、以下のコマンドを使用します。

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

  これらのコマンドは、ファイルシステムが使用可能でもクラスタのステータスからはそれが判断できなかったり、クラスタ上限量が失われた場合に便利です。

  184 ページの「fs2d メンバーシップ上限量の安定性」、208 ページの「CXFS サービスの再開」、208 ページの「クラスタ設定データベースのクリア」、および212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。
- CXFS サービスを開始および停止するには、GUI または以下の cmgr(1m) コマンドを使用します。

  ```
  cmgr> start cx_services on node hostname for cluster clustername
  cmgr> stop cx_services on node hostname for cluster clustername
  ```

  メタデータ・サーバではこのコマンドを実行しないようにしてください。実行すると、ファイルシステムが再設定されてしまいます。135 ページの「CLI を使った CXFS サービス・タスク」および98 ページの「GUI を使った CXFS サービス・タスク」を参照してください。

- ローカル・ノードのメタデータ・サーバを強制的に回復して、そのローカル・ノードのメンバーシップを許可および取消すには、GUI または以下の cmgr(1m) コマンドを使用します。

  ```
  cmgr> admin cxfs_start
  cmgr> admin cxfs_stop
  ```

  回復が完了するまでは、以降の admin cxfs_start を発行しないでください。ローカル・ノードは、回復が完了するまでメンバーシップに再設定できません。

  140 ページの「CLI を使ったローカル・ノードのメンバーシップ取消し」、140 ページの「CLI を使ったローカル・ノードのメンバーシップ許可」、105 ページの「GUI を使ったローカル・ノードのメンバーシップ取消し」、および105 ページの「GUI を使ったローカル・ノードのメンバーシップ許可」を参照してください。

ネットワーク・ツール

以下のネットワーク・ツールについて理解します。

- ネットワーク・ホストにパケットを送信するには、ping(1) コマンドを使用します。

  ```
  # /usr/etc/ping
  ```

- ネットワークのステータスを表示するには、netstat(1) コマンドを使用します。

  ```
  # /usr/etc/netstat
  ```

クラスタ/ノードのステータス・ツール

以下のクラスタ/ノードのステータス・ツールについて理解します。

- 実行されているクラスタ・デーモンを表示するには、ps(1) コマンドを使用します。

  ```
  # /sbin/ps -ef | grep cluster
  ```

  66 ページの「クラスタ・デーモンが実行されていることの確認」を参照してください。

- クラスタおよびファイルシステムのステータスを表示するには、以下のいずれかを使用します。
  - GUI:

    ```
    # /usr/sbin/cxdetail
    ```

    97 ページの「GUI を使ったクラスタの表示」を参照してください。

  - cluster_status(1m) コマンド:

    ```
    # /var/cluster/cmgr-scripts/cluster_status
    ```

166 ページの「cluster_status を使ったシステムのステータスのモニタ」を参照してください。

- clconf_info コマンド:

```
# /var/cluster/bin/clconf_info
```

- マウントされているファイルシステムを表示するには、mount(1m) コマンドまたは df(1) コマンドを使用します。

```
# /sbin/mount
```

```
# /usr/sbin/df
```

- 空きディスク・ブロックの数を報告するには、df(1) コマンドを使用します。

```
# /usr/sbin/df
```

- ボリュームを表示するには、df(1) コマンドを使用します。

```
# /sbin/xvm show vol/
```

『XVM Volume Manager Administrator's Guide』を参照してください。

パフォーマンス・モニタ・ツール

以下のパフォーマンス・モニタ・ツールについて理解します。

- システム・アクティビティをモニタするには、sar(1) コマンドを使用します。

```
# /usr/bin/sar
```

- ファイルシステム・バッファ・キャッシュ・アクティビティをモニタするには、bufview(1) コマンドを使用します。

```
# /usr/sbin/bufview
```

メモ: bufview は、使用中のノードではインタラクティブに使用せずに、バッチ・モードで実行してください。

- オペレーティング・システム・アクティビティ・データをモニタするには、osview(1) コマンドを使用します。

```
# /usr/sbin/osview
```

- XVM ボリュームの統計をモニタするには、xvm(1m) コマンドを使用します。

```
# /sbin/xvm change stat on {concatname|stripename|physname}
```

『XVM Volume Manager Administrator's Guide』を参照してください。

- システム・パフォーマンスをモニタするには、PCP (Performance Co-Pilot) を使用します。『Performance Co-Pilot User's and Administrator's Guide』、『Performance Co-Pilot Programmer's Guide』、および pmie(1) と pmieconf(1) のマン・ページを参照してください。

カーネルのステータス・ツール

以下のカーネルのステータス・ツールについて理解します(これについては、日本 SGI の営業担当員の支援が必要な場合があります)。

- カーネルのステータスを判断するには、icrash(1m) コマンドを使用します。

  # **/usr/bin/icrash**

  - mesglist: メッセージを受信者までトレースします。
  - sthread | grep cmsd: CXFS(カーネル)メンバーシップの状態を判断します。出力には、以下が表示される場合があります。
    - cms_nascent(): ノードが開始されてからクラスタに設定されていないことを示します。
    - cms_follower(): ほかのノードによって CXFS メンバーシップ(リーダー)が作成されるまでノードが待機していることを示します。
    - cms_leader(): ノードが CXFS メンバーシップ作成をリードしていることを示します。
    - cms_declare_membership(): ノードは CXFS メンバーシップを宣言する準備はできているが、リセットを待っていることを示します。
    - cms_stable(): CXFS メンバーシップが形成されており、安定していることを示します。
    - cms_shutdown(): ノードがシャットダウンされており、CXFS メンバーシップ内にないことを示します。
    - cms_dead(): ノードがダウンしていることを示します。
  - tcp_channels: ほかのノードとの接続のステータスを判断します。
  - sinfo: クライアント/サーバおよびファイルシステムを表示します。

- 役立つデバッグ情報を提供する内部カーネル・ルーチンを呼出すには、idbg(1m) コマンドを使用します。

  # **/usr/sbin/idbg**

ログ・ファイル

以下のログ・ファイルについて理解します。

- /var/adm/SYSLOG（Membership delivered を検索）
- /var/cluster/ha/log/cad_log
- /var/cluster/ha/log/clconfd_*hostname*
- /var/cluster/ha/log/cli_*hostname*
- /var/cluster/ha/log/cmond_log
- /var/cluster/ha/log/crsd_*hostname*
- /var/cluster/ha/log/diags_*hostname*
- /var/cluster/ha/log/fs2d_log
- /var/sysadm/salog

165 ページの「ログ・ファイルのモニタ」を参照してください。

クラスタのステータスの識別

問題が発生した場合、以下の質問に回答することによって、クラスタのステータスを識別します。

- クラスタ・デーモンは実行されているか。66 ページの「クラスタ・デーモンが実行されていることの確認」を参照してください。
- どのノードが CXFS メンバーシップ内にあるか。166 ページの「cluster_status を使ったシステムのステータスのモニタ」、172 ページの「CLI を使ったクラスタのステータスの照会」、および /var/adm/SYSLOG ファイルを参照してください。
- どのノードがクラスタ (fs2d) メンバーシップ内にあるか。各ノードの /var/cluster/ha/log/fs2d_log ファイルを参照してください。
- すべてのノードでデータベースが一致しているか。各ノードへのこのログインを判別し、/var/cluster/ha/log/fs2d_log ファイルおよびデータベース・チェックサムを調査します。
- さまざまなクライアント・ノードにログオンするか、詳細の表示されている「クラスタ表示(Cluster View)」画面を使用して、以下に回答します。
  - すべてのノードでデバイスが使用可能であるか。以下を使用します。

- xvm(1m) コマンド: 物理ボリュームを表示します。

  ```
  xvm> show -v phys/
  ```

- ls(1) コマンド: /dev/cxvm ディレクトリの内容をリストします。

  ```
  # ls /dev/cxvm
  ```

- hinv(1m) コマンド: ハードウェア・インベントリを表示します。

  ```
  # hinv
  ```

– すべてのノードにファイルシステムがマウントされているか。mount(1m) コマンドおよび clconf_info コマンドを使用します。

– どのノードが各ファイルシステムのメタデータ・サーバであるか。cluster_status(1m) コマンドまたは clconf_info コマンドを使用します。

メタデータ・サーバでは、clconf_info コマンドを使用します。

- メタデータ・サーバが回復中であるか。icrash(1m) コマンドを使用してメッセージを検索し、/var/adm/SYSLOG をチェックします。178 ページの「カーネルのステータス・ツール」を参照してください。以下のようなメッセージは、回復ステータスを示します。

  – 回復が進行中の場合

```
Mar 13 11:31:02 1A:p2 unix: ALERT: CXFS Recovery: Cell 1: Client Cell 0 Died, Recovering </scratch/p9/local>
```

  – 回復が完了している場合

```
Mar 13 11:31:04 5A:p2 unix: NOTICE: Signaling end of recovery cell 1
```

- (20 秒を超える) 長時間実行されているカーネル・メッセージがあるか。これを調べるには、icrash mesglist コマンドを使用します。

- ファイルシステムがマウントされていない場合、XVM にオンラインで表示されるか。次の xvm(1m) コマンドを使用できます。

  ```
  xvm> show vol/*
  ```

問題の検出

問題を見つけるには、以下を実行します。

- ログ・ファイルを調べます。
  - すべてのログ・ファイル内でエラーを検索します。165 ページの「ログ・ファイルのモニタ」を参照してください。問題の期間内のメッセージをすべて調べます。
  - エラーの原因を突止めます。エラーを引起こしたイベントを検索してみます。
- `icrash` コマンドを使用します。178 ページの「カーネルのステータス・ツール」を参照してください。
- CXFS クラスタ・マネージャ GUI で詳細情報パネルを使用し、具体的な設定情報を絞込みます。36 ページの「CXFS クラスタ表示」を参照してください。
- GUI で「接続性のテスト(Test Connectivity)」タスクを実行します。104 ページの「GUI を使った接続性のテスト」を参照してください。
- 各ノードで次のコマンドを入力して、クラスタのノードで現在の CXFS メンバーシップがどのように表示されているかを判別します。

  ```
  # /usr/cluster/bin/clconf_info
  ```

  このコマンドを実行すると、以下のフィールドが表示されます。
  - Node（ノード名）
  - Node ID（ノード ID）
  - Status（ステータス。UP または DOWN）
  - Age（このフィールドは有益ではないので無視してください）
  - Incarnation（このフィールドは有益ではないので無視してください）
  - Cell ID（セル ID）。クラスタにノードを追加するときに CXFS ソフトウェアによって動的に割当てられる番号です（ユーザがセル ID 番号を定義することはありません）。

例:

```
# /usr/cluster/bin/clconf_info
Membership since Fri Sep 10 08:57:36 1999
Node          NodeId     Status    Age   Incarnation     CellId
cxfs6           1001         UP      1             7          0
cxfs7           1002         UP      0             0          1
cxfs8           1003         UP      0             0          2
2 CXFS FileSystems
/dev/xvm/test1 on /mnts/test1  disabled  server   0 clients
/dev/xvm/test2 on /mnts/test2  disabled  server   0 clients
```

