

TruCluster Server

クラスタ・インストール・ガイド

Part Number: AA-RM86D-TE

2002 年 11 月

ソフトウェア・バージョン: TruCluster Server バージョン 5.1B

オペレーティング・システム: Tru64 UNIX バージョン 5.1B

本書では、管理者向けに TruCluster Server ソフトウェアを Tru64 UNIX オペレーティング・システム上にインストールする方法について説明します。この中では、インストール要件、構成手順、アップグレード手順、移行手順、クラスタ・メンバの追加についても説明しています。

© 2002 Hewlett-Packard Company

本書の著作権は日本ヒューレット・パッカート株式会社が保有しており、本書中の解説および図、表は日本ヒューレット・パッカートの文書による許可なしに、その全体または一部を、いかなる場合にも再版あるいは複製することを禁じます。

日本ヒューレット・パッカートは、弊社または弊社の指定する会社から納入された機器以外の機器で対象ソフトウェアを使用した場合、その性能あるいは信頼性について一切責任を負いかねます。

本書に記載されている事項は、予告なく変更されることがありますので、あらかじめご承知おきください。万一、本書の記述に誤りがあった場合でも、弊社は一切その責任を負いかねます。

本書で解説するソフトウェア(対象ソフトウェア)は、所定のライセンス契約が締結された場合に限り、その使用あるいは複製が許可されます。

COMPAQ, Compaq ロゴ, Digital ロゴは U.S. Patent and Trademark Office に登録されています。Alpha, AlphaServer, NonStop, TruCluster, および Tru64 は米国 Compaq Computer Corporation の商標です。

Microsoft, Windows および Windows NT は米国 Microsoft 社の登録商標です。Intel は米国 Intel 社の登録商標です。Motif, OSF/1, UNIX, The Open Group および X/Open は、The Open Group の米国ならびに他の国における商標です。

このドキュメントに記載されているその他の会社名および製品名は、各社の商標または登録商標です。

目次

まえがき

1 概要

1.1	バージョン 5.1B の新しいインストール機能と変更されたインストール機能	1-2
1.2	インストールのタイプ	1-3
1.3	インストール・コマンド	1-4
1.4	フル・インストールの概要	1-5
1.5	必要なライセンス	1-6
1.6	TruCluster Server サブセット	1-7
1.7	Tru64 UNIX のサブセット	1-8
1.8	一般的な注意事項	1-8

2 インストールの準備

2.1	必要なマニュアルおよび PAK	2-2
2.2	メンバ ID	2-2
2.3	IP 名と IP アドレス	2-3
2.3.1	クラスタ名と IP アドレス	2-4
2.3.2	メンバのホスト名と IP アドレス	2-6
2.3.3	各メンバのクラスタ・インターコネクト用 IP 名と IP アドレス	2-7
2.4	ハードウェア	2-9
2.5	ディスク	2-10
2.5.1	インストールに必要なディスク	2-11
2.5.1.1	Tru64 UNIX ディスク (プライベートまたは共用)	2-12
2.5.1.2	クラスタ単位のルート (/), /usr, /var ディスク (共用)	2-13

2.5.1.3	メンバ・ブート・ディスク (共用)	2-14
2.5.1.4	クォーラム・ディスク (共用)	2-16
2.5.2	LSM の考慮事項	2-17
2.5.3	2 ノード・クラスタの最小ディスク・レイアウト	2-18
2.5.4	推奨するディスクの空き容量	2-19
2.5.5	ローリング・アップグレードの準備のための追加のディスク容量の割り当て	2-21
2.6	クォーラム：メンバ・ポートと期待ポート	2-23
2.7	コンソール変数	2-26
2.8	サンプルのクラスタ構成チェックリスト	2-29

3 Tru64 UNIX オペレーティング・システムのインストールと構成

3.1	SRM ファームウェアのアップデート	3-3
3.2	Tru64 UNIX オペレーティング・システムのインストール ...	3-3
3.3	基本的なサービスの構成	3-5
3.3.1	ルーティング・デーモン	3-7
3.3.2	タイム・サーバ	3-7
3.3.3	ネーム・サーバ	3-8
3.3.4	NFS	3-8
3.3.5	NIS	3-9
3.3.6	DHCP	3-9
3.3.7	メール・サーバ	3-9
3.3.8	プリント・サーバ	3-10
3.3.9	ネットワーク・サービスの概要	3-10
3.4	Secure Shell ソフトウェアの構成	3-11
3.5	エンハンスド・セキュリティの構成 (オプション)	3-11
3.6	冗長ネットワーク・インタフェース用の NetRAIN の構成 (オプション)	3-12
3.7	クラスタのインストールに必要なディスクの構成	3-12
3.7.1	パーティション・サイズ	3-12

3.7.2	ディスク位置の特定	3-14
3.7.2.1	シンメトリックなストレージ構成	3-16
3.7.2.2	シンメトリックでないストレージ構成	3-17
3.8	LAN インターコネクトを使用する際のネットワーク・アダプタのデバイス名の取得	3-18
4	シングル・メンバ・クラスタの作成	
4.1	TruCluster Server のソフトウェア・ライセンス登録	4-2
4.2	TruCluster Server サブセットのロード	4-2
4.3	パッチの適用	4-3
4.4	clu_create コマンドの実行	4-3
4.4.1	クラスタ・インターコネクトの選択	4-4
4.4.2	clu_create が実行するタスク	4-5
4.5	シングル・メンバ・クラスタとしてのシステムのブート	4-6
4.6	重要な構成ファイルのオン・ディスク・バックアップ・コピーの作成	4-9
5	メンバの追加	
5.1	SRM ファームウェアのアップデート	5-3
5.2	インストレーション時における新しいメンバのブートの防止	5-3
5.3	clu_add_member コマンドの実行	5-3
5.4	新しいメンバのブート	5-5
5.5	重要な構成ファイルのオン・ディスク・バックアップ・コピーの作成	5-8
6	クラスタ・メンバの再インストール	
6.1	シングル・メンバ・クラスタの再作成	6-1
6.2	個々のクラスタ・メンバの再インストール	6-3
6.3	インストール構成ファイルの使用	6-3

7 ローリング・アップグレード

7.1	ローリング・アップグレードでサポートされている作業	7-4
7.2	ローリング・アップグレードでサポートされていない作業 ...	7-7
7.3	ローリング・アップグレードの手順	7-8
7.4	ローリング・アップグレードのステータス表示	7-17
7.5	各段階の取り消し	7-18
7.6	ローリング・アップグレードのコマンド	7-19
7.7	ローリング・アップグレードの各段階	7-24
7.7.1	準備段階	7-25
7.7.2	セットアップ段階	7-28
7.7.3	プリ・インストール段階	7-30
7.7.4	インストール段階	7-31
7.7.5	ポスト・インストール段階	7-32
7.7.6	ロール段階	7-33
7.7.7	スイッチ段階	7-34
7.7.8	クリーンアップ段階	7-35
7.8	タグ付きファイル	7-35
7.9	バージョン・スイッチ	7-39
7.10	ローリング・アップグレードとレイヤード・プロダクト	7-40
7.10.1	一般的なガイドライン	7-40
7.10.2	ブロッキング・レイヤード・プロダクト	7-41
7.11	ローリング・アップグレードと RIS	7-42

8 TruCluster Software バージョン 1.5 またはバージョン 1.6 からのアップグレード

8.1	別のクラスタを作成するか既存クラスタをアップグレードするかの決定	8-3
8.2	ストレージとクラスタのインターコネクトに関する制約事項	8-5

8.3	アップグレードの準備	8-6
8.3.1	一般的な要件	8-7
8.3.2	ハードウェアとストレージ・トポロジの要件	8-8
8.3.3	ファイル・システムの要件	8-11
8.4	オプション 2 およびオプション 3 で使用するアップグレード・スクリプト	8-13
8.5	オプション 1: 別のクラスタの作成 — 新しいシステムと新しいストレージ	8-16
8.6	オプション 2: 別のクラスタの作成 — 新しいシステムと既存ストレージの使用	8-19
8.6.1	オプション 2: アップグレード手順	8-19
8.6.2	オプション 2: 部分アップグレードの取り消し	8-24
8.7	オプション 3: 既存のシステムとストレージの使用	8-25
8.7.1	オプション 3: アップグレード手順	8-25
8.7.2	オプション 3: 部分アップグレードの取り消し	8-34
8.8	アップグレードのケース・スタディ	8-35
8.9	TruCluster Memory Channel Software クラスタのアップグレード	8-44

A 情報チェックリスト

B クラスタ・インストレーションのトラブルシューティング

B.1	LAN インターコネクトのトラブルシューティング	B-1
B.1.1	省略時の物理クラスタ・インターコネクト IP 名との競合	B-1
B.1.2	メンバをブートすると、そのメンバはクラスタに加わるが、マルチユーザ・モードになる前にハングしているように見える	B-2
B.1.3	メンバをブートすると、クラスタに加わるときにハングする	B-4
B.1.4	メンバをブートすると、"ics_ll_tcp" メッセージが出てパニックが発生する	B-5

B.1.5	メンバをブートすると, "ics_ll_tcp: ERROR: Could not create a NetRAIN set with the specified members" というメッセージが表示される	B-6
B.2	その他の問題への対処	B-7
B.2.1	クラスタ・ライセンスなしに新しいメンバをブートすると ATTENTION メッセージが表示される	B-7
C システム・ファイルの変更		
D インストレーションの例		
D.1	clu_create のログ	D-1
D.2	clu_add_member のログ	D-11
D.3	clu_upgrade のログ	D-15
E システム・ログ・ファイルのクラスタ関連メッセージ		
E.1	clu_create 実行後のスタートアップ・メッセージ	E-1
E.2	clu_add_member 実行後のスタートアップ・メッセージ	E-3
F アップグレード時の手動によるストレージの構成		
F.1	デバイスならびにストレージ構成情報の手動による収集	F-1
F.2	新規 Tru64 UNIX システムあるいは TruCluster Server クラスタ上のストレージの手動による構成	F-4

索引

例

D-1	Memory Channel インターコネクトに対する clu_create のログ・ファイルの例	D-1
D-2	LAN インターコネクトに対する clu_create のログ・ファイルの例	D-6
D-3	clu_add_member のログ・ファイルの例	D-12

D-4	clu_upgrade のログ・ファイルの例	D-15
E-1	clu_create 実行後のスタートアップ・メッセージ	E-1
E-2	clu_add_member 実行後のスタートアップ・メッセージ	E-4

図

2-1	クラスタのネットワーク・インタフェース	2-4
2-2	クラスタ単位のファイルとメンバ・ブート・パーティション	2-19
7-1	ローリング・アップグレード作業の流れ	7-3
8-1	8 ノード・クラスタのブロック図	8-47

表

1-1	TruCluster Server バージョン 5.1B へのアップグレード手順	1-3
1-2	TruCluster Server サブセットの内容	1-7
1-3	TruCluster Server サブセットのサイズ	1-8
2-1	準備作業	2-1
2-2	推奨するディスクの空き容量	2-20
2-3	クォーラム・ディスクを持つクラスタのための推奨ポート割り当て	2-25
2-4	クォーラム・ディスクを持たないクラスタのための推奨ポート割り当て	2-25
2-5	サンプルの Tru64 UNIX システム属性	2-30
2-6	サンプルのクラスタ属性	2-30
2-7	サンプルのメンバ属性	2-31
3-1	Tru64 UNIX のインストール	3-2
3-2	ネットワーク・サービスの要約	3-10
4-1	シングル・メンバ・クラスタの作成作業	4-1
5-1	メンバの追加作業	5-2
7-1	バージョン 5.1A および バージョン 5.1B で行えるローリング・アップグレード作業	7-6
7-2	ローリング・アップグレードの各段階における所要時間	7-8

7-3	各段階の取り消し	7-19
7-4	バージョン 5.1A からローリング・アップグレードする場合の 段階と clu_upgrade のバージョンとの対応	7-21
7-5	バージョン 5.1B からローリング・アップグレードする場合の 段階と clu_upgrade のバージョンとの対応	7-22
7-6	ブロッキング・レイヤード・プロダクト	7-41
A-1	Tru64 UNIX システム属性	A-1
A-2	クラスタ属性	A-1
A-3	メンバ属性	A-2
C-1	/etc/rc.config の変数	C-4
C-2	/etc/sysconfigtab の属性	C-8
C-3	/etc/sysconfigtab.cluster の属性	C-11

まえがき

本書では、次の作業の手順について説明します。

- クラスタのインストールの準備
- TruCluster Server バージョン 5.1B ソフトウェアの HP Tru64 UNIX バージョン 5.1B オペレーティング・システムへのインストールと、Memory Channel または LAN のいずれかをクラスタ・インターコネクトとして使用する新しいクラスタの作成
- メンバのクラスタへの追加
- TruCluster Server バージョン 5.1A から TruCluster Server バージョン 5.1B へのローリング・アップグレード
- TruCluster Production Server クラスタまたは Available Server 構成の TruCluster Server バージョン 5.1B クラスタへのアップグレード
- TruCluster Memory Channel Software クラスタから TruCluster Server バージョン 5.1B へのアップグレード

対象読者

本書は、TruCluster Server 製品のインストール、構成、および管理を行うシステム管理者を対象としています。UNIX に精通しており、ハードウェア、オペレーティング・システム、およびネットワークの構成と保守が行えることを前提としています。

変更内容

本書は、バージョン 5.1A をリリースしてから、次のように変更されています。

- バージョン 5.1A の『クラスタ LAN インターコネクト』に記載されていた情報は本書および『クラスタ・ハードウェア構成ガイド』、『クラスタ管理ガイド』、および『クラスタ概要』のマニュアルに統合されました。これにより、『クラスタ LAN インターコネクト』マニュアルは TruCluster Server のドキュメント・セットの一部ではなくなりました。

- 2.3.3 項では、LAN インターコネクトを用いたクラスタをインストールするための情報が追加されています。
- 3.8 節では、作成したクラスタが LAN インターコネクトを使う場合に、Tru64 UNIX インストール後に収集する必要がある情報について説明しています。
- 4.4.1 項では、LAN インターコネクトを用いたクラスタに最初のメンバを作成するために必要な情報を提供します。
- 5.3 節では、LAN インターコネクトを用いたクラスタにメンバを追加するための情報が追加されました。
- 7.1 節では、dupatch ノーロール機能について説明します。この機能を使うとパッチを各クラスタ・メンバについてロールする代わりに、クラスタ単位に適用することができます。また、『*Patch Kit Installation Instructions*』も参照してください。
- 7.3 節では、ロール段階で同時に複数のメンバを持つことができる、新しい並列ローリング・アップグレード機能について説明します。
- B.1 節では、誤った構成によって発生する、クラスタ LAN インターコネクトに関連した問題を解決するための提案を紹介します。

構成

本書の構成は、次のとおりです。

第 1 章	インストレーションのタイプと TruCluster Server キットの内容について説明します。
第 2 章	インストレーションを開始する前に読んで実行する必要のある事項を説明します。
第 3 章	第 1 クラスタ・メンバとなるシステムに Tru64 UNIX をインストールして構成する方法を説明します。
第 4 章	<code>clu_create</code> コマンドを実行して、シングル・メンバ・クラスタを作成する方法を説明します。
第 5 章	<code>clu_add_member</code> コマンドを実行して、クラスタにメンバを追加する方法を説明します。
第 6 章	第 1 クラスタ・メンバを再インストールし、追加のクラスタ・メンバを再インストールする方法を説明します。
第 7 章	ローリング・アップグレードの方法を説明します。

第 8 章	TruCluster Software バージョン 1.5 またはバージョン 1.6 からアップグレードする方法を説明します。
付録 A	クラスタの作成に必要な情報を収集するときに使用するチェックリストを載せてあります。
付録 B	クラスタ LAN インターコネクトの設定ミスによる問題など、インストール中に発生する可能性がある問題を処理するための提案を紹介します。
付録 D	<code>clu_create</code> , <code>clu_add_member</code> , および <code>clu_upgrade</code> のインストール・ログの例を示します。
付録 E	シングル・メンバ・クラスタを作成し、メンバをクラスタに追加した後に表示されるブート・メッセージの例を示します。
付録 F	TruCluster バージョン 1.5 またはバージョン 1.6 の製品からアップグレードする際に、第 8 章で説明する移行スクリプトを使用しないシステム管理者のために、手動によるストレージ情報収集および構成手順を説明します。

関連資料

クラスタの理解，構成，インストール，および管理については，次の TruCluster Server のマニュアルを参照してください。

- TruCluster Server 『*QuickSpecs*』 — TruCluster Server バージョン 5.1B について説明するとともに，製品の機能およびサポートするハードウェアに関する情報が記載されています。『*QuickSpecs*』は，TruCluster Server 『ソフトウェア仕様書 (SPD)』に置き換わるものです。
『*QuickSpecs*』の最新版は，次の URL から入手することができます。
http://tru64unix.compaq.com/docs/pub_page/spds.html
- 『クラスタ概要』 — TruCluster Server 製品を紹介するとともに，主なサブシステムについて説明しています。
- 『クラスタ・リリース・ノート』 — TruCluster Server の新機能に関する簡単な説明と，わかっている問題点およびその対処方法を記載しています。
- 『クラスタ・ハードウェア構成ガイド』 — クラスタ・メンバにするシステムの設定方法と，クラスタの共用ストレージを構成する方法を説明しています。
- 『クラスタ管理ガイド』 — クラスタ特有の管理作業について説明しています。

- 『クラスタ高可用性アプリケーション・ガイド』 — 既存のアプリケーションを TruCluster Server クラスタで実行する方法と、クラスタ対応のアプリケーションを作成する方法を記述しています。

TruCluster Server のマニュアルは World Wide Web にあり、次の URL から入手できます。

http://www.tru64unix.compaq.com/docs/pub_page/cluster_list.html

Tru64 UNIX バージョン 5.1B のドキュメント・セットにある次のマニュアルも参照してください。

- 『インストレーション・ガイド』
- 『インストレーション・ガイド — 上級ユーザ編』
- 『リリース・ノート』
- 『システム管理ガイド』
- 『ハードウェア管理ガイド』
- 『ネットワーク管理ガイド：接続編』
- 『ネットワーク管理ガイド：サービス編』
- 『セキュリティ管理ガイド』

Tru64 UNIX のドキュメントは次の URL でアクセスできます：

<http://tru64unix.compaq.co.jp/document/index.html>

オンライン・マニュアル

TruCluster Server の日本語版のドキュメント・セットは、Tru64 UNIX の「日本語追加機能」CD-ROM に、また、英語版のドキュメント・セットは「Tru64 UNIX Documentation」CD-ROM と Tru64 UNIX 「Associated Products Volume 2」CD-ROM にそれぞれ収録されており、Web ブラウザ (HTML 形式) または Adobe Acrobat Reader (PDF 形式) で読める形式になっています。英語版のリファレンス・ページも、HTML 形式のものが参照できます。

Alpha システムで作業をしている場合は、オペレーティング・システムにより HTML のドキュメントを表示するための Netscape ブラウザが提供されます。Windows PC または Macintosh でマニュアルを表示するには、Javascript 1.1 をサポートするブラウザが必要です。Netscape Navigator

バージョン 4.0 以降，Netscape Communicator，Microsoft Internet Explorer のいずれかを使用することをお勧めします。

ほとんどの相互参照はホット・リンクになっており，マニュアル間や，マニュアルからリファレンス・ページまでを移動できます。参照するマニュアルやリファレンス・ページは別ウィンドウで開かれるので，必要な情報を収集した後は，最初に参照していたマニュアルまたはリファレンス・ページに簡単に戻ることができます。

ドキュメントを参照する手順についての詳細は，Tru64 UNIX 『ドキュメント概要』を参照してください。

本書で使用する表記法

本書では、次の表記法を使用しています。

#	番号記号は root としてログインした場合のシステム・プロンプトを表します。
% cat	対話式の例における太字(ボールド体)は、ユーザが入力する文字を示します。
<i>file</i>	イタリック体(斜体)は、変数値、プレースホルダ、および関数の引数名を示します。
:	垂直の反復記号は、実際には存在する例の一部が省略されていることを示します。
Return	四角で囲まれたキー名はユーザがそのキーを押すことを示します。
cat(1)	リファレンス・ページの参照には、該当するセクション番号をカッコ内に示します。たとえば、cat(1) は、cat コマンドについての情報が、リファレンス・ページのセクション 1 に記載されていることを示します。

本書では、TruCluster Server 製品をインストールして構成する方法を説明します。

インストレーションを開始する前に本書に目を通し、インストールするタイプに関連した章をよくお読みください。一般的なインストール手順に慣れておくと、時間の節約になるだけでなく、問題の発生も防止できます。

SysMan を使って TruCluster Server 製品またはレイヤード・プロダクトをクラスタにインストールまたはアンインストールすることはできません。その代わりに、本書および『クラスタ管理ガイド』に記載されている手順を使用してください。

クラスタ・インターコネクトとして Memory Channel またはローカル・エリア・ネットワーク (LAN) ハードウェアを使用することができます。『クラスタ・ハードウェア構成ガイド』に、Memory Channel または LAN ハードウェアをクラスタ・インターコネクトとして設定するために必要な情報が説明されています。本書では、インターコネクト・ソフトウェアを設定するための手順を説明します。

注意

本書で説明する手順は、各システムのハードウェアおよびファームウェアが『クラスタ・ハードウェア構成ガイド』で説明するとおりインストールされ、構成されていることを前提としています。ソフトウェアのインストレーションは、ハードウェアとファームウェアのインストールおよび構成を完了し、すべてのストレージおよびクラスタ・インターコネクト・ハブまたはスイッチの電源を入れてから開始してください。

TruCluster Server をインストールする前に、『クラスタ・リリース・ノート』をお読みください。TruCluster Server のアー

キテクチャおよび用語について不慣れな場合は、『クラスタ概要』をお読みください。

この章の内容は、次のとおりです。

- バージョン 5.1B の新しいインストール機能と変更されたインストール機能 (1.1 節)
- インストールのタイプ (1.2 節)
- インストール・コマンド (1.3 節)
- フル・インストールの概要 (1.4 節)
- 必要なライセンス (1.5 節)
- TruCluster Server サブセット (1.6 節)
- Tru64 UNIX サブセット (1.7 節)
- 一般的な注意事項 (1.8 節)

1.1 バージョン 5.1B の新しいインストール機能と変更されたインストール機能

以下に、TruCluster Server バージョン 5.1B の新しい機能と変更された機能のうち、インストールまたはアップグレード処理に影響を与えるものを示します。

- ローリング・アップグレード中における並列ロール段階
先行メンバでロール段階を実行後、クラスタ構成によって、最大 4 メンバまで同時にロールすることができます。詳細については 7.3 節を参照してください。
- ノーロール・パッチ
クラスタが、短時間利用できなくなることが許される場合、各クラスタ・メンバにロールを行う代わりに、`dupatch` コマンドを使用してクラスタ単位でパッチを適用することができます。詳細については『*Patch Kit Installation Instructions*』を参照してください。

TruCluster Server バージョン 5.1B の新しい機能と変更された機能の全体については、『クラスタ概要』を参照してください。

1.2 インストレーションのタイプ

TruCluster Server のインストールには、次の 2 つの方法があります。

フル・インストール フル・インストールは、新しい TruCluster Server クラスタを作成する場合、または既存のクラスタを TruCluster Server バージョン 5.1B にアップグレードできない場合、またはアップグレードしない場合に行います。ただし、既存クラスタに対してフル・インストールを行うと、インストール中およびアプリケーションの構成中はそのクラスタを使用できません。

アップグレード・インストール アップグレード・インストールでは、現在の構成ができるかぎり保持されます。アップグレードの手順は、現在使用している製品の種類とバージョンによって異なります。表 1-1 に、TruCluster 製品の TruCluster Server バージョン 5.1B へのアップグレード手順を示します。

表 1-1: TruCluster Server バージョン 5.1B へのアップグレード手順

現在の製品	推奨する手順	コメント
TruCluster Server バージョン 5.1A	ローリング・アップグレード	第 7 章を参照。この章の手順で、ベース・オペレーティング・システムおよびクラスタ・ソフトウェアのローリング・アップグレードを実行する。
TruCluster Server バージョン 5.1	バージョン 5.1A へローリング・アップグレードした後、バージョン 5.1B へローリング・アップグレードする。	Tru64 UNIX バージョン 5.1B および TruCluster Server バージョン 5.1B のフル・インストールを実行するという方法もある (構成ファイル <code>.membern.cfg</code> を保存して、クラスタを再度作成する際に使用できる。構成ファイルについての詳細は、6.3 節を参照)。
TruCluster Server バージョン 5.0 または 5.0A	Tru64 UNIX バージョン 5.1B および TruCluster Server バージョン 5.1B のフル・インストール	

表 1-1: TruCluster Server バージョン 5.1B へのアップグレード手順 (続き)

現在の製品	推奨する手順	コメント
TruCluster Production Server Software , TruCluster Available Server Software バージョン 1.5 またはバージョン 1.6 のいずれか	Tru64 UNIX バージョン 5.1B と TruCluster Server バージョン 5.1B のフル・インストール	第 8 章を参照。この章では 3 つのアップグレード方法を説明する。 ^a
TruCluster Production Server Software または TruCluster Available Server Software の旧バージョン	Tru64 UNIX バージョン 5.1B と TruCluster Server バージョン 5.1B のフル・インストール	第 1 章～第 5 章を参照。バージョン 1.5 または 1.6 の製品にアップグレードしたのち、第 8 章に示すいずれかのアップグレード方法を使用する方法もある。
TruCluster Memory Channel Software	Tru64 UNIX バージョン 5.1B および TruCluster Server バージョン 5.1B のフル・インストール	第 8 章を参照。8.9 節で、この種類のクラスタを TruCluster Server バージョン 5.1B にアップグレードする方法を説明している。

^a既存の生産レベルのクラスタがある場合は、フル・インストールまたはアップグレードを行う代わりに、別にクラスタを作成して構成し、すべてのアプリケーションをテストしたのち、旧クラスタのデータを新しいクラスタに移行することもできる。この方法の詳細については、第 8 章で説明する。

1.3 インストール・コマンド

ここでは、クラスタの作成、クラスタへのメンバの追加、クラスタからのメンバの削除、およびローリング・アップグレードを行うコマンドを紹介します。

`clu_create`

Tru64 UNIX システムで `clu_create` コマンドを実行することにより、クラスタの最初のメンバを作成できる。`clu_create` コマンドをいつどのように使用するかは、`clu_create(8)` と第 4 章を参照。

`clu_add_member`

クラスタの現在のメンバで `clu_add_member` コマンドを実行することにより、クラスタにメンバを追加できる。`clu_add_member` コマンドをいつどのように使用するかは、`clu_add_member(8)` と第 5 章を参照。

`clu_delete_member`

クラスタの現在のメンバで
`clu_delete_member` コマンドを実行することにより、それ以外のメンバをクラスタから削除できる。`clu_delete_member` コマンドをいつどのように使用するかは、`clu_delete_member(8)` と『クラスタ管理ガイド』マニュアルを参照。

`clu_upgrade`

現在のクラスタの特定のメンバで
`clu_upgrade` コマンドを実行することにより、ローリング・アップグレードを実行できる。`clu_upgrade` コマンドをいつどのように使用するかは、`clu_upgrade(8)` と第 7 章を参照。

1.4 フル・インストールの概要

この節では、新しい TruCluster Server バージョン 5.1B クラスタを形成するために必要な主な手順の概要を説明します。

1. 『クラスタ・ハードウェア構成ガイド』を参照して、システムおよびストレージ・ハードウェアとファームウェアを構成します。
2. AdvFS ファイル・システムを選択し、1 つまたは複数のディスクに Tru64 UNIX をインストールします。ディスクは、第 1 クラスタ・メンバとなるシステムのプライベート・ディスク、またはシステムがアクセスできる共用バス上のディスクのいずれかです。

注意

Tru64 UNIX オペレーティング・システムは 1 つのシステムにのみインストールします。クラスタ・メンバになるシステムごとにオペレーティング・システムをインストールする必要はありません。

3. ネットワーク・サービスやタイム・サービスを含め、Tru64 UNIX システムを構成します。クラスタで使用するアプリケーションをロードして構成します。

4. TruCluster Server のライセンスとサブセットをロードします。
5. `clu_create` コマンドを実行して第 1 クラスタ・メンバのブート・ディスクを作成し、クラスタ単位¹のルート (/), /usr, および /var AdvFS ファイル・システムを作成します。
6. Tru64 UNIX システムを停止し、第 1 メンバのクラスタ・ブート・パーティションのあるディスクをブートします。システムのブートにより、シングル・メンバ・クラスタが形成され、クラスタ単位のルート (/), /usr, および /var の各ファイル・システムがマウントされます。
7. root としてログインし、`clu_add_member` コマンドを実行してメンバをクラスタに追加します。次のメンバを追加する前に、新しいメンバをブートしてください。

1.5 必要なライセンス

クラスタ単位のライセンスはありません。ライセンスはクラスタ・メンバごとにインストールする必要があります。

第 1 メンバの場合には、Tru64 UNIX をインストールしライセンスを登録してから、TruCluster Server ライセンス (TCS-UA) をロードして登録し、シングル・メンバ・クラスタを作成します。TruCluster Server サブセットは TruCluster Server ライセンスを登録しなくてもロードできますが、クラスタは TruCluster Server ライセンスをロードして登録するまで作成できません。

メンバをクラスタに追加するたびに、TruCluster Server ライセンスを登録しなければなりません。新しいメンバをブートすると、そのメンバは Tru64 UNIX のクラスタ単位のルート (/), /usr, および /var の各ファイル・システムを使用するため、Tru64 UNIX ライセンスも登録する必要があります。さらに、そのメンバで必要となる、その他すべてのアプリケーション・ライセンスも登録しなければなりません。

注意

TruCluster Server ライセンスがないシステムもブートできます。
その場合、システムはクラスタに参加して、マルチユーザ・モー

¹ クラスタ単位という言葉は、どのメンバが稼働しているかにかかわらず、それらのファイル・システムがすべてのクラスタ・メンバからアクセス可能でなければならないことを意味します。したがって、すべてのクラスタ・メンバから直接アクセスできるストレージになければなりません。

ドでブートしますが，ログインできるのは `root` のみ (最大 2 ユーザ) です。CAA (Cluster Application Availability) デモン `caad` は起動されません。システムからは，ライセンスのロードを促すライセンス・エラー・メッセージが表示されます。このポリシーにより，ライセンスのチェックが強制的に実行されますが，緊急時にはシステムのブート，ライセンス登録，および修復が行えます。

1.6 TruCluster Server サブセット

表 1-2 に，TruCluster Server サブセットを示します。

表 1-2: TruCluster Server サブセットの内容

サブセット名	説明	内容
TCRBASE540	TruCluster 基本コンポーネント	インストール・チェックおよびすべての必須クラスタ・コンポーネント
TCRMAN540	TruCluster リファレンス・ページ	リファレンス・ページと例
TCRMIGRATE540	TruCluster 移行コンポーネント	TruCluster Production Server Software バージョン 1.5 またはバージョン 1.6，あるいは TruCluster Available Server Software からアップグレードする場合に使用する移行ユーティリティ

表 1-3 に，Tru64 UNIX システムのルート (`/`)，`/usr`，および `/var` の各ファイル・システムで，TruCluster Server バージョン 5.1B の各サブセットに必要なディスク容量の概算をメガバイト (MB) で示します。クラスタを作成するために必要な最小ディスク・サイズおよび推奨する容量については，表 2-2 に示します。

表 1-3: TruCluster Server サブセットのサイズ

サブセット	ルート (/)ファイル・システム (MB)	/usrファイル・システム (MB)	/varファイル・システム (MB)	合計 (MB)
TCRBASE540	0.7	49.1	10.0	59.8
TCRMAN540	0	0.9	0	0.9
TCRMIGRATE540	0	1.1	0	1.1
合計 (MB)	0.7	51.1	10.0	61.8

1.7 Tru64 UNIX のサブセット

Tru64 UNIX をインストールする場合、ルート (/)、/usr、および /var の各ファイル・システムのタイプは AdvFS を選択しなければなりません。

Tru64 UNIX オペレーティング・システムをインストールするには、サイトのポリシーで禁止されていないかぎり、クラスタ内のシステムの種類に関係なくすべてのサブセットをロードしてください。後で別のシステムを追加したり、インストールしていないサブセットに依存するアプリケーションをインストールしたりすることがあります。

注意

クラスタにさまざまなタイプのシステムが存在する場合は、各ハードウェア構成のサポートに必要となる、オプションの Tru64 UNIX サブセットがロードされることを確認してください。たとえば、キーボードやグラフィック・カードを正しく動作させるには特定のサブセットが必要となるため、すべてのキーボード・サブセットとフォント・サブセットをロードします。

ソフトウェア・サブセットは後で追加することもできますが、この時点では1つのシステムのみを扱っているため、クラスタを作成する前にインストールするほうが簡単です。既存のクラスタにサブセットをインストールする方法については、『クラスタ管理ガイド』で説明しています。

1.8 一般的な注意事項

インストール時の一般的な注意事項としては、次のものがあります。

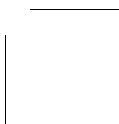
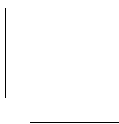
- 独自のカーネルを構築する場合、クラスタでは /vmunix はコンテキスト依存シンボリック・リンク (CDSL) になることに注意してください。

```
/vmunix -> cluster/members/{memb}/boot_partition/vmunix
```

CDSL は、他のシンボリック・リンクと同様に扱ってください。つまり、ファイルをコピーしてもリンクは保持されますが、ファイルを移動するとリンクは置き換えられます。カーネルを /vmunix に (コピーではなく) 移動すると、シンボリック・リンクは実際のファイルで置き換えられます。

CDSL については、『クラスタ概要』を参照してください。CDSL の使用方法および修復方法は、『クラスタ管理ガイド』で説明しています。

- ベース・オペレーティング・システムでは、Tru64 UNIX のインストールに必要なメモリの最低量が決まっています。クラスタでは、メンバごとにこの量より少なくとも 64 MB 多いメモリを用意する必要があります。たとえば、ベース・オペレーティング・システムに最低限必要なメモリが 128 MB である場合、クラスタで使用する各システムには、少なくとも 192 MB のメモリが必要です。
- TruCluster Server では、UNIX ファイル・システム (UFS) の読み取りがクラスタ単位で行えます。つまり、あるメンバが UFS ファイル・システムをマウントして、クラスタの他のメンバから読み取り専用でアクセスさせることができます。UFS ファイル・システムを読み取り/書き込みでマウントすることもできます。ただし、その場合にそのファイル・システムに読み取り/書き込みアクセスができるのは、そのファイル・システムをマウントしたメンバに限られます (ローカル使用のみ)。他のクラスタ・メンバはそのファイル・システムにアクセスできません。そのメンバがダウンした場合、フェイルオーバーは行われません。



インストールの準備

この章では、クラスタ・ソフトウェアをインストールする前に必要な作業について説明します。表 2-1 に、準備作業のガイドライン、および、詳細な説明のある参照先を示します。

表 2-1: 準備作業

作業	参照先
情報チェックリストを作成する。	付録 A
Tru64 UNIX および TruCluster Server ソフトウェアのインストールに必要なマニュアル類と PAK (Product Authorization Key) がそろっていることを確認する。	2.1 節
TruCluster Server 『クラスタ・リリース・ノート』に目を通す。	『クラスタ・リリース・ノート』
クラスタに組み込む予定のすべてのシステムについて、ハードウェアとファームウェアが、インストールを開始できる状態かどうかを確認する。(新バージョンの Tru64 UNIX および TruCluster Server では、通常、新しいバージョンの AlphaServer SRM ファームウェアを必要とする)。	TruCluster Server 『QuickSpecs』, 『クラスタ・ハードウェア構成ガイド』
割り当てるメンバ ID を決める。	2.2 節
必要な IP 名と IP アドレスを取得する。	2.3 節
クラスタ用にハードウェアをインストールする。	2.4 節 および 『クラスタ・ハードウェア構成ガイド』
Tru64 UNIX および TruCluster Server のインストールで使用するディスクとパーティションを決定する。将来のローリング・アップグレードに必要なスペースを確保する。	2.5 節
各クラスタ・メンバに割り当てるポートを決定し、クォーラム・ディスクを構成する場合は、そのディスクに割り当てるポートも決定する。	2.6 節および 『クラスタ管理ガイド』
コンソール変数を設定する。	2.7 節

2.1 必要なマニュアルおよび PAK

インストールを開始する前に、次のマニュアルを用意してください。

- 『クラスタ・ハードウェア構成ガイド』
- 『クラスタ・リリース・ノート』
- 『クラスタ概要』
- 『クラスタ高可用性アプリケーション・ガイド』
- 『クラスタ管理ガイド』
- Tru64 UNIX 『インストール・ガイド』
- Tru64 UNIX 『インストール・ガイド — 上級ユーザ編』
- Tru64 UNIX 『リリース・ノート』
- Tru64 UNIX 『システム管理ガイド』
- Tru64 UNIX 『ハードウェア管理ガイド』
- Tru64 UNIX 『ネットワーク管理ガイド：接続編』
- Tru64 UNIX 『ネットワーク管理ガイド：サービス編』
- Tru64 UNIX 『セキュリティ管理ガイド』
- メンバごとの Tru64 UNIX PAK (Product Authorization Key)
- メンバごとの TruCluster Server PAK

2.2 メンバ ID

クラスタ・メンバは、それぞれ固有のメンバ ID を持ち、その値は 1 ~ 63 の整数です。クラスタ・ソフトウェアはメンバ ID を使用して、クラスタの各メンバを識別します。クラスタにメンバを追加すると、省略時の値として、使用できる次のメンバ ID がインストール・プログラムにより提供されます。インストール時に、提供された省略時の設定を使用することも、任意の使用されていないメンバ ID を入力することもできます。

たとえば、メンバ `pepicelli` および `polishham` から構成される 2 ノード・クラスタがあるとします。クラスタ・ソフトウェアをインストールする際に、省略時のメンバ ID を使用すると、`pepicelli` にはメンバ ID として 1 が割り当てられ、`polishham` にはメンバ ID として 2 が割り当てられます。

```
pepicelli      member ID = 1
polishham      member ID = 2
```

メンバがクラスタに追加されるとき、そのメンバID が generic サブシステムの memberid 変数の値として、そのメンバの sysconfigtab ファイルに書き込まれます。

2.3 IP 名と IP アドレス

クラスタの作成およびメンバの追加を行う場合は、次の IP 名および IP アドレスの情報がが必要です。

- 省略時のクラスタ別名として使用される、クラスタ名と IP アドレス (2.3.1 項)
- 各クラスタ・メンバについて、それぞれの外部ネットワーク・インターフェースのためのホスト名、IP 名および IP アドレス (2.3.2 項)
- 各クラスタ・メンバについて、そのメンバの仮想クラスタ・インターコネクトのための IP 名と IP アドレス (2.3.3 項)

注意

Memory Channel インターコネクトでは、メンバごとに IP 名と IP アドレスがそれぞれ 1 つ必要です。インストール・プログラムは省略時の IP 名と IP アドレスを設定します。

LAN インターコネクトでは、メンバごとに IP 名と IP アドレスがそれぞれ 2 つ必要です。インストール・プログラムは省略時の IP 名と IP アドレスを設定します。

クラスタ・メンバはすべて、Memory Channel または LAN ハードウェアのどちらかを使用するように構成しなければなりません。同じクラスタ内で種類の異なるインターコネクトを混在させることはできません。

IP アドレスの割り当てについてのガイドラインは、Tru64 UNIX 『ネットワーク管理ガイド：接続編』で説明しています。RFC 1918 では、IANA (Internet Assigned Numbers Authority) により、次の IP アドレス空間ブロックがプライベート・インターネット用として予約されています。

```
10.0.0.0      - 10.255.255.255
172.16.0.0    - 172.31.255.255
```