- 各ノードで SYSLOG をチェックし、CXFS ファイルシステムが正しくマウントまたはアンマウントされていることを確認します。マウントまたはアンマウントできなかった場合、エラーがログに記録され、短い遅れの後で操作が再試行されます。
- アクティブなディスクを表示するには、sar(1) システム・アクティビティ・レポータを使用します。たとえば、以下の例では、アクティブなディスクを表示してそのディスク名を行の最後に配置し、10 秒の間毎秒ポーリングします。

  ```
  # sar -DF 1 10
  ```

  詳細については、sar(1) のマン・ページを参照してください。
- 使用中のバッファを表示するには、bufview(1) ファイルシステム・バッファ・キャッシュ・アクティビティ・モニタを使用します。bufview 内では、help サブコマンドを使用して、使用可能なサブコマンドを参照できます。たとえば、指定したフラグを持つものだけが表示されるように制限する f サブコマンドなどです。以下に、使用中のバッファを表示する場合の例を示します。

  ```
  # bufview
  f
  Buffer flags to display bsy
  ```

  詳細については、bufview(1) のマン・ページを参照してください。
- icrash(1M) IRIX システム・クラッシュ解析ユーティリティを使用します。詳細については、icrash(1M) のマン・ページを参照してください。
- クラスタ設定データベースのダンプを取得します。このようなダンプは、次のコマンドを使用して抽出できます。

  ```
  # /usr/cluster/bin/cdbutil -c 'gettree #' > dumpfile
  ```

問題の回避

この節では、以下の内容について説明します。

- 「適切な開始」
- 「設定の統一性」
- 「残っているクラスタの削除」
- 184 ページの「強制シャットダウン後の CXFS の再開」
- 184 ページの「`fs2d` メンバーシップ上限量の安定性」
- 185 ページの「GUI の使用」
- 185 ページの「ログ・ファイル名とサイズ」
- 185 ページの「Netscape と Brocade スイッチの GUI」
- 186 ページの「書込まれていないエクステントの追跡と排他書込みトークンに関するパフォーマンスの問題」
- 188 ページの「システム容量の効果的な使用」
- 188 ページの「ノード ID またはクラスタ ID を変更する前の再起動」

適切な開始

クラスタを設定する前に、必ず65 ページの「準備手順」の手順に従います。

設定の統一性

プール全体のノードの設定ファイルと、ネットワークの設定時には、すべて同じ設定を使用する必要があります。つまり、同じ名前を同じ順序で使用してください。52 ページの「システム・ファイルの設定」を参照してください。

残っているクラスタの削除

別の新しいクラスタの設定を始める前に、以前のクラスタのノードが CXFS メンバーシップに存在しないことを確認します。以下を入力します。

```
# icrash -e 'sthread | grep cmsd'
```

出力に cmsd カーネル・スレッドが表示される場合は、以下のコマンドを入力して強制的に CXFS をシャットダウンします。

```
# /usr/cluster/bin/cmgr -p
cmgr> admin cxfs_stop
```

その後、cmsd カーネル・スレッドをもう一度チェックします。

```
# icrash -e 'sthread | grep cmsd'
```

数秒待ってもまだ cmsd カーネルが存在している場合は、マシンを再起動するか、新しいクラスタ定義から除外してください。この状態では、マシンを新しいクラスタに設定することはできません。また、クラスタのその他の部分が新しい CXFS メンバーシップを形成できなくなる可能性があります。

強制シャットダウン後の CXFS の再開

強制シャットダウンをノードで実行した場合は、そのノードをクラスタに戻す前にノードで CXFS を再開する必要があります。ノードがクラスタ内にあってアクティブなことがまだデータベースに示されているときにこれを行うと、CXFS メンバーシップ・デーモンがノードによって再開されます。したがって、この操作は、データベースをリセットした後や、CXFS サービスを停止した後に行うことができます。

たとえば、開始するノードに以下を入力します。

```
# /usr/cluster/bin/cmgr -p
cmgr> stop cx_services on node localnode
cmgr> admin cxfs_start
```

154 ページの「強制 CXFS シャットダウン」も参照してください。

fs2d メンバーシップ上限量の安定性

設定プロセス中は、fs2d メンバーシップ上限量が常に安定した状態になければなりません。可能であれば、設定タスクの実行中は、複数のウィンドウを使用して各ノードの fs2d_log ファイルを表示してください。以下を入力します。

```
# tail -f /var/cluster/ha/log/fs2d_log
```

新しい上限量が出力されたら、メンバー・カウントをチェックします。通常の状況では、ノードを追加または削除するとメッセージが 2〜3 出力されますが、新しい上限量の適用後数秒以内に停止します。

十分な数のマシンが応答していない場合、上限量はなくなります。この場合、データベースは伝えられません。

fs2d メンバーシップ上限量の問題を検出した場合は、データベースにほかの変更を加える前にその問題を解決してください。正しい fs2d メンバーシップ上限量がないノード、またはすべてのノードで同時にクラスタ・インフラストラクチャ・デーモンを再開します。以下を入力します。

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

fs2d メンバーシップ上限量の問題を報告するときは、fs2d ログ・ファイルを提出してください。

GUI の使用

GUI を使用すると、クラスタとそのコンポーネントを CXFS クラスタ表示で簡単に表示できます。CXFS クラスタ表示を使用して実行状況を確認し、ノードを追加または削除するのが早すぎることがないようにしてください。ノードを定義した後は、別のノードを追加する前に、定義したノードがCXFS クラスタ表示に表示されるまで待つ必要があります。クラスタを定義した後は、ノードを追加する前に、定義したクラスタが表示されるまで待ってください。変更を加えるのが早すぎると、エラーが発生する可能性があります。

「表示(View)」メニューから、要素を選択してチェックできます。クラスタの詳細を表示するには、クラスタ名またはアイコンをクリックします。ステータスおよび設定の情報が、新しいウィンドウに表示されます。同じウィンドウに詳細を表示するには、以下を選択します。

「オプション(Options)」
>「詳細を表示(Show Details)」

詳細については、39 ページの「GUI の起動」を参照してください。

IRIX で GUI を実行している場合は、GUI アクションが行われているときに別の IRIX デスクトップに移動しないでください。移動すると、GUI がクラッシュする可能性があります。

ログ・ファイル名とサイズ

ログ・ファイルの名前は変更しないでください。ログ・ファイルの名前を変更すると、エラーが発生します。

ログ・ファイルは定期的にローテーションし、ディスクがいっぱいにならないようにします。156 ページの「ログ・ファイル管理」を参照してください。ディスク容量に関する問題が発生した場合は、冗長性の低いログ・レベルを選択することをお薦めします。100 ページの「GUI を使ったログ・グループの設定」または137 ページの「CLI を使ったログ・グループの設定」を参照してください。

Netscape と Brocade スイッチの GUI

IRIX ノード上の Netscape を通じて Brocade Web Tools V2.0 にアクセスする場合、Netscape を開始する前にまず以下のいずれかを入力する必要があります。

- sh(1) シェルまたは ksh(1) シェルの場合

  ```
  $ NOJIT=1; export NOJIT
  ```

- csh(1) シェルの場合

  ```
  % setenv NOJIT 1
  ```

これが実行されていないと、Netscape はコア・ダンプします。

書込まれていないエクステントの追跡と排他書込みトークンに関するパフォーマンスの問題

この節では、書込まれていないエクステントの追跡と排他書込みトークンに関するパフォーマンスの問題について説明します。

書込まれていないエクステントの追跡

ファイルシステムを定義するとき、書込まれていないエクステントの追跡をオン (unwritten=1) またはオフ (unwritten=0) にするかを指定できます。デフォルトではオンになります。

ほとんどの場合は、書込まれていないエクステントの追跡を使用してもパフォーマンスに影響が出ることはないので、セキュリティを高めるためにデフォルトを使用することをお薦めします。

ただし、ファイルが事前に割当てられており、さらにこれらの事前割当済みのエクステントが初めて 4 M バイト未満のレコードで書込まれるときは、書込まれていないエクステントの追跡がパフォーマンスに影響を及ぼします。これらの両方の状況に該当するときに CXFS で最適なパフォーマンスを得るには、unwritten=0（オフ）を使用してファイルシステムをビルドしなければならない場合があります。

メモ: unwritten=0 の使用に関しては、セキュリティ上の問題があります。詳細については、『IRIX Admin: Disks and Filesystems』を参照してください。

排他書込みトークン

適切なパフォーマンスを得るために、CXFS では排他書込みトークンを取得しないことをお薦めします。このため、以下のガイドラインを使用してください。

- ファイルを事前に割当てます。
- ファイルのサイズを最大サイズに設定し、切詰めなどによってサイズが変更されないようにします。
- ファイルには追加しないでください（つまり、O_APPEND は、開かれているときに真ではありません）。
- written としてエクステントをマークしないでください。

- アプリケーションに連続 preallocation コールの実行を許可しないでください。

これらのガイドラインに従ってもパフォーマンスの問題が解決されない場合は、MPI ジョブの実行前後と実行中に icrash stat コマンドを実行すると、役立つ情報を見つけることができます。詳細については、icrash(1m) のマン・ページを参照してください。

crontab ファイルによる過剰なファイルシステム・アクティビティの回避

デフォルトの root crontab ファイルには、以下のエントリが含まれます（行連結文字と改行により読みやすくしてあります）。

```
0       5       *       *       *       find / -local \
-type f '(' -name \ core -o -name dead.letter ')' \
-atime +7 -mtime +7 -exec rm -f '{}' ';'

 0 3 * * 0 if test -x /usr/etc/fsr; then (cd /usr/tmp; /usr/etc/fsr) fi
```

最初のエントリは、過去 7 日間アクセスされていない core または dead.letter という名前のファイルをすべて検索して削除する find(1) コマンドを実行します。

2 番目のエントリは、マウントされているファイルシステムの編成を向上させる fsr(1m) コマンドを実行します。

find コマンドは、すべてのローカル・ファイルシステムで毎晩実行されます。CXFS ファイルシステムはクラスタ内のすべてのノードでローカルと見なされるので、ノードが同じファイルシステムに同時にアクセスしようとすると、ファイルシステム・アクティビティが過剰に生成される場合があります。このため、以下の順序を使用して、1 つを除くすべてのノードでこれらの crontab エントリを無効にするか、または変更します。

1. root としてログインします。
2. vi などの好みのエディタを定義します。

   # **setenv EDITOR vi**

3. crontab ファイルを編集します。

   # **crontab -e**

4. find 行および fsr 行をコメント・アウトまたは削除します。

fsr コマンドは、メタデータ・サーバでのみ実行できるため、crontab ファイル内に残しておいても害はありませんが、実行されない可能性があります。

システム容量の効果的な使用

メタデータ・サーバとクライアントの接続が切断されないようにするには、メタデータ・サーバや、クラスタ内のノードに接続されているプライベート・サーバを購読しすぎないでください。空きメモリの量が不十分な場合、ノードでハートビートに遅れが発生することがあり、その結果、ノードは CXFS メンバーシップから除外されます。

システムでのメモリの空き容量を監視するには、`osview`(1) ツールを使用します。

193 ページの「論理スワップ・スペース不足」も参照してください。

ノード ID またはクラスタ ID を変更する前の再起動

ノード ID またはクラスタ ID を再定義する場合は、まず再起動しなければなりません。ここでの問題は、カーネルがまだ古い値を持っているために、CXFS メンバーシップの形成が妨げられる点です。ただし、最初に再起動すると元の値がクリアされるので、ノード ID またはクラスタ ID を再定義できます。

したがって、ノードで `cdbreinit` を使用してクラスタ設定データベースを再作成する場合は、ノード ID またはクラスタ ID を変更する前に再起動しなければなりません。212 ページの「クラスタ設定データベースの再作成」を参照してください。

使用されていないノードの削除

しばらくの間ノードを停止する場合は、クラスタおよびプールからそのノードを削除して、`fs2d` メンバーシップおよび CXFS メンバーシップの上限量の問題を回避します。以下の節を参照してください。

- 96 ページの「GUI を使ったクラスタ定義の変更」
- 124 ページの「CLI を使ったクラスタの変更およびファイルシステムの定義とマウント」
- 117 ページの「CLI を使ったノードの削除」

一般的な問題

以下は、一般的な問題とその解決方法です。

ファイルシステムにアクセスできない

ファイルシステムにアクセスできない場合は、以下をチェックします。

- ファイルシステムが有効になっているか。GUI、clconf_info コマンド、および cluster_status コマンドをチェックします。
- マウント・エラーが発生したか。

GUI が実行されない

GUI が実行されない場合は、以下をチェックします。

- ライセンスが正しくインストールされているか。66 ページの「ライセンスの確認」、195 ページの「ライセンス・エラー」、および47 ページの「必要な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・デーモンは実行されているか。66 ページの「クラスタ・デーモンが実行されていることの確認」を参照してください。
- tcpmux サービスおよび tcpmux/sgi_sysadm サービスが /etc/inetd.conf ファイル内で有効になっているか。
- inetd ラッパまたは tcp ラッパが干渉しているか。これは、connection refused メッセージまたは login failed メッセージによって示される場合があります。

ログ・ファイルがディスク容量を消費しすぎる

ログ・ファイルがディスク容量を消費しすぎる場合は、ログ・ファイルをローテーションします。156 ページの「ログ・ファイル管理」を参照してください。また、冗長性の低いログ・レベルを選択することもできます。以下を参照してください。

- 54 ページの「/etc/config/cad.options の設定」
- 55 ページの「/etc/config/fs2d.options の設定」
- 100 ページの「GUI を使ったログ・グループの設定」

ノードを定義できない

ノードを定義できない場合は、ホスト名解決問題が存在している可能性があります。52 ページの「ホスト名解決: /etc/sys_id、/etc/hosts、/etc/nsswitch.conf の設定」を参照してください。

システムがハングする

以下に、システムがハングする原因を示します。

- ディスク・ドライブがオーバーランした。
- ハートビートが切断された。この場合、withdrawl of node を示すメッセージが表示されます。
- 最後の手段として、システムのノンマスカブル割込み (NMI: non-maskable interrupt) を行った後、日本 SGI まで連絡してください。メモリのイメージを保存して後で解析できるよう、NMI はカーネルにノードをパニックさせるよう命令します。詳細については、ノードに対するオーナーのガイドを参照してください。

  vmcore.#.comp、unix.#、/var/adm/SYSLOG、およびクラスタ・ログ・ファイルを使用可能にします。

ノードは検出されるが、メンバーシップに設定されない

ノードが /var/adm/SYSLOG で検出されるのに Membership delivered メッセージを受取らない場合は、ネットワークの問題が存在する可能性があります。52 ページの「システム・ファイルの設定」を参照してください。

セル ID カウントと“ Membership Delivered” メッセージ

/var/adm/SYSLOG ファイルの Membership delivered メッセージには、新しい CXFS メンバーシップのメンバーであるノードのセル ID のリストが含まれます。

セル ID の後にはメンバーシップ・バージョンを表す数字が続き、ノードがメンバーシップに設定されてからメンバーシップが変更された回数を示します。

/var/adm/SYSLOG に Membership delivered メッセージが頻繁に記録される場合は、ネットワークの問題を示している可能性があります。

- 安定していてメンバーシップに残っているノードのメンバーシップ・バージョン番号は大きくなります。
- 問題のあるノードでは、メッセージがないか、またはメンバーシップ・バージョン番号が小さくなります。

52 ページの「システム・ファイルの設定」を参照してください。

ログインできない

ノードにログインできない場合は、現在のノードがその他のノードの .rhosts ファイルにリストされていれば、以下のいずれかのコマンドを使用できます。

```
# rsh hostname ksh -i
# rsh hostname csh -i
```

エラー・メッセージの理解

この節では、表示されることがあるいくつかのエラー・メッセージについて説明します。通常、メッセージの例は、まずタイプ順にリストされた後、メッセージ ID またはテキストのアルファベット順でリストされています。

以下の節が含まれます。

- 「通常メッセージ」
- 193 ページの「clconfd デーモンが異常終了する場合」
- 193 ページの「論理スワップ・スペース不足」
- 194 ページの「クラスタ名 ID 欠如エラー」
- 194 ページの「CXFS メンバーシップの取消し」
- 195 ページの「ライセンス・エラー」
- 195 ページの「IP アドレス・エラー」
- 196 ページの「SYSLOG エラー」
- 203 ページの「ログ・ファイル・エラー」

通常メッセージ

以下のメッセージが表示される場合があります。これらは通常のメッセージで、問題を示すものではありません。

```
NOTICE: Error reading mesg header 4 channel 1 cell 2
```

ノード 2 との接続用の MEMBERSHIP メッセージ・チャネル（チャネル 1。チャネル 0 は CXFS および XVM データのメイン・チャネルです）のエラー番号 4 (EINTR)。EINTR は、このメッセージ・チャネルが意図的に切断されていることを示すもので、それ自体にエラーがあることを示すわけではありません。その他のエラー・番号は実際のエラーで、ローカル・ノードはその他のノードに異常が発生したことを宣言します。これは情報メッセージです。修正処置は必要ありません。

```
NOTICE: Membership delivered.  Membership contains 0(21) 1(12) cells
```

ノード 0 およびノード 1 が CXFS メンバーシップに含まれています。ノード 0 は最後の 21 個の CXFS メンバーシップに含まれ、ノード 1 は最後の 12 個のメンバーシップに含まれます。ノードが上限量に追加されたり上限量から削除されると、そのたびにメンバーシップが形成されます。この数字は、上限量変更プロセスの間にメンバーシップが確認されたときにも増やされるので、連続する数字にはなりません。これは情報メッセージです。修正処置は必要ありません。

```
NOTICE: Resetting cells 0x4
```

この数字は、リセットが要求されたノード番号のビットマスクです。この場合、0x4 はノード 2 と同じです。This is an informative message; no corrective action is required.

```
CI_FAILURE, Cell 1 Machine cxfs1:  server has no information about a
machine that has reset capabilities for this machine
```

このノードにはリセット・メカニズムが提供されていないので、異常が発生しても自動的にリセットされません。リセット機能がない場合、このメッセージは無視してかまいません。ただし、CXFS クラスタ、特に重み付けされた 2 ノード・クラスタでは、リセット回線が強く推奨されています。IRIS FailSafe ではリセット回線は必須です。

```
NOTICE: Error reading mesg header 4 channel 1 cell 2
```

mesg header 4 というテキストは、これが単なる情報メッセージではないことを示します。

```
clconfd[16574]:  <<CI> E config 2> CI_ERR_NOTFOUND, Error reading CMS
status for machine tango, assuming machine is FailSafe-disabled in
cluster twango.
```

これは、クラスタが CXFS だけであり、IRIS FailSafe は使用されていないことを示します。

```
CI_CLCONFERR_INIT in ep_name() not binding socket
```

このメッセージはデーモンの開始前に表示されます。

```
clconfd[16574]:  <<CI> E clconf 0> CI_CLCONFERR_INIT, in ep_name():  not
binding socket
```

clconfd メッセージはデーモンの開始中に表示されます。

```
date <I0 clconfd clconf 610:0 clconfd_client.c:84> client registration:
clconfinfo, id 9119
date<I0 clconfd clconf 610:0 clconfd_service.c:781> sending reply
configuration and membership msg to client:  clconfinfo, id 9119
date <I0 clconfd clconf 610:0 clconfd_client.c:96> client
un-registration:  clconfinfo, id 9119
```

これらのメッセージは、clcon_info コマンドを実行する場合に発行されます。clconf_info コマンドは、まず clconfd にクライアントとして登録されます。続いて、その設定とメンバーシップのステータスの要求に対する応答メッセージを受信し、完了すると登録を解除します。

```
date <I0 clconfd clconf 610:0 clconfd_service.c:781 sending reply
configuration and membership msg to client:  cad, id 602
```

このメッセージは、cad デーモンがステータスについて定期的に clconfd をポーリングしていることを示します。clconf_info と異なり、cad は毎回登録と登録解除は行いません。これは、cad はデーモンであり、各要求の後に終了しないからです。cad または clconfd が再開されたときは、cad の登録/登録解除メッセージだけが表示されます。

clconfd デーモンが異常終了する場合

clconfd デーモンが開始直後に異常終了する場合は、CXFS ライセンスが適切にインストールされていないことを意味します。関連するエラー・メッセージについての詳細は、195 ページの「ライセンス・エラー」を参照してください。

CXFS を使用するには、各ノードにライセンスをインストールしなければなりません。システムの CPU の数を増やすと、新しいライセンスが必要になることがあります。47 ページの「必要なソフトウェアのインストール」を参照してください。

論理スワップ・スペース不足

以下の /var/adm/SYSLOG メッセージの例は、過剰に購読されているシステムを示します。

```
ALERT: inetd [164] - out of logical swap space during fork while
allocating uarea - see swap(1M)
Availsmem 8207 availrmem 427 rlx freemem 10, real freemem 9
```

188 ページの「システム容量の効果的な使用」を参照してください。

この場合は、クラスタ・デーモンでメモリ・リークも発生している可能性があり、クラスタ・デーモンの再開が必要なことがあります。

```
# /etc/init.d/cluster restart
```

クラスタ名 ID 欠如エラー

例:

```
Mar  1 15:06:18 5A:nt-test-07 unix: NOTICE: Physvol (name cip4) has no
CLUSTER name id: set to ""
```

このメッセージには以下の意味があります。

- XVM physvol としてラベルが付けられているディスクは、IRIX 6.5.6 でラベルが付けられた可能性があります。その後、システムは、新しいバージョンの VXM ラベル形式を使用する新しいバージョンにアップグレードされています。これは問題を示すわけではありません。
- XVM が、XVM クラスタの physvol ラベルが付いたこれらのディスクを検出したときに、クラスタ名がまだ設定されていませんでした。これは、このホストでノード/クラスタの初期化が完了する前に XVM がディスク・インベントリの初期スキャンを実行したときの通常の出力です。

  このメッセージは、XVM クラスタの physvol ラベルの付いたディスクは XVM によって認識されていますが、このノードがまだ CXFS メンバーシップに設定されていないことを示します。したがって、クラスタ名は空 ("") になっています。

  ノードまたはクラスタの初期化時に、XVM はディスク・インベントリを再スキャンして、XVM クラスタの physvol ラベルを検索します。この時点では、このホストに対してクラスタ名が設定されている必要があります。ノード/クラスタの初期化後にクラスタ名が空の場合は、クラスタの初期化に問題があることを示します。

  このディスクの XVM 要素の設定を初めて変更すると、ラベルが更新されて新しいラベル形式に切替えられ、これらの通知は表示されなくなります。

  詳細については、36 ページの「CXFS クラスタ表示」を参照してください。XVM についての詳細は、『XVM Volume Manager Administrator's Guide』を参照してください。