192.168.0.0 - 192.168.255.255

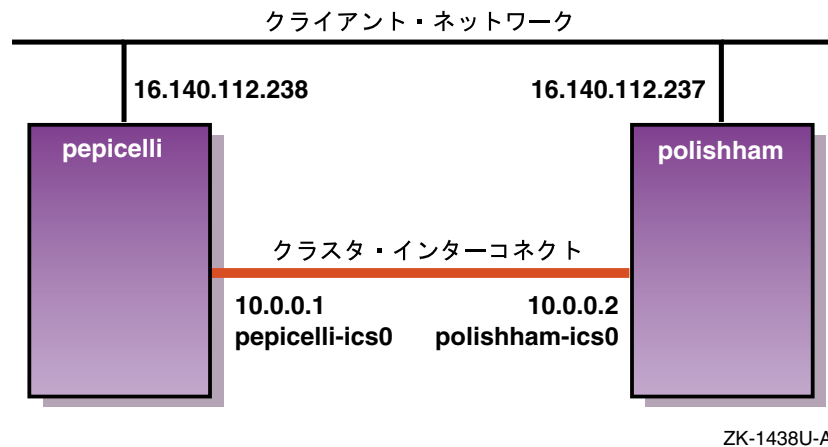
インストール・ソフトウェアがクラスタ・インターコネクト・ネットワークに省略時のアドレスを設定する際は、クラス C ネットワークのアドレスを使用します (サブネット・マスク 255.255.255.0)。

メンバのホスト名、クラスタ名、各メンバのクラスタ・インターコネクトに対応する名前など、ホスト名形式の文字列を使用する属性については、RFC 952 で定義され RFC 1123 で修正されたホスト名の標準命名規則に従い、文字として、英字の A ~ Z と a ~ z、数字の 0 ~ 9、'.' (ピリオド)、および '-' (ハイフン) のみを使用しなければなりません。ホスト名の最初の文字は、英字または (RFC 1123 での修正に従い) 数字でなければなりません。

以降の各項では、deli というサンプル・クラスタの作成に使用する名前とアドレスを示します。このクラスタには、pepicelli と polishham という 2 つのメンバがあり、クラスタ・インターコネクトに Memory Channel ハードウェアを使用しています。図 2-1 に、クラスタ deli のネットワーク・トポロジを示します。

図 2-1: クラスタのネットワーク・インタフェース

クラスタ名 (省略時のクラスタ別名): deli
省略時のクラスタ別名 IP アドレス: 16.140.112.209



2.3.1 クラスタ名と IP アドレス

各クラスタには、クラスタ名が付けられています。クラスタ名は、いずれかのクラスタ・メンバのホスト名ではなく、クラスタ全体に割り当てるホスト

2-4 インストールの準備

名です。この名前は、省略時のクラスタ別名に関連付けられたホスト名、つまりクラスタ全体に割り当てられた IP アドレスです。外部クライアントがこのクラスタ名を使用すれば、個々のクラスタ・メンバではなくクラスタにサービスを要求できます。

次に示す例は、クラスタ `deli` の完全修飾クラスタ名と省略時のクラスタ別名 IP アドレスです。

```
16.140.112.209    deli.zk3.dec.com    deli    # default cluster alias IP address
                                     # cluster host name
                                     # (default cluster alias name)
```

省略時のクラスタ別名 IP アドレスは、クライアントがルーティングできる有効なアドレスでなければなりません。クラスタのインターネット・サービスの中には、クラスタから発信して接続する場合に、このアドレスをソース・アドレスとして使用するものがあります。このアドレスが、クライアントが応答できるものでない場合、サービスは動作しません。

クラスタ別名アドレスは、ブロードキャスト・アドレスまたはマルチキャスト・アドレスとして使用してはなりません。クラスタ・インターコネクトが使用するサブネット内に存在してはなりません。また、クラスタ・メンバは IPv6 アドレスを使用したり公開したりできますが、クラスタ別名サブシステムは IPv6 アドレスをサポートしていません。したがって、IPv6 アドレスをクラスタ別名に割り当てることはできません。

RFC 1918 で定義されている次のいずれかのプライベート・アドレス空間のアドレスは、クラスタ別名 IP アドレスとして割り当てることができます。

```
10.0.0.0          - 10.255.255.255    (10/8 prefix)
172.16.0.0         - 172.31.255.255    (172.16/12 prefix)
192.168.0.0        - 192.168.255.255   (192.168/16 prefix)
```

ただし、省略時の設定では、クラスタ別名デーモン `aliasd` はこれらのプライベート・アドレス空間にある別名アドレスを通知しません。これらのアドレスを省略時のクラスタ別名に割り当てる場合は、シングル・メンバ・クラスタを作成した後に次の作業を行います。

1. `/etc/rc.config.common` ファイルの `CLUAMGR_ROUTE_ARGS` 変数の値を `resvok` に設定します。

```
# rcmgr -c set CLUAMGR_ROUTE_ARGS resvok
```
2. `cluamgr` コマンドを使用して、予約済みのアドレスを通知できることをクラスタ別名デーモン `aliasd` に知らせます。

```
# cluamgr -r resvok,start
```

/etc/rc.config.common ファイルで CLUAMGR_ROUTE_ARGS が resvok に設定されると、各クラスタ・メンバはブート時に cluamgr -r resvok コマンドを自動的に実行します。

2.3.2 メンバのホスト名と IP アドレス

各クラスタ・メンバには、ホスト名が付けられています。ホスト名は通常、システムのネットワーク・インタフェースに割り当てられた IP アドレスに関連付けられています。2.3.3 項で説明するクラスタ・インターコネクト・インタフェースの場合を除き、次に示す、クラスタ・メンバで使用するための名前および IP アドレスが必要です。

- 各メンバのホスト名

最初のクラスタ・メンバになるシステムに Tru64 UNIX をインストールする場合、および clu_add_member を実行してメンバをクラスタに追加する場合は、ホスト名を指定します。

- 各メンバの外部ネットワーク・インタフェースの構成時に使用する、インタフェースごとの IP 名と IP アドレス

最初のクラスタ・メンバになるシステムに Tru64 UNIX をインストールするときに、ネットワーク・インタフェースを構成します。

clu_create を実行する前に、そのシステムに別の外部ネットワーク・インタフェースを構成することもできます。ただし、クラスタ・インターコネクト・インタフェースは構成しないでください。このインタフェースは clu_create で構成します。

clu_add_member コマンドは、クラスタ・インターコネクト・インタフェースのみを構成します。追加のネットワーク・インタフェースは、新しいメンバを初めてブートするときに構成できます。構成するインタフェースのうち、少なくとも 1 つはシステムのホスト名を IP 名にすることをお勧めします。

たとえば、deli クラスタの各メンバ・システムには、外部ネットワーク・インタフェースが 1 つあります。この場合、各メンバのホスト名を、そのメンバの外部インタフェースの IP アドレスに関連づけられた IP 名として使用します。

```
16.140.112.238 pepicelli # First member's tu0 IP name and address.
                        # This is the Tru64 UNIX system.
                        # The host name is pepicelli.
16.140.112.237 polishham # Second member's tu0 IP name and address.
                        # The host name is polishham.
```


2.3.3 各メンバのクラスタ・インターコネクト用 IP 名と IP アドレス

クラスタには専用のクラスタ・インターコネクトが必要です。これは、クラスタ・メンバだけが使用する独立した別個の通信チャンネルです。クラスタ・インターコネクトが LAN の場合、すべてのメンバは 1 つのプライベート・サブネットに存在する必要があります。クラスタ・メンバはすべてクラスタ・インターコネクトに接続されていなければなりません。

- Memory Channel クラスタ・インターコネクトを使用するクラスタでは、各メンバ・システムの仮想クラスタ・インターコネクト・インタフェースに、IP 名と IP アドレスが必要です (クラスタ・メンバは、フェイルオーバーのための冗長物理インタフェースを持つことができますが、これらのインタフェースは 1 つの仮想ネットワーク・アドレスを共用しています)。メンバの Memory Channel クラスタ・インターコネクトは、`/etc/rc.config` ファイルの 1 つの `IFCONFIG` エントリと `NETDEV` エントリだけで指定され、システムにある物理インタフェースの数とは関係ありません。

省略時の設定では、インストール・プログラムにより、メンバのホスト名の短縮形に `-ics0` を付加した IP 名と、10.0.0 サブネット上の IP アドレスが提供され、アドレスのホスト部分にはメンバ ID が設定されます。たとえば、`pepicelli-ics0` と `10.0.0.1` のようになります。

注意

以前のリリースでは、サポートされるクラスタ・インターコネクト・デバイスは Memory Channel のみであり、メンバのホスト名の短縮形に付加される識別子は `-mc0` でした。

- LAN インターコネクトを使用するクラスタでは、メンバ間の通信が TCP/IP の 2 層で行われるので、以下の IP 名と IP アドレスが必要です。
 - メンバ・システムごとの仮想クラスタ・インターコネクト・デバイスの IP 名と IP アドレス。

省略時の設定により、インストール・プログラムは、仮想クラスタ・インターコネクトに対して 10.0.0 サブネット上の IP アドレスを割り当てます。その際、この IP アドレスのホスト部分にはそのメンバの ID が設定され、IP 名にはそのメンバのホスト名の短縮形に `-ics0` を付加した名前が設定されます。

- 物理 LAN インタフェース用のサブネットの異なる IP 名と IP アドレス。

省略時の設定により，インストール・プログラムは，物理クラスタ・インターコネクトに対して 10.1.0 サブネット上の IP アドレスを割り当てます。その際，この IP アドレスのホスト部分にはそのメンバの ID が設定され，IP 名は member`member-ID`-icstcp0 に設定されます。

注意

イーサネット・スイッチのアドレスは，通常，製造元が管理上の目的のために省略時の値を設定しています。このアドレスは，クラスタ・インストール・スクリプトが仮想クラスタ・インターコネクト・デバイスや物理 LAN インタフェースに設定する省略時の IP アドレスと競合する可能性があります。その場合は，クラスタ・インターコネクトの IP アドレスに，スイッチで使われていないアドレスを割り当てる必要があります。

省略時の設定により，インストール・プログラムは，仮想および物理 LAN インターコネクト・インタフェースの IP アドレスに Class C のサブネット・マスクを使用します。

クラスタ・インターコネクトの IP アドレスには .0 または .255 で終わるアドレスを割り当てることができません。このタイプのアドレスはブロードキャスト・アドレスであるとみなされます。このタイプのアドレスを持つシステムはクラスタに組み入れることができません。

次の例は，Memory Channel クラスタ・インターコネクトを使用して動作している deli クラスタの 2 つのメンバ，つまり pepicelli と polishham のクラスタ・インターコネクト用 IP 名と IP アドレスを示します。

```
10.0.0.1 pepicelli-ics0 # first member's virtual interconnect IP name and address
10.0.0.2 polishham-ics0 # second member's virtual interconnect IP name and address
```

また、次の例は、同じクラスタの2つのメンバがLAN インターコネクトを使って動作している場合の、メンバのクラスタ・インターコネクト用 IP 名および IP アドレスを示します。

```
# first member's cluster interconnect virtual interface IP name and address
10.0.0.1 pepicelli-ics0
# first member's cluster interconnect physical interface IP name and address
10.1.0.1 member1-icstcp0
# second member's cluster interconnect virtual interface IP name and address
10.0.0.2 polishham-ics0
# second member's cluster interconnect physical interface IP name and address
10.1.0.2 member2-icstcp0
```

クラスタ・インストレーション・スクリプトは、クラスタ・インターコネクトの仮想インタフェースと物理インタフェースの両方を、クラスタ・インタフェース (CLUIF) フラグでマークします。たとえば、次に示す `ifconfig -a` コマンドの出力では、クラスタ・インターコネクト仮想インタフェース (ics0) とクラスタ・インターコネクト物理インタフェース (ee0) がマークされています。

```
# ifconfig -a | grep -p CLUIF
ee0: flags=1000c63<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,SIMPLEX,CLUIF>
    inet 10.1.0.2 netmask ffffffff broadcast 10.1.0.255 ipmtu 1500
ics0: flags=1100063<UP,BROADCAST,NOTRAILERS,RUNNING,NOCHECKSUM,CLUIF>
    inet 10.0.0.2 netmask ffffffff broadcast 10.0.0.255 ipmtu 1500
```

次の例では、NetRAIN 仮想インタフェースであるクラスタ・インターコネクト物理インタフェース (nr0) が示されています。

```
# ifconfig -a | grep -p CLUIF
ee0: flags=1000c63<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,SIMPLEX,CLUIF>
    NetRAIN Virtual Interface: nr0
    NetRAIN Attached Interfaces: ( ee1 ee0 ) Active Interface: ( ee1 )
ee1: flags=1000c63<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,SIMPLEX,CLUIF>
    NetRAIN Virtual Interface: nr0
    NetRAIN Attached Interfaces: ( ee1 ee0 ) Active Interface: ( ee1 )
ics0: flags=11000c63<BROADCAST,NOTRAILERS,NOCHECKSUM,CLUIF>
    inet 10.0.0.2 netmask ffffffff broadcast 10.0.0.255 ipmtu 1500
nr0: flags=1000c63<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,SIMPLEX,CLUIF>
    NetRAIN Attached Interfaces: ( ee1 ee0 ) Active Interface: ( ee1 )
    inet 10.1.0.2 netmask ffffffff broadcast 10.1.0.255 ipmtu 1500
```

2.4 ハードウェア

Tru64 UNIX オペレーティング・システムのインストールの前に、ストレージ・ハードウェアおよびクラスタ・インターコネクト用 (Memory Channel または LAN) ハードウェアなど、クラスタ用のハードウェアをインストールします。

オペレーティング・システムをインストールまたはアップデートした後に、新しいハードウェア、たとえば、ネットワーク・アダプタなどを追加

する場合は、必ず `/genvmunix` をブートして、カスタマイズしたカーネルを構築してください。そうしない場合、システムのカーネル構成ファイルにそれらのハードウェア・オプションが組み込まれないため、TruCluster Server のインストール時に構築したカーネルで新しいハードウェアが認識されません。カーネルの構築については、Tru64 UNIX 『システム管理ガイド』で説明しています。

2.5 ディスク

この節では、次の事項について説明します。

- インストールに必要なディスク (2.5.1 項)
- LSM の考慮事項 (2.5.2 項)
- 2 ノード・クラスタの最小ディスク・レイアウト (2.5.3 項)
- ディスクの推奨空き容量 (2.5.4 項)
- ローリング・アップグレードの準備のために追加ディスク・スペースを割り当てる方法 (2.5.5 項)

注意

クラスタのルート (`/`)、`/usr`、および `/var` の各ファイル・システムは、高可用性である必要があるため、これらのファイル・システムはミラー化するようにしてください。RAID (Redundant Array of Independent Disk) コントローラまたは LSM (ソフトウェア RAID) を使用して、これらのファイル・システムをミラー化することができます。メンバのブート・パーティションやクォーラム・ディスクでは LSM (Logical Storage Manager) がサポートされていないので、この場合は RAID コントローラ (HSG80 など) を使用してミラー化してください。

ベース・オペレーティング・システムまたはクラスタに必要ないずれかのディスクに対して、Fibre Channel ストレージセットを使用する予定がある場合には、『クラスタ・ハードウェア構成ガイド』の Fibre Channel に関する章をお読みください。

Tru64 UNIX オペレーティング・システムをインストールする前に、ストレージ・ハードウェアをすべて構成してください。

2.5.1 インストールに必要なディスク

次の用途に割り当てるためのディスク (物理ディスクまたは論理ディスク) が必要です。

- Tru64 UNIX オペレーティング・システムを格納するための 1 つ以上のディスク。第 1 クラスタ・メンバとなるシステム上の 1 つ以上のプライベート・ディスク、またはシステムがアクセスできる共用バス上の 1 つ以上のディスク。
- クラスタ単位の AdvFS ファイル・システム、つまりルート (/), /usr, および必要に応じて /var を格納するための、共用バス上のディスク 1 つ以上。

別個のディスクを使用すれば、並行して処理される入出力が多くなるため、性能が向上します。また、さまざまな負荷分散の方法も提供されます。ディスク入出力分散についての詳細は、『クラスタ管理ガイド』を参照してください。

- 共用バス上の、メンバごとのブート・ディスク。メンバ・ブート・ディスクを共用バスに割り当てる理由は、次の 3 つ。
 1. メンバを追加する場合、現在のメンバが新しいメンバのブート・ディスクにアクセスできなければならない。
 2. ブート・ディスクをプライベート・バス上に設置した場合、そのメンバがブートできないときに、実行中のメンバからそのディスクをマウントしたり、問題を診断したりできない。
 3. clu_quorum のように、クラスタ管理コマンドのいくつかは、メンバがダウンしたときに、そのメンバのブート・ディスクにアクセスする必要がある。
- クラスタ内のメンバ数に応じて、クォーラム・ディスクとして動作する共用バス上のディスク 1 台。

この項で示す例では、システム・コンソール・デバイス (DK) の観点と、オペレーティング・システムのデバイス・スペシャル・ファイル (dsk) の観点から各ブート可能ディスクを説明します。ブート可能なディスクのコンソール・デバイス名がわかっている場合は、この項を読み進みながら付録 A のチェックリストに記入してください。ただし、どのスペシャル・ファイルが各物理デバイスに関連付けられるかは、ベース・オペレーティング・システム

ムをインストールするまでわかりません。ベース・オペレーティング・システムをインストールした後に、3.7 節の情報を参照してコンソール・デバイス名をデバイス・スペシャル・ファイル名にマップしてください。

以降の各項では、クラスタのインストールに必要なディスクについて詳しく説明します。

2.5.1.1 Tru64 UNIX ディスク (プライベートまたは共用)

ベース・オペレーティング・システムは、1 つ以上の物理ディスクまたは論理ディスクに、AdvFS ファイル・システムを使用してインストールします。ディスクは、最初のクラスタ・メンバとなるシステムの 1 つ以上のプライベート・ディスク、またはシステムがアクセスできる共用バス上の 1 つ以上のディスクのいずれかです。たとえば、次のように、DKA0 にベース・オペレーティング・システムをインストールし、オペレーティング・システムにより DKA0 が dsk0 にマップされるとします。

DKA0	dsk0a	root_domain#root
	dsk0g	usr_domain#usr
	dsk0h	var_domain#var

ベース・オペレーティング・システムは、クラスタの作成後も使用できるため、緊急時には、このベース・オペレーティング・システムをブートして、問題を診断し、解決することができます。

警告

ベース・オペレーティング・システムはすべての共用バス上のデータにアクセスできるため、クラスタの実行中には、ベース・オペレーティング・システムをブートしないでください。ベース・オペレーティング・システムは、クラスタが共用データへの複数システムのアクセスを制御するときに使用するバリアやロックを把握していません。クラスタが使用するディスクに Tru64 UNIX システムがアクセスすると、データが破損する恐れがあります。このため、ベース・オペレーティング・システムのブートは、クラスタ全体をシャットダウンした後にのみ行ってください。

制約事項: Tru64 UNIX オペレーティング・システム用に使用するディスクは、クラスタ単位のディスク、メンバ・ブート・ディスク、またはクォーラム・ディスクのいずれとしても使用できません。

2.5.1.2 クラスタ単位のルート (/), /usr, /var ディスク (共用)

クラスタの作成時には、インストール・スクリプトにより、Tru64 UNIX ディスクからクラスタ単位のファイル・システム用として指定したディスクに、Tru64 UNIX のルート (/), /usr, および /var の各ファイル・システムがコピーされます。

個々のメンバが立ち上がっているかダウンしているかに関係なく、クラスタ・メンバがディスクにアクセスできるようにするため、クラスタ単位のファイル・システムに使用する物理ディスクまたは論理ディスクを共用バス上で特定します。すべてのクラスタ・メンバからアクセスできなければならないファイル・システムがメンバのプライベート・バス上にある場合、そのメンバがクラッシュすると、残りのクラスタ・メンバはそのファイル・システムにアクセスできなくなります。

インストール時に、クラスタ単位のルート (/), /usr, および /var の各ファイル・システムを格納するディスク・パーティション名を指定します。各 AdvFS ファイル・システムは、別個のパーティションに格納しなければなりません。ただし、パーティションが同一のディスク上にある必要はありません。表 2-2 に、すべてのクラスタ・ファイル・システムに対して推奨する最小サイズを示します。

たとえば、次の例を参照してください。

dsk1b	cluster_root#root
dsk2c	cluster_usr#usr
dsk3c	cluster_var#var

注意

クラスタでは、/var を /usr の下に置くことはできません。
/var を /usr の下に置いて Tru64 UNIX をインストールすると、clu_create はそれらに対して別々にパーティションを作成します。

cluster_root#root ファイル・システムは、次の理由から b パーティションに配置することがよくあります。

- サイズ上の理由。b パーティションは通常は a パーティションより大きい。

- ブート不可。クラスタ・メンバは、`cluster_root#root` ファイル・システムをブートしない (各メンバは、クラスタ単位のルート (/) ファイル・システムではなく、`a` パーティションの最小ルート (/) ファイル・システムからブートされる)。

クラスタを作成して使用を開始した後に、さらにスペースが必要となった場合は、`cluster_root#root`、`cluster_usr#usr`、`cluster_var#var` のいずれかにボリュームを追加することができます。

注意

AdvFS ドメインのボリューム、および、すべてのクラスタ・メンバがアクセスしなければならないファイル・システムの LSM ボリュームにあるストレージ用に、すべてのメンバで共用しているバス上に物理デバイスを配置することを推奨します。

制約事項: クラスタ単位のルート (/) 、`/usr`、`/var` のいずれかのファイル・システムでディスクのパーティションが使用される場合、そのディスクは、メンバ・ブート・ディスクまたはクォーラム・ディスクとして使用できません。

2.5.1.3 メンバ・ブート・ディスク (共用)

それぞれのメンバには、ブート・ディスクとして、物理ディスクまたは論理ディスクがあります。ディスクは、メンバをクラスタに追加しているシステムにアクセス可能なバス上にある必要があります。このディスクは、共用バスに置いて、他のクラスタ・メンバからアクセスできるようにすることを推奨します。

例:

```
DKC400      dsk10      first member's boot disk [pepicelli]
DKC600      dsk12      second member's boot disk [polishham]
```

省略時の設定では、インストレーション・スクリプトにより、各メンバのブート・ディスクが次の 3 つのパーティションを格納するように再フォーマットされます。

- メンバのブート可能なルート・ファイル・システム用の `a` パーティション
- スワップ用の `b` パーティション
- クラスタの状態情報用の `h` パーティション (`cnx` の `fstype`)

メンバのブート・ディスクには、ファイル・システム `/usr` または `/var` はありません。

注意

新規メンバで不適切なディスクがブートされるのを避けるため、新規メンバのコンソールで、メンバのブート・ディスク用の `/dev/disk/dsk` スペシャル・ファイル名と、そのディスク用の DK デバイス名を取得する必要があります。

`clu_add_member` コマンドを使ってメンバを追加するとき、そのメンバのブート・ディスクに関する情報が表示されます。そこには、ディスクのシリアル番号 (ワールドワイド ID)、製造元、モデル番号、および物理的な位置 (バス/ターゲット/論理ユニット番号 (LUN)) が分かっている範囲で表示されます。物理的な位置情報 (バス/ターゲット/LUN) は、`clu_add_member` を実行したシステムから見える情報なので注意してください。システムとストレージの構成によっては、追加するメンバのコンソールで表示されるストレージ情報と違う場合があります。

`clu_add_member` によって得られる情報と 3.7 節に記載されている情報を使って、メンバ・ブート・ディスク用のスペシャル・ファイル名を物理デバイスにマップしてください。この情報は、付録 A の「メンバ属性」の表 (表 A-3) に記入してください。

メンバのディスクには 3 つ以上の必須パーティションを置くことができます。ただし、データ用にディスクを使用することはできません。

ディスクが次の必要条件を満たす場合、インストレーション・スクリプトはディスク・ラベルを変更しません。

- a パーティションのサイズが表 2-2 に示す最低必要条件を満たし、オフセットが 0 である。
- b パーティションのサイズが表 2-2 に示す最低必要条件を満たすとともに、そのオフセットがパーティション a のサイズより大きく、しかも、ファイル・システムのタイプが `swap` である。
- h パーティションのサイズがちょうど 1 MB であり、オフセットが b のサイズと b のオフセットを合わせた値よりも大きく、パーティショ

ンの終わりがディスクの最後に接していて、しかも、ファイル・システムのタイプが `cnx` である。

制約事項: メンバ・ブート・ディスクには、クラスタ単位のルート (`/`)、`/usr`、または `/var` ファイル・システムを置けません。メンバ・ブート・ディスクをクォーラム・ディスクとして使用することはできません。

推奨されてはいませんが、メンバのブート・パーティションにファイルセットを追加することは可能です (a パーティション)。ただし、メンバがクラスタから離れた場合、メンバのブート・パーティション・ドメインからマウントされたファイルセットはすべて強制的にアンマウントされます。それらのファイルセットは再配置することはできません。

2.5.1.4 クォーラム・ディスク (共用)

クォーラム・ディスクは、`h` パーティションにクラスタの状態とクォーラム情報を格納する物理ディスクまたは論理ディスクです。次に例を示します。

```
dsk7          quorum disk
```

クォーラム・ディスクを使用すると、クラスタ分断を防ぐとともに、クラスタの可用性が高められます。2 ノード・クラスタの場合は、クォーラム・ディスクを構成して、クォーラム・ディスクに 1 ポート与えてください。各メンバとクォーラム・ディスクがそれぞれ 1 ポートを持つ場合、ポートが 2 であるかぎり、クラスタはクォーラムを保持して、稼働することができます。このため、両方のメンバが稼働しているか、または一方のメンバが稼働していてクォーラム・ディスクが使用可能であれば、クラスタはクライアントの要求に応答できます。ただし、クォーラム・ディスクがない場合は、一方のメンバで障害が起きると、クラスタのクォーラムが失われて、クライアントの要求に応答できなくなります。

クォーラム・ディスクを構成するかどうかを決定する前に、本書の 2.6 節と、『クラスタ管理ガイド』のクォーラム・ディスクについての説明を参照してください。クォーラム・ディスクは、クラスタを作成した後でも構成できます。

注意

クラスタの移行中でも、クォーラム・ディスクにはクラスタ・メンバからアクセスがあるので、クォーラム・ディスクには入出力バリアは設けられません。こうした事情があるので、このディスクの残りのパーティションは、データ用に使用しないでください。

インストール時またはその後に `clu_quorum` コマンドを使用してクォーラム・ディスクを指定する場合、保存しておきたいデータを格納したディスクは指定しないでください。クォーラム・ディスクになったディスクはラベルが変更されるため、既存の生データまたはファイル・システムにはアクセスできなくなります。

制約事項: 1 つのクラスタに構成できるクォーラム・ディスクは 1 つだけです。クォーラム・ディスクを、すべてのクラスタ・メンバが直接接続されている共用バス上に配置します。そうでない場合、クォーラム・ディスクに直接接続されていないメンバは、直接接続されているメンバより前にクォーラム・ディスクを失うおそれがあります。メンバ・ブート・ディスクはクォーラム・ディスクとして使用できません。クラスタ単位のルート (/), /usr, /var のいずれかのファイル・システムを格納したディスクもクォーラム・ディスクとしては使用できません。

2.5.2 LSM の考慮事項

LSM を使用してクラスタ・ファイル・システムをミラー化する場合、Tru64 UNIX システムにあるルート (/) (`root_domain`), /usr (`usr_domain`), /var (`var_domain`) などのファイル・システムをミラー化しても、クラスタの `cluster_root`, `cluster_usr`, `cluster_var` ドメインに影響を与えません。たとえば、Tru64 UNIX システムの `root_domain` をミラー化した後、`clu_create` を実行してクラスタを作成しても、`cluster_root` は自動的にミラー化されません。したがって、LSM を使用して Tru64 UNIX システム上のファイル・システムではなく、TruCluster Server クラスタのファイル・システムをミラー化する場合は、Tru64 UNIX システムに LSM をセットアップした後、クラスタの最初のメンバをブートするまで、ファイル・システムのミラー化を行わないでください。

`clu_create` で単一メンバのクラスタを作成すれば、`volmigrate` コマンドを使用して `cluster_root`, `cluster_usr`, および `cluster_var` ドメインを LSM ボリュームに移動することができます。その際、オプションでそのボリュームをミラー化することもできます。これらのドメインを LSM ボリュームに移動する場合、`volmigrate` は、`volmigrate` 実行時に指定した新しいターゲット・ストレージに、`clu_create` で指定したディスク・パーティションのデータを移動するので注意してください (`clu_create` で指定したディスク・パーティションはボリュームでは使用されなくなり、他の用途

に使用できるようになります)。このため、`cluster_root`、`cluster_usr`、および `cluster_var` ドメインを LSM ポリ्यूームに移動するには、`clu_create` で指定するディスク・パーティションは一時的なストレージと考え、`volmigrate` に必要なストレージを別途準備する必要があります。

注意

`cluster_root`、`cluster_usr`、または `cluster_var` ドメインを LSM ポリ्यूームに移動するには `volmigrate` コマンドを使用しなければなりません。`volencap` コマンドは使用できません。

クラスタ内で LSM を使用方法についての詳細は、『クラスタ管理ガイド』を参照してください。また、クラスタ内で使用する `volmigrate` コマンドについては、`volmigrate(8)` を参照してください。

2.5.3 2 ノード・クラスタの最小ディスク・レイアウト

2 ノード・クラスタの場合、最小ディスク・レイアウトでは少なくとも 4 つのディスクを使用します (ただし、5 つ使用することをお勧めします)。

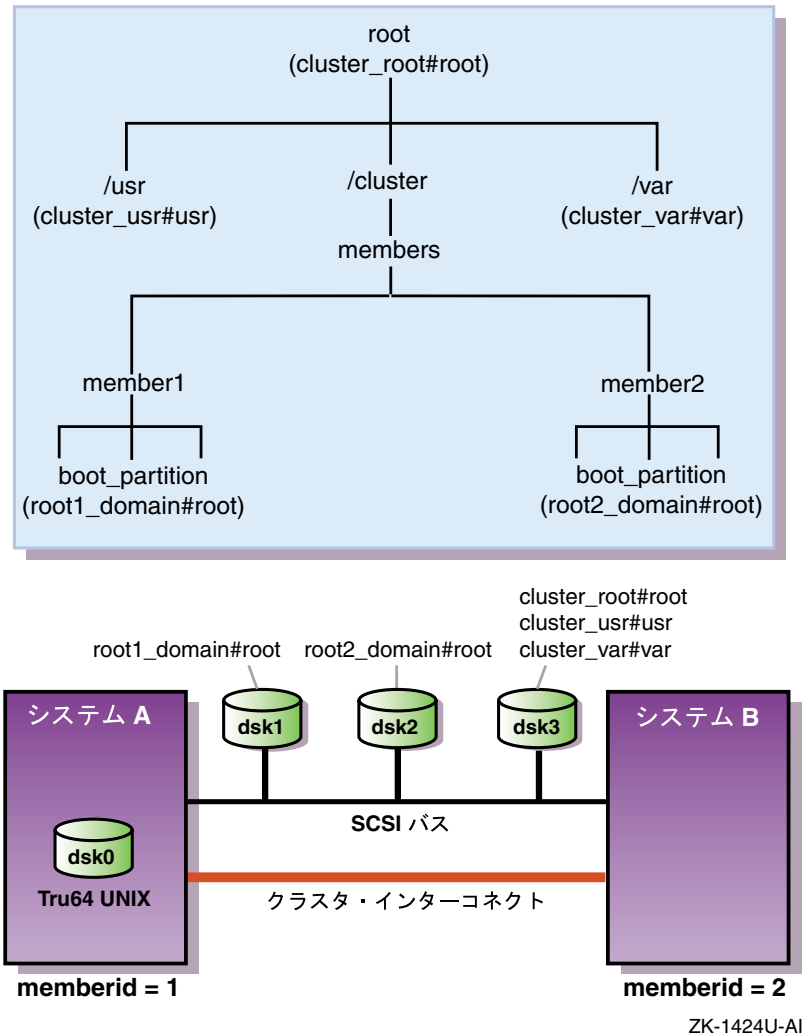
- 最初のクラスタ・メンバとなるシステムのためのディスク 1 台 (プライベート・ディスクまたは共用)。このディスクに Tru64 UNIX バージョン 5.1B オペレーティング・システムを格納する。
- クラスタ単位のルート (/), /usr, および /var の各ファイル・システムに使用する、共用バス上のディスク 1 台。
- メンバのブート・パーティションに使用する、共用バス上のディスク 2 台。
- クォーラム・ディスクとして使用する、共用バス上のディスク 1 台 (推奨)。

この最小ディスク・レイアウトでは、NSPOF (no-single-point-of-failure) 構成になっていません。すべてのメンバ・ブート・ディスクが共用バス上にありますが、クラスタ単位のルート (/), /usr, および /var の各ファイル・システムがミラー化されていません。これらの重要なファイル・システムのミラー化を推奨します。

NSPOF ハードウェア構成については『クラスタ・ハードウェア構成ガイド』で説明しています。

図 2-2 に、これらのディスクに格納されるファイル・システム間の関係と、これらのファイル・システムによって構成されるクラスタのディレクトリ構造を示します。クォーラム・ディスクにはファイル・システムが格納されないため、この図には示していません。

図 2-2: クラスタ単位のファイルとメンバ・ブート・パーティション



2.5.4 推奨するディスクの空き容量

表 2-2 に、ディスク・パーティションで必要とする推奨最小サイズを示します。メンバが 4 つ以下のクラスタでは、このサイズが妥当です。2.5.5 項

には、ローリング・アップグレードを実行する際に必要なディスク容量についての情報が記載されています。ディスク容量について計画する場合には、この項も参照してください。

表 2-2: 推奨するディスクの空き容量

ファイル・システム (種類)	パーティション	推奨最小サイズ	コメント
クラスタ・ルート (/) (AdvFS)	b	200 MB	<p>最小パーティション・サイズは 125 MB または $(1.125 \times \text{currently_used_root_size})$ より大きくなければならない。たとえば、Tru64 UNIX ルート (/) ファイル・システムが現在、そのパーティションの 158 MB を使用している場合、クラスタ単位のルートのパーティション・サイズは 177.75 MB になる ($1.125 \times 158 = 177.75$)。必ず必要なパーティション・サイズは 125 MB。</p> <p>容量に余裕がある場合、クラスタのルート・パーティションのサイズは Tru64 UNIX ルート・パーティションの 3 倍にする。</p> <p>TruCluster Server では、ルート・ファイル・システムへのボリュームの追加をサポートしている。クラスタ内では、AdvFS <code>addvol</code> コマンドを使用して <code>cluster_root#root</code> にボリュームを追加できる (AdvFS Utilities は別ライセンスである)。これにより、クラスタの規模の拡大に応じて、ルート・ファイル・システムを拡張できる。AdvFS ファイル・ドメインへのボリュームの追加についての詳細は、『クラスタ管理ガイド』を参照。</p>
クラスタ /usr (AdvFS)	g	1000 MB	<p>最小パーティション・サイズは 675 MB または $(1.125 \times \text{currently_used_usr_size})$ より大きくなければならない。絶対的な最小パーティション・サイズは 675 MB。</p>
クラスタ /var (AdvFS)	h	1000 MB	<p>最小パーティション・サイズは、360 MB または $(1.125 \times \text{currently_used_var_size})$ より大きくなければならない。絶対的な最小パーティション・サイズは 360 MB。</p>

表 2-2: 推奨するディスクの空き容量 (続き)

ファイル・システム (種類)	パーティション	推奨最小サイズ	コメント
メンバ・ブート・ディスク: ルート (AdvFS)	a	256 MB	メンバ・ブート・ディスクに必要なファイル・システムは、ルート、swap、およびクラスタ状態情報用の 1 MB のパーティション。ファイル・システムを格納していないディスクの場合、インストール・プロシージャにより、省略時の a パーティション・サイズが使用される。ファイル・システムを格納しているディスクの場合は、そのサイズで十分かどうか計算される。必要なパーティションが使用可能な場合、既存のパーティション・サイズを使用するかどうかを確認するプロンプトが表示される。
メンバ・ブート・ディスク: swap (スワップ)	b	システムによる	メンバ・ブート・ディスクで a パーティションと h パーティションを割り当てた後の残り部分。最小サイズは 256 MB。 共用ストレージ・バスのトラフィックを軽減するため、クラスタを作成した後にスワップを再構成して、各メンバの一次スワップ・パーティションをプライベート・ディスク上に配置することができる。クラスタ・メンバのスワップ領域再構成についての詳細は、『クラスタ管理ガイド』を参照。大量のメモリを装備したシステムのスワップ領域を構成する場合は、スワップ領域が十分であることを確認する。スワップ領域の計画についての詳細は、Tru64 UNIX 『インストール・ガイド — 上級ユーザ編』を参照。
メンバ・ブート・ディスク: CNX クラスタ状態用	h	1 MB ちょうど	h パーティション (fstype cnx) には、クラスタ状態情報が格納される。このパーティションは、インストール・プロシージャにより、各メンバのブート・ディスクに作成される。
クォーラム・ディスク	h	1 MB ちょうど	クォーラム・ディスクには、小容量のディスクを使用できる。すべてのクラスタ・メンバがクォーラム・ディスクから情報を検索できなければならないため、このディスクには入出力バリアがない。このため、このディスクにはファイル・システムを格納しない。

2.5.5 ローリング・アップグレードの準備のための追加のディスク容量の割り当て

ローリング・アップグレードでは、第 7 章で説明するように、次のようなディスク容量の要件があります。

- ルート (/) つまり `cluster_root#root` に 50 パーセント以上の空き容量がある。
- `/usr` つまり `cluster_usr#usr` に 50 パーセント以上の空き容量がある。
- `/var` つまり `cluster_var#var` に 50 パーセント以上の空き容量があり、さらに、オペレーティング・システムをアップデートする場合は、ベース・オペレーティング・システムの新しいバージョンのサブセットを格納するために 425 MB が追加が必要。

ローリング・アップグレード中は、以下のメンバ固有のディレクトリの内容が `/var` ファイル・システムにバックアップされます。

- `/cluster/members/membermemberID`
- `/usr/cluster/members/membermemberID`
- `/var/cluster/members/membermemberID`

これらのバックアップ・ファイルはローリング・アップグレードの完了前に削除されます。

すべてのクラスタ・メンバについてのディレクトリ内のファイルを保存するのに十分な空き容量が `/var` にない場合、ローリング・アップグレードは設定段階で失敗に終わります。

ローリング・アップグレードの際のバックアップ・ファイルに必要な容量を、以下の手順で最小限にすることができます。

- ローリング・アップグレードを始める前に、`/var/cluster/members/membermemberID/adm` ディレクトリのログ・ファイルおよびアカウント・ファイルをアーカイブし、メンバ固有のディレクトリから削除します。
- `/cluster/members/membermemberID/boot_partition` のメンバ・ブート・パーティションにデータを置かないように注意します。
- 各国語サポート (WLS) サブセットの独立した `i18n` ドメインがある場合は、そのドメインに 50 パーセント以上の空き容量が必要。
- タグ付きファイルは、メンバ・ブート・パーティションに置かれない。ただし、カーネルをブート・パーティションに移動する場合、プログラムに空き容量が必要になることがある。各メンバのブート・パーティションには、少なくとも 50 MB の空き容量を確保することを推奨する。

注意

メンバ・ルート・ドメインの最小ディスク・サイズは 256 MB です。

次の 2 つのオプションがあります。

- クラスタの作成時に、ルート (/)、/usr、/var、および i18n ドメイン (WLS サブセットの独立したドメインを使用する場合) のサイズを増やす。
- ローリング・アップグレードを開始する前に、addvol や dtadvfs のような AdvFS 管理ユーティリティを使用して、既存のドメインにボリュームを追加する (AdvFS Utilities は別ライセンスが必要)。

注意

addvol コマンドを使用してメンバのルート・ドメイン (メンバのブート・ディスクにある a パーティション) にボリュームを追加することはできません。その場合は、クラスタからそのメンバを外し、diskconfig または SysMan を用いて適切にディスクを設定した後、そのメンバをクラスタに戻す、という方法を取らなければなりません。

2.6 クォーラム：メンバ・ポートと期待ポート

クラスタに参加する可能性のあるシステムのポート数は、0 または 1 です。システムに割り当てられたポート数は、メンバ固有の /etc/sysconfigtab ファイルに cluster_node_votes の値として格納されています。各システムには、期待ポートと呼ばれる値もあり、/etc/sysconfigtab ファイルに cluster_expected_votes の値として格納されています。一般に、cluster_expected_votes の値は、潜在的なすべてのメンバの cluster_node_votes 値を合計したものです。クォーラム・ディスクを構成する場合は、クォーラム・ディスクに割り当てられるポートもこれに加えられます。