CXFS メンバーシップの取消し

/var/adm/SYSLOG の以下のメッセージは、カーネルによって実行された CXFS メンバーシップの取消しを示します。

```
Membership lost - withdrawing from cluster
```

この場合は、ローカル・ノードの CXFS メンバーシップをアクティブに許可する必要があります。105 ページの「GUI を使ったローカル・ノードのメンバーシップ許可」または140 ページの「CLI を使ったローカル・ノードのメンバーシップ許可」を参照してください。

ライセンス・エラー

/var/cluster/ha/log/clconf_hostname ログ・ファイルに以下のメッセージが記録されている場合は、CXFS ライセンスが適切にインストールされていなかったことを示します。

```
CXFS not properly licensed for this host.  Run
               '/usr/cluster/bin/cxfslicense -d'
       for detailed failure information.
```

clconfd デーモンが開始直後に異常終了する場合は、このエラーが存在します。

CXFS を使用するには、各ノードにライセンスをインストールしなければなりません。47 ページの「必要なソフトウェアのインストール」を参照してください。

IP アドレス・エラー

サイトにクラスタ ID 番号の競合がある場合は、以下のようなエラーが表示されます。

```
WARNING: mtcp ignoring  alive message from 1 with wrong ip addr 128.162.89.34
WARNING: mtcp ignoring  alive message from 0 with wrong ip addr 128.162.89.33
```

クラスタ ID 番号は一意でなければなりません。この問題を解決するには、クラスタ ID 番号を一意にします。

特定のノードに以前の設定の古い情報があるときに、クラスタ設定を再定義して CXFS サービスを開始すると、このエラーが発生する可能性があります。

問題を解決するには、まず183 ページの「残っているクラスタの削除」の手順を試してみてください。これらの手順で問題が解決されない場合は、古い情報を持つノードを再起動します。古い情報を持っているノードの判別方法として、古いノードではすべてのノードに問題があることが示されますが、最新のノードでは古いノードに問題があることしか示されないことを利用できます。また、古いノードの /var/cluster/ha/log/clconfd_ ログ・ファイルには、SGI_CMS_CONFIG_ID 異常に関するエラー・メッセージも記録されます。

エラー・メッセージが多すぎて古いノードを認識できない場合は、すべてのノードを再起動してください。

SYSLOG エラー

CXFS は、通常の処理と重大なエラーの両方を、/var/adm/SYSLOG のほかに各ログ・グループの個々のログにも記録します。

一般的に、/var/adm/SYSLOG ファイルのエラーは以下の形式になります。

timestamp priority_&_facility : hostname process[ID]: <internal_info> CODE message_text

例:

```
Sep  7 11:12:59 6X:cxfs0 cli[5830]: < E clconf 0> CI_IPCERR_NOSERVER, clconf
ipc: ipcclnt_connect() failed, file /var/cluster/ha/comm/clconfd-ipc_cxfs0
```

表8-1 に、この SYSLOG メッセージの各部分を示します。

表8-1 SYSLOG エラー・メッセージの形式

内容	部分	意味
Sep 7 11:12:59	タイムスタンプ	9 月 7 日午前 11 時 12 分
6X	機能とラベル	6X は情報メッセージを示します。syslogd(1M) および /usr/include/sys/syslog.h ファイルを参照してください。
cxfs0	ノード名 (Node name)	論理名が cxfs0 のノードは、プロセスが実行されているノードです。
cli[5830]	プロセス[ID]	メッセージを送信するプロセスは cli で、プロセス ID 番号は 5830 です。
<CI>E clconf 0	内部情報: メッセージ・ソース、ログ・サブシステム、およびスレッド ID	このメッセージはクラスタ・インフラストラクチャ (CI: Cluster Infrastructure) からのものです。E は、エラーであることを示します。clconf コマンドはログ・サブシステムです。0 は、マルチスレッド化されていないことを示します。

内容	部分	意味
CI_IPCERR_NOSERVER, clconf ipc	内部エラー・コード	メッセージのタイプに関する情報。この場合、サーバを示すメッセージはありません。通常メッセージの場合、エラー・コードは出力されません。
ipcclnt_connect() failed, file /var/cluster/ha/comm/clconfd-ipc_cxfs0	メッセージ・テキスト	clconfd-ipc_cxfs0 ファイルの接続に失敗しました。

以下の節では、メッセージ ID とテキストだけを示します。

cli エラー

cli メッセージについては、コマンドからの最後のメッセージ(CLI private command failed で始まります)だけに意味があります。その他の cli メッセージはすべて無視してかまいません。

以下に、cli デーモンのエラーの例を示します。

CI_ERR_INVAL, CLI private command: failed (Machine (cxfs0) exists.)

論理名 cxfs0 を使用して新しいノード定義を作成しようとしましたが、そのノード名はすでにクラスタ設定データベースに存在します。別の名前を選択してください。

CI_ERR_INVAL, CLI private command: failed (IP address (128.162.89.33) specified for control network is cxfs0 is assigned to control network of machine (cxfs0).)

ノード cxfs0 の2 つの異なるコントロール・ネットワークに対して同じ IP アドレスを指定しました。別の IP アドレスを使用してください。

CI_FAILURE, CLI private command: failed (Unable to validate hostname of machine (cxfs0) being modified.)

cxfs0 名の DNS 解決に失敗しました。この問題を解決するには、すべてのノードの /etc/hosts に cxfs0 のエントリを追加します。

CI_IPCERR_NOPULSE, CLI private command: failed (Cluster state is UNKNOWN.)

クラスタの状態が「不明(UNKNOWN)」で、コマンドを完了できませんでした。これは一時的なエラーです。ただし、エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

clconfd エラー

以下に、clconfd デーモンから送信されるエラーを示します。

```
CI_CONFERR_NOTFOUND, Could not access root node.
```

クラスタ設定データベースが存在しないかまたは壊れています。あるいはデータベース・デーモンが応答していません。データベースが存在することを確認してください。

エラーが表示される場合や、ダンプが空の場合は、データベースを再作成してください。詳細については、208 ページの「クラスタ設定データベースのクリア」を参照してください。

データベースが存在する場合は、クラスタ・デーモンを再開します。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

```
CI_ERR_NOTFOUND, Could not get Cellular status for local machine (cxfs1)
```

データベースが壊れているか、またはデータベースにアクセスできません。上記と同じアクションを実行してください。

```
CI_FAILURE, Call to open cdb for logging configuration when it is
already open.
```

これはソフトウェアの問題を示しており、デーモンを再開する必要があります。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

```
CI_FAILURE, Cell 1 Machine cxfs1:  server has no information about a
machine that has reset capabilities for this machine
```

このノードにはリセット・メカニズムが提供されていないので、異常が発生しても自動的にリセットされません。異常処理が適切に行われるようにするには、GUI または CLI を使用してノードの定義を変更し、リセット情報を追加します。84 ページの「GUI を使ったノードの定義」または113 ページの「CLI を使ったノードの変更」を参照してください。

```
CI_FAILURE, CMD(/sbin/umount -k /dev/xvm/bob1):  exited with status 1
(0x1)
```

/dev/xvm/bob1 ファイルシステムをアンマウントするときにエラーが発生しました。通常は、このメッセージの直前に umount(1M) コマンドからのメッセージが発行され、異常の理由についての詳細が提供されます。

```
CI_FAILURE, CMD(/sbin/clmount -o 'server_list=(cxfs0,cxfs1)'
/dev/xvm/bob2 /bob2):  exited with status 1 (0x1)
```

/dev/xvm/bob2 ファイルシステムをマウントするときにエラーが発生しました。通常は、このメッセージの直前に mount(1M) コマンドからのメッセージが発行され、異常の理由についての詳細が提供されます。

```
CI_FAILURE, CMD(/sbin/clmount -o 'server_list=(cxfs2,cxfs0)'
/dev/xvm/stripe4 /xvm/stripe4):  exited with status 1 (0x1)
```

最初に mkfs(1M) を実行せずにファイルシステムをマウントしようとしました。ファイルシステムをマウントする前に、mkfs(1M) を使用してファイルシステムを構築する必要があります。詳細については、mkfs(1M) のマン・ページを参照してください。

```
CI_FAILURE, Could not write newincarnation number to CDB, error = 9.
```

クラスタ設定データベースにアクセスできませんでした。操作を再試行してください。エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

問題が解決されない場合は、データベースをクリアして再起動し、データベースを再作成してください。208 ページの「クラスタ設定データベースのクリア」を参照してください。

```
CI_FAILURE, Exiting, monitoring agent should revive me.
```

このデーモンには新しいデータが必要です。自動的に再開されます。

```
CI_FAILURE, No node for client (3) of filesystem (/dev/xvm/bob1) on
(/bob1).
```

(このメッセージは何度も表示される場合があります。) クラスタ設定データベースに存在していないノードにファイルシステムがマウントされたままになっている可能性があります。以前にクラスタで使用されていて現在もファイルシステムをマウントしているノードを識別できる場合は、そのノードを再起動します。それ以外の場合は、クラスタ全体を再起動します。

```
CI_FAILURE, No node for server (-1) of filesystem (/dev/xvm/bob1) on
(/bob1).
```

(このメッセージは何度も表示される場合があります。) クラスタ設定データベースに存在していないノードにファイルシステムがマウントされたままになっている可能性があります。以前にクラスタで使用されていて現在もファイルシステムをマウントしているノードを識別できる場合は、そのノードを再起動します。それ以外の場合は、クラスタ全体を再起動します。

```
CI_ FAILURE, Node cxfs0:  SGI_CMS_HOST_ID(tcp,128.162.8 >9.33) error 149
(Operation already in progress)
```

カーネルはすでにこの情報を持っています。このメッセージは無視してかまいません。

```
CI_FAILURE, Unregistered from crs.
```

clconfd デーモンがリセット・デーモンに接続されていないので、異常ノードをリセットを処理できません。修正処置はありません。

```
CI_IPCERR_NOSERVER, Crs_register failed,will retry later.  Resetting not
possible yet.
```

clconfd デーモンがリセット・デーモンに接続できません。異常ノードのリセットを処理できません。リセット・デーモンのログ・ファイル (/var/cluster/ha/log/crsd_) でその他のエラー・メッセージがないかどうかをチェックしてください。

```
Clconfd is out of membership, will restart after notifying clients.
```

clconfd デーモンに、クラスタの現在の状態に関する十分な情報がありません。このデーモンは終了して、自動的に新しいデータで再開されます。

```
CMD(/sbin/clmount -o 'server_list=(cxfs2,cxfs0)' /dev/xvm/stripe4
/xvm/stripe4):  /dev/xvm/stripe4:  Invalid argument
```

最初に mkfs(1M) を実行せずにファイルシステムをマウントしようとしました。ファイルシステムをマウントする前に、mkfs(1M) を使用してファイルシステムを構築する必要があります。詳細については、mkfs(1M) のマン・ページを参照してください。

```
CMD(/sbin/clmount -o 'server_list=(cxfs0,cxfs1)' /dev/xvm/bob2 /bob2):
/dev/xvm/bob2:  Invalid argumentSep 9 14:12:43 6X:cxfs0 clconfd[345]:  <
E clconf 3> CI_FAILURE, CMD(/sbin/clmount -o 'server_list=(cxfs0,cxfs1)'
/dev/xvm/bob2 /bob2):  exited with status 1 (0x1)
```

最初のメッセージは clmount コマンド(内部 CXFS マウント・コマンド)からのもので、エラーを説明しています(無効な引数が発行されました)。2 番目のメッセージは、マウントに失敗したことを示します。

crsd エラー

以下に、crsd デーモンから送信されるエラーを示します。

```
CI_ERR_NOTFOUND, No logging entries found for group crsd, no logging
will take place - Database entry #global#logging#crsd not found.
```

クラスタ設定データベースに crsd ログ定義がありませんでした。これは、データベースを作成せずにクラスタ・プロセスを開始した場合に発生することがあります。212 ページの「クラスタ設定データベースの再作成」を参照してください。

```
CI_ERR_RETRY, Could not find machine listing.
```

crsd デーモンは、クラスタ設定データベースでローカル・ノードを見つけることができませんでした。ローカル・ノード定義をまだ作成していない場合、このメッセージは無視してかまいません。

```
CI_ERR_SYS:125, bind() failed.
```

/etc/services ファイルの sgi-crsd のポート番号が一意ではないか、またはこのファイルに sgi-crsd エントリがありません。このエントリの追加についての詳細は、54 ページの「/etc/services の設定」を参照してください。

```
CI_FAILURE, Entry for sgi-crsd is missing in /etc/services.
```

/etc/services ファイルに sgi-crsd エントリがありません。このエントリの追加についての詳細は、54 ページの「/etc/services の設定」を参照してください。

```
CI_FAILURE, Initialization failed, exiting.
```

このメッセージは、連続する複数のメッセージの最後に示されます。異常の原因を判別するには、このメッセージの前のメッセージを参照してください。

cmond エラー

以下に、cmond デーモンから送信されるエラーを示します。

```
Could not register for notification.cdb_error = 7
```

クラスタ・プロセスが開始されたときにクラスタ設定データベースが初期化されませんでした。212 ページの「クラスタ設定データベースの再作成」を参照してください。

```
Process clconfd:343 of group cluster_cx exited, status = 3.
```

clconfd プロセスがステータス 3 で終了しました。つまり、このプロセスは cmond によって再開されません。修正処置は必要ありません。

```
Process crsd:1790 of group cluster_control exited, status = 127
```

crsd プロセスがエラー（ゼロ以外）のステータスで終了しました。対応するデーモンのログでエラー・メッセージを検索してください。

fs2d エラー

以下に、fs2d デーモンから送信されるエラーを示します。

```
Error 9 writing CDB info attribute for node
#cluster#elaine#machines#cxfs2#Cellular#status
```

クラスタ設定データベースへの書込みの際に内部エラーが発生しました。操作を再試行してください。エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

問題が解決されない場合は、データベースをクリアして再起動し、データベースを再作成してください。208 ページの「クラスタ設定データベースのクリア」を参照してください。

```
Error 9 writing CDB string value for node
#cluster#elaine#machines#cxfs2#Cellular#status
```

クラスタ設定データベースへの書込みの際に内部エラーが発生しました。操作を再試行してください。エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

問題が解決されない場合は、データベースをクリアして再起動し、データベースを再作成してください。208 ページの「クラスタ設定データベースのクリア」を参照してください。

```
Failed to update CDB for node
#cluster#elaine#Cellular#FileSystems#fs1#FSStatus
```

クラスタ設定データベースへの書込みの際に内部エラーが発生しました。操作を再試行してください。エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

問題が解決されない場合は、データベースをクリアして再起動し、データベースを再作成してください。208 ページの「クラスタ設定データベースのクリア」を参照してください。

```
Failed to update CDB for node
#cluster#elaine#machines#cxfs2#Cellular#status
```

クラスタ設定データベースへの書込みの際に内部エラーが発生しました。操作を再試行してください。エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

問題が解決されない場合は、データベースをクリアして再起動し、データベースを再作成してください。208 ページの「クラスタ設定データベースのクリア」を参照してください。

```
Machine 101 machine_sync failed with lock_timeout error
```

fs2d デーモンがクラスタ設定データベースを同期できず、sync プロセスがタイムアウトしました。この操作は、fs2d によって自動的に再試行されます。

```
ALERT: CXFS Recovery:  Cell 0:  Server Cell 2 Died, Recovering
```

サーバ (cell 2) が異常終了したため、現在、システムがファイルシステムを回復しています。

一般メッセージ

```
CI_CONFERR_NOTFOUND, Logging configuration error:  could not read
cluster database /var/cluster/cdb/cdb.db, cdb error = 3.
```

クラスタ設定データベースが初期化されていません。212 ページの「クラスタ設定データベースの再作成」を参照してください。

```
WARNING: Error receiving messages from cell 2 tcpchannel 1
```

CXFS メンバーシップ・チャネル (チャネル 1。チャネル 0 は CXFS および XVM データのメイン・メッセージ・チャネルです) にエラーがあります。これは、チャネルを切断した結果か、またはノード (この場合は ID が 2 のノード) のエラーの可能性があります。修正処置はありません。

ログ・ファイル・エラー

CXFS では、各 CXFS デーモンのログを維持します。これらのログのカスタマイズについての詳細は、99 ページの「GUI を使ったログの設定」を参照してください。

以下に、ログ・メッセージの形式を示します。

*daemon*_log *timestamp internal_process: message_text*

例:

```
cad_log:Thu Sep  2 17:25:06.092  cclconf_poll_clconfd: clconf_poll failed with error CI_IPCERR_NOPULSE
```

表8-2 に、このメッセージの各部分を示します。

表8-2 ログ・エラー・メッセージの形式

内容	部分	意味
cad_log	タイムスタンプ	このメッセージは cad デーモンに関係しています。
Sep 2 17:25:06.092	タイムスタンプおよびプロセス ID	9 月 2 日午後 5 時 25 分、プロセス ID 92
cclconf_poll_clconfd	内部プロセス情報	内部プロセス情報
clconf_poll failed with error CI_IPCERR_NOPULSE	メッセージ・テキスト	クラスタのステータスの更新を取得するために clconfd デーモンと通信できませんでした。

cad メッセージ

以下に、/var/cluster/ha/log/cad_log メッセージの例を示します。

ccacdb_cam_open: failed to open connection to CAM server error 4

内部メッセージ。cad の動作は自動的に再開されるので無視してかまいません。

ccamail_cam_open: failed to open connection to CAM server error 4

内部メッセージ。cad の動作は自動的に再開されるので無視してかまいません。

ccicdb_cam_open: failed to open connection to CAM server error 4

内部メッセージ。cad の動作は自動的に再開されるので無視してかまいません。

cclconf_cam_open: failed to open connection to CAM server error 4

内部メッセージ。cad の動作は自動的に再開されるので無視してかまいません。

`cclconf_poll_clconfd: clconf_poll failed with error CI_IPCERR_NOCONN`

clconfd デーモンが実行されていないか、または外部の要求に応答していません。エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

`cclconf_poll_clconfd: clconf_poll failed with error CI_IPCERR_NOPULSE`

クラスタのステータスの更新を取得するために clconfd デーモンと通信できませんでした。エラーが解決されない場合は、クラスタ・デーモンを停止して再開してください。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

`cclconf_poll_clconfd: clconf_poll failed with error CI_CLCONFERR_LONELY`

clconfd デーモンは、クラスタの正確な状態を提供するための十分な情報を持っていません。このデーモンは新しいデータを使用して自動的に再開され、サービスを再開します。

`csrm_cam_open: failed to open connection to CAM server error 4`

内部メッセージ。cad の動作は自動的に再開されるので無視してかまいません。

`Could not execute notification cmd. system() failed. Error: No child processes`

cad がプロセスをフォークできなかったため、メール・メッセージは送信されませんでした。クラスタ・デーモンを停止して再開します。212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」を参照してください。

`error 3 sending event notification to client 0x000000021010f078`

GUI プロセスがクリーンアップせずに終了しました。

`error 8 sending event notification to client 0x000000031010f138`

GUI プロセスがクリーンアップせずに終了しました。

cli メッセージ

以下に、/var/cluster/ha/log/cli_*hostname* メッセージの例を示します。

`CI_CONFERR_NOTFOUND, No machines found in the CDB.`

ローカル・ノードは設定データベースに定義されていません。

`CI_ERR_INVAL, Cluster (bob) not defined`

bob というクラスタは設定データベースに存在しません。

```
CI_ERR_INVAL, CLI private command:  failed (Cluster (bob) not defined)
```

bob というクラスタは設定データベースに存在しません。

```
CI_IPCERR_AGAIN, ipcclnt_connect():  file
/var/cluster/ha/comm/clconfd-ipc_cxfs0 lock failed - Permission denied
```

CLI が root 以外のログインによって呼出されました。root としてログインしたときは、cmgr(1M) のみ使用してください。

```
CI_IPCERR_NOPULSE, CLI private command:  failed (Cluster state is
UNKNOWN.)
```

クラスタの状態を判別できませんでした。clconfd(1M) デーモンが実行されているかどうかをチェックしてください。

```
CI_IPCERR_NOPULSE, ipcclnt_pulse_internal():  server failed to pulse
```

クラスタの状態を判別できませんでした。clconfd(1M) デーモンが実行されているかどうかをチェックしてください。

```
CI_IPCERR_NOSERVER, clconf ipc:  ipcclnt_connect() failed, file
/var/cluster/ha/comm/clconfd-ipc_cxfs0
```

ローカル・ノード (cxfs0) は設定データベースに定義されていません。

```
CI_IPCERR_NOSERVER, Connection file
/var/cluster/ha/comm/clconfd-ipc_cxfs0 not present.
```

ローカル・ノード (cxfs0) は設定データベースに定義されていません。

crsd エラー

以下に、/var/cluster/ha/log/crsd_*hostname* メッセージの例を示します。

```
CI_CONFERR_INVAL, Nodeid -1 is invalid.
I_CONFERR_INVAL, Error from ci_security_init().
CI_ERR_SYS:125, bind() failed.
CI_ERR_SYS:125, Initialization failed, exiting.
CI_ERR_NOTFOUND, Nodeid does not have a value.
CI_CONFERR_INVAL, Nodeid -1 is invalid.
```

これらの各メッセージが表示された場合は、ノード定義にノード ID が指定されていなかったか、または設定データベースにノード定義を作成したときにそのノードでクラスタ・プロセスが実行されていませんでした。これは、必要なときにオプションの情報を使用できないことを示す警告です。

```
CI_ERR_NOTFOUND, SystemController information for node cxfs2 not found,
requests will be ignored.
```

ノード cxfs2 に対してシステム・コントローラ情報（オプションの情報）が指定されていませんでした。ノード定義を変更して、ノード cxfs2 に対してシステム・コントローラ情報を指定します。これは、必要なときにオプションの情報を使用できないことを示す警告です。この情報がないと、異常が発生してもノードはリセットされません。このため、クラスタが異常から正常に回復できなくなる可能性があります。

```
CI_ERR_NOTFOUND, SystemController information for node cxfs0 not found,
requests will be ignored.
```

ノード ID が 101 のノードに対してノード定義で指定されているオーナー・ノードが定義されていません。オーナー・ノードを定義してください。

```
CI_CRSERR_NOTFOUND, Reset request 0x10087d48 received for node 101, but
its owner node does not exist.
```

ノード ID が 101 のノードに対してノード定義で指定されているオーナー・ノードが定義されていません。オーナー・ノードを定義してください。

fs2d エラー

以下に、/var/cluster/ha/log/fs2d_*hostname* メッセージの例を示します。

```
Failed to copy global CDB to node cxfs1 (1), error 4
```