メンバ・ポート、クォーラム、および実行可能なクラスタの形成と保守は、簡単には説明できない複雑な問題です。クォーラムを計算する方法およびクラスタのメンバシップを保守する方法については、『クラスタ管理ガイド』で

説明しています。インストール時に潜在的なクラスタ・メンバに割り当てるポートの数を決める前に、このクォーラムに関する説明をお読みください。

次のリストでは、インストール時に `clu_create` と `clu_add_member` によって提供される省略時のノード・ポート設定について説明します。

注意

インストール時にノード・ポートの扱いがよくわからない場合は、間違ってクラスタが形成できなくなるという事態を避けるために、省略時の設定に手を加えないでください。ポートは、後ほどいつでも変更できます。

- 第1メンバのノード・ポートは必ず1です。
クラスタを形成できるのはポートを持つメンバのみであり、ポートを持たないメンバはクラスタへ参加できるだけです。そのため、クラスタを形成して自分自身が最初のメンバになるメンバには、ポートが必要です。
- クォーラム・ディスクの省略時のポート数は1です。0または1を割り当てることができます。クォーラム・ディスクを構成する場合は、1ポートを割り当ててください(シングル・メンバ・クラスタを作成し当分は別のメンバを追加しない場合を除いて、1ポートを割り当ててください。シングル・メンバ・クラスタにクォーラム・ディスクを構成した場合、ポートを割り当てたクォーラム・ディスクに障害が発生すると、クォーラムは失われることがあります)。クラスタに2つのメンバが存在する場合は、クォーラム・ディスクを構成し、そのディスクに1ポートを割り当てます。2番目のメンバをできるだけ早急に追加してください。
- 追加メンバ用に割り当てられる省略時のノード・ポートの値は、`clu_add_member` を実行するメンバで設定した期待ポートに基づいて決定されます。期待ポートの値が1の場合、`clu_add_member` によって割り当てられる省略時のポートは0になります。期待ポートが1より大きい場合は、1になります。有効な値は0と1です。

クォーラム・ディスクを持った2メンバ・クラスタを作成する場合、各メンバおよびクォーラム・ディスクに1ポートずつ割り当ててください。

2.3 節に示した `deli` クラスタの例では、インストール中に、第1メンバ(`pepicelli`)に1ポートが自動的に割り当てられ、1ポートのクォーラム・ディスクが構成され、2番目のメンバ(`polishham`)にも1ポートが割り当

てられました。このため、`cluster_expected_votes` の値は 3 であり、クォーラムには 2 ボート必要になります。

`clu_create` および `clu_add_member` で提供される省略時のボート値は、2 メンバ・クラスタ (2 つのボートを持つメンバとクォーラム・ディスク) の最大の可用性が得られるようにバイアスがかけられます。他のクラスタ構成の可用性を高めるには、表 2-3 および表 2-4 の提案を考慮に入れます。メンバ数が偶数 (2, 4, 6, 8) のクラスタでは、可用性を高めるためにクォーラム・ディスクの使用を強く推奨します。これは、特に、小規模なクラスタに当てはまります。しかしながら、クォーラム・ディスクを選択できない状況でも役立つように、表 2-4 に、クォーラム・ディスクを構成しないクラスタ用に最適なボートの割り当てを示します。

表 2-3: クォーラム・ディスクを持つクラスタのための推奨ボート割り当て

メンバ数	クォーラム・ディスクのボート	ノード・ボート割り当て (メンバ ID の昇順にコンマで区切ったノード)
1	0	1
2	1	1,1
3	0	1,1,1
4	1	1,1,1,1
5	0	1,1,1,1,1
6	1 ^a	1,1,1,1,1,1

^aメンバ数が偶数のクラスタでは、ボートを持つクォーラム・ディスクにより可用性が高まる。ただし、クラスタのメンバが 4 より多くなると、ボートを持つクォーラム・ディスクによってもたらされる可用性が相対的に小さくなる。

表 2-4: クォーラム・ディスクを持たないクラスタのための推奨ボート割り当て

メンバ数	ノード・ボート割り当て (メンバ ID の昇順にコンマで区切ったノード)
1	1
2	1,0
3 ~ 8	1, ..., 1

2.7 コンソール変数

次に示すのは、コンソール変数のリストです。これらの設定はクラスタ内のシステムにとって重要です。リストでは、`clu_create` または `clu_add_member` が変数を設定するのか、あるいはユーザが手動で変数を設定しなければならないのかを示しています。

`clu_create` または `clu_add_member` によってコンソール変数を設定できない場合は、ユーザが手動で設定する必要があります。

すべてのコンソール変数がすべてのシステムに適用されるわけではありません。

`boot_osflags`

```
set boot_osflags A
```

`boot_osflags` が A に設定されていると、システムは自動的にマルチユーザ・モードでブートします。クラスタの最初のメンバになるシステムをインストールする際、ベース・オペレーティング・システムが自動的にこの変数を設定します。メンバを追加する場合は、そのメンバを初めてブートする前に、この変数を設定してください。

`boot_reset`

```
set boot_reset on
```

`boot_reset` の値が `off` である場合、システム停止コマンドまたはブート・コマンドを実行すると、ウォーム・ブートだけが実行されます。値が `on` の場合には、フル・リセット (コールド・ブート) が実行されます。値を `on` に設定すると、システムはブートのたびに初期化された状態になります。

`clu_create` コマンドでは、クラスタの作成時に第 1 メンバの `boot_reset` が `on` に設定されます。`clu_add_member` コマンドでは一回限りのスクリプトが作成され、新しいメンバを初めてブートするときには、このスクリプトにより `boot_reset` が `on` に設定されます。

AlphaServer 8200 システムと 8400 システムの場合、この変数は手動で設定する必要があります。

`boot_reset` コンソール変数をサポートしないシステムでは、シャットダウン時に必要な初期化を行います。これらのシステムでは、ユーザによる操作の必要はありません。

bootdef_dev

```
set bootdef_dev boot_disk
```

bootdef_dev 変数には、ブート・コマンドをパラメータなしで入力した場合または AUTO_ACTION ブートが進行中の場合に、システムをブートするデバイスを順番に指定したリストを、コンマで区切り、スペースをいれずに指定します。

clu_create コマンドでは、クラスタの作成時に、第 1 メンバのブート・デバイスに設定されます。clu_add_member コマンドでは、一回限りのスクリプトが作成され、それにより新しいメンバの最初のブート時にブート・デバイスが設定されます。

AlphaServer 8200 システムおよび 8400 システムの場合、この変数は手動で設定する必要があります。

SCSI または Fibre Channel バスに接続され、複数バス・フェイルオーバー用に構成されている HS コントローラがハードウェア構成に含まれている場合、clu_create と clu_add_member で設定できるバスのパスは 1 つだけです。ブートする前にシステムを停止し、コンソールから手動で両方の値を設定しなければなりません。次に例を示します。

```
>>> set bootdef_dev device1,device2
```

各デバイスは、スペースではなくコンマで区切ります。コンソールから最初のデバイスをブートできなかった場合、リストの順序に従って、次のデバイスからブートが試行されます。デバイスのパスをすべて指定しなかった場合、メンバをブートしようとする「not connected」というコンソール・エラー・メッセージが表示されることがあります。

注意

システムでクラスタ・メンバのブート・ディスクとして Fibre Channel ディスクを使用している場合は、wwidmgr set または wwidmgr quickset コマンドのいずれかを使用してデバイスおよびポート・パス情報を構成していないかぎり、clu_create および clu_add_member のいずれでも bootdef_dev 変数を設定できません。また、それらの情報を構成している場合でも、clu_create および clu_add_member コマンドで設定できるのは 1 つのバスだけです。Fibre Channel を介してメンバのブート・ディスク

にアクセスする場合に `bootdef_dev` 変数に複数のパスを設定する手順については、『クラスタ・ハードウェア構成ガイド』の Fibre Channel に関する章を参照してください。

`boot_dev`

```
set boot_dev boot_disk
```

`boot_dev` 変数は、リブートに使用するディスクを指定します。

`clu_create` コマンドでは、クラスタの作成時に、第 1 メンバに対してこの変数が自動的に作成および設定されます。`clu_add_member` コマンドでは、一回限りのスクリプトが作成され、新しいメンバを初めてブートするときにこの変数が作成および設定されます。

AlphaServer 8200 システムと 8400 システムでは、この編集は手動で設定する必要があります。

`bus_probe_algorithm`

```
set bus_probe_algorithm new
```

コンソール変数 `bus_probe_algorithm` をサポートするシステムの場合、最初のシステムでこの変数に `new` を設定してから Tru64 UNIX をインストールしなければなりません。また、メンバを追加するときは、初めてブートする前にそのメンバでこの設定を行う必要があります。`bus_probe_algorithm` の値を `new` に設定すると、すべてのメンバ・システムで PCI (peripheral component interconnect) デバイスが必ず探索されます。

AlphaServer システム 800、1000、1000A、2000、2100、および 2100A では、コンソール変数 `bus_probe_algorithm` をサポートしています。これより新しいシステムでは、この変数を使用しません。

コンソール変数 `bus_probe_algorithm` を確認して、設定を行うには、次の手順に従ってください。

1. システムをシャットダウンしてコンソール・モードにし、システムの電源を一度オフにしてからオンにします。これにより、一時的なコンソール変数がすべてクリアされます。
2. `show` コマンドを入力します。

```
>>> show bus_probe_algorithm
```

この変数がサポートされる場合、コンソールにより変数名と現在の値 (old または new) が表示されます。

3. ファームウェアがコンソール変数 `bus_probe_algorithm` をサポートしており、現在の値が `old` である場合は、値を `new` に設定します。

```
>>> set bus_probe_algorithm new
```

2.8 サンプルのクラスタ構成チェックリスト

表 2-5、表 2-6、および表 2-7では、付録 A のチェックリストを使用して、サンプルの `deli` クラスタ構成に使用される情報の概要を示します。

表 2-5: サンプルの Tru64 UNIX システム属性

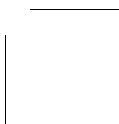
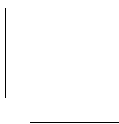
属性	値
ホスト名	pepicelli.zk3.dec.com
ホスト名の IP アドレス	16.140.112.238
Tru64 UNIX のルート・パーティション (たとえば, dsk0a)	dsk0a
Tru64 UNIX のコンソールからのルート・デバイス (たとえば, DKA0)	DKA0
Tru64 UNIX の /usr パーティション (たとえば, dsk0g)	dsk0g
Tru64 UNIX の /var パーティション (たとえば, dsk0h)	dsk0h

表 2-6: サンプルのクラスタ属性

属性	値
クラスタの完全修飾名	deli.zk3.dec.com
省略時のクラスタ別名 IP アドレス	16.140.112.209
クラスタ単位のルート・パーティション (たとえば, dsk1b)	dsk1b
クラスタ単位のルート (/) ディスク・デバイスのシリアル番号 (たとえば, WWID)	DEC RZ1CF-CF (C) DEC 50022303
クラスタ単位の /usr パーティション (たとえば, dsk2c)	dsk2c
クラスタ単位の /usr ディスク・デバイスのシリアル番号 (たとえば, WWID)	DEC RZ1CB-CS (C) DECQD2202330Y3WJL
クラスタ単位の /var パーティション (たとえば, dsk3c)	dsk3c
クラスタ単位の /var ディスク・デバイスのシリアル番号 (たとえば, WWID)	DEC RZ1CF-CF (C) DEC 50021480
クォーラム・ディスクを使用している場合のディスク・デバイス (たとえば, dsk7)	dsk7
クォーラム・ディスクを使用している場合のクォーラム・ディスクのシリアル番号 (たとえば, WWID)	DEC RZ28L-AS (C) DEC- JED716250N6TF6
クォーラム・ディスクを使用している場合にクォーラム・ディスクに割り当てられるポート数	1

表 2-7: サンプルのメンバ属性

属性	第 1 メンバ	第 2 メンバ	第 3 メンバ
メンバ・ホストの完全修飾名	pepi-cellli.zk3.dec.com	polish-ham.zk3.dec.com	
ホスト名の IP アドレス	16.140.112.238	16.140.112.237	
メンバ ID (memberid: 1 ~ 63)	1	2	
このメンバに割り当てられるポート数	1	1	
ブート・ディスク (たとえば, dsk10)	dsk10	dsk12	
コンソールからのブート・デバイス (たとえば, DKC400)	DKC400	DKC600	
ブート・デバイスのシリアル番号 (たとえば, WWID)	DEC RZ1CF-CF (C) DEC 50066053	DEC RZ1CF-CF (C) DEC 50066104	
ブート・デバイスの物理的位置 (bus/target/LUN)	bus-2-targ-11-lun-0	bus-2-targ-13-lun-0	
仮想クラスタ・インターコネクトの IP 名	pepicelli-ics0	polishham-ics0	
仮想クラスタ・インターコネクトの IP アドレス	10.0.0.1	10.0.0.2	
物理クラスタ・インターコネクトのデバイス名	Memory Channel		
物理クラスタ・インターコネクトの IP アドレス (LAN のみ)	n/a	n/a	
追加のネットワーク・インタフェースの IP 名	n/a	n/a	
追加のネットワーク・インタフェースの IP アドレス	n/a	n/a	



Tru64 UNIX オペレーティング・システムのインストールと構成

クラスタを作成する場合は、クラスタ・メンバになるすべてのシステムに Tru64 UNIX オペレーティング・システムをインストールする必要はありません。第 1 クラスタ・メンバになるシステムにだけ Tru64 UNIX オペレーティング・システムをインストールします。そのシステムを基にして第 1 クラスタ・メンバが作成されます。

`clu_create` コマンドを実行してシステムを第 1 クラスタ・メンバにするとき、必要となる Tru64 UNIX システムは 1 台のみです。その第 1 クラスタ・メンバをブートした後、`clu_add_member` を実行して追加メンバ用のブート・ディスクを作成します。それらの追加メンバには、ベース・オペレーティング・システムの個別コピーは必要ありません。ただし、Tru64 UNIX のライセンスは個別に必要です。

クラスタに各種のシステムが含まれる場合は、各種のハードウェア構成をサポートするために必要な、オプションの Tru64 UNIX サブセットをロードします。たとえば、キーボードとグラフィック・カードを正しく機能させるには特定のサブセットが必要になるため、キーボードとフォントのサブセットをすべてロードします。それらのサブセットは後からでもインストールできますが、Tru64 UNIX システムに十分なディスク・スペースが存在し、サイトのポリシーに違反しないのであれば、Tru64 UNIX オペレーティング・システムのインストール時にすべてのサブセットをロードしてください。

クラスタの作成は、Tru64 UNIX システムを完全に構成してから行うことにより、クラスタを作成する前に構成を確認できます。

ベース・オペレーティング・システムを構成するだけでなく、クラスタで使用する各種のレイヤード・プロダクトおよびアプリケーションも、ロードして構成しておきます。

ベース・オペレーティング・システム用に Fibre Channel ストレージセットを使用する場合は、TruCluster Server 『クラスタ・ハードウェア構成ガイド』の Fibre Channel に関する章をお読みください。

表 3-1 に、インストレーション作業を順序どおりに示すとともに、必要な情報の参照先も示しています。

注意

クラスタのローリング・アップグレードを実行する場合は、第 7 章へ進み、その章の指示に従ってください。

TruCluster Production Server Software または Available Server Software のバージョン 1.5 あるいはバージョン 1.6 から TruCluster Server バージョン 5.1B にアップグレードする場合は、第 8 章へ進み、その章の指示に従ってください。

TruCluster Memory Channel Software から TruCluster Server バージョン 5.1B にアップグレードする場合は、8.9 節へ進み、その節の指示に従ってください。

表 3-1: Tru64 UNIX のインストール

作業	参照先
すべてのストレージが正しくインストールされ、構成されていることを確認する。たとえば、HS コントローラを使用している場合、RAID セットおよびユニットが構成されていることを確認する。	『クラスタ・ハードウェア構成ガイド』
コンソール変数を確認する。	2.7 節
SRM ファームウェアを更新する。	3.1 節
Tru64 UNIX オペレーティング・システムをインストールする。	3.2 節および Tru64 UNIX 『インストレーション・ガイド』
基本的なサービスを構成する。	3.3 節および Tru64 UNIX 『ネットワーク管理ガイド：サービス編』
Secure Shell ソフトウェアを構成する。	3.4 節 および Tru64 UNIX 『セキュリティ管理ガイド』
エンハンスド・セキュリティを構成する (オプション)。	3.5 節および Tru64 UNIX 『セキュリティ管理ガイド』

表 3-1: Tru64 UNIX のインストール (続き)

作業	参照先
冗長ネットワーク・インタフェース用に NetRAIN を構成する (オプション)。	3.6 節 および 『クラス タ管理ガイド』
クラスタのインストールに必要なディ スクを構成する。	2.5 節, 3.7 節, およ び Tru64 UNIX 『システ ム管理ガイド』
LAN インターコネクトを使用するクラス タについて, ネットワーク・アダプタの デバイス名を取得する。	3.8 節

3.1 SRM ファームウェアのアップデート

Tru64 UNIX および TruCluster Server の新しいバージョンでは, 通常, AlphaServer SRM ファームウェアの新しいバージョンを必要とします。ファームウェアのアップデートは, ベース・オペレーティング・システムの Software Distribution Kit に含まれている Alpha Systems Firmware CD-ROM にあります。ファームウェアをアップデートする必要があるかどうかを判断するには, クラスタ内の各システムのタイプについて, TruCluster Server 『QuickSpecs』 およびファームウェアのリリース・ノートを参照してください。TruCluster Server 『QuickSpecs』の最新版は, 次の URL から入手できます。

http://www.tru64unix.compaq.com/docs/pub_page/spds.html

3.2 Tru64 UNIX オペレーティング・システムのインストール

注意

クラスタのインストールでは, Tru64 UNIX のルート (/), /usr, /var の各ファイル・システムがコピーされて, クラスタ単位のルート (/), /usr, /var の各ファイル・システムが作成されます。したがって, クラスタを作成する前に Tru64 UNIX システムを完全に構成しておくことをお勧めします。

Tru64 UNIX 『インストール・ガイド』で説明しているインストール・プロシージャを実行する前に、次のリストを参照し、これらの作業もインストール時に行ってください。

1. すべてのストレージ・デバイスに電源が入っていることを確認します。
2. クラスタ・インターコネクต์に使用するすべてのハブまたはスイッチに電源が入っていることを確認します。
3. 1 つ以上のディスクに Tru64 UNIX オペレーティング・システムをインストールします。ディスクは、第 1 クラスタ・メンバにするシステムのプライベート・ディスクまたはそのシステムがアクセスできる共用バス上のディスクのいずれかです。

注意

Tru64 UNIX オペレーティング・システムをインストールするときは、すべてのサブセットをロードしてください。

4. AdvFS (Advanced File System) ファイル・システムを使用します。
5. 各国語サポート (WLS) を独立したファイル・システムにインストールする場合、そのファイル・システムが存在するディスクは、すべてのクラスタ・メンバで共用するバス上に設置することをお勧めします。WLS のインストールについては、Tru64 UNIX 『インストール・ガイド — 上級ユーザ編』を参照してください。
6. LSM (Logical Storage Manager) を使用してクラスタ上のシステムをミラー化しようと計画している場合、LSM は Tru64 UNIX システム上に構成します。ルート・ディスク・グループ (rootdg) は、共用バス上にあるディスクに配置します。本書の 2.5.2 項と、TruCluster Server 『クラスタ管理ガイド』のクラスタに LSM を構成する方法についての章も参照してください。
7. ベース・オペレーティング・システムのインストール中にパッチ・キットを同時にインストールする場合は、パッチ・キットをインストールする前に TruCluster Server のサブセットをロードします。パッチの操作前に TruCluster Server キットがロードされていない場合は、TruCluster Server ソフトウェアのパッチがロードされません。Tru64

UNIX を初めてインストールする場合のパッチのインストール手順は、次のとおりです。

- a. Tru64 UNIX オペレーティング・システムをインストールして構成します。
 - b. `setld` コマンドを実行して、TruCluster Server キットをインストールします。
 - c. システムにパッチをインストールします。
 - d. `clu_create` コマンドを実行してシングル・メンバ・クラスタを作成します。
8. Tru64 UNIX オペレーティング・システムをインストールまたはアップデートした後に、新しいハードウェア、たとえば新しいネットワーク・アダプタなどを追加する場合は、必ず `/genvmunix` をブートして、カスタマイズしたカーネルを構築してください。そうしない場合、システムのカーネル構成ファイルにそれらのハードウェア・オプションが組み込まれないため、TruCluster Server のインストール時に構築したカーネルで新しいハードウェアが認識されません。カーネルの構成については、Tru64 UNIX 『システム管理ガイド』で説明しています。
9. この手順は、非同期転送モード (ATM) ネットワークに接続したシステムにのみ適用します。ATM LAN エミュレーション (LANE) のサポートを構成するには、`doconfig` で表示されるリストから必要なオプションを選択します。次に示す `doconfig` オプションのリスト (一部) では、LANE のサポートに必要なオプションにアスタリスク (*) が付加されています。

```
IP Switching over ATM (ATMIFMP)
* LAN Emulation over ATM (LANE)
  Classical IP over ATM (ATMIP)
* ATM UNI 3.0/3.1 Signalling for SVCs
```

3.3 基本的なサービスの構成

Tru64 UNIX 『ネットワーク管理ガイド：接続編』の説明に従って `netconfig` ユーティリティまたは SysMan Menu を実行し、システムの標準ネットワーク・インタフェース (1 つまたは複数) を構成します。 `ifconfig`, `ping`, `ftp`, および `telnet` などのネットワーク・ユーティリティを使用してネットワークが正しく構成されていることを確認します。

注意

この時点では、クラスタ・インターコネクトのインタフェースを構成しないでください。それらのインタフェースは、クラスタを作成するときに構成します。

次の基本的なサービスを構成します。

- ルーティング・デーモン。クラスタ別名のために `gated` は必須 (3.3.1 項)。
- タイム・サービス。ネットワーク・タイム・プロトコル (NTP) を推奨 (3.3.2 項)。
- ネーム・サーバ。たとえば、BIND (Berkeley Internet Name Domain) サーバなど (3.3.3 項)。
- クラスタがファイル・サーバになる場合は、NFS (ネットワーク・ファイル・システム) を構成 (3.3.4 項)。
- クラスタが NIS (ネットワーク情報サービス) のマスタ、スレーブ、クライアントのいずれかになる場合は、NIS を構成 (3.3.5 項)。
- クラスタが DHCP (Dynamic Host Configuration Protocol) サーバになる場合は、DHCP を構成 (3.3.6 項)。
- クラスタがメール・サーバになる場合は、メールを構成 (3.3.7 項)。
- クラスタがプリント・サーバになる場合は、印刷を構成 (3.3.8 項)。

各サービスを構成する場合は、以降の各項に目を通し、それらの情報も参考にして構成してください。

注意

クラスタを作成する前にこれらのサービスのいくつかを構成しなかった場合は、稼働中のクラスタにそれらのサービスを構成する方法を説明している『クラスタ管理ガイド』を参照してください。ただし、クラスタを作成する前に各サービスを構成したほうが簡単です。

3.3.9 項に Tru64 UNIX システムで推奨されるネットワーク構成が要約されているので、TruCluster Server のインストールを開始する前に参照してください。

3.3.1 ルーティング・デーモン

クラスタ別名ソフトウェアは、以下とともに動作するように設計されています。

- gated (省略時の設定)
- routed
- スタティック・ルーティング

ogated ルーティング・デーモンはサポートしていません。

gated ルーティング・デーモンの使用を推奨します。gated を使用すると、各クラスタ・メンバの別名デーモン aliasd は、そのメンバ固有の /etc/gated.conf.membern ファイルを作成します。

クラスタにおけるルーティング・デーモンの役割および gated 以外のルーティング機構に関する説明については『クラスタ管理ガイド』および『クラスタ概要』を参照してください。

3.3.2 タイム・サーバ

分散タイム・サービスを実行すると、各ファイル・システムとアプリケーションで使用されるタイム・スタンプをクラスタ全体で一貫させることができます。NTP (Network Time Protocol) デーモン (xntpd) などの分散タイム・サービスを構成することをお奨めします。NTP を使用すると、きわめて正確に同期をとることができるとともに、時刻ソースの信頼性を追跡することができます。NTP については、Tru64 UNIX 『ネットワーク管理ガイド：サービス編』および ntp_intro(7) を参照してください。

システム時刻の同期がとれていない場合、正確なタイム・スタンプを必要とするチェックはすべて失敗します。サイトで NTP を使用していない場合は、使用するタイム・サービスが、RFC 1035 『*Network Time Protocol (Version 3) Specification, Implementation and Analysis*』で定義されている細分性の仕様を満たしているかどうかを必ず確認してください。

各クラスタ・メンバのシステム時刻は数秒以上ずれていてはならないため、時刻の設定に `timed` デーモンは使用しないでください。

使用しているネットワーク上にタイム・サーバが存在せず、クラスタをタイム・サーバとして使用したい場合は、Tru64 UNIX 『ネットワーク管理ガイド：サービス編』 および `ntp_manual_setup(7)` を参照してください。Tru64 UNIX システムをタイム・サーバとして構成する場合は、クラスタを作成する前に構成してください。

クラスタを信頼できる時刻ソースとして機能させるには、クラスタを作成した後、時刻サービスを高可用性サービスにします。高可用性サービスのセットアップ方法については、『クラスタ高可用性アプリケーション・ガイド』を参照してください。

Tru64 UNIX システムで NTP を使用している場合、クラスタの作成時に Tru64 UNIX システムの NTP 設定が第 1 クラスタ・メンバに継承されます。メンバを追加すると、各メンバはそれぞれ他のメンバのピアになります。

3.3.3 ネーム・サーバ

クラスタは、ネーム・サーバまたはクライアント、あるいはその両方として機能することができます。ベース・オペレーティング・システムを BIND サーバとして構成すると、そのクラスタの中では、BIND デーモン `named` が自動的にシングル・インスタンスの高可用性サービスとして構成されます。

ネーム・サーバ (たとえば BIND) を Tru64 UNIX システム上に構成する場合は、`/etc/svc.conf` の `hosts` エントリで、`local` サービスを `bind` サービスと `yp` サービスの前に指定してください。次に例を示します。

```
hosts=local,bind,yp
```

3.3.4 NFS

TruCluster Server クラスタは非常に信頼できる NFS (ネットワーク・ファイル・システム) サービスをクライアントに提供できるので、クラスタを作成する前に、ベース・オペレーティング・システムを NFS サーバとして構成することをお勧めします。システムを NFS クライアントとして構成することもできます。

ベース・オペレーティング・システムを NFS サーバとして構成した場合、NFS `lockd` および `statd` デーモンは、クラスタ内のシングル・インスタ

ンス高可用性サービスとして構成されます。したがって、クラスタは可用性の高い NFS サーバとなります。

3.3.5 NIS

クラスタが NIS (ネットワーク情報サービス) のマスタ、スレーブ、クライアントのいずれかになる場合は、クラスタを作成する前に NIS を構成します。NIS の構成方法については、Tru64 UNIX 『ネットワーク管理ガイド：サービス編』を参照してください。

クラスタ・メンバは、すべて同じ NIS ドメインに入っていなければなりません。Tru64 UNIX オペレーティング・システムがマスタ・サーバとして構成されている場合は、すべてのクラスタ・メンバが NIS マスタとして構成されます。Tru64 UNIX オペレーティング・システムがスレーブ・サーバまたはクライアントとして構成されている場合は、すべてのクラスタ・メンバがスレーブまたはクライアントとして構成されます。

3.3.6 DHCP

ベース・オペレーティング・システムを DHCP (Dynamic Host Configuration Protocol) サーバとして構成した場合、DHCP デーモン `joind` は、クラスタ内のシングル・インスタンスの高可用性サービスとして構成されます。

警告

システムを DHCP クライアントとして構成しないでください。クラスタのメンバを DHCP クライアントとすることはできません。これは、各メンバのクラスタ・インターコネクトに関連付けられた IP 名および IP アドレスは、クラスタを形成するためには絶対的に不変でなければならないためです。

3.3.7 メール・サーバ

クラスタがメール・サーバになる場合は、クラスタを作成する前にメールの構成を行います。メールの構成方法については、Tru64 UNIX 『システム管理ガイド』および TruCluster Server 『クラスタ管理ガイド』を参照してください。

3.3.8 プリント・サーバ

クラスタがプリント・サーバになる場合は、クラスタを作成する前に印刷関連の構成を行います。プリンタの構成方法については、Tru64 UNIX 『システム管理ガイド』および TruCluster Server 『クラスタ管理ガイド』を参照してください。

3.3.9 ネットワーク・サービスの概要

TruCluster Server のインストールを開始する前に、Tru64 UNIX システムの推奨するネットワーク構成について、表 3-2 に要約を示します。また、この表では `clu_create` によって第 1 クラスタ・メンバに渡される構成情報と、`clu_add_member` によって追加のクラスタ・メンバに渡される構成情報も示しています。

表 3-2: ネットワーク・サービスの要約

サービス/デーモン	推奨事項	コメント
ルーティング・デーモン	<code>gated</code>	<code>ogated</code> と <code>routed</code> は使用しない。
タイム・サーバ	NTP	第 1 クラスタ・メンバは、ベース・オペレーティング・システムの構成を継承する。追加のメンバは自動的にピアとして構成される。
ネーム・サーバ	オプション (BIND または NIS、あるいはその両方)	すべてのクラスタ・メンバは、同じドメイン (DNS および NIS) 内になければならない。 <code>svc.conf</code> で、 <code>hosts=local,bind,yp</code> と設定する。ベース・オペレーティング・システムがネーム・サーバとして構成されている場合、そのネーム・サーバはクラスタ内の高可用性サービスとして構成される。クラスタが BIND サーバである場合、クラスタ別名がクラスタ単位の <code>/etc/resolv.conf</code> ファイル内で BIND サーバの名前になる。
NFS	オプション (ベース・オペレーティング・システム上にサーバとして設定することも、クライアントとして構成することもできる)	すべてのクラスタ・メンバは、 <code>lockd</code> デーモンおよび <code>statd</code> デーモンのクライアント・バージョンを実行する。サーバ・バージョンの <code>lockd</code> デーモンおよび <code>statd</code> デーモンは一度に 1 つのクラスタ・メンバで実行され、それらは高可用性サービスとして構成される。

表 3-2: ネットワーク・サービスの要約 (続き)

サービス/デーモン	推奨事項	コメント
NIS	オプション	すべてのクラスタ・メンバは、同じドメイン内になければならない。ベース・オペレーティング・システムがマスタ・サーバとして構成されている場合、NIS はクラスタ内の高可用性サービスとして構成される。ベース・オペレーティング・システムがスレーブ・サーバまたはクライアントとして構成されている場合は、すべてのクラスタ・メンバがスレーブまたはクライアントとして構成される。
DHCP	オプション (サーバのみ)	ベース・オペレーティング・システム上に構成された場合、DHCP はクラスタ内の高可用性サービスとして構成される。
メール	オプション	クラスタを作成する前に構成すること。
プリント・サーバ	オプション	クラスタを作成する前に構成すること。

3.4 Secure Shell ソフトウェアの構成

Secure Shell ソフトウェアは、接続先より送信されたデータを暗号化するだけでなく、ホストとユーザを認証するための機構も提供します。Secure Shell ソフトウェアを構成して、`rsh` などのコマンドが自動的に Secure Shell 接続を使用するようにできます。クラスタ・ソフトウェアは `rsh` を使用してクラスタ・メンバにコマンドを送信するため、Secure Shell ソフトウェアを有効にすることを推奨します。

クラスタ作成後に Secure Shell ソフトウェアを構成することもできますが、クラスタ作成前に行うほうが簡単です。

Secure Shell ソフトウェアについては Tru64 UNIX 『セキュリティ管理ガイド』を参照してください。

3.5 エンハンスト・セキュリティの構成 (オプション)

クラスタ内にエンハンスト・セキュリティを構成するには、次の手順に従います。

- Tru64 UNIX バージョン 5.1B をインストールするときに、エンハンスト・セキュリティ・サブセット (名前が OSFC2SECxxx および OSFXC2SECxxx の形式であるもの) を選択します。
- TruCluster Server のライセンスとサブセットをロードしてクラスタを作成する前に、Tru64 UNIX システムでエンハンスト・セキュリティを完全に構成しておいてください。クラスタを作成する前にエンハンスト・セキュリティを構成するほうが、クラスタを作成した後にクラスタ・メンバを個別に構成するよりも簡単です。また、Enhanced (C2) Security など一部の構成要素は、カーネルの再構築またはシェアード・ライブラリ置換に依存しているため、それらの構成要素をクラスタの作成後に構成した場合は、クラスタ全体をリブートする必要があります。

セキュリティ・オプションの選択と構成の方法については、Tru64 UNIX 『セキュリティ管理ガイド』を参照してください。

3.6 冗長ネットワーク・インタフェース用の NetRAIN の構成 (オプション)

システムに 2 つのネットワーク・インタフェースが存在する場合、NetRAIN により 2 つのインタフェースを用いた完全に冗長性のある構成を設定することができます。詳細については『クラスタ管理ガイド』を参照してください。

3.7 クラスタのインストールに必要なディスクの構成

Tru64 UNIX オペレーティング・システムをインストールしてブートした後、クラスタのインストール時に使用するディスクのサイズと位置を確認します。各メンバをそのブート・ディスクからブートしなければならないので、メンバ・ブート・ディスクの位置は特に重要です。メンバを間違ったブート・ディスクからブートすると、通常はブート中にパニックが発生します。パニックの種類によっては、他のクラスタ・メンバに影響を与えることがあります。

3.7.1 パーティション・サイズ

クラスタ・ファイル・システムおよびメンバ・ブート・ディスクとして使用するディスクのパーティションが、表 2-2 に示した最小サイズに関する必要条件を満たしていることを確認します。また 2.5.5 項の情報を使用して、将来のローリング・アップグレードに必要な容量を確保します。

クラスタ単位のルート (/), /usr, /var の各ファイル・システム用に使用するディスク (1 つまたは複数) には, clu_create コマンドを実行する前にディスク・ラベルを付けておく必要があります。これは, clu_create コマンドを実行すると, パーティション情報を入力するプロンプトが表示されるので, それらのパーティションが存在していなければならないためです。

ディスクのラベル付けおよび構成を行うには, disklabel コマンドおよび diskconfig コマンドを使用します。また, SysMan Station を使用してストレージ構成の表示および変更を行うこともできます。

注意

メンバ・ブート・ディスクとして使用するディスクおよびクォーラム・ディスクには, 使用前にラベルを付ける必要はありません。

次の例は, あるディスクに対して disklabel コマンドを実行したときの出力を示しています。このディスクの b, g, h の各パーティションに, 2 つのノードで構成されるクラスタのクラスタ単位のルート (/), /usr, /var の各ファイル・システムが保持されます。シリンダ情報は示されていません。

```
# disklabel -r dsk6
# /dev/rdisk/dsk6c:
type: SCSI
disk: RZ1DF-CB
label:
flags: dynamic_geometry
bytes/sector: 512
sectors/track: 168
tracks/cylinder: 20
sectors/cylinder: 3360
cylinders: 5273
sectors/unit: 17773524
rpm: 7200
interleave: 1
trackskew: 28
cylinderskew: 72
headswitch: 0          # milliseconds
track-to-track seek: 0 # milliseconds
drivedata: 0

8 partitions:

#          size      offset  fstype  fsize  bsize
a:       786432         0   unused      0      0
b:     1552131     786432   unused      0      0
c:     17773524         0   unused      0      0
d:       5400220    1572864   unused      0      0
e:       5400220    6973084   unused      0      0
f:       5400220   12373304   unused      0      0
g:       7595651    2338563   unused      0      0
h:       7839310    9934214   unused      0      0
```

3.7.2 ディスク位置の特定

SysMan Station と `hwmgr` コマンドを使用すると、デバイス・スペシャル・ファイル名を物理的な位置にマップできます。誤ったクラスタ・メンバ・ディスクをブートしないためには、ディスクをブートするコンソールで、正しいデバイスにデバイス・スペシャル・ファイルをマップすることが重要になります。

注意

クラスタに必要な任意のディスクに対して Fibre Channel ストレージセットを使用する場合は、『クラスタ・ハードウェア構成ガイド』の Fibre Channel に関する章をお読みください。

次の汎用 `hwmgr` コマンドを実行すると、Tru64 UNIX で認識されている各ディスクのデバイス名、物理位置、およびワールドワイド ID (WWID) が表示されます。

```
# hwmgr -get attr -a dev_base_name \
-a phys_location -a name -category disk|more
```

第 1 メンバのクラスタ・ブート・ディスクを見つけるには、第 1 クラスタ・メンバとして構成するシステムで `clu_create` コマンドを実行すればよいいため、問題にはなりません。`clu_create` コマンドにより、コンソール変数 `bootdef_dev` が正しいブート・ディスクに設定されます。ただし、別のシステム用のブート・ディスクを構成するには、システムで `clu_add_member` を実行します。別のシステムには、オペレーティング・システムがありません。そのシステムのストレージ構成について得られる情報は、そのコンソールからの情報のみです。`clu_add_member` に指定する `dsk` 名は、新しいメンバのコンソールからそのディスクをブートするときに使用する DK 名にマップする必要があります。

`clu_add_member` コマンドを使ってメンバを追加するとき、そのメンバのブート・ディスクに関する情報が表示されます。そこには、ディスクの製造元、モデル番号、物理的な位置 (バス/ターゲット/論理ユニット番号 (LUN))、およびシリアル番号 (WWID) が分かっている範囲で表示されます。`clu_add_member` コマンドは、次の `hwmgr` コマンドを使用して情報を収集します。


```
# hwmgr -get attr -a dev_base_name=new_member_boot_disk \  
-a serial_number -a manufacturer -a model -a phys_location
```

clu_add_member コマンドは、hwmgr コマンドの出力を再フォーマットし、次のフォーマットでディスクの位置情報を表示します。

```
Manufacturer:  
Model:  
Target:  
Lun:  
Serial Number:
```

Fibre Channel ストレージセットを使用している場合は、wwidmgr -show wwid コマンドを使用して、新しいメンバのコンソールからディスクの位置を特定できます (『クラスタ・ハードウェア構成ガイド』に、wwidmgr コマンドの使い方を示す例がいくつか記載されています)。

メンバ・ブート・ディスクのシリアル番号が取得できない場合、デバイス・スペシャル・ファイルをコンソール・デバイス名に確実にマップできるかどうかは、次に述べるように、ストレージ構成がシンメトリックであるかどうかに依存します。

- シンメトリックである場合：クラスタ内の各システムは物理的に同一で、構成もまったく同一である。バス 1、バス 2、バス 3、...バス n がすべてのシステム上で同じものであることが分かっている。コンソールから見ると、それぞれの DKB バスは、クラスタ全体にわたって物理的に同じバスを参照している。DKC、DKD、... DKZ についても同様である。そのような場合には、hwmgr コマンドまたは sysman hw_devices コマンドを使用して dsk スペシャル・ファイルに関連付けられたバス、ターゲット、論理ユニット番号 (LUN) を表示できる。その後、そのバス、ターゲット、および LUN を使用し、3.7.2.1 項の手順に従ってディスクのコンソール・デバイス名を生成できる。
- シンメトリックでない場合：あるシステム上のバス名 (たとえば DKB) は、別のシステムの DKB と物理的に同じバスではない。そのバスをベース・オペレーティング・システムがバス 1 とみなしても、そのバスがすべてのコンソールから DKB として参照される保証はない。こうしたことは、クラスタ内に、SCSI アダプタの搭載数が異なる各種のシステムが存在する場合、または、メンバ・ブート・ディスク用のバスに対してアダプタの挿入やバスのケーブル接続を不用意に行ってシンメトリックなストレージを構成した場合などに起こる。3.7.2.2 項の手順を参照。