ローカル・ノードとノード cxfs2 の間に通信上の問題があります。これらの 2 つのノードのコントロール・ネットワークをチェックしてください。

```
Communication failure send new quorum to machine cxfs2 (102) (error
6003)
```

ローカル・ノードとノード cxfs2 の間に通信上の問題があります。これらの 2 つのノードのコントロール・ネットワークをチェックしてください。

```
Failed to copy CDB transaction to node cxfs2 (1)
```

ローカル・ノードとノード cxfs2 の間に通信上の問題があります。これらの 2 つのノードのコントロール・ネットワークをチェックしてください。

```
Outgoing RPC to hostname :  NULL
```

このメッセージが表示される場合は、RPC の設定をチェックします。詳細については、rpcinfo(1m)、rpcinfo(1m)、および portmap(1m) のマン・ページを参照してください。

修正処置

この節では、以下の修正処置について説明します。

- 208 ページの「CXFS サービスの再開」
- 208 ページの「クラスタ設定データベースのクリア」
- 210 ページの「2 ノード・クラスタの回復」
- 209 ページの「再起動」
- 211 ページの「クラスタに再設定しない再起動」
- 212 ページの「クラスタ・インフラストラクチャ・デーモンの停止と再開」
- 212 ページの「クラスタ設定データベースの再作成」

CXFS サービスの再開

再起動後に CXFS サービスが再開されない場合は、GUI または CLI の停止機能を使用して開始フラグがオフにされている可能性があります。この場合、`/etc/init.d/cluster start` を発行してもサービスは再開されません。CXFS サービスを開始する必要があります。GUI または CLI を使用してサービスを再開すると、以降の再起動時に CXFS サービスが再開されるように設定されます。

詳細については、98 ページの「GUI を使った CXFS サービスの開始」または135 ページの「CLI を使った CXFS サービスの開始」を参照してください。

クラスタ設定データベースのクリア

クラスタのすべてのノードのクラスタ設定データベースをクリアするには、以下の手順に従います。次の手順に進む前に、各ノードで各手順を完了してください。

注意 : この手順を実行すると、すべての設定情報が削除されます。

1. すべてのノードで以下のコマンドを入力します。

   # **/usr/cluster/bin/cmgr -c 'admin cxfs_stop'**

2. すべてのノードで以下のコマンドを入力します。

   # **/etc/init.d/cluster stop**

注意: 任意のノードに対して手順 3 を実行する前に、各ノードで手順 1 と 2 を完了してください。

3. すべてのノードで以下のコマンドを入力します。

```
# /usr/cluster/bin/cdbreinit
```

メモ: 188 ページの「ノード ID またはクラスタ ID を変更する前の再起動」も参照してください。

4. すべてのノードで以下のコマンドを入力します。

```
# /etc/init.d/cluster start
```

5. すべてのノードで以下のコマンドを入力します。

```
# /usr/cluster/bin/cmgr -c 'admin cxfs_start'
```

カーネル内の古いと思われるクラスタ設定を取除くには、183 ページの「残っているクラスタの削除」を参照してください。必要に応じてノードを再起動します。

再起動

すべてのノードで個別に以下のコマンドを入力して、クラスタを再起動します。

```
# reboot
```

メモ: ノードの再起動時に常に CXFS サービスを再開したい場合は、GUI または CLI を使用して CXFS サービスを開始します。GUI についての詳細は98 ページの「GUI を使った CXFS サービスの開始」、CLI についての詳細は135 ページの「CLI を使った CXFS サービスの開始」をそれぞれ参照してください。

以下に、再起動が必要になることがある状況を示します。

- vnode が使用中であるために一部のクライアントがファイルシステムをマウントできない場合や、ノードをリセットしても問題が解決されない場合は、クラスタ内のすべてのノードの再起動が必要になる場合があります。
- 10 分以内に回復アクティビティが行われない場合は、該当するノードの再起動が必要になる場合があります。

2 ノード・クラスタの回復

以下のような状況であるとします。

1. 以下の 2 つのノードを持つ clusterA という名前のクラスタがあり、タイブレーカー・ノードはありません。
   - node1
   - node2
2. node1 が停止し、しばらく停止したままになります。
3. node2 が回復し、clusterA は引続き動作しています。

メモ: 初期 CXFS メンバーシップの形成後、既存のクラスタは、重み付けされた残りのノードの 50% まで低下する場合があります。詳細については、13 ページの「メンバーシップ上限量」を参照してください。

---

4. node2 が停止したために、clusterA に異常が発生します。
5. node2 が復帰します。ただし、クラスタの初期化には以下のいずれかが必要なので、clusterA は形成できません。
   - **50%** を超える重み付けされたノード
   - 重み付けされたノードの 50%。これらの 1 つはタイブレーカーです。

node2 が単独でクラスタを形成できるようにするには、以下を実行する必要があります。

1. GUI または CLI を使用して、node2 をタイブレーカー・ノードに設定します。
   - 99 ページの「GUI を使ったタイブレーカー・ノードの設定」を参照してください。
   - 136 ページの「CLI を使ったタイブレーカー・ノードの設定」を参照してください。
2. node2 の CXFS メンバーシップを取消します。
   - 105 ページの「GUI を使ったローカル・ノードのメンバーシップ取消し」を参照してください。
   - CLI で以下のコマンドを入力します。

     ```
     cmgr> admin cxfs_stop
     ```

     140 ページの「CLI を使ったローカル・ノードのメンバーシップ取消し」を参照してください。
3. node2 の CXFS メンバーシップを許可します。

- 105 ページの「GUI を使ったローカル・ノードのメンバーシップ許可」を参照してください。
- CLI で以下のコマンドを入力します。

  ```
  cmgr> admin cxfs_start
  ```

  140 ページの「CLI を使ったローカル・ノードのメンバーシップ許可」を参照してください。

4. タイブレーカー・ノード機能の設定を解除します。

注意: 重み付けされた 2 ノード・クラスタ内のタイブレーカー・ノードに異常が発生したり、管理者が CXFS サービスを停止した場合は、その他のノードによって強制シャットダウンが実行され、CXFS ファイルシステムはすべてアンマウントされます。データを確実に保護するため、特に重み付けされた 2 ノード・クラスタでは、CXFS システムに対してハードウェア・リセットを使用することが推奨されています。IRIS FailSafe ではリセットは必須です。

GUI または CLI を使用します。

- 99 ページの「GUI を使ったタイブレーカー・ノードの設定」
- 136 ページの「CLI を使ったタイブレーカー・ノードの設定」

メモ: node1 は現在もクラスタに設定されているので、クラスタはノードが停止していても通信を試みます。したがって、CXFS メンバーシップの形成とファイルシステムのマウントに多少時間がかかる場合があります。

クラスタに再設定しない再起動

chkconfig(1M) の cluster フラグは、ほかのクラスタ管理デーモン(主に複製されたクラスタ設定データベース)を制御します。これがオフの場合、データベース・デーモンは次回再起動されるときに開始されず、ほかのノードでクラスタ設定を変更してもデータベースのローカル・コピーは更新されません。これにより、後で問題が発生することがあります。特に、ノードの過半数でデータベース・デーモンが実行されていない場合は、問題が発生する可能性が高くなります。

クラスタ・デーモンが深刻な問題を引起していてマシンを再起動できない場合は、シングルユーザ・モードで起動して cluster フラグをオフにし、マルチユーザ・モードで再起動することで、ノードを回復できます。

```
# init 1
# /etc/chkconfig cluster off
# init 2
```

クラスタ・インフラストラクチャ・デーモンの停止と再開

クラスタ・インフラストラクチャ・デーモンを停止して再開するには、以下のコマンドを入力します。

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

これらのコマンドは、クラスタ・インフラストラクチャ・デーモンだけを制御します。

注意: クラスタ・インフラストラクチャ・デーモンを停止すると、ノードはデータベースの更新を受付けず、カーネル設定を更新しなくなります。これにより、好ましくない影響が出る場合があります。ほとんどの場合、インフラストラクチャ・デーモンは常に実行したままにすることが推奨されています。これらのコマンドは、指示を受けた場合にのみ使用してください。

208 ページの「CXFS サービスの再開」も参照してください。デーモンについての一般情報は、20 ページの「デーモン」を参照してください。

クラスタ設定データベースの再作成

初期クラスタ設定データベースを再作成するには、以下の手順に従います。

1. `fs2d` メンバーシップ上限量が正常なデータベースを持つノードによって保持されるようにして、不正なデータベースが伝えられるのを避けます。
2. 以下のコマンドを入力します。

   ```
   # /usr/cluster/bin/cdbreinit
   ```

メモ: 188 ページの「ノード ID またはクラスタ ID を変更する前の再起動」も参照してください。

初期設定チェックリスト

以下は、CXFS システムのインストールと設定の際に実行しなければならない手順のチェックリストです。

注意: CXFS は複雑な製品です。CXFS が最適にインストールおよび設定されるよう、必ず日本 SGI から初期セットアップ・サービスをご購入ください。

このチェックリストは、カスタマが直接使用することを目的としておらず、リファレンスとして提供されています。つまり、補足資料としての使用のみを目的としています。これらの手順を完了する前に、このマニュアル全体、特に173 ページの 第8章「トラブルシューティング」をよくお読みください。

[ ] サイトでのアプリケーションの用途、ストレージの設定、および I/O パターンを理解します（一部のアプリケーションでは CXFS の効果はありません。1 ページの「XFS と CXFS の比較」を参照してください）。

[ ] SAN ハードウェアを接続します。以下のドキュメントを参照してください（日本 SGI の営業担当員のみ入手可能です）。

- 『SGI. Storage Area Network Installation Instructions』
- 『SGI TP9400 RAID Installation and Upgrades Guide』
- 『SGI Total Performance 9100 Storage System Installation Instructions』
- 『IRIS FailSafe Version 2 Installation and Maintenance Instructions』（主にハードウェア設定情報が適用されます）

[ ] プライベート・ネットワークがあるかどうか。これは必要条件です。パフォーマンスおよび制御の点から、ハブではなくスイッチが推奨されます。

[ ] リセット回線があるかどうか。CXFS ではリセット機能はオプションですが、データの整合性を保つために強く推奨されています。リセットは、重み付けされた 2 ノード・クラスタおよび IRIS FailSafe では必須です。11 ページの「ハードウェア・コンポーネント」および17 ページの「ハードウェア・リセット」を参照してください。

[ ] このマニュアルを読みます。

[ ] ネットワークが使用可能であることを確認します。47 ページの 第2章「CXFS ソフトウェアの初期インストールとシステムの準備」を参照してください。

[ ] CXFS ソフトウェアをインストールします。47 ページの「必要なソフトウェアのインストール」を参照してください。

[ ] 必要に応じて設定ファイルを変更します。/etc/sys_id ファイルの内容が /etc/hosts ファイルのプライマリ名と一致することを確認します。52 ページの「システム・ファイルの設定」を参照してください。