3.7.2.1 シンメトリックなストレージ構成

次の手順は、新しいメンバのコンソールで `show device` コマンドを使用して表示したディスクの名前に、`/dev/disk/dsk*` スペシャル・ファイル名をマップする方法を説明しています。この手順は、`hwmgr` コマンドを実行するシステムが、実行後にディスクのブートを試みるシステムと同じタイプである場合にのみ採用できます。つまり、モデル・システム、種類、および数も同じストレージ・アダプタが両方のマシンの同じスロットに装着されており、同じ共用ストレージに接続されている場合です。後ほどクラスタ・メンバ・ブート・ディスクとして構成するディスクを含んでいるバスは、どちらのコンソールでも同じように表示されます。2つのシステムがこうした必要条件を満たしているかどうかがよくわからない場合は、3.7.2.2 項に示すシンメトリックでない場合の手順を参照してください。

1. Tru64 UNIX システムで `hwmgr` コマンドまたは SysMan Station を使用して、`dsk` ファイル名を物理デバイスのバス、ターゲット、および LUN にマップします。共用ストレージのデバイス・スペシャル・ファイル名は、Tru64 UNIX システムおよびクラスタのものと同一になります。たとえば、`dsk13` のバス、ターゲット、および LUN を取得するには次のように入力します。

```
# hwmgr -view devices | grep dsk13
56: /dev/disk/dsk10c DEC RZ1CF-CF (C) DEC bus-1-targ-11-lun-0c
```

このディスクは、バス 1、ターゲット 11、LUN 0 にあります。

2. そのディスクをブートするシステムのコンソールで、次のように `show device` コマンドを入力します。

```
>>> show device
      :
      :
dskb100.1.0.12.0      DKB100      RZ28M  1104
      :
      :
```

このコンソールから見た場合、SCSI ディスクのクラス・ドライバ指定子は、`DK` です。バスの番号は、ファームウェアでバスが検出された順序になります。最初に検出されたバスは `A`、次に検出されたバスは `B` で、以下同様に続きます。同じように、最初のターゲットは `0`、第 2 のターゲットは `100`、第 3 のターゲットは `200` となります。`hwmgr` の `bus-1-targ-1-lun-0` という出力は、通常、`DKB100` に変換されます。

注意

Fibre Channel ストレージセットを使用している場合は、コンソール・コマンド `wwidmgr -show wwid` を使用することで、ユーザ定義 ID (UDID) とそれに対応するワールドワイド ID (WWID) をリストにして表示させることができます。

3. `dsk` 名、`DK` 名、および `WWID` を表 A-3 の「メンバ属性」の表に記入してください。

3.7.2.2 シンメトリックでないストレージ構成

ストレージ構成がシンメトリックでない場合は、あるシステムのオペレーティング・システムから見たストレージ (`dsk`) を別のシステムのコンソールから見たストレージ (`DK`) に確実にマップできるコマンドはありません。

ディスクへのマップを、コンソールまたはオペレーティング・システムのどちらから行うかは、ユーザの判断によります。次の手順では、オペレーティング・システムから作業を開始し、他のシステムのコンソールに移動しています。

1. `hwmgr` コマンドを使用して、ブート・ディスクとして使用するディスクのアクティビティ・ライトを点滅させます。たとえば、スペシャル・ファイル `/dev/disk/dsk13` を介してアクセスするディスクのライトを点滅させるには、次のように入力します。

```
# hwmgr -flash light -dsf dsk13
```
2. ライトが点滅しているディスクを見つけ、ストレージ・キャビネット内での位置を把握します。
3. そのキャビネットから、そのディスクをブート・ディスクとして使用するシステムへのケーブルをたどります。

この時点で、ディスクの SCSI バス上の位置およびシステム内の SCSI アダプタの位置が判明します。システムによるアダプタの番号付け方法が分かっているれば、そのアダプタの SCSI バス番号も推定できます。クラスタを構成したときにストレージ・マップを描いていれば、その情報が役立ちます。さまざまなディスクがある場合は、ディスクのモデル番号と `WWID` も参考になります。

次の例では、追跡したディスクが、システムの 2 番目の SCSI アダプタの 2 番目のデバイスであると想定しています。

そのディスクをブートするシステムのコンソールで、次のように `show device` コマンドを入力します。

```
>>> show device
      :
dkb100.1.0.12.0      DKB100      RZ28M  1104
      :
```

このコンソールから見た場合、SCSI ディスクのクラス・ドライバ指定子は、DK です。バスの番号は、ファームウェアでバスが検出された順序になります。最初に検出されたバスは A、次に検出されたバスは B で、以下同様に続きます。同じように、最初のターゲットは 0、第 2 のターゲットは 100、第 3 のターゲットは 200 となります。2 番目の SCSI アダプタ上の第 2 ターゲットは、DKB100 となります。そのディスクが RZ28M であると分かっている場合は、その情報も選択項目を絞り込むのに役立ちます。ただし、当然ながら、そのバス上のディスクがすべて RZ28M である場合は除きます。

注意

Fibre Channel ストレージセットを使用している場合は、コンソール・コマンド `wwidmgr -show wwid` を使用することで、ユーザ定義 ID (UDID) とそれに対応するワールドワイド ID (WWID) をリストにして表示させることができます。

4. dsk 名、DK 名、および WWID を表 A-3 の「メンバ属性」の表に記入しておいてください。

3.8 LAN インターコネクトを使用する際のネットワーク・アダプタのデバイス名の取得

`clu_create` コマンドまたは `clu_add_member` コマンドを実行する前に、構成するメンバに存在する適切なイーサネット・ネットワーク・アダプタの名前を調べます。以下の条件に合えば、アダプタとして適切です。

- インストールされていること
- 構成されていないこと

- 100 Mb/秒または 1000 Mb/秒の全二重オペレーション用のスイッチ・ポートおよび他のメンバの LAN インターコネクト・ネットワーク・アダプタと互換性があるように設定されていること

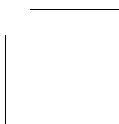
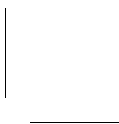
クラスタ・インストール・コマンドでは、物理的なイーサネット・ネットワーク・アダプタの名前または NetRain 仮想インタフェースの名前のいずれかを使用できます。

警告

クラスタ・インストール・コマンドは、LAN インターコネクトに対して自動的に NetRAIN 仮想インタフェースを構成します。したがって、`clu_create` スクリプトを実行する前に手動で NetRAIN デバイスを作成しないでください。手動で作成した場合の結果については、『クラスタ管理ガイド』を参照してください。

候補となるネットワーク・アダプタのデバイス名を調べる場合は、クラスタの最初のメンバになるシステムで、`ifconfig -a` コマンドを実行します。また、速度と転送モードを調べる場合は、`hwmgr -get attr -cat network` コマンドを使用します。

クラスタに追加するシステムのデバイス名を調べる場合は、まずシステムを『Tru64 UNIX Operating System Volume 1』CD-ROM からブートしてください。ブート中に、イーサネット・アダプタの UNIX デバイス名がスクロールされながらコンソール画面に表示されます。システムのブート後の UNIX シェルで、`ifconfig -a` コマンドでネットワーク・アダプタのデバイス名を表示させたり、`hwmgr -get attr -cat network` コマンドでそれらのプロパティを表示させたりすることができます。



シングル・メンバ・クラスタの作成

Tru64 UNIX システムをインストールおよび構成したら、この章の指示に従って、このシステムをクラスタの第 1 メンバにします。表 4-1 に、作業を実行順に示すとともに、必要な情報の参照先を示します。

注意

第 1 メンバのブート・ディスクまたはクォーラム・ディスクに対して Fibre Channel ストレージセットを使用する場合は、『クラスタ・ハードウェア構成ガイド』の Fibre Channel に関する章をお読みください。

表 4-1: シングル・メンバ・クラスタの作成作業

作業	参照先
クラスタの作成に必要な情報を収集する。	第 2 章
Tru64 UNIX をインストールおよび構成する。	第 3 章
TruCluster Server のライセンスを登録する。	4.1 節
TruCluster Server サブセットをロードする。	4.2 節
必要に応じてパッチをあてる。	4.3 節
clu_create コマンドを実行する。	4.4 節
第 1 メンバのクラスタ・ブート・ディスクをブートする。	4.4 節および 4.5 節
重要な構成ファイルのオン・ディスク・バックアップ・コピーを作成する。	4.6 節
シングル・メンバ・クラスタのフル・バックアップを実行する。	Tru64 UNIX 『システム管理ガイド』

4.1 TruCluster Server のソフトウェア・ライセンス登録

TruCluster Server キットには、ライセンス PAK (Product Authorization Key) が含まれています。この PAK は、TruCluster Server のライセンスを登録するときに使用します。TruCluster Server のライセンスについては、1.5 節を参照してください。PAK がない場合には、弊社のサービス担当者に連絡してください。

ライセンス PAK の登録についての詳細は、Tru64 UNIX 『*Software License Management*』、`lmf(8)`、および `lmfsetup(8)` を参照してください。

4.2 TruCluster Server サブセットのロード

TruCluster Server キットをロードするには、次の手順に従います。

1. スーパユーザ (root) としてログインします。
2. TruCluster Server ソフトウェア・バージョン 5.1B のキットが含まれているデバイスまたはディレクトリをマウントします。
3. キットを配置するディレクトリを指定する `setld -l` コマンドを入力します。たとえば、CD-ROM を `/mnt` にマウントする場合には、次のように入力します。

```
# setld -l /mnt/TruCluster/kit
```

次のサブセット・インストール・オプションから 1 つを選択できます。

- All mandatory subsets only (すべての必須サブセットのみ)
- All mandatory and selected optional subsets (すべての必須サブセットと選択したオプション・サブセット)
- All mandatory and all optional subsets (すべての必須サブセットとすべてのオプション・サブセット)

“All mandatory and all optional subsets” オプションを選択することをお勧めします。

オプションを選択すると、サブセットをシステムにコピーする前に、インストール・プロシージャにより十分なファイル・システム・スペースがあるかどうかを確認されます。

4.3 パッチの適用

最初のクラスタ・メンバとなるシステムに、パッチ・キットをインストールします。パッチをあてる場合、ベース・オペレーティング・システム・パッチおよびクラスタ・パッチの両方を適用する必要があります。これは、TruCluster Server サブセットがシステム上に存在する場合にのみ行います。

クラスタ・サブセットがロードされる以前に、システムにパッチがあてられていた場合はシステムに再度パッチをあてる必要があります。最新の Tru64 UNIX および TruCluster Server パッチをインストールすることを推奨します。

パッチは、`clu_create` コマンドを実行する前に適用してください。

パッチの詳細については『*Patch Kit Installation Instructions*』を参照してください。パッチは次のアドレスからダウンロードすることができます。

<http://ftp.support.compaq.com/patches/.new/unix.shtml>

4.4 `clu_create` コマンドの実行

`/usr/sbin/clu_create` コマンドを実行すると、Tru64 UNIX システムからクラスタの第 1 メンバが作成されます。

このコマンドは、`/usr` ファイル・システム・ディスクの空き容量が `vmunix` (通常、この製品の合計は 100 MB 以下) のサイズの最低 3 倍はあることを確認します。ディスクの容量が足りない場合、カスタマイズされたクラスタ・カーネルを構築するのに十分な容量がないことを告げるメッセージが表示されます。ディスク容量が足りないため、カーネルを構築することができなくても、`clu_create` コマンドは正常に終了します。その後、クラスタ `genvmunix` をブート・ディスクからブートし、シングル・メンバ・クラスタにおいてカスタマイズされたカーネルの構築を試みることができます。

注意

`clu_create` コマンドは、`vdump` および `vrestore` コマンドを使用して、クラスタ単位のルート (`/`)、`/usr`、および `/var` の各ファイル・システムを操作します。Tru64 UNIX システム上にあるこれらファイル・システムのいずれかに、ネットワーク・ファイル・システム (NFS) がマウントされていて、そのファイル・システムの NFS サーバがダウンしている場合、ファイル・

システムのダンプを取ろうとすると、`vdump` はハングします。
`automount` デーモンを実行する場合には、NFS ファイル・システムがマウントおよびアンマウントされ、同様にハングする可能性があるので注意してください。

`clu_create` コマンドを実行する前に、すべての NFS ファイル・システムをアンマウントするか、またはそれらがアクセスできることを確認します。あるいは、`clu_create` コマンドを実行する前に、Tru64 UNIX システムをリブートすることにより、すべての不要なマウントを解除することもできます。`automount` デーモンを実行するシステムでは、`clu_create` コマンドより先に `/sbin/init.d/nfsmount stop` コマンドを実行すると、自動マウントを無効にすることができます。

`/usr/sbin/clu_create` コマンドを実行します。このコマンドは、シングル・メンバ・クラスタの作成に必要な情報を入力するためのプロンプトを表示します。付録 A のチェックリストの情報を使用して、プロンプトに回答します。このコマンドには各質問へのオンライン・ヘルプもあります。関連するヘルプ・メッセージを表示するには、プロンプトで `help` または疑問符 `?` を入力します。

4.4.1 クラスタ・インターコネクトの選択

`clu_create` コマンドはクラスタ・インターコネクトのタイプ (LAN または Memory Channel) を入力するように求めます。このとき、Memory Channel アダプタがインストールされていると、省略時の値として Memory Channel が示されます。

- クラスタ・インターコネクトのタイプに Memory Channel を指定すると、`clu_create` は省略時の物理クラスタ・インターコネクト・インタフェースとして `mchan0` を割り当てます。
- LAN を指定すると、`clu_create` コマンドから物理クラスタ・インターコネクト・デバイス名を入力するように求められますが、その場合は、以下のオプションがあります。
 - `tu0`, `ee0`, または `alt0` のように、1 つのネットワーク・インタフェースのデバイス名を指定する。

- 複数のネットワーク・インタフェースのデバイス名を繰り返し指定する。`clu_create` コマンドでは、これらのインタフェースを NetRAIN 仮想インタフェースに構成するという指定ができる。

注意

すでに存在する NetRAIN デバイスのデバイス名 (たとえば、`/etc/rc.config` ファイルで定義されているもの) を指定すると、`clu_create` コマンドから、その NetRAIN デバイスを物理クラスタ・インターコネクト・デバイスとして再定義するかどうかの確認を求められます。"yes" と答えると、`clu_create` コマンドは、`/etc/rc.config` ファイルから NetRAIN デバイスの定義を削除し、そのデバイスを `/etc/sysconfigtab` ファイルの `ics_ll_tcp` スタンザにクラスタ・インターコネクト・デバイスとして定義します。

その後、`clu_create` コマンドは `membermember-ID-icstcp0` という形式で物理クラスタ・インターコネクト・デバイスの IP 名を作成します。省略時には、このデバイスに IP アドレスとして `10.1.0.member-ID` を割り当てます。

`clu_create` コマンドがクラスタ・インターコネクトを定義するために書き込む `/etc/sysconfigtab` 属性のリストは、表 C-2 を参照してください。

4.4.2 `clu_create` が実行するタスク

`clu_create` コマンドでは、次のタスクが実行されます。

- クラスタ単位のルート (`/`)、`/usr`、`/var` の各ファイル・システム、および第 1 メンバのブート・ディスクの設定
- クォーラム・ディスクの構成 (オプション)
- クラスタ・コンポーネントでのカーネルの構築

カーネルの構築が成功すると、`clu_create` により新しいカーネルが第 1 メンバのブート・ディスクにコピーされます。カーネルの構築に失敗すると `clu_create` により警告メッセージが表示されますが、第 1 メンバの作成は続行されます (ブート・ディスクからクラスタ `genvmunix`

をブートして、シングル・メンバ・クラスタ上のカーネルを構築することができます)。

- ブート関連コンソール変数 `bootdef_dev` と `boot_reset` の設定および `boot_dev` の作成と設定

`clu_create` で変数が設定でき、カーネルの構築に成功した場合は、`clu_create` によってシステムがリブートされます。この時点でシステムをリブートしない場合には、シングル・メンバ・クラスタとしてシステムをブートする準備ができたなら、4.5 節を参照してください。

`clu_create` で変数を設定できない場合は、システムを停止して、変数を 2.7 節で指定した値に設定し、4.5 節の情報を参考にしてシステムをシングル・メンバ・クラスタとしてブートします。

注意

第 1 メンバのブート・ディスクが、デュアル SCSI または Fibre Channel バスに接続された HS コントローラを介してアクセスされたり、複数バス・フェイルオーバー用に構成されていたりする場合、あるいは、システムが AlphaServer 8200 または 8400 の場合は、システムを停止し、`bootdef_dev` コンソール変数の設定方法について、2.7 節を参照してください。

`clu_create` コマンドでは、インストールのログ・ファイルが `/cluster/admin/clu_create.log` に書き込まれます。ログ・ファイルには、すべてのインストール・プロンプト、応答、およびメッセージが含まれています。システムをシングル・メンバ・クラスタとしてブートする前に、このログ・ファイルにエラーがないかどうかを確認してください。`clu_create` のログ・ファイルの例は、D.1 節に示しています。

4.5 シングル・メンバ・クラスタとしてのシステムのブート

システムを手動でブートする場合は、シングル・メンバ・クラスタとしてシステムをリブートする前に、`clu_create` のログ・ファイル `/cluster/admin/clu_create.log` の次の項目を確認します。

1. 新しいメンバのカーネルが正常に作成され、このメンバのブート・パーティションにコピーされたことを確認します。`clu_create.log` ファイルで、次のようなメッセージを探してください。

```
*** PERFORMING KERNEL BUILD ***
Working....Tue May  8 15:54:11 EDT 2001
```

```
The new kernel is /sys/PEPICELLI/vmunix
Finished running the doconfig program.
```

```
The kernel build was successful and the new kernel
has been copied to this member's boot disk.
Restoring kernel build configuration.
```

2. ブート関連コンソール変数が、Tru64 UNIX ブート・ディスクではなく、第1メンバのクラスタ・ブート・ディスクをブートする設定になっていることを確認します。ログで、次のようなメッセージを探してください。

```
Updating console variables
Setting console variable 'bootdef_dev' to dsk10
Setting console variable 'boot_dev' to dsk10
Setting console variable 'boot_reset' to ON
Saving console variables to non-volatile storage
```

3. 新しいカーネルが適切な場所にあり、コンソール変数が正しく設定されている場合は、新しく作成したそのメンバ・ブート・ディスクから、シングル・メンバ・クラスタとしてシステムをリブートします。次のように入力します。

```
# shutdown -r now
```

4. カーネルの構築に失敗したか、または、コンソール変数を設定できなかった場合は、以下のようにしてシステムを停止させます。

```
# shutdown -h now
```

次の手順を実行します。

- a. `clu_create` でコンソール変数を設定できなかった場合は、コンソール変数を 2.7 節で指定した値に設定します (この後の手順では、コンソール変数が正しく設定されていることを前提としています)。
- b. カーネルの構築に成功したら、`bootdef_dev` が Tru64 UNIX ディスクではなく第1メンバのクラスタ・ブート・ディスクに設定されていることを確認して、第1メンバのブート・ディスクから `vmunix` をブートします。

```
>>> boot
```

- c. カーネルの構築に失敗、または、第1メンバのブート・ディスクから `vmunix` をブートできなかった場合、`bootdef_dev` が Tru64

UNIX ディスクではなく第 1 メンバのクラスタ・ブート・ディスクに設定されていることを確認してから、`genvmunix` をブートします。

```
>>> boot -file genvmunix
```

システムがマルチユーザ・モードになったら、ログインしてカーネルを構築します。構築が成功した場合は、新しいカーネルを `/sys/HOSTNAME/vmunix` から `/vmunix` にコピー (cp) します。カーネルを `/vmunix` に移動する (mv) 場合は、`/vmunix` コンテキスト依存シンボリック・リンク (CDSL) を上書きします。次にシステムをブートして、カスタマイズしたカーネルで動作するようにします。

```
# doconfig -c HOSTNAME
# cp /sys/HOSTNAME/vmunix /vmunix
# shutdown -r now
```

5. 省略時のクラスタ別名に予約したネットワーク IP アドレスを割り当てている場合は、その別名アドレスの通知方法について 2.3.1 項を参照してください。
6. LSM を使用する場合は 2.5.2 項を参照してください。

このノードをシングル・メンバ・クラスタとしてブートする場合は、`/etc/ftpusers` などのアクセス関連ファイルがすべてクラスタ・メンバによって共用されるのに対し、`/etc/securettys` などのその他のファイルはメンバ固有のファイルへの CDSL で置き換えられます。これは、`ftpusers` などのファイルは、クラスタ単位の要素であるユーザ・アカウントを処理するのに対し、`securettys` などのファイルはメンバ固有の情報、この場合は `tty` デバイスを処理するためです。

クラスタ・メンバとして最初にブートする場合は、`clu_check_config` コマンドが実行され、複数の重要なクラスタ・サブシステムの構成が検査されます。これらのサブシステムが正しく構成され、正しく動作しているかどうかを確認するには、`/cluster/admin` ディレクトリの `clu_check_config` ログ・ファイルを調べます。問題がある場合は、`clu_check_config(8)` を参照し、このコマンドでテストされた内容を表示してください。初期テストで発生したサブシステムの障害の詳細は、このコマンドを冗長モードで実行して表示させることができます。サブシステムの構成方法については、Tru64 UNIX 『システム管理ガイド』と TruCluster Server 『クラスタ管理ガイド』を参照してください。

次の一連のコマンドは、クラスタの初期構成や主なサブシステムをすばやく参照する際に便利です。

```
# clu_get_info -full | more
# clu_quorum
# cfsmgr | more
# drdmgr `ls /etc/fdmns/* | grep dsk | sed 's/[a-z]$//'` |\
  uniq | sort` | more
# cluamgr -s DEFAULTALIAS
# caa_stat
```

4.6 重要な構成ファイルのオン・ディスク・バックアップ・コピーの作成

クラスタ・メンバは、次のファイルの情報を使用するため、クラスタの第 1 メンバをブートした後、これらのファイルが不注意で変更された場合に備えて、オン・ディスク・コピーを作成しておくことをお勧めします。メンバ固有のファイルについては、この例では、第 1 メンバのメンバ ID を 1 (memberid=1) と想定しています。

- /etc/sysconfigtab.cluster:
cp /etc/sysconfigtab.cluster /etc/sysconfigtab.cluster.sav
- /etc/rc.config.common:
cp /etc/rc.config.common /etc/rc.config.common.sav
- /etc/sysconfigtab — このファイルは CDSL であり、ターゲットは次のとおりです。

```
../cluster/members/{memb}/boot_partition/etc/sysconfigtab
```

バックアップ・コピーを作成するには、第 1 メンバの boot_partition/etc ディレクトリに移動して、その sysconfigtab ファイルのコピーを作成します。たとえば、次のように入力します。

```
# cd /cluster/members/member1/boot_partition/etc
# cp sysconfigtab sysconfigtab.sav
```

- /etc/rc.config — このファイルは CDSL であり、ターゲットは次のとおりです。

```
../cluster/members/{memb}/etc/rc.config
```

バックアップ・コピーを作成するには、第 1 メンバの etc ディレクトリに移動して、その rc.config ファイルのコピーを作成します。たとえば、次のように入力します。

```
# cd /cluster/members/member1/etc  
# cp rc.config rc.config.sav
```

さらに、シングル・メンバ・クラスタのフル・バックアップも実行しておくことをお勧めします。

メンバの追加

クラスタにメンバを追加するには、任意の現在のクラスタ・メンバで `clu_add_member` コマンドを実行します。このコマンドにより、クラスタ単位のルート (/), /usr, および /var の各ファイル・システムに、新しいメンバ用のディレクトリとファイルが作成されます。このコマンドでは、新しいメンバのブート・ディスクがラベル付けされて作成されます。現在のメンバは新しいメンバのブート・ディスクに書き込む必要があるため、このブート・ディスクには、現在のメンバがアクセスできる必要があり、たとえば共用バス上などに置きます。

注意

新しいメンバそれぞれにベース・オペレーティング・システムをインストールする必要はありません。ただし、ブート後に、新しいメンバそれぞれの Tru64 UNIX ライセンスを登録しなければなりません。

複数のメンバを追加する場合は、1 つのメンバを追加したら、`clu_add_member` を実行して別のメンバを追加する前に、先に追加したメンバをクラスタにブートします。

メンバ・ブート・ディスクに対して Fibre Channel ストレージセットを使用する場合は、TruCluster Server 『クラスタ・ハードウェア構成ガイド』の Fibre Channel に関する章をお読みください。

メンバをクラスタに追加するには、この章の指示に従ってください。表 5-1 に、作業を順番に示し、必要な情報の参照先を示します。

表 5-1: メンバの追加作業

作業	参照箇所
新しいメンバの情報を収集する。	第 2 章
新しいメンバとなるシステムで、コンソール変数 <code>boot_osflags</code> , <code>boot_reset</code> , および <code>bus_probe_algorithm</code> を設定する。	2.7 節
SRM ファームウェアを更新する。	5.1 節
付録 A の「メンバ属性」チェックリストに、新しいメンバのブート・ディスク用の DK コンソール・デバイス名, <code>dsk</code> スペシャル・ファイル名, および WWID (わかる場合) が記入されていることを確認する。新しいメンバのブート・ディスクの物理的な位置がわからない場合は、探しておく。 ^a	2.5.1 項, 3.7 節, 付録 A
インストール中はシステムがブートできないことを確認する。	5.2 節
現在のクラスタ・メンバで <code>clu_add_member</code> コマンドを実行する。	5.3 節
新しいメンバのコンソールから、新しいメンバをクラスタにブートする。	5.4 節
重要な構成ファイルのオン・ディスク・バックアップ・コピーを作成する。	5.5 節

^a`clu_add_member` コマンドは、ディスクのタイプ、位置、および WWID に関して自分の把握している情報をすべて表示してから終了する。

注意

`clu_add_member` を実行する前に、現在のメンバが完全に構成されているかどうか、つまり、アプリケーションのインストール、ネットワーク・インタフェースおよびネットワーク・サービスの構成、TruCluster Server ライセンスのインストールなどが行われているかどうかを確認します。

`clu_create` を実行する前にベース・オペレーティング・システムを完全に構成した場合は、`clu_add_member` を実行する準備ができています。クラスタの作成前にベース・オペレーティング・システムを完全に構成しなかった場合は、`clu_add_member` を実行する前にクラスタを完全に構成してください。これは、新しいメンバは、現在のメンバからその構成の大部分を継承するためです。

5.1 SRM ファームウェアのアップデート

Tru64 UNIX および TruCluster Server の新しいバージョンでは、通常、AlphaServer SRM ファームウェアの新しいバージョンを必要とします。ファームウェアのアップデートは、ベース・オペレーティング・システムの Software Distribution Kit に含まれている Alpha Systems Firmware CD-ROM にあります。ファームウェアをアップデートする必要があるかどうかを判断するには、クラスタ内の各システムについて、TruCluster Server『*QuickSpecs*』およびファームウェアのリリース・ノートを参照してください。必要に応じて、ファームウェアをアップデートします。TruCluster Server『*QuickSpecs*』の最新版は、次の URL で入手できます。

http://www.tru64unix.compaq.com/docs/pub_page/spds.html

5.2 インストール時における新しいメンバのブートの防止

新しいメンバになるシステムを停止するか、または電源をオフにします。システムを停止したら、次の手順に従います。

1. コンソール変数 `auto_action` を `halt` に設定します。

```
>>> set auto_action halt
```
2. コンソール変数 `bootdef_dev` を空文字列に設定します。

```
>>> set bootdef_dev ""
```

これらの作業が必要なのは、`clu_add_member` を使用して新しいメンバのブート・ディスクとして構成するディスクから、システムをブートできないようにするためです。

5.3 `clu_add_member` コマンドの実行

`/usr/sbin/clu_add_member` コマンドを実行します。このコマンドは、シングル・メンバ・クラスタを作成するために必要な情報の入力を求めるプロンプトを表示します。付録 A のチェックリストの情報を使用して、プロンプトに応答します。このコマンドには、それぞれの質問に関するオンライン・ヘルプもあります。関連するヘルプ・メッセージを表示するには、プロンプトで `help` または疑問符 `?` を入力します。

`clu_add_member` コマンドを実行するクラスタ・メンバが LAN インターコネクトを使用するように構成されていた場合、`clu_add_member` コマン

ドは LAN インターコネクトの物理クラスタ・インターコネクト・デバイス名を入力するように求めます。この場合、以下のいずれかの方法を探ることができます。

- `tu0`, `ee0`, または `alt0` のように、1 つのネットワーク・インタフェースのデバイス名を指定する。
- 複数のネットワーク・インタフェースのデバイス名を繰り返し指定する。`clu_add_member` コマンドでは、これらのインタフェースを NetRAIN 仮想インタフェースに構成するという指定ができる。

注意

指定する NetRAIN デバイスのデバイス名が、その時点で `clu_add_member` コマンドを実行しているメンバの物理クラスタ・インターコネクト・デバイスとして定義されていると、追加しようとしているメンバ上で同じ NetRAIN デバイス (デバイス名と使用するアダプタが同一) を使用するかどうかの確認を求められます。"yes" と答えると、`clu_add_member` コマンドは、このデバイスを `/etc/sysconfigtab` ファイルの `ics_ll_tcp` スタンザにクラスタ・インターコネクト・デバイスとして定義します。

その後、`clu_add_member` コマンドは `membermember-ID-icstcp0` という形式で物理クラスタ・インターコネクト・デバイスの IP 名を作成します。また省略時には、このデバイスに IP アドレスとして `10.1.0.member-ID` を割り当てます。

`clu_add_member` コマンドがクラスタ・インターコネクトを定義するために書き込む `/etc/sysconfigtab` 属性リストについては、表 C-2 を参照してください。

`clu_add_member` コマンドにより、新しいメンバのブート・ディスクの構成、クラスタ単位のファイル・システムでのファイルの追加と変更、および TruCluster Server ライセンス PAK を登録するオプションの提供が行われます。

注意

TruCluster Server ライセンスのないシステムもブートできます。この場合、システムは、クラスタに参加してマルチユーザ・モードでブートしますが、ログインできるのは `root` のみで、最高で 2 ユーザまでです。CAA (cluster application availability) デーモン `caad` は起動されません。システムによりライセンス・エラー・メッセージが表示されて、ライセンスの登録が促されます。この設定ではライセンス・チェックが強制実行されますが、緊急時にはシステムのブートおよび修理ができます。

この時点では、TruCluster Server ライセンスだけを登録します。Tru64 UNIX ライセンス PAK は登録しないでください。Tru64 UNIX ライセンス PAK およびその他の必要なライセンス PAK は、新しいメンバを初めてブートした後に登録します。

`clu_add_member` コマンドを実行すると、`/cluster/admin/clu_add_member.log` にインストレーションのログ・ファイルが書き込まれます。続行する前に、このログ・ファイルにエラーがないかどうかを確認してください。`clu_add_member` のログ・ファイルの例は、D.2 節に示しています。

注意

3.7.2 項で説明されているように、`clu_add_member` コマンドから、新しいメンバのブート・ディスクが置かれている物理的な位置のヒントを得ることができます。この情報は画面に表示され、`clu_add_member.log` ファイルに書き込まれます。

5.4 新しいメンバのブート

`clu_add_member` を実行したら、新たにインストールしたメンバのコンソールで次の作業を行います。

1. 新しいメンバのコンソールで、コンソール変数 `boot_osflags` を A に設定して、システムがマルチユーザ・モードでブートされるようにします。

```
>>> set boot_osflags A
```

2. 新しいメンバのコンソールで、新しいメンバのブート・ディスクから `genvmunix` をブートします。

```
>>> boot -file genvmunix new_member_boot_disk
```

コンソールでは、ブート・ディスクに対して正しい DK デバイス名を指定するよう注意してください。 `clu_add_member` で指定した `dsk` スペシャル・ファイル名は使用しないでください。たとえば、新しいメンバのブート・ディスクのコンソール・デバイス名が `DKC600` である場合は、次のようになります。

```
>>> boot -file genvmunix DKC600
```

警告

新たにインストールしたブート・ディスクの位置がわからない場合、物理的な位置を特定するには 3.7.2 項を参照してください。

必ず正しいディスクをブートするよう注意するとともに、ブート・ディスクを他のディスクにコピーすることや、そのコピーしたディスクからメンバをブートすることはできないことにも注意してください。メンバは、`clu_add_member` で作成したディスクからブートしなければなりません。メンバのブート・ディスクを変更するには、6.2 節の指示に従ってメンバを再インストールする必要があります。

3. ブート中に、新しいメンバで次のタスクが自動的に実行されます。
 - a. ロードされたサブセットがすべて構成される。
 - b. カスタマイズされたカーネルを構築する。
 - カーネルの構築が正常に終了したら、メンバのブート・パーティションに新しいカーネルをコピーする。
 - 構築に失敗した場合、システムがマルチユーザ・モードになっているときは、`doconfig` を実行してカーネルを構築できる。
`/sys/HOSTNAME/vmunix` から `/vmunix` へ新しいカーネルをコピー (cp) します。カーネルを `/vmumix` へ移動 (mv) すると、`/vmunix CDSL` が上書きされます。

- c. スクリプトを実行して、ユーザが追加のネットワーク・インタフェースを構成できるようにする。clu_add_member コマンドでは、クラスタ・インターコネクトのインタフェースのみが構成されます。追加のインタフェース、つまり、新しいメンバのホスト名と関連付けられたパブリック・ネットワーク・インタフェースを少なくとも 1 つ構成することをお勧めします。

注意

新しいメンバのホスト名と関連付けられたインタフェースがない場合で、共通デスクトップ環境 (CDE) を使用する場合は、初めてログインすると CDE によってエラー・ダイアログ・ボックスが表示されます。ダイアログ・ボックスの表示を見てから OK をクリックして、ログアウトします。フェールセーフ・セッションを選択し、再びログインしてください。ネットワーク・インタフェースを構成し、それをシステムのホスト名に関連付けます。

- d. 変数 boot_reset と bootdef_dev を設定し、コンソール変数 boot_dev を作成および設定する。
- 4. システムは、これ以降もマルチユーザ・モードでブートされます。マルチユーザ・モードでのシステムのブートが終了した場合には、Tru64 UNIX ライセンスおよびその他の必要なアプリケーション・ライセンスを登録します。clu_add_member を実行したときに TruCluster Server ライセンスを登録しなかった場合は、ここで登録します。
 - 5. このメンバではまだ genvmunix を実行しているため、システムをリブートしてカスタム・カーネルを使用します。メンバをクラスタに追加する手順では、このリブートが必要です。

```
# shutdown -r now
```

注意

新しいメンバのブート・ディスクが、デュアル SCSI または Fibre Channel バスに接続された HS コントローラを介してアクセスされたり、複数バス・フェイルオーバー用に構成されていたり、あるいは、システムが AlphaServer 8200 または

8400 の場合は、システムを停止し、`bootdef_dev` コンソール変数の設定方法について、2.7 節を参照してください。

クラスタ・メンバとして最初にブートする場合は、`clu_check_config` コマンドが実行され、複数の重要なクラスタ・サブシステムの構成が検査されます。これらのサブシステムが正しく構成され、正しく動作しているかどうかを確認するには、`/cluster/admin` ディレクトリの `clu_check_config` ログ・ファイルを調べます。問題がある場合は、`clu_check_config(8)` を参照し、このコマンドでテストされた内容を確認してください。テストで発生したサブシステムの障害の詳細は、このコマンドを詳細 (verbose) モードで実行して表示できます。サブシステムの構成方法については、Tru64 UNIX 『システム管理ガイド』と TruCluster Server 『クラスタ管理ガイド』を参照してください。

5.5 重要な構成ファイルのオン・ディスク・バックアップ・コピーの作成

クラスタ・メンバは、次のファイルの情報を使用するため、クラスタの追加のメンバをそれぞれブートした後、これらのファイルが不注意で変更された場合に備えて、オン・ディスク・コピーを作成しておくことをお勧めします。メンバ固有のファイルについては、この例では、メンバ 2 (memberid=2) を使用しています。バックアップ・コピーを作成する際には、新しいメンバの正しいメンバ ID と置き換えてください。

- `/etc/sysconfigtab.cluster:`

```
# cp /etc/sysconfigtab.cluster /etc/sysconfigtab.cluster.sav
```

メンバが追加されるたびに、クォーラム・ポート情報が更新されるため、各メンバの追加後には、このファイルのバックアップ・コピーを作成します。

• `/etc/sysconfigtab` — このファイルは CDSL であり、ターゲットは次のとおりです。

```
../cluster/members/{memb}/boot_partition/etc/sysconfigtab
```

バックアップ・コピーを作成するには、新しいメンバの `boot_partition/etc` ディレクトリに移動して、その `sysconfigtab` ファイルのコピーを作成します。たとえば、次のように入力します。

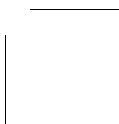
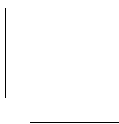

```
# cd /cluster/members/member2/boot_partition/etc
# cp sysconfigtab sysconfigtab.sav
```

- /etc/rc.config — このファイルは CDSL であり，ターゲットは次のとおりです。

```
../cluster/members/{memb}/etc/rc.config
```

バックアップ・コピーを作成するには，新しいメンバの `etc` ディレクトリに移動して，その `rc.config` ファイルのコピーを作成します。たとえば，次のように入力します。

```
# cd /cluster/members/member2/etc
# cp rc.config rc.config.sav
```



クラスタ・メンバの再インストール

この章では、次の作業の手順を説明します。

- シングル・メンバ・クラスタの再作成 (6.1 節)
- 個々のクラスタ・メンバの再インストール (6.2 節)
- インストール構成ファイルを用いた、クラスタの再作成および単一メンバの再インストールの自動化 (6.3 節)

6.1 シングル・メンバ・クラスタの再作成

この手順では、現在のクラスタからメンバを削除した後、ベース・オペレーティング・システムをブートし、`clu_create` コマンドを再実行して新しいクラスタを作成します。新しく作成するクラスタの構成は現在のクラスタと同じでも同じでなくてもかまいません。6.3 節に示す手順では、構成ファイルを用いて現在のクラスタと同じ構成のクラスタを作成しています。

シングル・メンバ・クラスタを再作成する手順は、次のとおりです。

1. どのディスクが Tru64 UNIX のブート・ディスクであることを特定します。

- a. 次の例で示すように、`/etc/fdmns` ディレクトリで、Tru64 UNIX ブート・ディスクのスペシャル・ファイル名を探します (通常は `root_domain` ディレクトリにあります)。

```
# ls /etc/fdmns/root_domain  
/dev/disk/dsk0a
```

- b. ディスクがプライベート・バス上にある場合は、SysMan Station を使用して、どのシステムにそのディスクがあるかを特定します。
- c. 次の例で示すように、ディスクのライトを点滅させます。

```
# hwmgr -flash light -dsf /dev/disk/dsk0
```

ディスクの `dsk` デバイス・スペシャル・ファイル名をその物理的位置にマップする方法については、3.7.2 項で説明しています。

2. Tru64 UNIX ディスクに直接アクセスできるクラスタ・メンバで、その他のクラスタ・メンバをすべて削除します。クラスタ・メンバの削除については、『クラスタ管理ガイド』を参照してください。これらのメンバを削除する前に構成ファイルを保存する場合は、6.3 節を参照してください。この時点で、このシステムがクラスタの唯一のメンバとなります。

警告

クラスタを再作成する前に、すべてのクラスタ・メンバを削除してください。削除しない場合、古いメンバ・ブート・ディスクが新しいクラスタにブートされる可能性があります。

3. このメンバのクラスタ・ブート・ディスクのデバイス・スペシャル・ファイル名を決定し、忘れないようにメモをとっておきます。たとえば、このシステムがメンバ 1 の場合には次のように入力します。

```
# ls /etc/fdmns/root1_domain  
/dev/disk/dsk10a
```

4. システムを停止します。

```
# shutdown -h now
```

5. システム・コンソールから、Tru64 UNIX オペレーティング・システムをマルチユーザ・モードでブートします。

```
>>> boot UNIX_disk
```

6. このシステムはクラスタから削除されなかったため、そのクラスタ・メンバ・ブート・ディスクは、それにアクセスできるシステムからまだブートできます。システムから偶然にこのディスクがブートされることを防ぐため、ディスク・ラベルをゼロ (クリア) にします。たとえば、dsk10 のラベルをゼロにするには、次のように入力します。

```
# disklabel -z dsk10
```