[ ] その他のシステム設定手順を実行し、再起動します。以下を参照してください。

  [ ] 52 ページの「システム・ファイルの設定」

  [ ] 58 ページの「(オプション) 自動再起動用の設定」

  [ ] 58 ページの「ネットワーク・インタフェースの設定」

  [ ] 61 ページの「シリアル・ポートの設定」

  [ ] 62 ページの「システムの再起動」

[ ] 小さいクラスタ (3 ノード) を完全に設定します。65 ページの 第3章「クラスタの初期設定」を参照してください。

[ ] /var/cluster/ha/logs ディレクトリのデーモン・ログ・ファイル内でエラーを検索します。

[ ] すべてが正常な場合は、残りのノードを追加します。問題がある場合は、173 ページの 第8章「トラブルシューティング」を参照してください。

[ ] ファイルシステムを設定します。65 ページの 第3章「クラスタの初期設定」を参照してください。

用語集

CXFS データベース

クラスタ設定データベースを参照してください。

CXFS メンバーシップ

クラスタでアクティブにファイルシステムを共有している CXFS ノードのグループ。クラスタで定義されているノードのサブセットの場合があります。ノードは起動処理中に CXFS メンバーシップを要求し、要求が受付けられると、クラスタのファイルシステムを共有できます。カーネル空間メンバーシップとも呼びます。CXFS メンバーシップは、`fs2d` メンバーシップおよび FailSafe メンバーシップとは異なります。FailSafe についての詳細は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

FailSafe メンバーシップ

クラスタでアクティブにリソースを共有しているノードのグループ。クラスタで定義されているノードのサブセットの場合があります。FailSafe メンバーシップは、CXFS メンバーシップおよび `fs2d` メンバーシップとは異なります。FailSafe についての詳細は、『IRIS FailSafe Version 2 Administrator's Guide』を参照してください。

fs2d メンバーシップ

`fs2d` にアクセス可能で、クラスタ設定データベースの更新を受信できるプール内のノードのグループ。プール内で定義されたノードのサブセットの場合があります。ユーザ空間メンバーシップとも呼びます。

オーナー TTY 名

システム・コントローラが接続されているオーナー・ホストの端末ポート (TTY) のデバイス・ファイル名。オーナー・ホストからノードをリモートで制御できるよう、このケーブルのもう一方はシステム・コントローラ・ポートを持つノードに接続します。

オーナー・ホスト

ノードの電源サイクルなどのノードの制御をリモートで行うことができるシステム。ランタイム時には、オーナー・ホストがプール内のノードとして定義されていなければなりません。

重み

メンバーシップ重みを参照してください。

回復

あるノード上のサービスが中断した場合に、最初のノードから別のノードにメタデータ・サーバが移動するプロセス。

カーネル空間メンバーシップ

CXFS メンバーシップを参照してください。

クラスタ

クラスタとして定義されているプール内のノードのセット。クラスタは単純な名前で識別されます。この名前は、プール内で一意でなければなりません。たとえば、クラスタとノードで同じ名前を使用することはできません。

クラスタ内のすべてのノードはプールにも存在しますが、プール内のすべてのノードがクラスタに存在するとはかぎりません。つまり、クラスタは、プール内のノードのサブセットで構成されている場合があります。各プールのクラスタは 1 つだけです。

クラスタ管理者

クラスタの管理とメインテナンスの担当者。

クラスタ設定データベース

すべてのノードとクラスタに関する設定情報が含まれます。

クラスタ・ドメイン

ファイルシステムがローカル・ノードだけでなくクラスタ全体に適用されるという XVM での概念。ローカル・ドメインも参照してください。

クラスタ・ノード

ノードを参照してください。

クラスタ・メンバーシップ

fs2d メンバーシップを参照してください。

コントロール・ネットワーク

ネットワーク・インタフェース（通常は Ethernet）を通じてノードを接続するネットワーク。CXFS が、接続されているノードにネットワークを通じてハートビート・メッセージとコントロール・メッセージを送信できるよう

にします。CXFS では、コントロール・ネットワーク上で最も優先度が高いネットワーク・インタフェースが使用されます。コントロール・ネットワーク上にある優先度の高いすべてのネットワーク・インタフェースに異常が発生した場合は、優先度の低いネットワーク・インタフェースが使用されます。

コントロール・メッセージ

クラスタ・ノードに対する操作を要求したり、クラスタ・ノードに関する情報を配布するためにクラスタ・ソフトウェアがクラスタ・ノード間で送信するメッセージ。コントロール・メッセージおよびハートビート・メッセージは、コントロール・ネットワークに接続されているノードのネットワーク・インタフェースを通じて送信されます。

ノードが専用の CXFS ノードではない場合、ノードのコントロール・ネットワークは、コントロール・メッセージを受付けないように設定してください。コントロール・メッセージを受付けるように設定すると、CXFS がノードをリセットしたときに、そのマシンでほかのジョブを実行するエンド・ユーザのジョブが予期せず強制終了される可能性があります。

再設定

管理操作のために、あるノードから別のノードにメタデータ・サーバを移動するプロセス。最初のノード上のその他のサービスが中断されることはありません。

システム・コントローラ・ポート

リモートでノードの電源サイクルを行う方法を提供するノード上のポート。クラスタ設定データベース (CDB) でシステム・コントローラ・ポートが有効または無効のどちらに設定されているかによって、システム・コントローラ・ポートで CXFS が処理を実行できるかどうかが決まります。システム・コントローラ・ポートの情報はプール内のノードの場合はオプションですが、そのノードがクラスタに追加される場合は必須です。システム・コントローラ・ポートの情報がないと、そのノードで実行されているリソースは高可用性になりません。

上限量

クラスタの形成に必要なノードの数。CXFS メンバーシップの場合、初期メンバーシップ上限量の形成には、クラスタ内の重み付けされたノードの過半数 (>50%) が必要です。既存のメンバーシップ上限量を維持するには、クラスタ内の重み付けされたノードの半数 (50%) が必要です。`fs2d` メンバーシップの場合、メンバーシップ上限量の形成および維持には、プール内のノードの 50% が必要です。

セル ID

ノードに関連付けられている数字。CXFS ソフトウェアによって使用され、メッセージに表示されます。

タイブレーカー・ノード

CXFS のタイブレーカーとして識別されるノード。クラスタ内のちょうど半数のノードが動作していて相互に通信できる場合にクラスタの CXFS メンバーシップを計算する処理で使用されます。デフォルトのタイブレーカー・ノードはありません。

通知コマンド

クラスタおよびノードの変更や異常をクラスタ管理者に通知するために使用されるコマンド。このコマンドは、クラスタ内のすべてのノードに存在する必要があります。

データベース

クラスタ設定データベースを参照してください。

同時実行

CXFS と IRIS FailSafe を同時に実行する機能。詳細については、27 ページの「IRIS FailSafe 2.1 との同時実行」を参照してください。

ドメイン

クラスタ・ドメインおよびローカル・ドメインを参照してください。

ノード

個々のコンピュータ。すべてのノードで、同バージョンまたは直前のバージョンの IRIX オペレーティング・システム (OS: Operating System) が実行されている必要があります（6.5.11 と 6.5.12 など）。ノードは、ストレージ・システムとクラスタ内のノードを接続する SAN に接続されています。ノードが属することのできるクラスタは 1 つだけです。

このノードという用語は、Origin システムのノードとは異なる意味を持ちます。

ノード ID

プール内のノードで一意の 1～32767 の範囲の整数。数字を指定しないと、CXFS によって自動的に ID が計算されます。デフォルトの ID は、マシンのシリアル・ナンバーなどのマシン固有の情報を基にした 5 桁の数字で、連続する数字ではありません。ノードを定義した後にノード ID 番号を変更しないでください。

ノード・タイムアウト

この時間内にノードからハートビートが受信されない場合、ノードはダウンしていると見なされます。CXFS を正しく動作させるには、ノード・タイムアウト値をハートビート周期の少なくとも 10 倍に設定してください。

ノード・メンバーシップ

クラスタでアクティブな（CXFS メンバーシップを持っている）ノードのリスト。

ハートビート周期

ハートビート・メッセージの周期。CXFS を正しく動作させるには、ノード・タイムアウト値をハートビート周期の少なくとも 10 倍に設定してください。ハートビートの数が多いほど（ハートビート周期が小さいほど）、ネットワークの速度が遅くなる可能性が高くなります。

ハートビート・メッセージ

クラスタ・ソフトウェアがノード間で送信する、ノードが動作中であることを示すメッセージ。コントロール・メッセージおよびハートビート・メッセージは、コントロール・ネットワークに接続されているノードのネットワーク・インタフェースを通じて送信されます。

プール

ネットワークによって相互に結合され、クラスタ設定データベースでノードとして定義されているノードのセット全体。通常、これらのノードは互いに近い位置にあり、同じ目的に使用されます。プール内の各ノードには、複製されたクラスタ設定データベースが格納されています。

クラスタに追加できるノードはすべてプールの一部ですが、プール内のすべてのノードをクラスタの一部にする必要はありません。プールは 1 つだけです。ほかのプールを存在させることはできますが、各プールはお互いから分離した状態になり、ノードやクラスタ定義は共有されません。

ポート・パスワード

システム・コントローラ・ポートのパスワード。通常は、ファームウェアまたはジャンパ・ワイヤで一度だけ設定します。ポート・パスワードは、ノードの root パスワードとは異なります。

メタデータ

ファイルの名前、サイズ、場所、アクセス権など、ファイルを記述する情報。

メタデータ・サーバ

クラスタ内のすべてのノードに代わってメタデータの更新を調整するノード。
See also メタデータ

メンバーシップ

`fs2d` メンバーシップおよび CXFS メンバーシップを参照してください。

メンバーシップ重み

CXFS メンバーシップ上限量を計算するためにノードに割当てられる数字（通常は 0 または 1）。

メンバーシップ・バージョン

ノードのセル ID に関連付けられている数字。ノードがメンバーシップに設定されてから CXFS メンバーシップが変更された回数を示します。

ユーザ空間メンバーシップ

fs2d メンバーシップを参照してください。

ローカル・ドメイン

ファイルシステムがクラスタではなくローカル・ノードだけに適用されるという XVM での概念。クラスタ・ドメインも参照してください。

ログ・グループ

同じログ設定を使用する 1 つまたは複数の CXFS プロセスのセット。通常、ログ・グループは `gcd` などの 1 つのデーモンに対応します。

ログ設定

ログ設定は、ログ・レベルとログ・ファイルの 2 つの部分で構成され、どちらもログ・グループに関連付けられます。クラスタ管理者は、ログ出力の場所と量をカスタマイズしたり、すべてのノードまたは 1 つのノードだけを対象にしたログ設定を指定できます。たとえば、`crsd` ログ・グループを設定して、ノード `foo` だけについては詳細レベル 10 のメッセージを `crsd-foo` ログに記録し、その他のすべてのノードについては最小レベル 1 のメッセージだけを `crsd` ログに書込むことができます。