7. clu_create コマンドを実行して、シングル・メンバ・クラスタを作成します。

```
# /usr/sbin/clu_create
```

clu_create コマンドの実行については、第 4 章を参照してください。

6.2 個々のクラスタ・メンバの再インストール

メンバを再インストールするには、そのメンバを停止し、別のメンバで次の作業を行います。

1. クラスタからメンバを削除するには、`clu_delete_member` コマンドを実行します。クラスタからのメンバの削除については、『クラスタ管理ガイド』を参照してください。
2. クラスタにメンバを再び加えるには、`clu_add_member` コマンドを実行します。`clu_add_member` コマンドの実行についての詳細は、第5章を参照してください。構成ファイルの使用についての詳細は、`clu_add_member(8)` と 6.3 節を参照してください。

6.3 インストール構成ファイルの使用

`clu_create` および `clu_add_member` は、TruCluster Server クラスタを新規に作成するとき、その構成ファイルを `/cluster/admin` ディレクトリに書き込みます。このファイルの名前は `.membern.cfg` で、*n* はクラスタ・メンバの ID を表します。これらのコマンドが正しく実行されるたびに、それぞれのメンバの構成ファイルに現在の構成情報が追加されます。構成ファイルとその使用の制約事項については、`clu_create(8)` および `clu_add_member(8)` を参照してください。

以下の例は、各国語サポート (WLS) の機能を持つとともにクラスタ・インターコネクトとして LAN ハードウェアを使用するクラスタを作成した際に、`clu_create` が作成した構成ファイルです。読みやすいようにコメント行は折り返してあります。

```
# clu_create saved configuration values:
# date: Tue May 15 15:47:14 EDT 2001 hostname \
# pepicelli.zk3.dec.com
# Previously saved value in this file have been \
# converted to comment lines
clu_alias_ip=16.140.112.209
clu_boot_dev=dsk10
clu_il8n_dev=dsk14
clu_ics_dev=ics0
clu_ics_host=pepicelli-ics0
clu_ics_ip=10.0.0.1
clu_mem_votes=1
clu_memid=1
clu_name=deli.zk3.dec.com
clu_nr_dev=nr0
```

```
clu_phys_devs=ee0,ee1
clu_quorum_dev=dsk7
clu_quorum_votes=1
clu_root_dev=dsk1b
clu_tcp_host=member1-icstcp0
clu_tcp_ip=10.1.0.1
clu_usr_dev=dsk2c
clu_var_dev=dsk3c
```

インストール構成ファイルには `variable=value` という組み合わせで記録されています。この情報を使用することで、`clu_create` および `clu_add_member` の実行するタスクを自動化することができます。`-c` オプションと構成ファイル名を指定してこれらのコマンドを実行すると、これらのコマンドはその構成ファイルを入力として使用します (情報を手動で入力する必要がなくなります)。

`clu_create` または `clu_add_member` は、構成ファイルを読み取る際に必要な名前と値のペアを発見できないと、必要な情報の入力を促すためのプロンプトを表示して、構成ファイルの読み取りに戻ります。したがって、たとえば、クラスタの最初のメンバを削除してしまった後でも、`clu_create` が作成した構成ファイル (通常は `/cluster/admin/.member1.cfg`) を用いて `clu_add_member -c` を実行することができます。`clu_add_member` コマンドは、`clu_create` の作成した構成ファイルに必要な情報が入っていないければ、プロンプトを表示します。

注意

構成ファイルの作成と読み取りはどちらもプログラムが行います。特別なことがなければ、構成ファイルは手動で編集しないでください。

構成ファイルを使用すれば、既存のクラスタを簡単に作成し直すことができます。ただし、構成ファイル内の情報 (ホスト名、IP アドレス、ディスク・スペシャル・ファイル名など) は正確でなければなりません。ディスク・デバイスは見つかった順序で命名されるため、構成ファイルを用いてクラスタを再作成する場合は、以前と同じシステムで `clu_create -cmember_conf_file` を実行することになります。また、ストレージ構成ファイルも変更されていないことが前提になります。さらに、メンバの追加は、最初にクラスタを作成した場合と同じ順序で行わなければなりません。各メンバについて、以前に実行した場合と同じメンバ上で `clu_add_member`

-c *member_conf_file* を実行します (構成ファイルのコメント行 `date:` には、`clu_create` または `clu_add_member` が実行された日付と実行されたホスト名が記録されています)。

既存のクラスタが以下の要件を満たしていれば、現在の `clu_create` および `clu_add_member` 構成ファイルを保存することで、クラスタの再作成を自動化できます。

- クラスタをアップグレードする場合に現在と同じディスク、ホスト名、および IP アドレスを使用する。
- ディスク・ストレージの構成が、そのクラスタを新規に作成してメンバを追加した時点と同じである。つまり、`clu_create` および `clu_add_member` 構成ファイルに記録されているディスク名が、最後にコマンドを実行したときと同じデバイスを参照している。クラスタを作成してからデバイスを追加または削除していると、最初に Tru64 UNIX をインストールしたときのハードウェア構成と現在の構成が一致していないことがあります。クラスタ単位の `root (/)`、`/usr`、`/var` ファイル・システム、メンバ・ブート・ディスク、またはクォラム・ディスクのいずれかに使用しているディスクの名前がこの違いに関係していると、問題になることがあります。

注意

この節の残りの部分では、クラスタ全体を再作成する方法について説明します。特に説明しませんが、構成ファイルを使用してクラスタをメンバに再度追加することもできます。

以下に、現在のクラスタから構成ファイルを保存した後、その構成ファイルを使用して TruCluster Server を再作成する手順を示します。

1. 現在のクラスタのフル・バックアップを取ります。
2. どのディスクが Tru64 UNIX のブート・ディスクになっているかを調べます。最初にクラスタ・メンバとなったシステムのプライベート・ディスクがブート・ディスクになっていることがよくあります。
 - a. 次の例で示すように、`/etc/fdmns` ディレクトリを調べて、Tru64 UNIX のブート・ディスクに対応するスペシャル・ファイル名を探します (通常は `root_domain` ディレクトリの中)。

```
# ls /etc/fdmns/root_domain  
/dev/disk/dsk0a
```

- b. SysMan Station を使用して、どのシステムにそのディスクがあるか調べます。
- c. 次の例で示すように、そのディスクのライトを点滅させます。

```
# hwmgr -flash light -dsf /dev/disk/dsk4
```

ディスクの dsk デバイス・スペシャル・ファイル名からその物理的位置を割り出すときは、3.7.2 項を参照してください。ブートはシステムのコンソールから行うので、このディスクの物理的位置を知る必要があります。

3. Tru64 UNIX オペレーティング・システム・ディスクにアクセスできるクラスタ・メンバ上でディスクをマウントし、現在の構成ファイルとライセンス PAK をそのディスクに保存します。次に例を示します。

```
# mount root_domain#root /mnt  
# mkdir /mnt/config_files /mnt/licenses  
# cp /cluster/admin/.member*.cfg /mnt/config_files  
# for i in `lmf list | grep -v Product | awk '{print $1}'`  
do  
  lmf issue /mnt/licenses/$i.license $i  
done
```

注意

その他に、次のような情報も保存しておくとい良いでしょう。

- サイト固有の CAA プロファイルおよび処理スクリプト
- 各メンバの /etc/rc.config ファイル
- 各メンバのクラスタ別名構成ファイル (/etc/clu_alias.config)
- システム構成ファイルに対する修正
- /etc/fstab
- /etc/fdmns/* の再帰リスト (ls -R)

一言でいえば、クラスタを作成した以降に行った変更のうち、その作業を繰り返したくないものは保存しておく、ということです。何を保存してよいかわからない場合は、`sys_check -all` コマンドを使用して、システムの構成情報

を収集してください(その場合も CAA のプロファイルとスクリプト, クラスタ別名構成ファイル, およびメンバ固有のファイルは手動で保存しなければなりません)。

4. クラスタを停止させます。

```
# shutdown -c now
```

5. システム・コンソールから Tru64 UNIX オペレーティング・システムをマルチユーザ・モードでブートします。

```
>>> boot UNIX_disk
```

注意

この手順では, ベース・オペレーティング・システムをインストールして新規にクラスタを作成した後, そのクラスタのローリング・アップグレードをまだ実行していないことを前提としています。ローリング・アップグレードをすでに実行していると, このディスクに置かれているベース・オペレーティング・システムは, クラスタをシャットダウンした時点とバージョンが異なります (sizer -v コマンドを使用するとオペレーティング・システムのバージョンが表示されます)。

オペレーティング・システムが最新バージョンでない場合は, 次の手順を実行します。

- a. システムをシングルユーザ・モードにします。
- b. TruCluster Server サブセットを削除します。
- c. アップデート・インストールを実行して Tru64 UNIX のバージョンを最新版にします。
- d. TruCluster Server サブセットの最新バージョンをロードします。
- e. (オプション) 新バージョンのベース・オペレーティング・システムおよびクラスタ・ソフトウェアに対してパッチ・キットがある場合は, clu_create をまだ実行していないこの時点で Tru64 UNIX システムにパッチをあて

ます。ここでパッチをあてておけば、後からパッチ・キットをクラスタにローリングする必要がありません。

6. 保存しておいた必要なライセンスを登録します (Tru64 UNIX と TruCluster Server のライセンスはすでにアクティブです)。次の例では、`lmf list` を実行することで不要な `*.license` ファイルがすでに `/licenses` から削除されていることを前提としています。

```
# for i in /licenses/*.license
do
    lmf register - < $i
done
# lmf reset
```

7. 保存してある構成ファイルの中からどの構成ファイルを `clu_create` で使用するかを決めます。次に、その構成ファイルの名前を指定して `clu_create -c` を実行します。次に例を示します。

```
# cd /config_files
# grep clu_create *.cfg
.member1.cfg:# clu_create saved configuration values:
# /usr/sbin/clu_create -c /config_files/.member1.cfg
```

8. 最初のクラスタ・メンバをブートした後、保存しておいたメンバ構成ファイルを使用して、残りのメンバをクラスタに追加します。

警告

メンバの追加は、元のクラスタに追加したときと同じ順序で、しかも同じホストから行ってください。そうしないと、デバイス名が元のクラスタと同じ名前にならないことがあります。

構成ファイルをそれぞれ調べて、`clu_add_member` コマンドをどのメンバで実行したかを調べます。各ファイル内の最新の `# date` コメントを調べると、コマンドが実行された時刻とホストがわかります。次の短いスクリプトを実行すると、構成ファイル名、`clu_add_member` が実行されたホスト、および追加されたメンバ名が表示されます。

```
#!/bin/ksh
cd /config_files
for i in `grep -l unix_host .member*.cfg`
do
    print '\n' $i
```

```
tail -21 $i | grep -E '^# date|^unix_host'
done
```

3 メンバから構成されるサンプル・クラスタでこのスクリプトを実行すると、出力は次のようになります。

```
.member2.cfg
# date: Tue May 15 17:46:48 EDT 2001 hostname pepicelli.zk3.dec.com
unix_host=polishham.zk3.dec.com
```

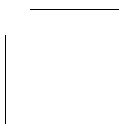
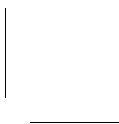
```
.member3.cfg
# date: Tue May 15 18:09:32 EDT 2001 hostname polishham.zk3.dec.com
unix_host=provolone.zk3.dec.com
```

この情報を基に、クラスタの最初のメンバ `pepicelli` で `.member2.cfg` ファイルを指定して `clu_add_member` を実行し、2 番目のメンバ `polishham` を追加します。`polishham` をブートした後、そこで `.member3.cfg` ファイルを指定して `clu_add_member` を実行すると、3 番目のメンバ `provolone` が追加されます。

たとえば、クラスタの 2 番目のメンバを追加する場合は、`pepicelli` で次のコマンドを実行します。

```
# /usr/sbin/clu_add_member -c /config_files/.member2.cfg
```

必ず、新しく追加したメンバをブートしてから、次のメンバを追加してください。



ローリング・アップグレード

ローリング・アップグレードは、クラスタの稼働中にソフトウェアをアップグレードする方法です。クラスタ・メンバを一度に1つずつアップグレードして運用に戻し、複数のバージョンのベース・オペレーティング・システム、クラスタ、および各国語サポート (WLS) ソフトウェアが混在する環境を透過的に管理します。サービスにアクセスするクライアントからは、ローリング・アップグレードが進行中であることがわかりません。

ローリング・アップグレードでは、一連の手順を順序に沿って実行します。この手順を段階といいます。ローリング・アップグレードを制御するコマンドはこの順序で実行する必要があります。

この章の前半では、ローリング・アップグレードの実行、ローリング・アップグレードのステータス表示、およびローリング・アップグレードにおける段階の取り消し (複数段階可) について説明します。ローリング・アップグレードがどのように働くかについては、7.6 節およびそれに続く節で説明します。

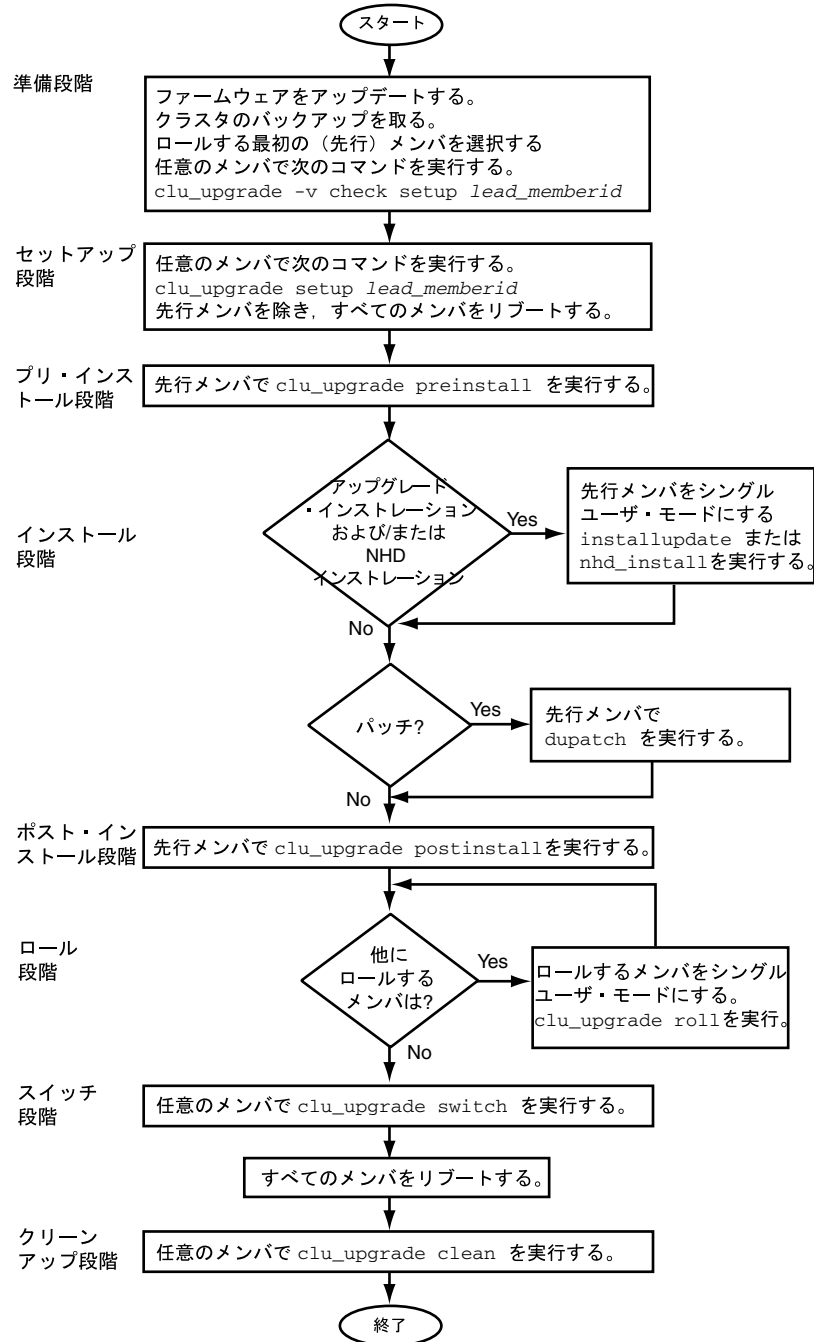
この章で説明する内容は次のとおりです。

- 1 つのローリング・アップグレードで実行できる作業または作業の組み合わせ (7.1 節)
- ローリング・アップグレードで実行できない作業 (7.2 節)
- ローリング・アップグレード方法 (7.3 節)
- ローリング・アップグレードのステータス表示方法 (7.4 節)
- ローリング・アップグレード段階の取り消し方法 (7.5 節)
- ローリング・アップグレードで使用するコマンド (7.6 節)
- ローリング・アップグレードの段階 (7.7 節)
- ローリング・アップグレードをサポートする 2 つの機構 — タグ付きファイル (7.8 節) とバージョン・スイッチ (7.9 節)
- ローリング・アップグレードとレイヤード・プロダクト (7.10 節)

- ローリング・アップグレードと RIS (7.11 節)

図 7-1 は、バージョン 5.1B クラスタで開始するローリング・アップグレードの実行に必要な作業と段階を簡単なフロー・チャートにまとめたものです。

図 7-1: ローリング・アップグレード作業の流れ



ZK-1667U-AI

7.1 ローリング・アップグレードでサポートされている作業

ローリング・アップグレードで行うことができる作業は、クラスタで現在実行しているベース・オペレーティング・システムとクラスタ・ソフトウェアのバージョンによって異なります。この章では主に、TruCluster Server バージョン 5.1B クラスタで開始するローリング・アップグレードについて説明します。ただし、TruCluster Server バージョン 5.1A からバージョン 5.1B へローリング・アップグレードする準備をされている方のために、2 つのバージョンの間にあるローリング・アップグレードの違いについても示していきます。

ローリング・アップグレードで行うことができる基本的な作業は、次のとおりです。

- クラスタの Tru64 UNIX ベース・オペレーティング・システムと TruCluster Server ソフトウェアのアップグレード。このタイプのローリング・アップグレードでは、インストールされているバージョンを次のバージョンへアップグレードします。

ベース・オペレーティング・システムとクラスタ・ソフトウェアをローリング・アップグレードする際は、あるバージョンからすぐ次のバージョンへローリングできるだけです。バージョンをスキップすることはできません。実施可能なアップグレード・パスについては表 1-1 を参照してください。

注意

ローリング・アップグレードでは、クラスタで現在使用しているファイル・システムとディスクがアップデートされます。ただし、クラスタの作成に使用した Tru64 UNIX オペレーティング・システムを含むディスクはアップデートされません (clu_create を実行したオペレーティング・システム)。クラスタが停止した場合のように、緊急時に元のオペレーティング・システムをブートすることはできますが、クラスタをアップデートするたびに最新のクラスタと元のオペレーティング・システムの相違点が増加していくことに注意してください。

- クラスタで現在使用している Tru64 UNIX ベース・オペレーティング・システムと TruCluster Server ソフトウェアのパッチ。
- NHD (New Hardware Delivery) キットのインストール (クラスタで TruCluster Server バージョン 5.1A 以降が動作していることが必要)。

パッチ・キットまたは NHD キットによるローリングでは、ベース・オペレーティング・システムおよびクラスタ・ソフトウェアの新リリースによるローリングと同じ手順を使用します。違いは、インストール段階で実行するコマンドです。

- ベース・オペレーティング・システムおよびクラスタ・ソフトウェアをアップグレードする場合は、インストール段階で `installupdate` コマンドを使用します
- パッチ・キットをロールする場合は、インストール段階で `dupatch` を実行します。インストール段階で、複数のパッチ・キットをロールするために `dupatch` を複数回実行することができます。

クラスタのノーロール・パッチを実行する場合は、`clu_upgrade` コマンドは実行せずに、`dupatch` コマンドをマルチユーザ・モードで起動しているクラスタ・メンバから実行します。

ノーロール・パッチはパッチを迅速に適用し、リブートの数を減らすことができます。ノーロール・パッチは1回の操作でクラスタにパッチをあてることができます。ただし、操作を完了するにはクラスタ全体をリブートする必要があるため、クラスタはその間利用することができません。詳細については、インストールしようとしているパッチ・キットに添付の『*Patch Kit Installation Instructions*』を参照してください。

- NHD キットをインストールするには、インストール段階で `nhd_install` を実行します。

この章では、ローリング・アップグレードという用語を、ソフトウェア・キットのロール手順全体を表す用語として使用しています。

図 7-1 に示すように、1 回のローリング・アップグレードで複数の作業を行うことができます。

クラスタでバージョン 5.1A またはバージョン 5.1B が動作している場合は、ローリング・アップグレードとして、表 7-1 に示す作業を行うことができます。

表 7-1: バージョン 5.1A および バージョン 5.1B で行えるローリング・アップグレード作業

バージョン 5.1A から バージョン 5.1B へのアップデート・インストール バージョン 5.1B から次のリリースへのアップデート・インストール
バージョン 5.1A のパッチ バージョン 5.1B のパッチ
NHD (New Hardware Delivery) キットの バージョン 5.1A クラスタへのインストール NHD キットの バージョン 5.1B クラスタへのインストール
ベース・オペレーティング・システムおよびクラスタ・ソフトウェアのバージョン 5.1A から バージョン 5.1B へのアップデート・インストールと、その後に行う バージョン 5.1B のパッチ ベース・オペレーティング・システムおよびクラスタ・ソフトウェアのバージョン 5.1B から次のリリースへのアップデート・インストールと、その後に行う次のリリースのパッチ ^a
バージョン 5.1A のクラスタへの NHD インストールと、その後に行うバージョン 5.1A のパッチ バージョン 5.1B のクラスタへの NHD インストールと、その後に行う バージョン 5.1B のパッチ
バージョン 5.1A からバージョン 5.1B へのアップデート・インストールと、その後に行う バージョン 5.1B への NHD キットのインストール ベース・オペレーティング・システムとクラスタ・ソフトウェアのバージョン 5.1B から次のリリースへのアップデート・インストールと、その後に行う次のリリースへの NHD キットのインストール ^b
バージョン 5.1A から バージョン 5.1B へのアップデート・インストールと、その後に行う バージョン 5.1B への NHD キットのインストール、およびその後に行うバージョン 5.1B のパッチ バージョン 5.1B から次のリリースへのアップデート・インストールと、その後に行う次のリリースへの NHD キットのインストール、およびその後に行う次のリリースへのパッチ ^b

^a1 回のローリング・アップグレードで、ベース・オペレーティング・システムとクラスタ・ソフトウェアのアップグレードを新しいソフトウェアのパッチと組み合わせて行うことができます。つまり、インストール段階において、最初のメンバで `installupdate` を実行した後に、`dupatch` を使用して、新しくインストールしたソフトウェアにパッチをあてることができる。他のメンバをロールすると、それらのメンバに新しいソフトウェアとパッチが自動的に適用される。

ただし、現在のソフトウェアにパッチをあててからベース・オペレーティング・システムとクラスタ・ソフトウェアをアップグレードするという作業は、1 回のローリング・アップグレードで実行できない。この場合は、ローリング・アップグレードを 2 回行う必要がある。

表 7-1: バージョン 5.1A および バージョン 5.1B で行えるローリング・アップグレード作業 (続き)

^bバージョン 5.1A または バージョン 5.1B のクラスタに NHD キットがすでにインストールされている場合にのみ可能。

7.2 ローリング・アップグレードでサポートされていない作業

ローリング・アップグレード中に実行できない作業、または実行しないことを推奨する作業のリストを以下に示します。

- `/var/adm/update` ディレクトリにあるファイルは削除も変更もしないでください。このディレクトリにあるファイルは、ロール処理にとって非常に重要です。削除するとローリング・アップグレードが失敗するおそれがあります。
- インストール段階では、`installupdate` コマンドより先に `dupatch` コマンドを実行することはできません。ローリング・アップグレードを実行する前に現在のソフトウェアにパッチをあてるには、ローリング・アップグレード操作を 2 回 (現在のソフトウェアに対するパッチを 1 回とアップデート・インストレーションを 1 回) 実行する必要があります。
- ベース・オペレーティング・システムとクラスタ・ソフトウェアをローリング・アップグレードする際は、バージョンをスキップできません。あるバージョンからすぐ次のバージョンへのローリングができるだけです。実施可能なアップグレード・パスについては表 1-1 を参照してください。
- 以下のサブセットの追加または削除の際に、`/usr/sbin/setld` コマンドを使用しないでください。
 - ベース・オペレーティング・システム・サブセット (接頭文字 `OSF` のついたもの)。
 - TruCluster Server サブセット (接頭文字 `TCR` のついたもの)。
 - 各国語サポート (WLS) サブセット (接頭文字 `IOSWW` のついたもの)。

ロール中にこれらのサブセットを追加または削除すると、タグ付きファイルが一致なくなります。

- ロール中にレイヤード・プロダクトをインストールしないでください。最初にロールするメンバに製品の新バージョンをインストールできること、および複数のバージョンが混在するクラスタでレイヤード・プロダ

クトが動作できることが、レイヤード・プロダクトのドキュメントに特記されていない場合は、新しいレイヤード・プロダクトまたは現在インストールされているレイヤード・プロダクトの新バージョンをローリング・アップグレード中にインストールしないでください。

レイヤード・プロダクトおよびローリング・アップグレードについての詳細は 7.10 節を参照してください。

7.3 ローリング・アップグレードの手順

この節の手順では、特に明記しない限りコマンドはマルチユーザ・モードで実行します。各段階の手順は、それぞれの対応する項で詳しく説明します。ローリング・アップグレードを実行する場合は、その前に 7.7 節で各段階の詳しい説明を参照されることをお勧めします。

ローリング・アップグレードの段階の中には、他の段階よりも時間のかかるものがあります。表 7-2 に、各段階を完了するまでのおよその所要時間をまとめています。

表 7-2: ローリング・アップグレードの各段階における所要時間

段階	所要時間
準備	プログラム制御ではない。
セットアップ	45 ~ 120 分。 ^a
プリ・インストール	15 ~ 30 分。 ^a
インストール	単一システムで <code>installupdate</code> 、 <code>dupatch</code> 、 <code>nhd_install</code> 、またはこれらのコマンドの有効な組み合わせを実行する時間。
ポスト・インストール	1 分未満。
ロール (メンバごと)	パッチ：5 分未満。 アップデート・インストレーション：メンバの追加にかかる時間とほぼ同じ。 ^b
スイッチ	1 分未満。
クリーンアップ	30 ~ 90 分。 ^a

^aこれらの段階では、ローリング・アップグレードに必要なタグ付きファイルの作成、検査、または削除を実行する。これらの段階の所要時間は、コマンドを実行するメンバの処理速度、ストレージの処理速度、コマンドを実行するメンバがルート (/)、`/usr`、および `/var` ファイル・システムの CFS サーバであるかどうかによって異なる。`clu_upgrade` コマンドを実行するメンバにこれらのファイル・システムを移動する方法もある。

表 7-2: ローリング・アップグレードの各段階における所要時間 (続き)

^b 先行メンバのロール後、並列ロールによって同時に複数のロールを実行し、クラスタのロールにかかる時間を短縮する。

以下の手順で、TruCluster Server バージョン 5.1A クラスタをバージョン 5.1B にアップグレードします。すでにバージョン 5.1B になっているクラスタをアップグレードする場合も同じです。

1. ローリング・アップグレードを実行するために、次のようにクラスタを準備します (7.7.1 項)。

- a. 先行メンバ (最初にロールするメンバ) として使用するクラスタ・メンバを選択します。この手順の例では、`memberid` が 2 のメンバを先行メンバとして使用します。メンバのホスト名は、`provolone` です。
- b. クラスタのバックアップをとります。
- c. インストール段階でアップデート・インストレーションを実行する場合は、クラスタにインストールされている表 7-6 のブロッキング・レイヤード・プロダクトを削除します。
- d. 任意のクラスタ・メンバで次のように `clu_upgrade -v check setup lead_memberid` コマンドを実行し、クラスタをアップグレードする準備ができているかどうかを確認します。

```
# clu_upgrade -v check setup 2
```

ファイル・システムでより多くの空き容量が必要な場合は、`addvol` などの AdvFS Utilities を使用して、必要なボリュームをドメインに追加します。必要なディスク・スペースについては、7.7.1 項を参照してください。AdvFS ドメインの管理方法については、Tru64 UNIX 『AdvFS 管理ガイド』を参照してください。

- e. 各システムのファームウェアが新しいソフトウェアをサポートしていることを確認します。ローリング・アップグレードの前に、必要に応じてファームウェアをアップデートします。

2. セットアップ段階を実行します (7.7.2 項)。

注意

現在のクラスタがバージョン 5.1A 以降である場合に、このインストール段階でベース・オペレーティング・システムとクラスタ・ソフトウェアをアップグレードする予定があれば、新しい TruCluster Server キットの入っているデバイスまたはディレクトリをマウントし、その後で `clu_upgrade setup` を実行します。setup コマンドを実行すると、そのキットが `/var/adm/update/TruClusterKit` ディレクトリにコピーされます。

現在のクラスタがバージョン 5.1A 以降である場合に、インストール段階で NHD キットをインストールする予定があれば、新しい NHD キットの入っているデバイスまたはディレクトリをマウントし、その後で `clu_upgrade setup` を実行します。setup コマンドを実行すると、そのキットが `/var/adm/update/NHDKit` ディレクトリにコピーされます。

次の例のように、任意のメンバで `clu_upgrade setup lead_memberid` コマンドを実行します。

```
# clu_upgrade setup 2
```

`clu_upgrade` コマンドで表示されるメニューを 7.7.2 項に示します。

セットアップ段階が完了すると、`clu_upgrade` はプロンプトを表示して、先行メンバ以外の全クラスタ・メンバのリポートを求めます。

3. 先行メンバ以外のすべてのクラスタ・メンバを一度に 1 つずつリポートします。これらのメンバがリポートされるか停止されるまで、プリ・インストール段階を開始しないでください。
4. プリ・インストール段階を実行します (7.7.3 項)。

先行メンバで、次のコマンドを実行します。

```
# clu_upgrade preinstall
```

現在のクラスタがバージョン 5.1A 以降である場合は、`preinstall` コマンドを実行するときに、セットアップ段階で作成されたタグ付きファイルがあるかどうかを、必要に応じて確認することができます。

- セットアップ段階を完了したばかりで、タグ付きファイルを削除するような操作をまだ行っていない場合は、このテストを省略してもかまいません。
- セットアップ段階を完了してから時間が経過して、どうすべきかがよくわからなくなった場合は、`preinstall` を実行して、タグ付きファイルが正しいかどうかをテストしてください。

5. インストール段階を実行します (7.7.4 項)。

注意

インストール段階では、新しいソフトウェアを先行メンバにロードし、実質的にそのメンバをロールします。ロール段階を実行すると、この新しいソフトウェアはクラスタ内の他のメンバに広まります。

`clu_upgrade` コマンドは、インストール段階ではソフトウェアをロードしません。ソフトウェアのロードはどのコマンド (`installupdate` , `dupatch` , `nhd_install`) を実行するかで決まります。

バージョン 5.1A およびバージョン 5.1B で行えるローリング・アップグレード作業とその組み合わせについては、表 7-1 を参照してください。

- a. `installupdate` コマンドの使用方法については、Tru64 UNIX の『インストール・ガイド』を参照してください。

`dupatch` コマンドの使用方法については、パッチ・キットに添付されている Tru64 UNIX と TruCluster Server の『*Patch Kit Installation Instructions*』を参照してください。

`nhd_install` コマンドの使用方法については、NHD キットに添付されている Tru64 UNIX の『*New Hardware Delivery Release Notes and Installation Instructions*』を参照してください。

- b. インストールするソフトウェアが、シングルユーザ・モードでインストール・コマンドを実行しなければならないソフトウェアである場合は、次のようにして、システムを停止した後、シングルユーザ・モードでブートします。

```
# shutdown -h now
>>> boot -fl s
```

注意

システムを停止してリブートすることで、クラスタに提供するサービスを最小限に抑えることができます。その結果、稼働中のクラスタがこのシングルユーザ・モードで動作するメンバに依存する度合も最小限に抑えることができます。特に、サービスのフェイルオーバを完結させるために、クラスタ・メンバを一度は「DOWN」ステータスにしなければならないサービスがあれば、メンバを停止させることで、その要求を満たすこともできます。最初にクラスタ・メンバを停止しないと、サービスが期待どおりにフェイルオーバしません。

システムがシングルユーザ・モードになったら、次のコマンドを実行します。

```
# init s
# bcheckrc
# lmf reset
```

- c. `installupdate` , `dupatch` , または `nhd_install` を実行します。

複数のパッチ・キットをロールするには、1 回のインストール段階で `dupatch` コマンドを複数回実行します。ただし、パッチ処理が完了して、クラスタの使用中に問題が発生した場合、問題を特定することが難しくなる場合がありますので注意してください。

`installupdate` コマンドの前に `dupatch` コマンドを実行することはできません。ローリング・アップグレードを実行する前に現在のソフトウェアにパッチをあてるには、ローリング・アップグレード操作を 2 回 (現在のソフトウェアに対するパッチを 1 回とアップデート・インストレーションを 1 回) 実行する必要があります。

6. (オプション) 新しいカスタム・カーネルを使用して先行メンバを最終的にリブートしてから他のメンバをロールするまでの間に、次のテストを手動で実行することもできます。

- a. ロール済みの先行メンバから共用ルート (/) ファイル・システムをサービスできることを確認します。

- i. 次のように `cfsmgr` コマンドを使用し、その時点でルート・ファイル・システムをサービスしているクラスタ・メンバを調べます。

```
# cfsmgr -v -a server /  
  
Domain or filesystem name = /  
Server Name = polishham  
Server Status : OK
```

- ii. 次のコマンドを実行して、ルート (/)・ファイル・システムを先行メンバに再配置します。

```
# cfsmgr -h polishham -r -a SERVER=provolone /
```

- b. 先行メンバからクライアントにアプリケーションをサービスできることを確認します。クラスタからクライアントへサービスするすべての重要なアプリケーションを先行メンバからサービスすることも確認します。

テストする内容と方法を決定します。次の例のように、ロールを続ける前に重要なアプリケーションをすべて検査して、先行メンバからクライアントにこれらのアプリケーションが正しくサービスされることを確認することをお勧めします。

- CAA サービスを先行メンバに手動で再配置します。たとえば、`cluster_lockd` というアプリケーション・リソースを `provolone` 先行メンバに再配置するには、次のコマンドを実行します。

```
# caa_relocate cluster_lockd -c provolone
```

- 次のように、省略時のクラスタ別名に対する選択優先順位属性 `selp` を一時的に変更して、先行メンバがその別名宛てに送信された要求をすべてサービスするように強制します。

```
# cluamgr -a alias=DEFAULTALIAS,selp=100
```

この操作で、省略時のクラスタ別名に宛てられた接続要求とパケットは、先行メンバが最終的にすべて受信するようになります。

他のメンバまたは他のクライアントから `telnet` や `ftp` などのサービスを使用して、先行メンバで別名宛の通信が処理されることを確認します。クラスタで提供する重要なサービスにクライアントからアクセスできるかどうかを検査します。

問題がない場合は、先行メンバで別名の属性を元の値に再設定します。

7. ポスト・インストール段階を実行します (7.7.5 項)。

先行メンバで次のコマンドを実行します。

```
# clu_upgrade postinstall
```

8. ロール段階を実行します (7.7.6 項)。

ロール前のクラスタ・メンバをロールします。¹

ロールされていないメンバの数 (クォーラム・ディスクが構成されている場合はクォーラム・ディスクも含む) がクラスタ・クォーラムを維持するのに十分な場合、複数のメンバを同時にロールすることができます (並列ロール)。

メンバをロールするには、以下の手順を実行します。

- a. 次のようにして、メンバ・システムを停止させた後シングルユーザ・モードでブートします。

```
# shutdown -h now
>>> boot -fl s
```

- b. システムがシングルユーザ・モードになったら、次のコマンドを実行します。

```
# init s
# bcheckrc
# lmf reset
```

- c. メンバをロールします。

```
# clu_upgrade roll
```

¹ 先行メンバはインストール段階ですでにロールされています。したがって、先行メンバに対してはロール段階を実行しません。

並列ロールを実行する場合、`clu_upgrade roll` コマンドの `-f` オプションを使います。このオプションをつけると、許可の入力を求めることなく自動的にメンバをリブートします。

```
# clu_upgrade -f roll
```

ロール・コマンドは、メンバをロールすることによってクォーラムが失われないことを確認します。ロールの結果クォーラムが失われると判断した場合、メンバはロールされず、エラー・メッセージが表示されます。現在ロール中のメンバの 1 つがクラスタに戻り、クォーラム・ポートが利用可能になると、メンバをロールすることができます。

ロールが進むと、メンバはリブート可能になります。`-f` オプションを使うと、プロンプトが表示されずに自動的にリブートされます。`-f` オプションを使わない場合、`clu_upgrade` はプロンプトを表示し、この時点でリブートを行うかどうかについて質問してきます。リブート前に特に調べたいことがなければ、**yes** と入力します (**yes** と入力してから実際にリブートが行われるまで 30 秒程度かかることがあります)。

ロール段階を完了するのに必要な時間を短縮するには、並列ロールを実行します。たとえば、クォーラム・ディスクのある 8 メンバで構成されるクラスタでは、先行メンバをロールした後、4 メンバを同時にロールすることができます。

- i. メンバでロール段階を開始します (先行メンバはインストール段階でロールされているため、先行メンバについてはロール段階は実行しません)。
- ii. 以下のようなメッセージが表示されたら、次のメンバについてロール段階を開始します

```
*** Info ***
You may now begin the roll of another cluster member.
```

以下のような内容で始まるメッセージが表示されたら、メンバ・ポートに占める現在ロール中のメンバ数に起因する可能性があります。

```
*** Info ***
The current quorum conditions indicate that beginning
a roll of another member at this time may result in
the loss of quorum.
```

この場合、以下の選択肢があります。

- メンバがロール段階を完了するまで待ってから次のメンバに対してロールを開始する。
- メンバ・ポートに含まれないロールされていないメンバが存在する場合は、そのメンバに対してロール段階を開始する。

- d. クラスタのすべてのメンバがロールされるまで、ロールを続けます。それぞれのロール段階について、開始できることを示すメッセージが表示されてから、ロール段階を開始します。

最後のメンバをロールすると、以下のようなメッセージが表示されます

```
*** Info ***  
This is the last member requiring a roll.
```

注意

実際のロールは、リブート中に行われます。`clu_upgrade roll` コマンドは、リブート中に実行する `it(8)` スクリプトをセットアップします。リブートが始まると、`it` スクリプトはメンバをロールして、カスタマイズしたカーネルを構築した後、そのメンバを再度リブートして、新しくカスタマイズしたカーネルが実行されるようにします。メンバが新しくカスタマイズしたカーネルをブートすると、ロールは終了し、タグ付きファイルを使用しないで動作するようになります。

-
9. スイッチ段階を実行します (7.7.7 項)。

すべてのメンバをロールした後、任意のメンバで `switch` コマンドを実行します。

```
# clu_upgrade switch
```

10. クラスタ・メンバを一度に 1 つずつリブートします。

11. クリーンアップ段階を実行します (7.7.8 項)。

任意のメンバで次のコマンドを実行し、クラスタのタグ付きファイル(.old..)を削除して、アップグレードを完了します。

```
# clu_upgrade clean
```

7.4 ローリング・アップグレードのステータス表示

clu_upgrade コマンドでは、ローリング・アップグレードのステータスを表示するための次のようなオプションを提供しています。ステータス・コマンドは、いつでも実行することができます。

- ローリング・アップグレードの全体的なステータスを表示するには、次のコマンドを実行します。

```
clu_upgrade -v または clu_upgrade -v status
```

- 各段階を実行できるかどうかを判断するには、次のコマンドを実行します。

```
clu_upgrade check [stage]
```

stage の値を指定しない場合、clu_upgrade では、次の段階を実行できるかどうかを検査されます。

- 各段階が処理中であるかまたは完了しているかを確認するには、次のコマンドを実行します。

```
clu_upgrade started stage または clu_upgrade completed stage
```