ログ・ファイル

特定のログ・グループに関する通知が含まれるファイル。ログ・ファイルは、ログ・グループのログ設定の一部です。

ログ・レベル

関連付けられているログ・グループのログ・ファイルに CXFS が書込むログ・メッセージの数を制御する数字。ログ・レベルは、ログ・グループのログ設定の一部です。

索引

A

age 181
atime 24
AutoLoad 起動パラメータ 58

C

CAD オプション・ファイル 54
cdbreinit 188, 209, 212
cdbutil 182
cell ID 181
clconf_info 181
clconfd プロセス 20
clconfd-scripts ディレクトリ 144
CLI
  設定タスクを参照
  概要 33, 43
  シェルおよび 46
  終了 44
  スクリプトおよび 44
  ヘルプ 44
CLI のプロンプトの表示 43
cluster_admin サブシステム 20
cluster_control サブシステム 20
cluster_mgr
  CLIを参照
cluster_services サブシステム 20
cluster_status 166
cmgr
  CLIを参照
  スクリプト・モード 141
cms_dead() 178
cms_declare_membership() 178
cms_follower() 178
cms_leader() 178
cms_nascent() 178
cms_shutdown() 178
cms_stable() 178
cmsd カーネル・スレッド 183
crontab の制約 146
crsd プロセス 20
ctime 24
curses インタフェース 166
CXFS クラスタ表示 36
CXFS クラスタ・マネージャ GUI
  タスクセット 43
CXFS サービス 98
CXFS サービスの開始 154
CXFS サービスの停止
  CLI 135
  GUI 98
CXFS シャットダウン
  強制
    CLI 154
  通常
    CLI 153
    GUI 98
CXFS ソフトウェアのインストール 47
CXFS マネージャ
  設定タスク、CLIを参照
  CXFS メンバーシップの許可 140
  CXFS メンバーシップの取消し 140
  概要 33
CXFS メンバーシップ 7
  許可
    CLI 140
    GUI 105
  現在の情報 181
  取消し
    CLI 140
    GUI 105

CXFS メンバーシップの許可
  CLI 140
  GUI 105
CXFS メンバーシップの取消し
  CLI 140
CXFS メンバーシップの分割 17
CXFS メンバーシップの問題 184
cxfs-*-mount.sh スクリプト 144
cxfs-*-umount.sh スクリプト 144

E

/etc/config/cad.options ファイル 54
/etc/config/fs2d.options ファイル 55
/etc/init.d 212
/etc/mtab 2
/etc/services ファイル 54

F

FailSafe 切替え
  ノード 116
FailSafe クラスタの切替え
  CLI 127
  GUI 83, 96
FailSafe タスクセットからの切替え 83
FailSafe 同時実行 27
FailSafe ノードの CXFS への切替え 116
FailSafe ノードの切替え
  CLI 116
  GUI 83, 93
FailSafe メンバーシップ 8
filesystem reorganizer 2
find および crontab 146
fs2d オプション・ファイル 55
fs2d メンバーシップ 8
fs2d_log 184
fsr 146
fsr_xfs(1M) 2

G

GUI
  概要 33
  タスク
    設定タスクを参照
GUI の URL 41
GUI への Web アクセス 41

I

I/O レート保証 3
icrash 183
incarnation 181
IP アドレスおよびコントロール・ネットワーク 85
IRIS FailSafe 同時実行 27
IRIS FailSafe との同時実行 27
IRISconsole 27

J

java
  インストール・ファイル 50
  サポートされているリリース・レベル 51
java_plugin 51

L

LAN 4

M

mkpart 86, 111
mtime 24

N

NFS 159
NFS エクスポート 144
NVRAM 変数 58

O

Origin 200 26
Origin 2000 26

P

PC と GUI の使用 41
physvol 194

Q

quota 2
Quota 159

R

RealTime 159
reorganizer 2
/.rhosts ファイル 57
routed 60

S

SAMBA 159
SAN 1

V

/var/cluster/clconfd-scripts ディレクトリ 144

W

Web での CXFS マネージャ 41

X

xfs_fsr (fsr_xfs) (1) 2
xfsdump および xfsrestore 162
XVM 158

あ

「アクティブ」クラスタのステータス 171
アンマウント 161
アンマウント・スクリプト 144

い

インストール 47
インストールされるサブシステム 49

お

重み付け 13
重み付けされた 2 ノード・クラスタ 32

か

回復
  重み付けされた 2 ノード・クラスタ 32
  メタデータ・サーバの 151
  用語 9
書込みとメタデータの流れ 24
管理 143
管理シャットダウン 153
カーネル空間メンバーシップ 7

き

強制 CXFS シャットダウン
  CLI 154
強制シャットダウン 154
強制シャットダウンおよび再開 156, 184

く

クラスタ 7
クラスタ設定データベースのシャットダウン 152
クラスタ定義
  CLI 121
  GUI 95
クラスタの再設定 183
クラスタのステータス 171
クラスタの定義
  CLI 121, 131
  GUI 95, 97
クラスタの表示 36
  CLI 133
  GUI 97
クラスタ表示 36
クラスタへの再設定
  強制 CXFS シャットダウン 155
  通常 CXFS シャットダウン 154
クラスタ・データベース上限量 184
クラスタ・デーモンの再開 212
クラスタ・ドメイン 158
クラスタ・ノード 7
  ノードを参照 84
クラスタ・マネージャのツール 33
クラスタ・メンバーシップ 7

こ

コマンド・ライン・インタフェース
  CLIを参照
コントロール・ネットワーク　8
  ノードに対する定義　85

さ

再開
  CXFS サービス　208
再起動　62, 209
再起動（回避）　156
再設定　9
  重み付けされた 2 ノード・クラスタ　32

し

システム
  ステータス　165
  表示　19
  ファイル　52
システム・コントローラのステータス　171
システム・ソフトウェア
  通信パス　22
「実行中」ノードの状態　168
シャットダウン　153
  強制終了された CXFS クラスタ設定データベース　154
  クラスタ・データベース　152
  通常 CXFS
    GUI　98
  通常の CXFS データベース　153
シャットダウンおよび再開　156, 184
上限量　13, 184
初期クラスタ設定
  CLI　73
  GUI　68
シリアル接続
  CLI　139
  GUI　104
シリアル・ポート設定　61
診断タスク
  設定タスクを参照

す

スクリプト　141
ステータス
  クラスタ　171
  システム、概要　165
  システム・コントローラ　171
  ノード　168
  リソース　171
ストレージ・エリア・ネットワーク (SAN: Storage Area Network)　1

せ

設定タスク
  CLI　107
    CXFS サービスの停止　135
    CXFS メンバーシップの許可　140
    CXFS メンバーシップの取消し　140
    FailSafe クラスタの切替え　127
    クラスタの定義　121, 131
    クラスタの表示　133
    クラスタの変更　124
    クラスタ・タスク　121
    サービスの開始　135
    タイブレーカー・ノードの設定　136
    デフォルト　108
    ネットワークの接続性のテスト　139
    ノードの削除　117
    ノードの定義　109
    ノードの表示　120
    ノードの変更　113
    ノードのリセット　116
    ノード・タスク　109
    ファイルシステムのアンマウント　127
    ファイルシステムの再マウント　128
    ファイルシステムの削除　130
    ログ・グループの設定　137
    ログ・パラメータの設定　137
    ログ・パラメータの表示　137
    ログ・パラメータの変更　138
  GUI
    CXFS サービスの開始　98

CXFS サービスの停止　98
CXFS メンバーシップの許可　105
FailSafe クラスタの切替え　96
FailSafe ノードの切替え　93
クラスタの定義　95, 97
クラスタの表示　97
クラスタの変更　96
クラスタへのノードの追加　90
新規クラスタの設定　71
新規ファイルシステムの設定　72
接続性のテスト　104
ノードの削除　94
ノードの定義　84
ノードの表示　94
ノードの変更　91
ノードのリセット　90
ファイルシステムのアンマウント　103
ファイルシステムの削除　104
ファイルシステムの変更　103
ファイルシステムのマウント　102
メタデータ・サーバの定義　101
メンバーシップの取消し　105
ログ・グループ定義の表示　100
ログ・グループの設定　100
ログ・パラメータの設定　99

た

タイブレーカー・ノード
　CLI　136
　GUI　99
タスク　42
ダンプ　162
断片化解消プログラム　2

ち

チャネル・トラフィック　191

つ

通常 CXFS シャットダウン
　GUI　98
通信パス　22
通知
　GUI　96
ツリー表示　36
ツール　33

て

「停止」ノードの状態　168
ディスク・バッファ　5
データ管理　158
データベース
　クリア　208
　ダンプ　182
データベースのクリア　208
データベースのシャットダウン　152
データベースのダンプ　182
データ・フロー　24
デーモンの再開　212

と

ドメイン　158
トラブルシューティング　173
トークン　24

な

名前付きパイプと CXFS　6

ね

ネットワーク　8
ネットワークの接続性
　CLI　139
　GUI　104
ネットワークの分割とハードウェアのリセット　18
ネットワーク分割　18
ネットワーク・インタフェースの設定　58
ネットワーク・セグメント　8
ネットワーク・ファイルシステムの比較　4
ネットワーク・ルーティング　60

の

ノード

重み付け　13
状態　168
ステータス　168
タスク
設定タスクを参照
定義　7
ノードの削除
CLI　117, 124
GUI　90, 94
ノードの自動再起動　58
ノードのステータス
強制 CXFS シャットダウン　155
クラスタ設定データベースのシャットダウン　152
通常 CXFS シャットダウン　154
ノードの追加
GUI　90
ノードの表示
CLI　120
GUI　94
ノードの変更
GUI　91
ノードを自動的に再起動する　58
ノード・タスク
設定タスクを参照

は

パイプ（名前付き）　6
バックアップ　159
パーティション　86, 92, 110–111
ハートビート・ネットワーク　8, 85

ひ

「非アクティブ」
クラスタのステータス　171
ノードの状態　168
ピアツーディスク・モデル　5

ふ

ファイルシステム (CXFS) と IRIS FailSafe　28
ファイルシステム削除
CLI　130
GUI　104
ファイルシステムのアンマウント
CLI　127
GUI　103
ファイルシステムの拡張　161
ファイルシステムの再マウント
CLI　128
ファイルシステムの削除
CLI　130
GUI　104
ファイルシステムの変更
CLI　113
GUI　103
ファイルシステムのマウント
GUI　102
ファイルシステムのメインテナンス　160
ファイルシステム変更
GUI　103
ファイルシステム・アンマウント
CLI　127
GUI　103
ファイルシステム・タスクセット　72
ファイルシステム・マウント
CLI　124
GUI　102
ファイルへのスワップ　3
復元　162
「不明」ノードの状態　168
プラットフォーム　26
プール　7

へ

ヘルプ
CLI　44
GUI の　71

ほ

ホスト名
コントロール・ネットワーク　85
ボリューム管理　158

ま
マウント　160
マウントポイントのネスト　3
マウント・オプションのネスト　3
マウント・スクリプト　144

め
命名の制約　52
メインテナンス　143
メタデータ
　例　4
メタデータ・クライアント　18
メタデータ・サーバ
　CLI　124
　回復　151
　定義　9
　発見　149
メタデータ・サーバからのファイルシステムの削除　153
メタデータ・サーバ定義
　GUI　101
メタデータ・サーバの移動　153
メンバーシップ
　クラスタ・メンバーシップまたはCXFSメンバーシップを参照
メンバーシップの取消し
　GUI　105

ゆ
ユーザ空間メンバーシップ　8

よ
読取りとメタデータの流れ　24

ら
ライセンス・エラー　195

り
リセット
　データベース　208
　ノード　90
リセットのためのレース　18
リセット・ハードウェア　17
リセット・レース　18
リソース
　ステータス　171
リモート・リセット接続　18

る
ルーティング　60

ろ
ログ・グループ　99
ログ・ファイル
　管理　156
　モニタ　165
　リスト　165
ログ・ファイルのローテーション　156
ログ・レベル　100
ローカル・エリア・ネットワーク (LAN: Local Area Network)　4
ローカル・ドメイン　158
ローカル・ノード　154