- メンバがロール済みかどうかを確認するには、次のコマンドを実行します。

```
clu_upgrade check roll memberid
```

- レイヤード・プロダクトに対してタグ付きファイルが作成されているかどうかを確認するには、次のコマンドを実行します。

```
clu_upgrade tagged check [prod_code [prod_code ...]]
```

製品コードを指定しないと、クラスタ内のすべてのタグ付きファイルが検査されます。

注意

ロール中、クラスタには2つのバージョンのclu_upgrade コマンドが存在することがあります。2つのバージョンとは、ロール

前のメンバで使用する旧バージョン，および新バージョン (アップデート配布キットまたはパッチ・キットに含まれる場合) です。
`status` コマンドで表示される情報がロール前のメンバとロール後のメンバでは異なることがあります。そのため，2 つのメンバで `status` コマンドを実行した場合は，出力表示の形式が異なることがありますので注意してください。

`installupdate` を実行した後で `clu_upgrade status` を実行すると，`clu_upgrade` はメッセージを表示して，インストール段階が完了していることを通知してきます。しかし，実際には，`clu_upgrade postinstall` コマンドを実行するまでインストール段階は完了していません。

7.5 各段階の取り消し

`clu_upgrade undo` コマンドでは，スイッチ段階を完了していないローリング・アップグレードを取り消すことができます。スイッチ段階とクリーンアップ段階を除いて，どの段階でも取り消すことができます。段階の取り消しは正しい順序で行わなければなりません。たとえば，プリ・インストール段階を完了した後でローリング・アップグレードを取り消す場合は，プリ・インストール段階を取り消してから，セットアップ段階を取り消します。

注意

段階を取り消す前に，関連するバージョンの『クラスタ・リリース・ノート』を読み，段階の取り消しに関連する制約事項があるかどうかを調べておくようお勧めします。

段階を取り消すには，取り消す段階を指定して `undo` コマンドを実行します。`clu_upgrade` コマンドでは，指定した段階が取り消し可能であるかどうかを確認します。段階を取り消すための条件については，表 7-3 を参照してください。

表 7-3: 各段階の取り消し

取り消す段階	コマンド	説明
セットアップ	<code>clu_upgrade undo setup</code>	<p>このコマンドは、先行メンバで実行する。また、セットアップ段階を取り消す場合は、どのメンバもタグ付きファイルを使用して実行できない。</p> <p>セットアップ段階を取り消す前に、<code>clu_upgrade -v status</code> コマンドを実行し、タグ付きファイルを使用して実行しているメンバを検索する。次に、<code>clu_upgrade tagged disable memberid</code> コマンドを実行して、これらのメンバのタグ付きファイルを無効にする(タグ付きファイルとその操作に使用するコマンドについては、7.8 節を参照)。</p> <p>タグ付きファイルを使用して実行中のメンバがない場合は、先行メンバで <code>clu_upgrade undo setup</code> コマンドを実行する。</p>
プリ・インストール	<code>clu_upgrade undo preinstall</code>	このコマンドは、先行メンバで実行する。
インストール	<code>clu_upgrade undo install</code>	<p>このコマンドは、先行メンバ以外のメンバで実行する。</p> <p>先行メンバを停止し、この先行メンバのブート・ディスクにアクセスできるメンバで <code>clu_upgrade undo install</code> コマンドを実行する。コマンドが完了した後で、先行メンバをブートする。</p>
ポスト・インストール	<code>clu_upgrade undo postinstall</code>	このコマンドは、先行メンバで実行する。
ロール	<code>clu_upgrade undo roll memberid</code>	<p>このコマンドはロール段階を取り消すメンバ以外のメンバで実行できる。</p> <p>ロール段階を取り消すメンバを停止する。停止したメンバのブート・ディスクにアクセスできる他のメンバで <code>clu_upgrade undo roll memberid</code> コマンドを実行する。コマンドが完了した後で、停止したメンバをブートする。これらの操作により、メンバでタグ付きファイルが使用される。</p>

7.6 ローリング・アップグレードのコマンド

ローリング・アップグレード全体の流れは、`clu_upgrade(8)` で説明されている `clu_upgrade` コマンドを使用して制御します。このコマンドを使用すれば、正しい順序で各段階を実行することができます。インストール段階では、`installupdate`、`dupatch`、および `nhd_install` のうちの 1 つまたは複数を使用して、ソフトウェアのロードとインストールを実行することができます。これらのコマンドはローリング・アップグレードに対応しています。つまり、ローリング・アップグレードのインストール段階とロール段階でどの動作が許可されているかを判断できるようになっています。

ローリング・アップグレードを開始すると、クラスタは以前のリリースからソフトウェアを実行します。ローリング・アップグレードの最初の部分では、以前からクラスタにインストールされている `clu_upgrade` コマンドを実行します。ローリング・アップグレード中に新しいバージョンがインストールされた場合、コマンドのバージョンの違いにより画面表示や動作が多少異なることがあります。

インストールするキットにアップグレード・コマンドの新しいバージョンが含まれていた場合にローリング・アップグレードのどの段階でその新バージョンが使用できるようになるかを、以下の2つの表に示します。²

- 表 7-4 は、バージョン 5.1A からバージョン 5.1B へローリング・アップグレードする場合、パッチ・キット、または NHD キットを適用する場合、またはバージョン 5.1B のベース・オペレーティング・システムおよびクラスタ・ソフトウェアへローリング・アップグレードした後に続けて新ソフトウェアのパッチをあてる場合の、コマンドと段階との対応を示しています。
- 表 7-5 は、オペレーティング・システムとクラスタ・ソフトウェアをバージョン 5.1B から次のリリースへローリング・アップグレードする場合、バージョン 5.1B パッチ・キット、または NKD キットを適用する場合、および、次のリリースのベース・オペレーティング・システムおよびクラスタ・ソフトウェアへローリング・アップグレードした後に続けて新しいソフトウェアにパッチをあてる場合の、コマンドと段階との対応を示しています。

² `clu_upgrade version` コマンドは、`clu_upgrade` のバージョン番号を表示します。この `clu_upgrade` バージョン番号はオペレーティング・システムのバージョン番号に対応していません。

表 7-4: バージョン 5.1A からローリング・アップグレードする場合の段階と clu_upgrade のバージョンとの対応

段階	バージョン 5.1A	次のリリース ^a	説明
準備	X		現在インストールされている (旧) バージョンの clu_upgrade が常に行われる。
セットアップ	X		現在インストールされている (旧) バージョンの clu_upgrade が常に行われる。 アップデート・インストールを行う場合は clu_upgrade の新バージョンが TruCluster Server キットから読み取られて /usr/sbin/clu_upgrade にインストールされ、旧バージョンと交換される。この交換はタグ付きファイルを作成する前に行われるため、それ以降のローリング・アップグレードでは、どのメンバもすべて新しい clu_upgrade を使用することになる。
プリ・インストール		X	ローリング・アップグレードでアップデート・インストールを行う場合は、どのメンバも、セットアップ段階の途中でインストールされた新バージョンの clu_upgrade を使用する (それ以外の場合は、どのメンバも現バージョンの clu_upgrade を継続して使用する)。
インストール		X	ローリング・アップグレードでアップデート・インストールを行う場合は、どのメンバも、セットアップ段階の途中でインストールされた新バージョンの clu_upgrade を使用する アップデート・インストールの途中で、installupdate が新バージョンのものに置き換えられる。 パッチ・キットを適用すると、常に最新バージョンの dupatch がインストールされる。 パッチ・キットに新しいバージョンの clu_upgrade が含まれている場合にそのパッチを適用すると、新バージョンがインストールされ、どのクラスタ・メンバもすべてポスト・インストール段階で新バージョンを使用する。

表 7-4: バージョン 5.1A からローリング・アップグレードする場合の段階と clu_upgrade のバージョンとの対応 (続き)

段階	バージョン 5.1A	次のリリース ^a	説明
ポスト・インストール		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの clu_upgrade がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。
ロール		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの clu_upgrade がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。
スイッチ		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの clu_upgrade がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。
クリーンアップ		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの clu_upgrade がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。

^aTru64 UNIX および TruCluster Server のバージョン 5.1B、バージョン 5.1A のパッチ・キット、およびバージョン 5.1A への NHD キットのインストール。

表 7-5: バージョン 5.1B からローリング・アップグレードする場合の段階と clu_upgrade のバージョンとの対応

段階	バージョン 5.1B	次のリリース ^a	説明
準備	X		この段階では、現在インストールされている (旧) バージョンの clu_upgrade が常に行われる。

表 7-5: バージョン 5.1B からローリング・アップグレードする場合の段階と `clu_upgrade` のバージョンとの対応 (続き)

段階	バージョン 5.1B	次のリリース ^a	説明
セットアップ	X		<p>この段階では、現在インストールされている (旧) バージョンの <code>clu_upgrade</code> が常に行われる。</p> <p>アップデート・インストレーションを実行する場合は、<code>clu_upgrade</code> の新バージョンが TruCluster Server キットから読み取られて <code>/usr/sbin/clu_upgrade</code> にインストールされ、旧バージョンと交換される。この交換はタグ付きファイルを作成する前に行われるため、それ以降のローリング・アップグレードでは、どのメンバもすべて新しい <code>clu_upgrade</code> を使用することになる。</p>
プリ・インストール		X	<p>ローリング・アップグレードでアップデート・インストレーションを行う場合は、どのメンバも、セットアップ段階の途中でインストールされた新バージョンの <code>clu_upgrade</code> を使用する (それ以外の場合は、どのメンバも現バージョンの <code>clu_upgrade</code> を継続して使用する)。</p>
インストール		X	<p>ローリング・アップグレードでアップデート・インストレーションを行う場合は、どのメンバも、セットアップ段階の途中でインストールされた新バージョンの <code>clu_upgrade</code> を使用する。</p> <p>アップデート・インストレーションの途中で、<code>installupdate</code> が新バージョンのものに置き換えられる。</p> <p>パッチ・キットを適用すると、常に最新バージョンの <code>dupatch</code> がインストールされる。</p> <p>パッチ・キットに新しいバージョンの <code>clu_upgrade</code> が含まれている場合にそのパッチを適用すると、新バージョンがインストールされ、どのクラスタ・メンバもすべてポスト・インストール段階で新バージョンを使用する。</p>

表 7-5: バージョン 5.1B からローリング・アップグレードする場合の段階と `clu_upgrade` のバージョンとの対応 (続き)

段階	バージョン 5.1B	次のリリース ^a	説明
ポスト・インストール		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの <code>clu_upgrade</code> がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。
ロール		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの <code>clu_upgrade</code> がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。
スイッチ		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの <code>clu_upgrade</code> がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。
クリーンアップ		X	セットアップ段階またはインストール段階のいずれかで新しいバージョンの <code>clu_upgrade</code> がインストールされた場合は、どのメンバーもすべて新バージョンを使用する。

^a次のリリースの Tru64 UNIX および TruCluster Server、バージョン 5.1B のパッチ・キット、およびバージョン 5.1B への NHD のインストール。

7.7 ローリング・アップグレードの各段階

この節では、ローリング・アップグレードの各段階について説明します。

注意

以降の各項では、各段階の概要を説明します。ローリング・アップグレードを実行する場合は、7.3 節で示す手順に従ってください。

- 準備段階 (7.7.1 項)
- セットアップ段階 (7.7.2 項)
- プリ・インストール段階 (7.7.3 項)
- インストール段階 (7.7.4 項)
- ポスト・インストール段階 (7.7.5 項)
- ロール段階 (7.7.6 項)

- スイッチ段階 (7.7.7 項)
- クリーンアップ段階 (7.7.8 項)

7.7.1 準備段階

コマンド	実行場所	実行レベル
<code>clu_upgrade -v check setup lead_memberid</code>	任意のメンバ	マルチユーザ・モード

準備段階では、クラスタの重要なデータをすべてバックアップし、クラスタをロールする準備ができているかどうかを検査します。ローリング・アップグレードの開始前に、次の操作を行います。

1. 最初にロールするメンバを、1 つ選択します。このメンバを先行メンバと呼びます。先行メンバは、ルート (/)、/usr、/var、および i18n (使用されている場合) ファイル・システムに対して直接アクセスできる必要があります。

先行メンバで重要なアプリケーションを実行できることを確認します。これらのアプリケーションをテストできるのは、インストール段階でこのメンバをアップデートしてから他のメンバをロールするまでの間です。問題が発生した場合は、次へ進む前にこのメンバで問題を解決します。問題を解決できない場合は、ローリング・アップグレードを取り消して、クラスタをロール前の状態に戻します (ローリング・アップグレードの各段階を取り消す方法については、7.5 節を参照)。

2. クラスタ単位のルート (/)、/usr、および /var ファイル・システム、およびこれらのファイル・システムに含まれるメンバ固有のファイルすべてのバックアップをとってください。クラスタに独立した i18n ファイル・システムが含まれている場合は、このファイル・システムのバックアップもとります。また、他にも重要なユーザ・データやアプリケーション・データを格納したファイル・システムがある場合は、そのファイル・システムのバックアップもとってください。

注意

ローリング・アップグレード中に実行するクラスタの増分バックアップまたフル・バックアップは、タグ付きファイルで実行しているメンバ以外のメンバだけを対象にしてください。タグ付きファイルを使用しているメンバからバックアッ

ブをとると、`.old..` ファイルの内容のみのバックアップをとることになります。先行メンバではタグ付きファイルを使用しないので、ローリング・アップグレード中は、先行メンバ (またはロール済みの他のメンバ) からクラスタのバックアップをとることができます。

通常のサイトでは、バックアップ手順が自動化されています。クラスタのローリング・アップグレード中に自動バックアップが実行されることがわかっている場合は、そのバックアップが先行メンバまたはロール済みのメンバで実行されることを確認してください。

-
3. インストール段階で `installupdate` コマンドを実行する場合は、表 7-6 に示す、クラスタにインストールされているブロッキング・レイヤード・プロダクトを削除します。
 4. `clu_upgrade -v check setup lead_memberid` を実行し、次の内容を確認します。
 - ローリング・アップグレードが進行中でないこと
 - すべてのメンバで同じバージョンのオペレーティング・システムとクラスタ・ソフトウェアを実行していること
 - タグ付きファイルを使用して実行されているメンバがないこと
 - ディスクに適切な空き容量があること
 5. 各システムのファームウェアが新しいソフトウェアをサポートしていることを確認します。必要があれば、ローリング・アップグレードを開始する前にファームウェアをアップデートします。

クラスタにはオペレーティング・システムとクラスタ・ソフトウェアが 2 組あるため、ローリング・アップグレード中でもクラスタを運用することができます (共用構成ファイルは 1 つだけなので、あるメンバで加えられた変更を他のメンバからも確認することができます)。この方法では、同じクラスタで同時に 2 種類のバージョンのベース・オペレーティング・システムを実行することができます。ただし、クラスタ単位のルート (`/`)、`/usr`、および `/var` ファイル・システムに、また、各国語サポート (WLS) サブセットの独立したドメインがある場合は `i18n` ファイル・システムに、それぞれ必要な空きディスク容量があるかどうかをアップグレード前に確認する必要があります。

ローリング・アップグレードを実行するには、次の空きディスク容量が必要です。

- ルート (/) ファイル・システムつまり `cluster_root#root` に 50 パーセント以上の空き容量。
- /usr ファイル・システムつまり `cluster_usr#usr` に 50 パーセント以上の空き容量。
- /var ファイル・システムつまり `cluster_var#var` に 50 パーセント以上の空き容量があり、さらに、オペレーティング・システムをアップデートする場合は、新バージョンのベース・オペレーティング・システムのサブセットを格納するために別途 425 MB が必要。
- WLS サブセットの独立した `i18n` ドメインがある場合は、そのドメインに 50 パーセント以上の空き容量。
- メンバ・ブート・パーティションにはタグ付きファイルは置かれない。ただし、プログラムがカーネルをブート・パーティションに移動する場合に空き容量が必要になることがあるので、各メンバのブート・パーティションには最低 50 MB の空き容量の確保を推奨。

注意

`addvol` コマンドを使用してメンバのルート・ドメイン (メンバのブート・ディスク上のパーティション) にボリュームを追加することはできません。ボリュームを追加する場合は、クラスタからそのメンバを削除して、`diskconfig` または `SysMan` を用いてディスクを適切に構成した後、そのメンバをクラスタに戻さなければなりません。

-
- パッチ・キットをインストールする場合は、パッチ・キットに添付されている『*Patch Kit Installation Instructions*』を参照し、このパッチ・キットのインストールに必要な空きディスク容量を確認する。
NHD キットをインストールする場合は、NHD キットに添付されている『*New Hardware Delivery Release Notes and Installation Instructions*』を参照し、このパッチ・キットのインストールに必要な空きディスク容量を確認する。

ファイル・システムにより多くの空き容量が必要な場合は、`addvol` などの AdvFS Utilities を使用して、必要に応じてドメインにボリュームを追加しま

す。AdvFS ドメインの管理方法については、Tru64 UNIX 『AdvFS 管理ガイド』を参照してください (AdvFS Utilities は別ライセンスが必要)。クラスタ単位のルート (/) ドメインは拡張することができます。

注意

clu_upgrade コマンドは、ローリング・アップグレードを開始するときにディスク容量を確認します。ただし、ローリング・アップグレード中は、クラスタ・メンバがディスク容量を消費することを防止できないので、後の段階でディスク容量が不足することもあります。

必要なディスク容量は場合に応じて変わってきます。ローリング・アップグレード中にメンバがディスク容量を大量に消費することがわかっている場合には、アップグレードの前に容量を追加しておいてください。

7.7.2 セットアップ段階

コマンド	実行場所	実行レベル
clu_upgrade setup lead_memberid	任意のメンバ	マルチユーザ・モード

セットアップ段階では、clu_upgrade check setup コマンドを実行してタグ付きファイルを作成し、クラスタをロールする準備をします。

clu_upgrade setup lead_memberid では、次のタスクが実行されます。

- ローリング・アップグレードのログ・ファイル /cluster/admin/clu_upgrade.log が作成される (D.3 節に clu_upgrade.log ファイルのサンプルがある)。
- 7.7.1 項の -v check setup テストが実行される。
- アップデート・インストレーション、パッチ・キットのインストール、NHD キットのインストール、またはその組み合わせのうち何を実行するかを確認するためのプロンプトが表示される。TruCluster Server バージョン 5.1B の clu_upgrade コマンドが表示するメニューの例を次に示す。

What type of rolling upgrade will be performed?

Selection	Type of Upgrade
1	An upgrade using the installupdate command
2	A patch using the dupatch command
3	A new hardware delivery using the nhd_install command
4	All of the above
5	None of the above
6	Help
7	Display all options again

Enter your Choices (for example, 1 2 2-3):

- アップデート・インストレーションを指定すると、関連キットがディスクにコピーされる。
 - アップデート・インストレーションを実行する場合は、クラスタ・キットが /var/adm/update/TruClusterKit にコピーされ、installupdate コマンドがインストール段階でそのキットを使用できるようになる (installupdate コマンドはインストール段階でオペレーティング・システム・キットを /var/adm/update/OSKit にコピーする)。clu_upgrade コマンドは、TruCluster Server キットの場所を絶対パス名で指定するためのプロンプトを表示する。TruCluster Server バージョン 5.1B クラスタでアップデート・インストレーションを含むローリング・アップグレードを実行する場合は、clu_upgrade setup コマンドを実行する前に、TruCluster Server キットをマウントすること。
 - TruCluster Server バージョン 5.1B クラスタで NHD インストレーションを実行する場合は、nhd_install コマンドを使用して NHD キットを /var/adm/update/NHDKit にコピーする。

警告

/var/adm/update にあるファイルは、ロール処理にとって非常に重要です。このディレクトリのファイルは削除も変更もしないでください。削除したり変更したりすると、ローリング・アップグレードが失敗するおそれがあります。

- OSF (ベース), TCR (クラスタ), および IOS (各国語サポート) 製品に対して必要なタグ付きファイル・セットが作成される。

警告

アップグレード中にレイヤード・プロダクトのタグ付きファイルを作成する必要がある場合は、7.8 節を参照してください。

- 先行メンバを除くすべてのメンバで `sysconfigtab` 変数が `rolls_ver_lookup=1` に設定される。`rolls_ver_lookup=1` に設定されると、そのメンバでタグ付きファイルが使用される。そのため、他のメンバが `.old.` ファイルを使用して動作している間に、先行メンバを現在のリリースからアップグレードすることができる。
- 先行メンバ以外のすべてのクラスタ・メンバをリブートするためのプロンプトが表示される。`setup` コマンドが完了した後で、クラスタがクォーラムを保持できるようにこれらのメンバを一度に1つずつリブートする。その場合、複数のバージョンが混在するクラスタ内でタグ付きファイルを使用するメンバごとにリブートする必要がある。リブートが完了すると、先行メンバ以外のすべてのメンバがタグ付きファイルを使用して実行される。

7.7.3 プリ・インストール段階

コマンド	実行場所	実行レベル
<code>clu_upgrade preinstall</code>	先行メンバ	マルチユーザ・モード

プリ・インストール段階の目的は、クラスタの先行メンバで `installupdate`、`dupatch`、および `nhd_install` コマンドを実行する準備ができているかどうかを確認することです。

`clu_upgrade preinstall` コマンドでは、次の内容が実行されます。

- このコマンドが先行メンバで実行されていること、実行中の先行メンバがタグ付きファイルを使用していないこと、および実行中の他のメンバがタグ付きファイルを使用していることが確認される。
- 必要に応じて、タグ付きファイルが存在すること、タグ付きファイルが製品のインベントリ・ファイルに一致すること、および各タグ付きファイルの AdvFS プロパティが正しく設定されていることが確認される (この処理には、やや時間がかかるが、セットアップ段階でタグ付きファイルを作成する操作よりは早く終了する。各段階の所要時間については、表 7-2 を参照)。

- 先行メンバ固有のファイルのバックアップ・コピーがディスク上に作成される。

7.7.4 インストール段階

コマンド	実行場所	実行レベル
installupdate	先行メンバ	シングルユーザ・モード
dupatch	先行メンバ	シングルユーザ・モードまたはマルチユーザ・モード
nhd_install	先行メンバ	シングルユーザ・モード

現在のクラスタが TruCluster Server バージョン 5.1B または バージョン 5.1A で動作している場合は、表 7-1 に示す作業を単独または組み合わせて実行することがきます。

インストール段階は、clu_upgrade preinstall コマンドが完了してから clu_upgrade postinstall コマンドを実行する直前までを指します。

注意

installupdate を実行した後で clu_upgrade status を実行すると、clu_upgrade はメッセージを表示して、インストール段階が完了していることを通知します。しかし、実際には clu_upgrade postinstall コマンドを実行するまでインストール段階は完了していません。

installupdate コマンドまたは nhd_install コマンドを実行するには、先行メンバをシングルユーザ・モードに設定する必要があります。また、dupatch コマンドを実行する場合もシングルユーザ・モードに設定することをお勧めします。システムをシングルユーザ・モードに設定する場合は、システムを停止してから、シングルユーザ・モードでリブートします。

シングルユーザ・モードに設定されたシステムで、init s、bcheckrc、および lmf reset コマンドを実行してから installupdate、dupatch、または nhd_install コマンドを実行します。これらのコマンドの使用方法については、Tru64 UNIX 『インストール・ガイド』、Tru64 UNIX

および TruCluster Server の『Patch Kit Installation Instructions』，および Tru64 UNIX『New Hardware Delivery Release Notes and Installation Instructions』を参照してください。

注意

複数のパッチをインストールするには，dupatch コマンドを複数回実行します。ただし，パッチ処理が完了してクラスタの使用中に問題が発生した場合，問題の特定が困難になります。

インストール段階では，installupdate コマンドより先に dupatch コマンドを実行することはできません。ローリング・アップグレードを実行する前に現在のソフトウェアにパッチをあてるには，ローリング・アップグレード操作を 2 回 (現在のソフトウェアに対するパッチを 1 回とアップデート・インストールを 1 回) 実行する必要があります。

アップデート・インストールを含むローリング・アップグレードの一環として NHD インストールを行う場合は，nhd_install を手動で実行する必要がありません。installupdate コマンドが NHD キットをインストールします。それ以外の場合は，セットアップ段階で clu_upgrade がコピーした nhd_install コマンドを使用します。このコマンドは /var/adm/update/NHDKit/nhd_install にあります。

7.7.5 ポスト・インストール段階

コマンド	実行場所	実行レベル
clu_upgrade postinstall	先行メンバ	マルチユーザ・モード

ポスト・インストール段階では，先行メンバでアップデート・インストール，パッチ，または NHD インストールが完了したことを確認します。アップデート・インストールを実行した場合は，先行メンバが新しいバージョンのベース・オペレーティング・システムにアップグレードされたことを clu_upgrade postinstall コマンドで確認します。

7.7.6 ロール段階

コマンド	実行場所	実行レベル
<code>clu_upgrade roll</code>	ロール中のメンバ	シングルユーザ・モード

インストール段階では先行メンバをアップグレードしましたが、他のメンバはロール段階でアップグレードします。

多くのクラスタ構成では、複数のメンバを並列でロールし、クラスタ・アップグレードに必要な時間を短縮することができます。並列的にロールできるメンバの数は、ロールされていないメンバ(クォーラム・ディスクが構成されている場合は、クォーラム・ディスクを含む)がクォーラムを維持するのに十分なポートを持つ必要があるという要件によってのみ制限されます。並列ロールは、先行メンバがロールされた後にだけ実行することができます。

`clu_upgrade roll` コマンドでは、次のタスクが実行されます。

- メンバが先行メンバでないこと、メンバがロール前であること、およびメンバがシングルユーザ・モードであることが確認される。メンバをロールすることにより、クォーラムが失われないことが確認される。
- メンバ固有のファイルのバックアップがとられる。
- リブート時にロールを実行するための `it(8)` スクリプトがセットアップされる。
- メンバがリブートされる。ブート中に `it` スクリプトによりメンバがロールされ、カスタマイズされたカーネルが作成される。そのカーネルで、リブートされる。

注意

ローリング・アップグレードの途中でクラスタにメンバを追加しなければならない場合は、ロールを完了したメンバからそのメンバを追加してください。

すべてのメンバをロールし終える前にあるメンバがダウンして修復もリブートもできない場合は、クラスタのロールを完了させるために、そのメンバを削除しなければなりません。ただし、1つのメンバを除いてすべてのメンバのロールが完了しているという状態で、ロールの完了していないそのメンバ

がロール段階でリブートする前にダウンした場合は、そのメンバを削除してから他のクラスタ・メンバをリブートする必要があります。`clu_upgrade` はリブート中も動作して、クラスタに現在あるメンバの数に対してロールされたメンバがどれだけあるかを追跡します。`clu_upgrade` は、それら 2 つの値が同じになると、ロール段階が完了したものとしてマークします。そのため、1 つのメンバを除いてすべてのメンバをロールした場合、ロールされていないメンバを削除して他のメンバをリブートすることにより、ロール段階が完了し、ローリング・アップグレードが継続できます。

7.7.7 スイッチ段階

コマンド	実行場所	実行レベル
<code>clu_upgrade switch</code>	任意のメンバ	マルチユーザ・モード すべてのメンバが起動している ^a

^aこの要件は、スイッチ・コマンドに `-f` オプションを使うことにより上書きできる。ただし、`-f` が機能するには、すべてのメンバのブート・ディスクがアクセス可能である必要がある。

スイッチ段階では、ソフトウェアのアクティブ・バージョンを新バージョンに設定します。その結果、ローリング・アップグレード中に無効にしておいた新機能が有効になります (アクティブなバージョンと新バージョンについての説明は、7.9 節を参照してください)。

`clu_upgrade switch` コマンドでは、次のタスクが実行されます。

- すべてのメンバのロールが完了したこと、すべてのメンバで同じバージョンのオペレーティング・システムとクラスタ・ソフトウェアが実行されていること、およびタグ付きファイルを使用して実行されているメンバがないことが確認される。
- 各メンバの `sysconfigtab` ファイルと実行中のカーネルに新しいバージョンの ID が設定される。
- すべてのクラスタ・メンバでアクティブ・バージョンが新バージョンに設定される。

注意

スイッチ段階の終了後は、クラスタ・メンバを一度に 1 つずつリブートする必要があります。

7.7.8 クリーンアップ段階

コマンド	実行場所	実行レベル
<code>clu_upgrade clean</code>	任意のメンバ	マルチユーザ・モード

クリーンアップ段階では、タグ付き (.Old..) ファイルをクラスタから削除して、アップグレードを完了します。

`clu_upgrade clean` コマンドでは、次のタスクが実行されます。

- スイッチ段階が完了したこと、すべてのメンバで同じバージョンのオペレーティング・システムとクラスタ・ソフトウェアが実行されていること、およびタグ付きファイルを使用して実行されているメンバがないことが確認される。
- すべての .Old.. ファイルが削除される。
- `clu_upgrade` コマンドで作成したディスク上のバックアップ・アーカイブが削除される。
- ディレクトリが存在する場合、`/var/adm/update/TruClusterKit`、`/var/adm/update/OSKit`、および `/var/adm/update/NHDKit` が再帰的に削除される。
- アップデート・インストレーションを実行した場合、アップデート管理ユーティリティ (`updadmin`) を実行して、アップデート・インストレーション中に保存されたファイルを管理するためのオプションが表示される。
- このアップグレードに対するアーカイブ・ディレクトリ (`/cluster/admin/clu_upgrade/history/release_version`) が作成され、`clu_upgrade.log` ファイルがこのアーカイブ・ディレクトリに格納される。

7.8 タグ付きファイル

ローリング・アップグレードでは、一度に1つのクラスタ・メンバのソフトウェアがアップデートされます。ロール中にクラスタで2つのバージョンのソフトウェアを動作させるには、セットアップ段階で `clu_upgrade` を実行し、タグ付きファイルを作成します。

タグ付きファイルは、現在のファイルをコピーして、元のファイル名の先頭に `.Old..` を追加してその名前にするとともに、AdvFS プロパティ (`DEC_VERSION_TAG`) を設定したファイルです。たとえば、`vdump` コマンドのタグ付きファイルの名前は、`/sbin/.Old..vdump` となります。タグ付きファイルは、元のファイルと同じファイル・システムに作成されるため、ローリング・アップグレードを開始する前に、十分なディスク容量があることを確認する必要があります。

メンバがタグ付きファイルを使用して実行されるかどうかは、そのメンバの `sysconfigtab rolls_ver_lookup` 変数によって制御されます。アップグレード・コマンドは、メンバをタグ付きファイルを使用して実行する必要がある、その値を 1 に、また、メンバをタグ付きファイルを使用して実行してはならない場合は、その値を 0 にそれぞれ設定します。

メンバの `sysconfigtab rolls_ver_lookup` 属性を 1 に設定すると、パス名を特定する場合に、指定したファイルと対応する `.Old..filename` というファイル名のコピーがあるかどうか、およびこのコピーに `DEC_VERSION_TAG` プロパティが設定されているかどうかを調べます。両方の条件が満たされると、要求したファイル操作が透過的に変更され、`.Old..filename` ファイルが使用されます。たとえば、ロールが完了していないメンバで `vdump` コマンドを実行すると、`/sbin/.Old..vdump` が実行され、ロール済みのメンバで実行すると、`/sbin/vdump` が実行されます。先行メンバ (最初にロールするメンバ) だけは、タグ付きファイルを使用して実行されません。

注意

ディレクトリに対するファイル・システム操作では、このタグ付きファイルの制約を受けることはありません。たとえば、ローリング・アップグレード中にクラスタ・メンバのディレクトリで `ls` を実行すると、両方のバージョンのファイルがリストされます。ただし、ローリング・アップグレードを完了する前のメンバと完了した後のメンバでは、`ls -ail` コマンドを実行した場合の出力が異なります。以下の例は、`ls -ail` コマンドを、ロールされていないメンバで最初に行なった後、ロールされたメンバで行なった例です (`awk` コーティリティは、各ファイルの i ノード、サイズ、タイム・スタンプ (月日)、およびファイル名だけを出力するために使用しています)。

次に示す出力は、タグ付きファイルを使用して動作している
ロール前のクラスタ・メンバで `ls` コマンドを実行した場合のも
のです。タグ付きファイルとタグの付いていないファイルは同
一です (i ノード、サイズ、タイム・スタンプが同じ)。このメン
バで `hostname` コマンドを実行すると、タグの付いている方 (i
ノード 3643) が実行されます。

```
# cd /sbin
# ls -ail hostname .Old..hostname ls .Old..ls init .Old..init |\
awk '{printf("%d\t%d\t%s %s\t%s\n",$1,$6,$7,$8,$10)}'
```

3643	16416	Aug 24	.Old..hostname
3648	395600	Aug 24	.Old..init
3756	624320	Aug 24	.Old..ls
3643	16416	Aug 24	hostname
3648	395600	Aug 24	init
3756	624320	Aug 24	ls

次に示す出力は、ロールした後にタグ付きファイルを使用しない
で動作しているクラスタ・メンバで `ls` コマンドを実行した場合の
ものです。タグ付きファイルとタグの付いていないファイルは
異なります (i ノード、サイズ、タイム・スタンプが異なる)。こ
のメンバで `hostname` コマンドを実行すると、タグの付いてい
ない方 (i ノード 1370) が実行されます。

```
# cd /sbin
# ls -ail hostname .Old..hostname ls .Old..ls init .Old..init |\
awk '{printf("%d\t%d\t%s %s\t%s\n",$1,$6,$7,$8,$10)}'
```

3643	16416	Aug 24	.Old..hostname
3648	395600	Aug 24	.Old..init
3756	624320	Aug 24	.Old..ls
1187	16528	Mar 12	hostname
1370	429280	Mar 12	init
1273	792640	Mar 12	ls

セットアップ段階でタグ付きファイルを作成した後は、ロールしたメンバか
ら `tar` のような管理タスクを実行することをお勧めします。先行メンバで
は、タグ付きファイルを実行しないので常にコマンドを実行できます。

セットアップ段階で自動的にタグ付きファイルが作成されるファイルを判
断する規則は、次のとおりです。

- タグ付きファイルは、ベース・オペレーティング・システム (OSF),
TruCluster Server (TCR), および各国語サポート (IOS) の製品コード
を持つインベントリ・ファイルに対して作成されます。各製品のサブ
セットには、3 文字の製品コードから始まる名前が指定されていま
す。たとえば、TruCluster Server のサブセット名は、TCRBASE540,

TCRMAN540 , TCRMIGRATE540 のように TruCluster Server の 3 文字の製品コードから始まります。

- 省略時の設定では、他のレイヤード・プロダクトに関連するファイルに対してはタグ付きファイルを作成しません。タグ付きファイルは、ローリング・アップグレード中にタグ付きファイルをサポートするように変更されたレイヤード・プロダクトだけに対して作成されます。

警告

最初にロールするメンバに製品の新バージョンをインストールできること、および複数のバージョンが混在するクラスターでレイヤード・プロダクトが動作できることが、レイヤード・プロダクトのドキュメントに特記されていない場合は、新しいレイヤード・プロダクトまたは現在インストールされているレイヤード・プロダクトの新バージョンをローリング・アップグレード中にインストールしないでください。

`clu_upgrade` コマンドには、タグ付きファイル进行操作するためのさまざまな `tagged` コマンド・オプション (`check` , `add` , `remove` , `enable` , および `disable`) を指定することができます。タグ付きファイル进行操作の場合は、次の点に注意してください。

- 通常、ローリング・アップグレード中にタグ付きファイルを手動で追加したり、削除したりする必要はありません。`clu_upgrade` コマンドでは、必要に応じて `tagged` コマンドが呼び出され、タグ付きファイルの作成と削除が制御されます。
- `clu_upgrade tagged` コマンドを実行する場合、`check` , `add` , `remove` コマンドは、先行メンバのように、タグ付きファイルを使用しないで動作しているメンバで実行してください。`disable` および `enable` コマンドはどのメンバでも実行できます。
- タグ付きファイルに関する `check` , `add` , または `remove` 操作の指定は、製品全体を表す製品コードを対象とします。`clu_upgrade tagged` コマンドは、指定した製品のすべてのインベントリ・ファイルを対象にします。たとえば、次のコマンドでは、TCR カーネル・レイヤード・プロダクト (TruCluster Server のサブセット) に対して作成されたタグ付きファイルがすべて正しいかが検査されます。

```
# clu_upgrade tagged check TCR
```

1 つの .Old.. ファイルを誤って削除した場合、このファイルを再作成するために、レイヤード・プロダクト全体に対してタグ付きファイルを作成する必要があります。たとえば、vdump コマンドが OSF 製品の OSFADVFSxxx サブセットに含まれる場合に /sbin/.Old..vdump を誤って削除すると、次のコマンドを実行して、レイヤード・プロダクト全体のタグ付きファイルを再作成する必要があります。

```
# clu_upgrade tagged add OSF
```

- enable コマンドと disable コマンドは、クラスタ・メンバごとにタグ付きファイルを使用可能または使用不能にする場合に実行します。通常、ローリング・アップグレード中には、enable コマンドまたは disable コマンドを使用する必要はありません。

disable コマンドは、セットアップ段階を取り消す場合に便利です。セットアップ段階を取り消す場合は、どのメンバもタグ付きファイルを使用して実行できないため、disable コマンドで、現在タグ付きファイルを使用して実行されているクラスタ・メンバのタグ付きファイルを無効にします。たとえば、ID が 3 のメンバのタグ付きファイルを無効にするには、コマンドを次のように指定します。

```
# clu_upgrade tagged disable 3
```

enable コマンドは、disable コマンドを誤って実行した場合に使用します。

7.9 バージョン・スイッチ

バージョン・スイッチは、アクティブ・バージョンのオペレーティング・システムから新しいバージョンのオペレーティング・システムへの移行を管理する場合に使用します。アクティブ・バージョンとは、現在使用中のバージョンを指します。クラスタでバージョン・スイッチを使用する目的は、すべてのメンバがアップデートされるまで、本質的に互換性がない新機能を導入しないようにするためです。たとえば、新バージョンでカーネル構造が変更されており、現在の構造と互換性がない場合、通常は、すべてのクラスタ・メンバがその新構造をサポートするバージョンにアップデートされるまで、クラスタ・メンバで新構造を使用しません。

ローリング・アップグレードの開始時には、どのメンバでもアクティブ・バージョンは新バージョンと同じです。ローリング・アップグレードが進む

につれて各メンバで新バージョンがアップデートされます。すべてのメンバでローリング・アップグレードが終了した後のスイッチ段階で、すべてのメンバのアクティブ・バージョンが新バージョンに設定されます。アップグレードが完了すると、再びすべてのメンバのアクティブ・バージョンが新バージョンと同じになります。次の簡単な例に、アクティブ・バージョン 1 と新バージョン 2 を使用して、ローリング・アップグレードの過程でバージョンが変わる様子を示します。

```
All members at start of roll:   active (1)  = new (1)
Each member after its roll:    active (1) != new (2)
All members after switch stage: active (2)  = new (2)
```

バージョンの移行を管理するには、`clu_upgrade` コマンドに `versw` コマンド (`versw(8)` を参照) を指定します。`clu_upgrade` コマンドを使用して、各メンバをロールするときにバージョン・スイッチのすべてのアクティビティを管理します。すべてのメンバをロールした後のスイッチ段階で、次のコマンドを実行して新しいソフトウェアへの移行を完了させます。

```
# clu_upgrade switch
```

7.10 ローリング・アップグレードとレイヤード・プロダクト

ここでは、レイヤード・プロダクトとローリング・アップグレードの相互作用を次の項に分けて説明します。

- 一般的なガイドライン (7.10.1 項)
- ブロッキング・レイヤード・プロダクト (7.10.2 項)

7.10.1 一般的なガイドライン

`clu_upgrade setup` コマンドは、クラスタでオペレーティング・システムをローリング・アップグレードするための準備を行います。`clu_upgrade setup` コマンドを実行してから最初のメンバを新しいバージョンにロールするまでの間は、`setld` コマンドを用いてクラスタにソフトウェアをロードしないでください。`clu_upgrade setup` コマンドを実行してからクラスタ・メンバを新しいバージョンにロールするまでの間にソフトウェアをインストールすると、新しいファイルは `clu_upgrade setup` で処理されなくなります。その結果、最初のクラスタ・メンバをロールする際に、その新しいファイルが上書きされてしまいます。

ソフトウェアをロードしなければならない場合は、次のような方法で行ってください。

- 少なくとも 1 つのメンバがロールされるまで待つ。
- ロールされたメンバでソフトウェアをインストールする。

7.10.2 ブロッキング・レイヤード・プロダクト

ブロッキング・レイヤード・プロダクトは、`installupdate` コマンドによるアップデート操作を妨げる製品です。`installupdate` コマンドを実行するローリング・アップグレードの開始前にクラスタからブロッキング・レイヤード・プロダクトを削除する必要があります。クラスタにパッチをあてるため、または NHD キットをインストールするためだけにローリング・アップグレードを実行する場合は、ブロッキング・レイヤード・プロダクトを削除する必要はありません。

表 7-6 は、このリリースのブロッキング・レイヤード・プロダクトをまとめたものです。

表 7-6: ブロッキング・レイヤード・プロダクト

製品コード	説明
3X0	Open3D
4DT	Open3D
ATM	Atom Advanced Developers Kit
DCE	Distributed Computing Environment
DNA	DECnet
DTA	Developer's Toolkit (プログラム分析ツール)
DTC	Developer's Toolkit (C コンパイラ)
MME	マルチメディア・サービス
O3D	Open 3D
PRX	PanoramiX Advanced Developers Kit

注意

3 文字の製品コードは、サブセット名の最初の 3 文字です。たとえば、`ATMBASExxx` という名前のサブセットは、ブロッキング・レイヤード・プロダクトの 1 つである ATM 製品 (Atom Advanced

Developers Kit) の一部です。OSFATMBIN_{xxxx} というサブセットは名前に ATM を含みますが、このサブセットはブロッキング・レイヤード・プロダクトの一部ではありません。これは OSF 製品 (ベース・オペレーティング・システム) のサブセットです。

ブロッキング・レイヤード・プロダクトをローリング・アップグレードの一環として削除すると、すべてのメンバからブロッキング・レイヤード・プロダクトが削除されます。ブロッキング・プロダクトに依存するサービスは、ロールが完了し、ブロッキング・レイヤード・プロダクトが再インストールされた後で使用できるようになります。

7.11 ローリング・アップグレードと RIS

ローリング・アップグレードのインストール段階を実行する場合、CD-ROM またはリモート・インストレーション・サービス (RIS) サーバからベース・オペレーティング・システムのサブセットをロードすることができます。

注意

RIS は、ベース・オペレーティング・システムのサブセットをロードする場合だけに使用できます。

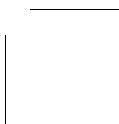
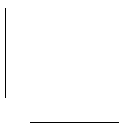
RIS を使用するには、先行メンバおよび省略時のクラスタ別名の両方を RIS サーバに登録する必要があります。オペレーティング・システム・ソフトウェアの登録では、各ホスト名のハードウェア・アドレスが必要です。そのため、省略時のクラスタ別名を RIS サーバに登録するには、省略時のクラスタ別名に対するハードウェア・アドレスを生成する必要があります (RIS サーバの /etc/bootptab ファイルまたは /var/adm/ris/clients/risdb ファイルにすでに登録されているアドレスは受け付けられません)。

クラスタでクラスタ別名の仮想 MAC (vMAC) 機能を使用する場合は、省略時のクラスタ別名のハードウェア・アドレスとしてその仮想ハードウェア・アドレスを RIS サーバに登録します。クラスタで vMAC 機能を使用しない場合は、『クラスタ管理ガイド』の「vMAC」の節で説明するアルゴリズムを使用し、省略時のクラスタ別名に対するハードウェア・アドレスを手動で生成することもできます。

vMAC アドレスは、接頭文字 (省略時では AA:01) と、別名に対する 16 進数形式の IP アドレスで構成されます。たとえば、省略時のクラスタ別名が deli で、この IP アドレスが 16.140.112.209 である場合、省略時のクラスタ別名に対する vMAC アドレスは AA:01:10:8C:70:D1 で表されます。アドレスの作成方法は、次のとおりです。

省略時の vMAC 接頭文字:	AA:01
クラスタ別名の IP アドレス:	16.140.112.209
16 進数形式の IP アドレス:	10.8C.70.D1
この別名の vMAC :	AA:01:10:8C:70:D1

8 つの 16 進数で構成される任意の文字列を vMAC の省略時の接頭文字 (AA:01) に追加して、ハードウェア・アドレスを作成することもできます (例 AA:01:00:00:00:00)。このアドレスが RIS サーバからサービスされる領域内で一意であることを確認してください。クラスタが複数ある場合に次の別名を追加するには、16 進数の任意の文字列の値を大きくしてください (vMAC アルゴリズムを使用すると、ネットワーク内で一意のアドレスを作成できる可能性が高いので便利です)。



TruCluster Software バージョン 1.5 またはバージョン 1.6 からのアップグレード

この章では、TruCluster Production Server Software あるいは TruCluster Available Server Software バージョン 1.5 またはバージョン 1.6 から TruCluster Server バージョン 5.1B にアップグレードする際の注意事項について説明します (この章の主な焦点は Production Server および Available Server クラスタのアップグレードですが、8.9 節で、Memory Channel Software を実行するクラスタのアップグレードも説明しています)。

注意

この章では、便宜上、バージョン 1.5 または 1.6 の TruCluster Production Server Software クラスタ、あるいは TruCluster Available Server Software を指すのに「ASE」という用語を使用し、新しい TruCluster Server バージョン 5.1B クラスタを指すのに「クラスタ」という用語を使用します。

この章では、次の 3 つのアップグレード・パスについて説明します。これらのパスは、TruCluster Production Server Software あるいは TruCluster Available Server Software バージョン 1.5 またはバージョン 1.6 からのアップグレードに使用します。

- オプション 1: 新しいシステムと新しいストレージ・ハードウェアを使用して別のクラスタを作成する。新しいクラスタが完全に構成されてテストされた後、データを ASE から新しいクラスタに移行する。
- オプション 2: 新しいシステムとクラスタおよびテスト・アプリケーションを作成するのに十分な新しいストレージを使用して、別のクラスタを作成する。新しいクラスタが完全に構成されてテストされた後、ASE から新しいクラスタへ古いストレージを物理的に移動する。
- オプション 3: 既存のハードウェアとストレージを使用したまま、ASE をアップグレードする。ASE から 1 つのメンバを削除して取り外し、

そのシステムで Tru64 UNIX バージョン 5.1B をインストールおよび構成して、ASE 内の他のシステムをシャットダウンする。ASE のストレージを Tru64 UNIX バージョン 5.1B システムに接続して、そのストレージを構成し、シングル・メンバ・クラスタを作成したのち、他のシステムを新しいクラスタに追加する。

注意

バージョン 1.5 より前の TruCluster Production Server Software または TruCluster Available Server Software 製品では、Tru64 UNIX バージョン 5.1B と TruCluster Server バージョン 5.1B のフル・インストールを行う必要があります。

TruCluster 製品に対してサポートされているアップグレード・パスのリストについては、表 1-1 を参照してください。

これとは別に、現在の構成と用途に応じて独自のカスタマイズ手順を設計することもできます。8.8 節に、オプション 3 を一部変更した手順をケース・スタディとして示しています。他のケース・スタディについては以下の URL をご覧ください。そこには、社内の運用レベルのクラスタを TruCluster Production Server バージョン 1.6 からパッチ・キットを含めた TruCluster Server バージョン 5.0A にアップグレードする場合の説明が記載されています。

<http://www.tru64unix.compaq.com/docs/highavail/index.htm>

この章で説明する内容は次のとおりです。

- 別のクラスタを作成するか、または既存クラスタをアップグレードするかの決定方法 (8.1 節)
- ストレージとクラスタ・インターコネクトに関する制約事項のリスト (8.2 節)
- アップグレードの準備方法 (8.3 節)
- オプション 2 およびオプション 3 で使用されるアップグレード・スクリプトの説明 (8.4 節)
- オプション 1: 別のクラスタの作成 — 新しいシステムと新しいストレージを使用 (データのみ移行) (8.5 節)

- オプション 2: 別のクラスタの作成 — 新しいシステムを使用し、既存のストレージを移行 (物理ストレージを移動) (8.6 節)
- オプション 3: 既存 ASE のアップグレード — 既存のシステムおよびストレージを使用 (8.7 節)
- オプション 3 の手順を一部変更した手順を使用したアップグレードのケース・スタディ (8.8 節)
- TruCluster Server Memory Channel Software 製品を実行しているクラスタをアップグレードするためのアプローチ (8.9 節)

アップグレードの準備については、第 1 章 ~ 第 5 章を参照してください。

8.1 別のクラスタを作成するか既存クラスタをアップグレードするかの決定

アップグレードのための多数のアプローチの中から、この章では、ASE からクラスタにアップグレードするための 3 つの方法について説明します。

- オプション 1: すべて新しいシステム・ハードウェアとストレージで別個のクラスタを作成する。このクラスタは、テスト専用設計された 2 ノードの最小構成から本格的な運用レベルのクラスタまで作成できる。

この方法 (新しいシステムと新しいストレージ) では、現在のハードウェア構成の制約、あるいは、Tru64 UNIX バージョン 4.x オペレーティング・システムまたはバージョン 1.5/1.6 の TruCluster Software 製品における制約を受けずに、新しいクラスタを作成できます。たとえば、HSG80 コントローラと Fibre Channel を使用して新たに NSPOF (no-single-point-of-failure) クラスタを作成できます。

次に、以前の ASE を新しいバージョン 5.1B クラスタと並行して実行します。この構成では、既存の ASE からクライアントにサービスを提供する一方で、新しいクライアントでアプリケーションを広範囲にテストできます。新しいクラスタに満足した場合は、以前の ASE から新しいクラスタに (ストレージ・ハードウェアではなく) アプリケーション・データを移行します。

新しいシステムおよび新しいストレージとともに、別個のクラスタを並行して使用すると、共用されるハードウェアがなく、ASE から新しいクラスタに移されるストレージもないため、リスクは最も低くなります。新しいクラスタには継続使用されるハードウェアがないため、たとえ

ば、ハードウェアをマルチパス構成して冗長化するなど、TruCluster Server バージョン 5.1B のすべての機能を活用できます。

別個のクラスタの作成方法については、8.5 節で説明します。

- オプション 2: 新しいストレージをいくつか使用して別個のクラスタを作成するが、ASE から新しいクラスタに既存のストレージを移す。

この方法 (新しいシステムと古いストレージ) では、ASE に影響を与えることなく、新しいシステムを構成して、アプリケーションのテストを行うという限度において、ある程度の独立が提供されます。ただし、現在の ASE ストレージ構成の制約事項が、新しいクラスタにも適用されます。これらの制約事項のために、このクラスタでは新しい機能が利用できないこともあります。たとえば、古いストレージがマルチパス化をサポートしていないかもしれません。

8.6 節に、既存の ASE から新しいクラスタへストレージを移す方法と、新しいクラスタでストレージを構成する方法を示します。その節では、ストレージの移行と構成のために使用するスクリプトの処理も説明します。

注意

考慮しなければならないことの 1 つは、データの移行です。ユーザのサイトで、データのコピー (オプション 1)、またはストレージの物理的な移動 (オプション 2) の利点と不利な点を判断します。たとえば、ASE に大量の共用ストレージがあり、データのバックアップを取って、新しいクラスタにリストアするのに、たいへんな時間がかかるために無理な場合は、その環境にとって、ストレージを物理的に移動するのがより良い解決方法であると決定します。

オプション 1 では、ASE とクラスタの両方が、アプリケーション・データを完全に複製したセットを持つことができるため、クライアント・サービスを ASE からクラスタへ切り替える前に、アプリケーション環境を広範囲にわたってテストして調整することができます。

オプション 2 では、ASE のストレージをクラスタへ物理的に接続した後、新しいクラスタで、ASE から移したすべてのアプリケーション・データにアクセスできるようになります。

ある時点で ASE を停止し、物理ストレージ・デバイスを新しいクラスタへ移す必要があります。

- オプション 3: 既存の ASE ハードウェアを新しいクラスタのベースとして使用し、Memory Channel ハードウェア (希望する場合) とストレージを追加して、必要に応じてストレージとシステムを移行する。

この方法 (既存のシステムおよびハードウェア) では、金銭的なコストを最も低く抑えることができます。ただし危険性は、完全に別個のクラスタを実行する場合よりも本質的に高まります。稼働中の ASE からメンバを削除する必要があるため、クライアントに対するサービスに利用できるシステムの数は一時的に減少します。アプリケーションのテストは、マルチ・メンバ・クラスタではなくシングル・メンバ・クラスタで行われるため、アプリケーション・フェイルオーバー・テストは実施できません。

この方法については、8.7 節で説明します。そこでは、スクリプトを使用して ASE が認識しているストレージの移行と構成を容易に行う手順についても説明します。

どの方法を採用するのがよいかわからない場合は、決定する前に章全体をお読みになることをお勧めします。この章の情報を出発点として使用して、ニーズに合ったアプローチと手順を決定してください。8.8 節と以下の URL に、アップグレードのケース・スタディが記載されています。

<http://www.tru64unix.compaq.com/docs/highavail/index.htm>

これらのケース・スタディを読めば、どのアプローチを用いてアップグレードすればよいかを決定するのに役立ちます。

8.2 ストレージとクラスタのインターコネクトに関する制約事項

この節では、ASE から TruCluster Server クラスタへアップグレードする場合の一般的な制約事項を示します。既存のハードウェアをアップグレードするときには制約が厳しくなりますが、どのようなアップグレード形態であっても、同一のストレージ・バスまたは同一のクラスタ・インターコネクトにおいて ASE と新しいクラスタの両方を同時にアクティブにすることはできません。

- **ASE** と新しいクラスタのシステムを同一のストレージ・バス上で実行してはならない。

ストレージは、ASE または新しいクラスタのいずれかによってアクセスされ、両方によってアクセスされることはありません。ASE と新しいクラスタの両方からシステムが同一のストレージ・デバイスにアクセスできる場合、データが破壊される可能性があります。ストレージを移行するときには、ASE システムを共用ストレージから物理的に分離するか、またはシステムを停止し、電源をオフにしてください。

- **ASE** のシステムと新しいクラスタのシステムを同一のクラスタ・インターコネクト上で実行してはならない。

クラスタ・インターコネクト・ハードウェアは ASE または新しいクラスタのいずれかに稼働状態のまま接続できますが、両方には絶対に接続しないでください。ASE と新しいクラスタがどちらも同じクラスタ・インターコネクトに接続されると、システムのブート不良のために、さまざまな問題が発生するおそれがあります (マシン・チェックなど)。システムを移行するときには、既存の ASE システムをクラスタ・インターコネクトから物理的に切り離すか、またはシステムを停止し、電源をオフにしてください。

- 新しいクラスタの第 1 メンバになる **ASE** システムを停止したら、既存デバイスを新しいクラスタ上の新スタイルのデバイス名にすべてマップするまで、ストレージ・トポロジを変更してはならない。

アップグレードを開始してからデバイスを新スタイルのデバイス名にマップするまでに、ストレージ・トポロジを変更すると、Tru64 UNIX バージョン 5.1B システムにのみ認識されているデバイスが導入されます。そのため、デバイスのマッピングが正しいかどうかを確認することが困難になります。この制約は、主に、既存のハードウェアを使用してアップグレードする場合に適用されます。ただし、既存のストレージを別個のクラスタに物理的に接続する場合は、この制約が適用されます。

8.3 アップグレードの準備

以降の各項で、アップグレードのための準備を行う方法について説明します。

- 一般的な要件 (8.3.1 項)
- ハードウェアおよびストレージ・トポロジに関する要件 (8.3.2 項)
- ファイル・システムに関する要件 (8.3.3 項)

8.3.1 一般的な要件

準備を始める前に、既存の ASE と、TruCluster バージョン 5.1B の新しい機能とアーキテクチャとの相違点を理解しておく必要があります。相違点を理解したら、ユーザのサイト固有のニーズに応じたアップグレード計画を立案できます。

- 新しい AdvFS フォーマット、拡張 SCSI サポート、および新しいデバイス命名規則については、Tru64 UNIX 『*Tru64 UNIX 概要*』および『*システム管理ガイド*』を参照してください。
- TruCluster Server バージョン 5.1B の推奨構成の詳細と、動作上および NSPOF (no-single-point-of-failure) に関する TruCluster Server バージョン 5.1B のクラスタ構成と以前の TruCluster 構成との重要な相違点については、TruCluster Server 『*クラスタ・ハードウェア構成ガイド*』および『*クラスタ概要*』を参照してください。
- バージョン 5.1B クラスタで高可用性アプリケーションを実行する方法については、『*クラスタ高可用性アプリケーション・ガイド*』を参照してください。TruCluster Server では、高可用性サービスを提供するときに ASE パラダイムを使用しません。つまり、`asemgr` コマンドも `asecdb` データベースもありません。代わりに、TruCluster Server では、クラスタ・ファイル・システム (CFS)、CAA (cluster application availability)、およびクラスタ別名機能を使用して、高可用性アプリケーションを提供します。

注意

Oracle や Informix などのサード・パーティから出ているアプリケーションを使用する場合は、そのアプリケーションのベンダにお問い合わせください。

-
- 新しいクラスタでハードウェア RAID を使用してファイル・システムをミラー化する場合、アップグレードは、RAID ハードウェアを追加して、マルチパス化を実施するよい機会となります。
 - 綿密な計画を作成して、構成図を作成します。現在の環境の構成マップを作成するには、`cluster_map_create -full` コマンドを実行します。マップの表示および印刷を行うには、クラスタ・モニタ (`cmon`) を使用します。

ストレージを移行する場合は、すべてのケーブルとストレージにラベルを付けます。必要なハードウェアを入手します。サイト固有のファイルのコピーを作成します。いつ、何をバックアップするかを決定します。アップグレード・オプションについて説明している節の手順を参照します。その後、ユーザのサイトおよび選択したアップグレード方法に適合する詳細な手順を作成します。別個のクラスタを作成する場合でも、手順を参照すると、ストレージの移行に関する概念を得ることができます。

以下の URL に、メンバが 2 つの社内クラスタをオプション 3 でアップグレードする際の、管理者のテキスト・ログが載っています。

http://www.tru64unix.compaq.com/docs/highavail/migration/migration_log.htm

ここでは、クラスタが TruCluster Software Production Server バージョン 1.6 からパッチ・キットを含めた TruCluster Server バージョン 5.0A へアップグレードされています。これはバージョン 5.1B へのアップグレードではありませんが、計画のアプローチ方法と基本的な処理手順は同じです。

- 付録 A のチェックリストを使用して、新しいクラスタのホスト名、ディスク、および IP アドレスを記録します。

8.3.2 ハードウェアとストレージ・トポロジの要件

次のリストに、既存の ASE からのアップグレードに影響を与える可能性がある、最も一般的なハードウェアの要件を示します。これらの要件は、新しいクラスタで現在の ASE ハードウェアのいくつかまたはすべてを使用する、オプション 2 およびオプション 3 のアップグレード・パスに当てはまります。TruCluster Server バージョン 5.1B ハードウェア構成に関する正確な情報については、『クラスタ・ハードウェア構成ガイド』を参照してください。

- (オプション 2) 新しいクラスタを構成する際には、ASE から新しいクラスタにストレージを接続するために、各システムに空いている SCSI アダプタが必要になります (代替方法としては、ASE アダプタをストレージとともに移動します)。
- (オプション 2 およびオプション 3) アップグレード・パスがストレージの物理的な移行を伴う場合、ASE は、シンメトリックな共用ストレージ構成でなければなりません (ASE サービスが使用する各共用デバイスは、同一のスペシャル・ファイル名ですべての ASE メンバに認識されています。たとえば、すべてのメンバで、rz17c は同一の物理デバイスを参照します)。

さらに、Production Server 環境には、ASE が 1 つだけ含まれていなければなりません。

これらが要件となっているのは、8.4 節で説明するスクリプトの自動化のためです。このスクリプトでは、現在の ASE 環境が、一意の AdvFS ドメイン名、LSM ボリューム、および ASE サービスが使用するすべての共用ストレージのデバイス・スペシャル・ファイル名を持っている必要があります。これらのスクリプトにより、ASE サービスが現在使用しているストレージを新しいクラスタへ移動する処理を自動化できます (これらの制約を満足していない ASE のあるサイトでは、付録 F で説明する、手動によるデバイス名マッピングおよびストレージ構成を使用することができます)。

現在の ASE の構成マップを作成していない場合またはマップが古い場合は、先に `cluster_map_create(8)` で示される手順を実行してから、メンバの 1 つで次のコマンドを実行し、`/etc/CCM` ファイルを作成します。

```
# cluster_map_create cluster_name -full
```

クラスタの構成マップを表示するには、クラスタ・モニタ (`cmon`) を使用します。移行スクリプトを使用してアップグレード中にストレージをマップできるかどうかをこの情報から判断します (`cluster_map_create` コマンドと `cmon` コマンドの詳細については、TruCluster Software Products バージョン 1.5 またはバージョン 1.6 の『管理ガイド』を参照してください)。

- (オプション 3) ASE システムは、TruCluster Server バージョン 5.1B でサポートしているシステムでなければなりません。サポートされるハードウェアについては、TruCluster Server バージョン 5.1B の『QuickSpecs』を参照してください。
- (オプション 3) 新しいクラスタで追加のストレージ・ハードウェアが必要な場合は、アップグレードを開始する前にこのハードウェアを ASE に追加することをお勧めします。新しいクラスタには、クラスタ単位のファイル・システム、メンバ・ブート・ディスク、およびオプションのクォーラム・ディスク用の共用ストレージを設置する必要があります。HSZ または HSG タイプのストレージ・デバイスをこの共用バスに設置して、クォーラム・ディスクとメンバ・ブート・パーティションをミラー化することをお勧めします (詳細については、2.5 節で説明している、クラスタ単位のファイル・システムのミラー化に関する注意を参照してください)。

注意

クォーラム・ディスクは ASE のタイブレーカ・ディスクに類似している所がありますが、重要な違いがあります。ASE のタイブレーカ・ディスクは ASE サービスに参加していなければなりません。クォーラム・ディスクには、重要なデータは保存しないようにしてください。

システムのプライベート・バスに別個のディスクを用意し、Tru64 UNIX バージョン 5.1B をそのディスクにインストールすることをお勧めします。ASE で使用されるオペレーティング・システムを格納しているディスクにバージョン 5.1B の Tru64 UNIX オペレーティング・システムをインストールするのは、できるだけ避けてください。ASE に戻す場合、以前のオペレーティング・システムを再インストールしてこのシステムに ASE 環境を再作成するよりも、ASE オペレーティング・システムのディスクをブートする方が簡単です。

注意

TruCluster Server では、0 ~ 15 の SCSI ID をサポートしています。アップグレードを開始する前に利用可能な SCSI ID がなかった場合は、Tru64 UNIX オペレーティング・システムをブートし、既存のデバイス名を新しいデバイス名にマップした後にストレージを追加できます。クラスタを作成する前に、ストレージをすべて Tru64 UNIX に接続して、認識された状態にしなければなりません。

Tru64 UNIX バージョン 5.1B および TruCluster Server バージョン 5.1B のメンバ・ブート・ディスクを LUN 0 に設置する必要はありません。

- (オプション 3) 新しいクラスタで LSM を使用する場合は、2.5.2 項を参照してください。共用ドライブ上に rootdg ディスク・グループ用として少なくとも 1 つの利用可能なパーティションが必要となります (冗長構成にするには複数のパーティションが必要です)。クラスタ単位のルート (/) ファイル・システムは、サイズ上の理由から通常はデバイスの b パーティションに配置しますが、a パーティションを使用することもできます。

ハードウェア RAID を用いてクラスタのファイル・システムをミラー化する場合、アップグレードは、RAID ハードウェアを追加してマルチパス化を行うよい機会となります。TruCluster Server バージョン 5.1B クラスタでのストレージ構成については、『クラスタ・ハードウェア構成ガイド』を参照してください。

8.3.3 ファイル・システムの要件

TruCluster Server バージョン 5.1B クラスタは、AdvFS ファイル・システムを使用します。UFS はクラスタ単位の読み取り専用アクセスだけがサポートされています。クラスタ・メンバは UFS ファイル・システムを読み取り/書き込みでマウントできます。ただし、そのファイル・システムにアクセスできるのは、そのメンバに限られます。

注意

アップグレード時のファイル・システムの移行を簡単に行うために、TruCluster Server バージョン 5.1B では、UFS ファイル・システムの読み取り/書き込みがサポートされています。バージョン 5.1B クラスタに UFS ファイル・システムを読み取り/書き込みアクセスでマウントする場合は、特に指定しなくても `mount` コマンドに `-o server_only` 引数を使用されます。これらのファイル・システムはパーティション化されたファイル・システムとして扱われます。つまり、そのファイル・システムは、それをマウントしたメンバしかアクセスできません。他のクラスタ・メンバはこのファイル・システムで読み取りも書き込みも実行できません。リモート・アクセスもフェイルオーバーもできません。UFS ファイル・システムを、すべてのクラスタ・メンバから読み取り専用でアクセスできるようにマウントする場合は、読み取り専用と明示してマウントしなければなりません。

ファイル・システムをマウントする際に `-o server_only` 引数を明示して指定すると、AdvFS ファイル・システムをパーティション化されたファイル・システムとしてマウントできます。

ファイル・システムのパーティション設定については、『クラスタ管理ガイド』と `mount(8)` に記載されています。

新しいクラスタで読み取り/書き込みアクセスを行う予定のデータが現在の ASE で UFS ファイル・システムを通して読み取り/書き込みが行われている場合は、アップグレードを開始する前に、これらのファイル・システムを AdvFS に移行することをお勧めします (新しいクラスタの 1 メンバのみに UFS ファイル・システムへの読み取り/書き込みアクセスを制限できる場合は、ファイル・システムをパーティション化できます)。データを移行する場合、その方法は、現在の構成において、新しいストレージに AdvFS ドメインを作成するための十分なストレージがあるかどうかや、現在のストレージを再使用するために、バックアップしてリストアする必要があるかどうかによって異なります。移行ストラテジのひとつとして、UFS の読み取り/書き込みに対して TruCluster Server のローカル・サポートを使用するという方法もあります。

注意

アップグレード・パスがオプション 1 (新しいシステムと新しいストレージ) の場合には、ASE で使用されているファイル・システムを変換したり、変更したりする必要はありません。ただし、ファイル・システムのデータ移行ストラテジとしては、全メンバに読み取りと書き込みの両方ができるファイル・システムを提供するために、新しいクラスタでは AdvFS ファイル・システムを採用する必要があります。

バージョン 5.0 以降のオペレーティング・システムでは、ドメイン・バージョン番号 4 (DVN4) というディスク構造で AdvFS ドメインを作成します。DVN4 では、2 T バイト (TB) を超えるクォータ値が可能で、ディレクトリに何千ものファイルが置かれている場合の性能が向上します。バージョン 5.0 より以前に作成されたドメインは DVN3 を使用しています。これらのドメインは新しいバージョンでも認識されますが、新しいディスク構造へ自動的にアップグレードされるわけではありません。

TruCluster Server へアップグレードする作業が完了すれば、AdvFS DVN3 ドメインを DVN4 に変換できます。変換するかしないか、また、いつ変換するかは、ユーザの都合によります。Tru64 UNIX バージョン 5.1B は DVN3 と DVN4 の両方のフォーマットを認識します (`mkfdmn -V3` オプションを使用すると、新しいクラスタ上に古いスタイルの AdvFS ドメインを作成することができます)。

現在のストレージを再使用する場合、既存の ASE での UFS から AdvFS へのファイル・システム変換は、都合の良いときに行うことができますが、ファイル・システムのフォーマットは DVN3 です。現在の ASE で UFS から AdvFS へ変換しない場合は、アップグレードの一環として変換できますが、ダウンしている時間が長くなります。また、アップグレード中に問題が発生しても、ASE に戻すことは困難です。

フォーマットを変換する際、ファイル・システム上で `vdump/vrestore` ユーティリティを実行するのに時間がかかります。新しいクラスタ上で新しい AdvFS フォーマットを活用したい場合は、次のような UFS の変換方法があります。

- 移行前に UFS ファイル・システムを AdvFS DVN3 ドメインへ変換してから、移行後に AdvFS DVN4 へ変換する。変換を 2 回実行することになるが、クラスタに移行すると直ちにデータの読み取り/書き込みが可能になる (ASE で UFS を AdvFS に変換し、新しいクラスタでしばらくの間継続して AdvFS DVN3 フォーマットを使用した後、DVN4 の機能を利用する必要が生じたときにドメインを DVN4 に変換することもできる)。
- 移行後に AdvFS へ変換する。変換は 1 回だけであるが、変換を行うまでは、読み取り/書き込みがローカル使用に限定され、クラスタ単位では読み取りしかできない (UFS ファイル・システムを新クラスタの AdvFS ドメインに移行する際は、ローカル使用限定の UFS 読み取り/書き込み機能を使用して制御することができる)。

8.4 オプション 2 およびオプション 3 で使用するアップグレード・スクリプト

次に、オプション 2 およびオプション 3 のアップグレード手順で使用する `TCRMIGRATE540` サブセット内のスクリプトを示します。スクリプトおよび関連するユーティリティ・プログラムは、「Tru64 UNIX Associated Products Volume 2」CD-ROM の TruCluster Server バージョン 5.1B ディレクトリに用意されており、`TCRMIGRATE540` サブセット内にあります。ASE の各メンバにサブセットをロードするには、`setld` コマンドを使用します。ユーティリティおよび関連ライブラリは、`/usr/opt/TruCluster/tools/migrate` ディレクトリにインストールされます。

スクリプトは Korn シェル (`ksh`) スクリプトです。シェル・プログラミングの知識があり、スクリプトで何をしているのかを正確に知りたい場合は、実

行する前にスクリプトのコードを調べることができます (付録 F で、ストレージ移行スクリプトに代わる方法として、デバイス名のマッピングとストレージの構成を手動で行う手順を説明しています)。

`clu_migrate_check`

(オプション 3) ハードウェア、ファームウェア、およびファイル・システムの一般的なタイプ・チェックを行います。

オプション 3 のアップグレードを開始する前に、現在の ASE の各メンバで `clu_migrate_check` を実行します。

`clu_migrate_save`

(オプション 2 およびオプション 3) ディレクトリを作成し、現在のシステム、ASE 構成、および ASE が使用する共用ストレージに関する情報を保存します。

新しいクラスタを作成した後、すべての共用ストレージがまだ ASE に接続されている間に、オンラインの ASE サービスが提供されている ASE の各メンバで `clu_migrate_save` を実行します。
`clu_migrate_save` スクリプトにより、ASE サービスが現在使用しているストレージを新しいクラスタへ移行するために必要な情報がすべて収集されます (これには、このストレージに関連する AdvFS ドメインまたは LSM ボリュームの移行も含まれます)。スクリプトが現在の構成に対して行う唯一の変更は、各ディスクの `rz*` スペシャル・ファイル名をディスクのラベルの `label:` フィールドに書き込むことです。ただし、スクリプトは元のディスク・ラベルを保存します。このラベルは、`clu_migrate_configure` を実行する際にリストアすることができます。`rz*` 名を `label:` フィールドに書き込むと、`clu_migrate_configure` により各ディスク・デバイスをその新しいスタイルの `dsk*` デバイス名にマップできるようになります。

`clu_migrate_save` スクリプトでは、データを新しいクラスタ (オプション 2) か Tru64 UNIX バージョン 5.1B システム (オプション 3) のどちらかに自動的にコピーすることを選択できます。スクリプトは `/var/TruCluster_migration` ディレクトリにある情報を、ターゲット・システム上に各メンバ独自のディレクトリを割り当てて、保存します。このとき、各メンバの `/etc/rc.config` にある `CLUSTER_NET` 変数を使用して、次のように命名します。

`/var/TruCluster_migration/CLUSTER_NET`

次に、clu_migrate_configure スクリプトによりこれらのディレクトリの情報が使用されて、物理ストレージ・デバイス名がマップされるとともに、新しいクラスタ上または Tru64 UNIX システム上のストレージが構成されます。

clu_migrate_configure

(オプション 2 およびオプション 3) 新しい TruCluster Server バージョン 5.1B クラスタ (オプション 2) または Tru64 UNIX バージョン 5.1B システム (オプション 3) 上のストレージを構成します。

ASE システムの電源をオフにして共用ストレージを新しいクラスタに接続した後、clu_migrate_configure を実行すると、以前 ASE で管理していたストレージが自動的に構成されます。clu_migrate_configure スクリプトにより、ASE メンバ上での clu_migrate_save による出力から収集された情報がマージされます。次にストレージの構成、つまり、各ディスクの label: フィールドに書き込まれた古いスタイルのデバイス名から新しいスタイルの dsk デバイス名へのマップ、LSM ボリュームのインポート、AdvFS ドメインの再作成、マウント・ポイントのテスト、および /etc/fstab および /etc/exports へのエントリの追加が行われます。ストレージの構成が完了すると、このスクリプトにより、clu_migrate_save によって上書きされた、保存されている label: フィールド値がすべて復元されます。

clu_migrate_configure スクリプトでは、すべての共用ディスク・デバイスについてデバイス名マッピングが行えますが、ASE で管理していないストレージは構成されません。共用ストレージが ASE で管理されていなかった場合、ASE メンバの /etc/fstab ファイルにあるそのストレージのエントリは移行されません。

このスクリプトでは、既存の AdvFS ファイル・システムは新しい AdvFS フォーマットに変換されません。新しい AdvFS フォーマットを使用するには、アップグレードを完了してから AdvFS ファイル・システムを変換します。

clu_migrate_configure スクリプトの実行中に、vollogcnvt を実行するように指示するメッセージが表示されることがあります。これは、ブロック変更ロギング (BCL) を使用するボリュームを検出したため、ダーティ・リージョン・ロギング (DRL) に変換する必要がある

ことを示します。変換には `vollogcnvt` ユーティリティを使用します。詳細については、`vollogcnvt(8)` および Tru64 UNIX 『*Logical Storage Manager*』を参照してください。

`clu_migrate_recover`

(オプション 2 およびオプション 3) デバイスの予約を解除して、ASE メンバの LSM 構成を復元します。正常に行われているアップグレードでは、このスクリプトは実行しません。

`clu_migrate_recover` スクリプトは、ASE システムで実行され、8.7.2 項に示す回復手順の一部として ASE に戻します。このスクリプトは、アップグレードが正常に完了できなかった場合のみ実行します。

注意

アップグレード中に AdvFS ファイル・システムを新しい AdvFS フォーマットに変換した場合は、`clu_migrate_recover` スクリプトでそれらを古いフォーマットに戻すことはできません。これは、手動で行う必要があります。

8.5 オプション 1: 別のクラスタの作成 — 新しいシステムと新しいストレージ

別のクラスタを作成すると、現在の運用環境に影響を与えることなくアプリケーションとシステム構成をテストできます。

別のクラスタを設定できるかどうかは、現在のハードウェア構成、および新しいハードウェア購入の可否によって異なります。新しいハードウェアを購入でき、別のクラスタを構成するスペースがある場合には、この方法をお勧めします。

次に、この方法の一般的な手順について概略を説明します。

1. 『クラスタ・ハードウェア構成ガイド』の情報を参照して、新しいクラスタのハードウェア構成を設計します。次の点について考慮する必要があります。

- NSPOF (no-single-point-of-failure) クラスタを作成する (NSPOF クラスタを作成するための基本的なハードウェアの必要条件については『クラスタ・ハードウェア構成ガイド』を参照)。
 - 高速の RAID コントローラとマルチパス化を使用して、クォーラム・ディスクまたはメンバ・ブート・パーティションのミラーリングを実行する (TruCluster Server は、これらのファイル・システムに対して LSM ミラーリングをサポートしていない)。
 - 拡張の余地を残しておく。たとえば、構成時にまだ空き PCI (Peripheral Component Interconnect) スロットがあるシステムを注文しておき、拡張可能なネットワークおよびストレージ・トポロジを作成する。
2. ハードウェアがサイトに届いたら、すべてのファームウェアとコンソールの構成を完了させ、物理的に接続したのち、本書の情報を参照して TruCluster Server クラスタを作成します。

注意

新しいクラスタ用に IP アドレスを選択する際には、現在 ASE で使用されているアドレスを使用してはなりません (新しいクラスタが動作している場合、クライアントでは、すべての NFS マウント要求を省略時のクラスタ別名か、または `/etc/exports.alias` の中に名前が指定されている別名に宛てて送信する必要があります)。

3. 新しいクラスタの機能を習得してください。
- 『クラスタ管理ガイド』の情報を参照して、クラスタを構成および管理します。たとえば、データのバックアップおよびリストア・プロシージャの作成とテスト、ハードウェアの監視、クラスタの調整および負荷分散などを行います。
- 『クラスタ高可用性アプリケーション・ガイド』の情報を参照して、高可用性アプリケーションを作成してテストします。使用する予定の重要な各アプリケーションについて、どちらの方法がそのアプリケーションに適しているのかを判断します。アプリケーションとフェイルオーバー・ポリシを構成したら、強制的にフェイルオーバーさせてみて、

期待どおりの結果が得られるかどうかを確認します。満足できるまで、変更と調整を繰り返します。

4. データの移行: 新しいクラスタは、現在の ASE とは完全に独立しているため、主な作業は、ASE から新しいクラスタへのアプリケーション・データの移行です。

注意

データの移行を開始する前に、ASE と新しいクラスタの両方のバックアップをとっておくことをお勧めします。

ASE で使用されているデータを新しいクラスタにいつ、どのように移動するかは、ユーザが決定しなければなりません。ここでは、推奨する手順やツールは示しません。次に、データ移行の方針を決める際に考慮する必要がある重要なポイントについて説明します。

- 標準の `vdump/vrestore` ユーティリティかアプリケーション固有のデータ移行ツールを使用するかは、現在の環境、アプリケーション、およびデータの量によって決まります。また、アプリケーション・データのいくつかまたはすべてが UFS フォーマットで保存されている場合は、新しいクラスタ上の AdvFS フォーマットにリストアする必要があります (UFS から AdvFS への変換についての詳細は 8.3.3 項を参照)。

データベース・ベンダは通常、データベースのリモート・コピーをアップデートできるアプリケーションを提供しています。リモート・アップデート機能のあるデータベース・アプリケーションを使用している場合は、その機能を使用して、ASE からクラスタへデータを移行することができます。アップデート (および新しいクラスタでのテスト) が完了すると、データベース・サービス用の IP アドレスを新しいクラスタに移動します。

- 現在の ASE で、複数のアプリケーションのサービスを行っている場合は、すべてのアプリケーション・データを同時に移行するのか、あるいは、ASE と新しいクラスタの両方からクライアントに対してサービスを行いながら、一度に 1 つのアプリケーションを移行するのかを決定しなければなりません。アプリケーションを一度に移行する場合には、それらを移行する順番も決定します。

- 移行の途中で、ASE とクラスタの両方がクライアントに対してサービスを行っている場合は、移行期間中にバックアップとリカバリを行う手順を考えます。

8.6 オプション 2：別のクラスタの作成 — 新しいシステムと既存ストレージの使用

この方法はオプション 1 に似ていますが、ある時点で、既存の ASE ストレージ・デバイスを新しいクラスタに物理的に移動する点が異なります。この節では、次の事項について説明します。

- アップグレード手順 (8.6.1 項)
- 部分アップグレードの取り消し手順 (8.6.2 項)

8.6.1 オプション 2：アップグレード手順

次に、オプション 2 のための一般的な手順について概略を説明します。

1. 『クラスタ・ハードウェア構成ガイド』の情報を参照して、新しいクラスタのハードウェア構成を設計します。その際、次の点について考慮する必要があります。
 - NSPOF (no-single-point-of-failure) クラスタを作成する (NSPOF クラスタを作成するための基本的なハードウェアの必要条件については『クラスタ・ハードウェア構成ガイド』を参照)。ASE からのストレージ・ハードウェアを使用しているため、その構成での NSPOF 制約事項が新しいクラスタに影響する。たとえば、ディスクの中にはマルチパス化をサポートしていないものもある。
 - 新しいクラスタには、クラスタの作成およびメンバの追加に必要な共用ストレージがなければならない。さらに、ASE ストレージを新しいクラスタに物理的に移動する前に行うアプリケーション・テストのために、追加の共用ストレージも必要。
 - 新しいクラスタでは、アプリケーション・データ用に現在の ASE からのストレージ・ハードウェアを使用するため、新しいクラスタ・システムに、ASE のストレージ・トポロジと互換性のあるストレージ・アダプタがあることを確認する。ASE システムから新しいシステムにアダプタを移す場合は、そのアダプタが新しいシステムでサポートされていることを確認する。

- 高速の RAID コントローラおよびマルチパス化を使用して、クォーラム・ディスクまたはメンバ・ブート・パーティションをミラーリングする (TruCluster Server では、これらのファイルに対して LSM ミラーリングをサポートしていない)。
 - 拡張の余地を残しておく。たとえば、構成時にまだ空き PCI スロットがあるシステムを注文しておき、拡張可能なネットワークおよびストレージ・トポロジを作成する。
2. ハードウェアがサイトに届いたら、すべてのファームウェアとコンソール構成を完了させ、物理的に接続したのち、本書の情報を参照して TruCluster Server クラスタを作成します。

注意

新しいクラスタ用に IP アドレスを選択する際には、現在 ASE で使用されているアドレスを使用してはなりません。

3. TruCluster Server クラスタの機能を習得してください。

『クラスタ管理ガイド』の情報を参照して、クラスタを構成および管理します。たとえば、データのバックアップおよびリストア・プロセスの作成およびテスト、ハードウェアの監視、クラスタの調整および負荷分散などを行います。

『クラスタ高可用性アプリケーション・ガイド』の情報を参照して、高可用性アプリケーションを作成してテストします。たとえば、2 つのサブシステム間の相違が理解できるまで、CAA とクラスタ別名を使用します。アプリケーションとフェイルオーバー・ポリシーを構成したら、強制的にフェイルオーバーさせてみて、期待どおりの結果が得られるかどうかを確認します。満足できるまで、変更と調整を繰り返します。
4. この時点まで、ASE と TruCluster Server クラスタは完全に独立しているため、一方に対して行ったことがもう一方に影響することはありません。次の手順からは、ストレージを ASE から新しいクラスタに移動する準備を開始します。作業を続行する前に、次のことを行ってください。
 - a. 8.4 節のユーティリティ・スクリプトに関する説明と、8.7.1 項のオプション 3 の手順を読んでおいてください。ASE からクラスタにストレージを移動する際に、そのスクリプトと手順の一部を使用します。

- b. 新しいクラスタの `.rhosts` を編集して、ASE の各メンバから root アクセスできるようにします。これにより、`clu_migrate_save` が、ASE メンバからクラスタ上の `/var/TruCluster_migration` ディレクトリに、情報を自動的にコピーできるようになります。
 - c. 新しいクラスタのバックアップをとります。何か問題が起こった場合にも、リカバリが可能です。
5. ASE の各メンバで、TCRMIGRATE540 サブセットをロードします。このサブセットは、「Tru64 UNIX Associated Products Volume 2」CD-ROM の TruCluster Server バージョン 5.1B ディレクトリに入っています。次の例では、CD-ROM を `/mnt` にマウントしていると想定しています。
- ```
setld -l /mnt/TruCluster/kit TCRMIGRATE540
```
- 移行スクリプト、ユーティリティ・プログラム、およびライブラリが `/usr/opt/TruCluster/tools/migrate` ディレクトリにインストールされます。
6. すべてのストレージのケーブルとアダプタにラベルを付けます。これは、ストレージを ASE に再接続する必要が生じた場合のためです。
7. すべての LSM ベースのサービスが ASE でオンラインになっていることを確認します。
8. ASE サービスを実行する ASE の各メンバで、`clu_migrate_save` を実行します。
- ```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_save
```
9. TruCluster Server バージョン 5.1B クラスタで次の処理を行います。
- a. `clu_migrate_save` で作成されたファイルが `/var/TruCluster_migration` ディレクトリにコピーされていることを確認します。
 - b. クラスタをシャットダウンして、停止します。

```
# shutdown -c now
```
 - c. クラスタの各メンバの電源をオフにします。

注意

ストレージの電源をオフにするかしないかは、ユーザの判断によります。

10. ASE の全メンバで次の処理を行います。

- a. 通常のバックアップ手順に従い、すべてのデータのバックアップを取ります。次の手順からは、ASE からクラスタにストレージ・デバイスを移行します。何らかの理由で ASE に戻す必要がある場合、これが現在の構成のフル・バックアップを行う最後の機会です。
- b. ASE サービスをすべてオフラインにします。データのバックアップ時にオフラインにする必要がない場合は、この時点でオフラインにします。
- c. 各システムをシャットダウンして、停止します。
- d. 各システムの電源をオフにします。

11. TruCluster Server クラスタで次の処理を行います。

- a. 共用ストレージを ASE からクラスタ・システムに接続します。

注意

TruCluster Server バージョン 5.1B では、ストレージがクラスタ・メンバにシンメトリックに接続されていなくても利用することができます。ただし、ASE のストレージには、高可用性のアプリケーション・データが含まれているため、ストレージはすべてのメンバから直接アクセスできるように接続することをお勧めします。

- b. 共用ストレージの電源がオフになっている場合は、その電源をオンにします。
- c. クラスタ・メンバの電源を投入します。
- d. 各コンソールで `show dev` コマンドを実行し、ASE が使用するディスク・デバイスが認識されていることを可能な限り確認しま

す。クラスタをブートする前に、ハードウェア関連の問題を発見して、修正してください。

- e. クラスタをブートします。新しいストレージがシンメトリックに構成されていない場合は、新しいストレージに直接接続されているすべてのメンバを確実にブートしてください。

注意

ブート・フェーズでは、クラスタ・メンバは新しいデバイスを検出して、スペシャル・デバイス・ファイルを作成します。

- 12. クラスタの1つのメンバで `clu_migrate_configure` を実行します。このコマンドにより、ストレージ・デバイスがオペレーティング・システムに認識されているかどうかを確認します。また、このコマンドでは古いスタイルのデバイス名が新しいスタイルのデバイス名にマップされて、ストレージが構成されます。
- 13. クラスタで、次の作業を行います。
 - a. アプリケーションを、完全なストレージでテストします。アプリケーションがデータを認識できるかどうか、およびデータを使用できるかどうかを確認します。以前のアプリケーション・テストでは、アプリケーションが問題なく実行できることを確認しました。完全なストレージで行うこのテストの目的は、クライアントにサービスを提供するときに使用するデータを、アプリケーションが認識して操作できるかどうかを確認することです。

注意

解決できない問題を発見し、ASE に戻すことを決定した場合は、8.6.2 項の手順に従ってください。

- b. クライアントへのサービスを開始します。NFS クライアントでは、NFS マウント要求を、省略時のクラスタ別名、または `/etc/exports.aliases` の中に名前が指定されている別名に宛てて送信する必要があります。

14. (オプション作業) 移行用のディレクトリおよび移行サブセットを削除します。クラスタの 1 つのメンバで次の作業を行います。

- a. 移行用のディレクトリを削除します。

```
# rm -rf /var/TruCluster_migration
```

- b. 移行用のツールを削除します。

```
# setld -d TCRMIGRATE540
```

8.6.2 オプション 2: 部分アップグレードの取り消し

ストレージの移行時またはアップグレードの完了後に、解決できない問題が発生した場合は、次の手順に従って ASE 構成に戻します。

1. クラスタを停止します。
2. クラスタの各メンバの電源をオフにします。

注意

ストレージの電源をオフにするかどうかは、ユーザの判断です。

3. ストレージを、ASE に再接続します。以前とまったく同様に接続されていることを確認してください。
4. 共用ストレージの電源がオフになっている場合は、その電源をオンにします。
5. すべての ASE システムの電源をオンにします。
6. ASE メンバをブートします。
7. `clu_migrate_save` を実行した ASE の各メンバ上で `clu_migrate_recover` コマンドを実行します。

```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_recover
```
8. すべてのメンバが動作し、回復したときに、いずれかのメンバ上で `asemgr` コマンドを実行して、すべてのサービスをオンラインに設定します。

9. (オプション作業) 移行ディレクトリおよび移行サブセットを削除します。ASE の各メンバで次の作業を行います。

- a. 移行ディレクトリを削除します。

```
# rm -rf /var/TruCluster_migration
```

- b. 移行ツールを削除します。

```
# setld -d TCRMIGRATE540
```

8.7 オプション 3: 既存のシステムとストレージの使用

この節では、既存のハードウェアを引き続き使用し、既存のバージョン 1.5 またはバージョン 1.6 の TruCluster Production Server Software クラスタあるいは TruCluster Available Server Software (ASE) を TruCluster Server バージョン 5.1B にアップグレードする方法を説明します。

- アップグレード手順 (8.7.1 項)
- 部分アップグレードの取り消し手順 (8.7.2 項)

8.7.1 オプション 3: アップグレード手順

次の手順では、8.4 節で説明したスクリプトを使用して、いくつかの移行作業を自動化しています。スクリプトを使用する場合、現在の ASE 構成は、8.3.2 項に示すハードウェアおよびストレージのトポロジ要件を満たしていなければなりません。

次のようなアップグレード方法があります。

- 自動ストレージ構成: 現在のシステム構成の確認、情報の保存、および新しいクラスタでのストレージの自動構成を、すべてスクリプトを使用して行う。
- 手動ストレージ構成: スクリプトを使用して現在のシステム構成を確認し、情報を保存するが、新しいクラスタ上では `clu_migrate_configure -x` を実行する。`-x` オプションにより、すべての構成情報が表示されるが、ストレージを構成する代わりに、`clu_migrate_configure` が通常使用する構成コマンドの一覧がリストされるだけである。その後、コマンドの順序を調べ、ユーザ独自の一連のコマンドを作成して手動で実行するか、あるいは、`-x` オプションを指定しないで `clu_migrate_configure` を再実行することができる。

どちらの方法でも、ほとんどの手順は同じです。新しいクラスタでストレージを手動で構成するユーザだけの手順には、「ストレージ手動構成のみ」と記しています。clu_migrate_configure を使用して新しいクラスタでストレージを構成するユーザだけの手順には、「ストレージ自動構成のみ」と記しています。

使用している環境にどちらの方法が適しているかを判断する前に、手順全体を参照することをお勧めします。

1. ASE の各メンバで、TCRMIGRATE540 サブセットをロードします。このサブセットは「Tru64 UNIX Associated Products Volume 2」CD-ROM の TruCluster Server バージョン 5.1B ディレクトリにあります。次の例では、CD-ROM を /mnt にマウントしていると想定しています。

```
# setld -l /mnt/TruCluster/kit TCRMIGRATE540
```

移行スクリプト、ユーティリティ・プログラム、およびライブラリが /usr/opt/TruCluster/tools/migrate ディレクトリにインストールされます。

2. 現在の ASE の各メンバで、clu_migrate_check を実行します。

```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_check
```

出力を 8.3.2 項に示す必要条件と照合して、各システムのハードウェアとファームウェアが、アップグレードできるよう適切に構成されていることを確認します。

3. すべてのストレージのケーブルとアダプタにラベルを付けます。
4. 新しいクラスタの第 1 メンバにする ASE メンバ・システムを決定します。
5. ASE の残りのメンバに ASE サービスを手動で再配置します。
6. 新しいクラスタの第 1 メンバになるシステムを ASE から削除します。
7. 新しいクラスタの第 1 メンバになるシステムで、次の作業を行います。
 - a. システムをシャットダウンし、停止します。
 - b. システム・コンソールで次の作業を行います。
 - i. show dev コマンドを実行し、すべての共用ストレージに関する出力を記録します。
 - ii. すべてのコンソール変数と値を記録します。

iii. 次のコンソール変数を設定します。

```
>>> set auto_action halt
>>> set bootdef_dev ""
>>> set boot_reset on
```

bus_probe_algorithm 変数をサポートするシステムについては、次のように設定します。

```
>>> set bus_probe_algorithm new
```

コンソール変数についての詳細は、2.7 節を参照してください。bus_probe_algorithm 変数を使用しないシステムでこの変数を設定しても障害にはなりません。この変数は、次の init または電源をオフにしたのちにオンにしたときに、クリアされます。

- c. システムの電源をオフにします。
- d. すべてのケーブルと接続にラベルが付けられていることを確認した後で、システムからすべての共用ストレージのケーブル (SCSI や Fibre Channel など) を外し必要に応じてアダプタをターミネートします。既存のケーブルが Y ケーブルまたはトライリンク・アダプタでターミネートされている場合は、ターミネータを追加する必要はありません。SCSI アダプタのターミネートについては、『クラスタ・ハードウェア構成ガイド』を参照してください。
- e. システムに 1 つ以上の Memory Channel クラスタ・インターコネクト・アダプタを取り付けている場合は、ケーブルを外します。
- f. クラスタ・インターコネクトを変更するには、『クラスタ・ハードウェア構成ガイド』の手順に従って作業します。この時点では、アダプタをケーブルに接続しないでください。
- g. システムの電源をオンにします。
- h. コンソールで show config コマンドを使用して、コンソールおよびアダプタのファームウェア・リビジョン番号をチェックし、Tru64 UNIX バージョン 5.1B と互換性があるかどうかを確認します。互換性がない場合には、必要に応じてファームウェアをアップデートします。
- i. 第 3 章の指示に従って、Tru64 UNIX バージョン 5.1B のフル・インストールを行います。

注意

ASE で使用するオペレーティング・システムを格納したディスクは、決して重ね書きしないでください。後で問題が発生した場合にも、これらのディスクの内容が損なわれていなければ、このシステムを直ちに ASE に戻すことができます。

ディスクを上書きする必要がある場合は、Tru64 UNIX バージョン 5.1B をインストールする前に現在のオペレーティング・システムのバックアップを取っておいてください。

8. Tru64 UNIX バージョン 5.1B システムで、次の作業を行います。
 - a. Tru64 UNIX オペレーティング・システムを完全に構成します。第 3 章の指示に従ってください。
 - b. アプリケーションをインストールします。
 - c. Tru64 UNIX バージョン 5.1B システムで `/.rhosts` を編集して、ASE の残りのメンバから `root` アクセスできるようにします。こうすることにより、残りの ASE メンバからの情報を `clu_migrate_save` で自動的にコピーできます。
 - d. TruCluster Server バージョン 5.1B ライセンスとサブセットをインストールします。

注意

作業を続行する前に、バージョン 5.1B のシステムのバックアップを取っておくことをお勧めします。この後の手順で問題が起こった場合には、Tru64 UNIX をインストールおよび構成してからアプリケーションをインストールし、TruCluster Server サブセットをロードするよりも、速くこの時点まで復元することができます。

9. すべての LSM ベースのサービスが ASE でオンラインになっていることを確認します。

10. ASE サービスを実行する ASE の各メンバで、clu_migrate_save を実行します。

```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_save
```

11. Tru64 UNIX システムで、次の作業を行います。

- a. clu_migrate_save で作成されたファイルが、
/var/TruCluster_migration ディレクトリにコピーされていることを確認します。
- b. システムを停止して、電源をオフにします。

12. ASE のすべてのメンバで、次の作業を行います。

- a. 通常のバックアップ手順に従い、すべてのデータのバックアップを取ります。次の手順からは、ASE から Tru64 UNIX バージョン 5.1B システムにストレージ・デバイスを移動します。何らかの理由により ASE に戻す必要がある場合、これが現在の構成のフル・バックアップを行う最後の機会です。
- b. ASE サービスをすべてオフラインにします。データのバックアップ時にオフラインにする必要がなかった場合は、この時点でオフラインにします。
- c. 各システムをシャットダウンして、停止します。
- d. 各システムのコンソールで、次のコンソール変数を設定します。

```
>>> set auto_action halt
>>> set bootdef_dev ""
>>> set boot_osflags A
>>> set boot_reset on
```

bus_probe_algorithm 変数をサポートするシステムでは、次のように設定します。

```
>>> set bus_probe_algorithm new
```

コンソール変数についての詳細は、2.7 節を参照してください。
bus_probe_algorithm 変数を使用しないシステムでこの変数を設定しても障害にはなりません。この変数は、次の init または電源をオフにしたのちにオンにしたときに、クリアされます。

- e. 各システムの電源をオフにします。

注意

指示があるまで、これらのシステムの電源はオンにしないでください。

13. Tru64 UNIX バージョン 5.1B システムで、次の作業を行います。
 - a. すべての共用ストレージをシステムに接続します。ストレージのケーブルは、システムが ASE の一部であったときとまったく同様に接続してください。
 - b. クラスタ・インターコネクト・ケーブルをシステムに接続します。クラスタ構成が Memory Channel ハブか、または、イーサネット・ハブあるいはスイッチを使用する場合、ケーブルをハブあるいはスイッチに接続します。ハブもスイッチも使用しない場合は、ケーブルのみを接続します。
 - c. システムの電源をオンにします。
 - d. コンソールで `show dev` コマンドを実行し、すべての共用デバイスが認識されていることを可能な限り確認します。表示されたデバイス・リストと、ストレージを分離する前に保存した情報とを比較します。システムをブートする前に、ハードウェア関連の問題を発見して、修正してください。

また、保存されているその他のコンソール情報も、現在のコンソールのものと比較します。
 - e. システムをマルチユーザ・モードでブートして、ログインします。
14. ストレージ自動構成のみ：Tru64 UNIX バージョン 5.1B システムで `clu_migrate_configure` を実行します。このコマンドにより、ストレージ・デバイスがオペレーティング・システムに認識されているかどうかを確認します。また、このコマンドでは古いスタイルのデバイス名が新しいスタイルのデバイス名にマップされ、ストレージが設定されます。


```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_configure
```
15. ストレージ手動構成のみ：Tru64 UNIX バージョン 5.1B システムで `clu_migrate_configure -x` を実行します。


```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_configure -x
```

-x オプションにより、すべての構成情報が表示されますが、ストレージを構成する代わりに、clu_migrate_configure が通常使用する構成コマンドの一覧がリストされるだけです。その後、コマンドの順序を調べ、ユーザ独自の一連のコマンドを作成して手動で実行するか、あるいは、-x オプションを指定しないで clu_migrate_configure を再実行することができます。

表示された一連のコマンドを調べたのち、ストレージを自動的に構成することを決定した場合は、clu_migrate_configure を実行します。

```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_configure
```

表示された一連のコマンドを調べたのち、ストレージを手動で構成することを決定した場合は、それを今実行します (ストレージを手動で構成する方法については、F.2 節で説明しています)。作業が終了したら、次のコマンドを使用して、ストレージの構成を調べ、元のディスク・ラベルを復元 (オプション作業) します。

- a. 次の LSM コマンドを実行して、LSM の構成を表示します。

```
# voldisk list
# volprint -thA
```

- b. 各 AdvFS ドメインについて、showfdmn domain コマンドを実行します。
- c. 各 AdvFS ドメインについて、showfsets domain コマンドを実行し、ドメインのファイル・セットが適切であることを確認します。
- d. (オプション作業) clu_migrate_save は、実行した各 ASE メンバに対して、/var/TruCluster_migration/CLUSTER_NET/Packids ファイルを作成しています。Packids ファイルには、そのメンバで認識されている共用デバイスの元のディスク・ラベルが含まれています。元のディスク・ラベルの packid (label:) フィールドに値が含まれていた場合は、元のラベルを復元できます。元のラベルを復元するには、restore_packids スクリプトを使用します。ディレクトリを /usr/opt/TruCluster/tools/migrate/utils に変更して、次のコマンドを実行します。

```
# ./restore_packids -f \
/var/TruCluster_migration/CLUSTER_NET/Packids
```

`CLUSTER_NET` は、各メンバの値に置き換えてください。たとえば、`CLUSTER_NET` 値が `mcclu14` である ASE メンバの場合、コマンドは次のようになります。

```
# ./restore_packids -f \  
/var/TruCluster_migration/mcclu14/Packids
```

16. Tru64 UNIX バージョン 5.1B システムで、次の作業を行います。
 - a. 第 4 章の手順に従い、`clu_create` コマンドを実行してシングル・メンバ・クラスタを作成します。
 - b. システムを停止し、それをシングル・メンバ・クラスタとしてブートします。
17. シングル・メンバの TruCluster Server バージョン 5.1B クラスタで次の作業を行います。
 - a. アプリケーション用の CAA プロファイルおよびスクリプトを設定します。アプリケーション・プロファイルの作成と CAA の使用については、『クラスタ高可用性アプリケーション・ガイド』、『クラスタ管理ガイド』、および `caa_profile(8)` を参照してください。
 - b. 省略時のクラスタ別名以外のクラスタ別名を使用する場合は、それらのクラスタ別名を指定し、加わります。クラスタ別名の構成については、『クラスタ管理ガイド』および `cluamgr(8)` を参照してください。
 - c. アプリケーションを、完全なストレージでテストします。アプリケーションがデータを認識できるかどうか、およびデータを使用できるかどうかを確認します。以前のアプリケーション・テストでは、アプリケーションが問題なく実行できることを確認しました。完全なストレージで行うこのテストの目的は、クライアントにサービスを提供するときに使用するデータを、アプリケーションが認識して操作できるかどうかを確認することです。

注意

解決できない問題を発見し、ASE に戻すことを決定した場合は、8.7.2 項の手順に従ってください。

- d. クライアントへのサービスを開始します。NFS クライアントでは、すべての NFS マウント要求を省略時のクラスタ別名または `/etc/exports.aliases` の中に名前が指定されている別名に宛てて送信する必要があります。
18. 残りの ASE メンバを一度に 1 つずつクラスタに追加します。各システムで次の作業を行います。
- a. システムの電源がオフになっていることを確認します。
 - b. クラスタ・インターコネクト・アダプタ (Memory Channel またはイーサネット) の追加または交換が必要な場合は、アダプタを取り付けます。
 - c. システムを共用ストレージに接続します。
 - d. クラスタ・インターコネクト・ケーブルを接続します。
 - e. システムの電源をオンにします。
 - f. `show config` コンソール・コマンドを使用して、コンソールとアダプタ・ファームウェアのリビジョン番号が Tru64 UNIX バージョン 5.1B と互換性があるかどうかを確認します。互換性がない場合は、必要に応じてファームウェアをアップデートします。
 - g. 第 5 章で説明する手順に従い、現在のクラスタ・メンバで `clu_add_member` コマンドを実行して、新しいメンバのブート・ディスクを作成します。その後、新しいメンバをクラスタにブートします。
19. (オプション作業) 移行ディレクトリを削除し、移行サブセットを削除します。いずれかのクラスタ・メンバで、次の作業を行います。
- a. 移行ディレクトリを削除します。

```
# rm -rf /var/TruCluster_migration
```
 - b. 移行ツールを削除します。

```
# setld -d TCRMIGRATE540
```

8.7.2 オプション 3: 部分アップグレードの取り消し

ストレージの移行時またはアップグレードの完了後に、解決できない問題が発生した場合は、次の手順に従って ASE 構成に戻します。

1. すべてのシステムを停止します。
2. すべてのシステムの電源をオフにします。

注意

ストレージの電源をオフにするかどうかは、ユーザの判断です。

3. Memory Channel アダプタを取り付けていた場合は、それらを取り外します。
4. ストレージがすべてのシステムに接続されていない場合は、ストレージを再接続します。ASE に接続されていたのと全く同様にストレージが接続されていることを確認します。
5. 共用ストレージの電源がオフになっていた場合は、電源をオンにします。
6. システムの電源をオンにします。
7. コンソール変数を以前の値に戻します。
8. アップグレード中に SCSI のワイド・アドレス (ID 8 ~ 15) を使用した場合、保存してある以前の設定値に復元します。
9. ASE の各メンバで、旧バージョンのオペレーティング・システムをマルチユーザ・モードでブートします。
10. `clu_migrate_save` を実行した ASE の各メンバ上で `clu_migrate_recover` コマンドを実行します。

```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_recover
```
11. すべてのメンバが動作し、回復したときに、いずれかのメンバ上で `asemgr` コマンドを実行して、すべてのサービスをオンラインに設定します。

12. (オプション作業) 移行ディレクトリおよび移行サブセットを削除します。ASE の各メンバで次の作業を行います。

a. 移行ディレクトリを削除します。

```
# rm -rf /var/TruCluster_migration
```

b. 移行ツールを削除します。

```
# setld -d TCRMIGRATE540
```

8.8 アップグレードのケース・スタディ

この節では、4 つのメンバを持つ TruCluster Production Server バージョン 1.6 のクラスタを TruCluster Server バージョン 5.0A へアップグレードする方法について説明します。例として、類似したハードウェアで移行をテストしてから、実際に運用しているバージョン 1.6 の複数のクラスタのアップグレード方法を決定する場合をとりあげます。

注意

ケース・スタディのターゲットにしているバージョンはバージョン 5.0A ですが、TruCluster Server のどの 5.x バージョンにアップグレードする場合でも、基本手順は同じです。

ここで述べるケース・スタディ以外にも、社内の運用レベルのクラスタを TruCluster Production Server バージョン 1.6 からパッチ・キットを含めた TruCluster Server バージョン 5.0A にアップグレードする場合の説明については、以下の URL にあるリンクをたどってください。

<http://www.tru64unix.compaq.com/docs/highavail/index.htm>

アップグレード前のクラスタは、次のハードウェアとソフトウェアで構成されています。

- 4 台の AlphaServer GS140 ラックマウント・システム (各システムに、8 つの CPU、8GB の RAM、および 14 個の KGPSA-BC アダプタを装備)。
- 14 の Fibre Channel ストレージ・エリア・ネットワーク・スイッチ (16 ポート)。

- 14 の StorageWorks ESA 12000 ストレージ・アレイ Fibre Channel キャビネット (各キャビネットに、二重冗長 HSG80 アレイ・コントローラおよび 36GB のディスク 48 台を装備)。
- 2 つの StorageWorks ESL9326D エンタープライズ・テープ・ライブラリ。
- 2 つの Memory Channel II ハブ。
- Tru64 UNIX バージョン 4.0F (パッチつき)。
- TruCluster Production Server バージョン 1.6 (ASE サービスと LSM は使用しない)。

このクラスタは、Tru64 UNIX バージョン 5.0A と TruCluster Server バージョン 5.0A で構成されて出荷されたものですが、アップグレードをテストするために、あえて Tru64 UNIX バージョン 4.0F と Production Server バージョン 1.6 とともにロードされるようにしています。

- 112 DRD (Distributed Raw Disk) デバイスを使用した Oracle 8.1.6 のシングル・インスタンス。

バージョン 5.0A のクラスタで同じバイナリを実行し、アップグレードが成功したかどうかをテストします。

アップグレード後のクラスタは、Tru64 UNIX バージョン 5.0A 上で TruCluster Server バージョン 5.0A を実行します。

3 通りの標準的なアップグレード方法の中からオプション 3 の修正バージョンでアップグレードします。次の事項が決定されました。

- ストレージまたは Memory Channel のケーブルを外さない。
この規模のクラスタでは多数のケーブル接続が必要なため、ケーブルを外さずに作業時間の短縮とリスクの軽減を図ります。
- ストレージ、Memory Channel ハブ、およびシステムの電源を切らない。
実際のアップグレードはコンピュータ・センタからリモートで制御するため、電源を切断するかわりに必要に応じてシステムをコンソール・プロンプトで制御します。ただし、この場合、管理者が誤ってシステムを起動しないようにする必要があります。
- 同時にすべてのシステムを停止する。

ストレージと Memory Channel ハブが接続され、電源が入った状態になっているため、Tru64 UNIX バージョン 5.0A のインストール中には、Production Server クラスタの実行を継続することはできません。

- アップグレード前に、Tru64 UNIX バージョン 5.0A をインストールするシステムのローカル・ディスクのファイル・システムに `clu_migrate_save` からのすべての出力をコピーする。Tru64 UNIX バージョン 5.0A システムをインストールしてから `clu_migrate_configure` を実行するまでの間に、ファイル・システムをマウントし、ファイルを `/var/TruCluster_migration` にコピーする。

オプション 2 とオプション 3 では、新しいクラスタ (オプション 2) またはストレージおよび Memory Channel と物理的に接続されていない現在のクラスタのメンバ (オプション 3) に `clu_migrate_save` の出力がネットワークを介してコピーされます。前述の手順では、Tru64 UNIX バージョン 5.0A のインストール前にバージョン 1.6 の Production Server クラスタ全体をシャットダウンするため、ファイルをバージョン 5.0A のシステムにコピーする必要があります。

アップグレード前のクラスタは、少しカスタマイズされています。LSM が使用されておらず、ASE サービスも定義されていません。Oracle 8.1.6 のシングル・インスタンスは、アップグレードが成功したかどうかのテストとして使用されます。このアプリケーションを新しいクラスタで実行し、このクラスタのすべてのストレージにアクセスできる場合は、アップグレードが成功したものとみなすことができます。

実際のアップグレード手順の概要は、次のとおりです。

1. TruCluster Software バージョン 1.6 のクラスタの 4 つのメンバすべてに TCRMIGRATE505 サブセットをロードします。
2. すべてのメンバで `clu_migrate_check` を実行します。
3. 1 つのメンバを先行メンバ (最初にアップグレードするシステム) として選択します。 `/var/TruCluster_migration` ディレクトリを作成します。
4. 先行メンバの `/var/TruCluster_migration` ディレクトリを `clu_migrate_save` で作成するデータ・ファイルの `rcp` の格納先として指定し、すべてのメンバで `clu_migrate_save` を実行します。

`disklabel -r` を使用してディスク・ラベルを検査します。ディスクの現在のデバイス・スペシャル・ファイル名を表す `@rzxxx` 文字列がディスク・ラベルに含まれていることを確認します。

5. バージョン 1.6 クラスタのすべてのメンバをシャットダウンしてコンソール・モードにします。

注意

この例では、バージョン 5.0A でサポートされる HSG80 のマルチバス・フェイルオーバを利用するように Fibre Channel スイッチ・ファブリックをあらかじめ構成しておきます。このように構成していない場合は、SAN スイッチのケーブルを接続し直します。このアップグレード方法では、すべてのシステムをコンソール・プロンプトで制御する間、各 HSG80 コントローラが透過的フェイルオーバからマルチバス・フェイルオーバに変更されて、再起動されます。

6. HSG80 コンソールで、5 つのディスクを使用してストレージセット (RAID レベル 5) を作成し、次のパーティションを設定します (サイズはパーセンテージで表します。パリティを除いた後の平均合計空き容量は、145GB です)。

- Tru64 UNIX バージョン 5.0A ディスク : 6% (約 8.7 GB)
- TruCluster Server バージョン 5.0A ディスクのルート (/), /usr/, および /var : 6% (約 8.7 GB)
- 各メンバのブート・ディスク : 15% (約 21.8 GB)

GS140 には 8 GB のメモリが含まれるため、各システムに多くのスワップ領域が必要。

- クォーラム・ディスク : 1% (約 1.45 GB)
- 残りのパーティション (将来使用するために予約されている) : 27% (約 39.3 GB)

HSG80 コンソールで使用するコマンドの概要は、次のとおりです。

注意

この例では、ストレージとブート・ディスクを構成するためのコンソール・コマンドと HSG80 コマンドの一部を紹介します。HSG80 コントローラの配下でブート・ディスクを使用する場合は、TruCluster Server 『クラスタ・ハードウェア構成ガイド』で説明する手順に従ってください。

```
HSG14 BOT> show unit
```

LUN	Uses	Used by
D0	R1	
D1	R2	
D2	R3	
D3	R4	
D100	R5	
D101	R6	
D102	R7	
D103	R8	

```
HSG14 BOT> locate d103      ! verify the disk to be deleted
HSG14 BOT> locate cancel
HSG14 BOT> delete unit d103 ! delete the existing unit
HSG14 BOT> show r8         ! check how much space is available
```

Name	Storageset	Uses	Used by
R8	raidset	DISK21100 DISK31100 DISK41100 DISK51100 DISK61100	
Switches:			
POLICY (for replacement) = BEST_PERFORMANCE			
RECONSTRUCT (priority) = NORMAL			
CHUNKSIZE = 256 blocks			
State:			
UNKNOWN -- State only available when configured as a unit			
Size: 284389020 blocks			

```
HSG14 BOT> create_partition r8 size=6
HSG14 BOT> create_partition r8 size=6
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=1
HSG14 BOT> create_partition r8 size=largest
```

Name	Storageset	Uses	Used by
R8	raidset	DISK21100 DISK31100 DISK41100 DISK51100 DISK61100	

```

Switches:
  POLICY (for replacement) = BEST_PERFORMANCE
  RECONSTRUCT (priority) = NORMAL
  CHUNKSIZE = 256 blocks
State:
  UNKNOWN -- State only available when configured as a unit
Size:      284389020 blocks
Partitions:
  Partition number      Size              Starting Block      Used by
  -----
    1                   17062907 ( 8736.20 MB)          0
    2                   17062907 ( 8736.20 MB)    17062912
    3                   42657787 ( 21840.78 MB)    34125824
    4                   42657787 ( 21840.78 MB)    76783616
    5                   42657787 ( 21840.78 MB)   119441408
    6                   42657787 ( 21840.78 MB)   162099200
    7                   2843643 ( 1455.94 MB)    204756992
    8                   76788375 ( 39315.64 MB)   207600640

HSG14 BOT> add unit d4 r8 part=1 ! Tru64 UNIX V5.0A disk
HSG14 BOT> add unit d5 r8 part=2 ! TruCluster V5.0A disk
HSG14 BOT> add unit d6 r8 part=3 ! member 1 boot disk
HSG14 BOT> add unit d7 r8 part=4 ! member 2 boot disk
HSG14 BOT> add unit d8 r8 part=5 ! member 3 boot disk
HSG14 BOT> add unit d9 r8 part=6 ! member 4 boot disk
HSG14 BOT> add unit d10 r8 part=7 ! quorum disk
HSG14 BOT> add unit d11 r8 part=8 ! remaining space

HSG14 BOT> show r8

Name          Storageset          Uses          Used by
-----
R8            raidset
              DISK21100      D10
              DISK31100      D11
              DISK41100      D4
              DISK51100      D5
              DISK61100      D6
              D7
              D8
              D9

Switches:
  POLICY (for replacement) = BEST_PERFORMANCE
  RECONSTRUCT (priority) = NORMAL
  CHUNKSIZE = 256 blocks
State:
  NORMAL
  DISK21100 (member 0) is NORMAL
  DISK31100 (member 1) is NORMAL
  DISK41100 (member 2) is NORMAL
  DISK51100 (member 3) is NORMAL
  DISK61100 (member 4) is NORMAL
Size:      284389020 blocks
Partitions:
  Partition number      Size              Starting Block      Used by
  -----
    1                   17062907 ( 8736.20 MB)          0      D4
    2                   17062907 ( 8736.20 MB)    17062912      D5
    3                   42657787 ( 21840.78 MB)    34125824      D6
    4                   42657787 ( 21840.78 MB)    76783616      D7
    5                   42657787 ( 21840.78 MB)   119441408      D8
    6                   42657787 ( 21840.78 MB)   162099200      D9
    7                   2843643 ( 1455.94 MB)    204756992     D10
    8                   76788375 ( 39315.64 MB)   207600640     D11

```



```
HSG14 BOT> show unit
```

LUN	Uses	Used by
D0	R1	
D1	R2	
D2	R3	
D3	R4	
D4	R8	(partition)
D5	R8	(partition)
D6	R8	(partition)
D7	R8	(partition)
D8	R8	(partition)
D9	R8	(partition)
D10	R8	(partition)
D11	R8	(partition)
D100	R5	
D101	R6	
D102	R7	

```
HSG14 BOT> set d4 id=100 ! create user-defined identifiers (UDIDs)
HSG14 BOT> set d5 id=101
HSG14 BOT> set d6 id=1
HSG14 BOT> set d7 id=2
HSG14 BOT> set d8 id=3
HSG14 BOT> set d9 id=4
```

7. 先行メンバ (メンバ 1) のコンソールで, `wwidmgr` コマンドを使用して, コンソール・デバイス名を Tru64 UNIX バージョン 5.0A ディスクのユーザ定義 ID (UDID) にマップします。この ID は HSG80 上で作成されます。また, このメンバのブート・ディスクをコンソール・デバイス名にマップし, `bootdef_dev` を設定します。

```
P00>>> set mode diag
Console is in diagnostic mode
P00>>> wwidmgr -quickset -udid 100 # Tru64 UNIX Version 5.0A disk
P00>>> wwidmgr -quickset -udid 1 # member 1 boot disk
:
Disk assignment and reachability after next initialization:

6000-1fe1-0005-9dc0-0009-0010-4628-00c6
via adapter: via fc nport: connected:
dgm1.1001.0.7.7 kgpsam0.0.0.7.7 5000-1fe1-0005-9dc3 Yes
dgm1.1002.0.7.7 kgpsam0.0.0.7.7 5000-1fe1-0005-9dc1 No
dgn1.1003.0.10.7 kgpsan0.0.0.10.7 5000-1fe1-0005-9dc2 No
dgn1.1004.0.10.7 kgpsan0.0.0.10.7 5000-1fe1-0005-9dc4 Yes
:
P00>>> init
:
P00>>> show device
:
kgpsam0.0.0.7.7 PGM0 WWN 1000-0000-c922-09f9
dgm100.1001.0.7.7 $1$DGA100 HSG80 V85F
dgm1.1001.0.7.7 $1$DGA1 HSG80 V85F
dgm1.1002.0.7.7 $1$DGA1 HSG80 V85F
dgn1.1003.0.10.7 $1$DGA1 HSG80 V85F
dgn1.1004.0.10.7 $1$DGA1 HSG80 V85F
```

```

:
P00>>> set bootdef_dev dgm1.1001.0.7.7
P00>>> init
:

```

8. 残りの各メンバについては、各コンソールで `wwidmgr` コマンドを使用して、それぞれのブート・ディスクの UDID をコンソール・デバイス名にマップし、`bootdef_dev` を設定します。

```

(member 2)
P00>>> set mode diag
Console is in diagnostic mode
P00>>> wwidmgr -s quickset udid 2
P00>>> init
:
P00>>> set bootdef_dev dgm2.1001.0.7.7
P00>>> init
:
(member 3)
P00>>> set mode diag
Console is in diagnostic mode
P00>>> wwidmgr -quickset -udid 3
P00>>> init
:
P00>>> set bootdef_dev dgm3.1001.0.7.7
P00>>> init
:
(member 4)
P00>>> set mode diag
Console is in diagnostic mode
P00>>> wwidmgr -quickset -udid 4
P00>>> init
:
P00>>> set bootdef_dev dgm4.1001.0.7.7
P00>>> init
:

```

注意

この例では、クラスタを作成するために必要なディスクの初期構成時に、起動可能なディスクだけに UDID を割り当てていますが、他のディスクにも UDID を割り当てることをお勧めします。UDID とディスクを関連付けると、`hwmgr` などのユーティリティを使用してデバイスを追跡しやすくなります。1000 の UDID を使用できるため、制限する必要はありません。

また、この例では、コンソールで `bootdef_dev` 環境変数を使用して各起動デバイスにパスを 1 つだけ設定していま

す。クラスタの作成後、各メンバに複数のブート・パスを設定します。

9. 先行メンバのコンソールで、Tru64 UNIX バージョン 5.0A をインストールします。基本的なネットワーク・サービスとタイム・サービスを構成します。TruCluster Server バージョン 5.0A のサブセットをロードします。
10. Tru64 UNIX バージョン 4.0F の `usr_domain#usr` を `/mnt` にマウントし、`clu_migrate_save` で収集されたストレージ情報を含む移行ディレクトリをバージョン 5.0A システムの `/var/TruCluster_migration` にコピーします。
11. `clu_migrate_configure -x` を実行します。`clu_migrate_configure` で実行されるコマンドを検査します。
12. `clu_migrate_configure` を実行します (この TruCluster Software バージョン 1.6 の Production Server クラスタでは、ASE サービスも LSM も使用しないため、`clu_migrate_configure` で、`/etc/fstab` へのエントリの追加、ファイル・システムのマウント、および LSM ボリュームの作成が行われることはありません)。

新しいスタイルの `dsk` デバイス名を Oracle のテスト・データベースで使用する `drd` リンクにマップするシェル・スクリプトの入力として、`clu_migrate_configure` ログ・ファイルを使用します。

注意

Tru64 UNIX バージョン 5.0A をインストールしてバージョン 5.0A クラスタを作成するためのディスク・デバイスは、バージョン 1.6 クラスタのシャットダウン後に作成されます。そのため、`clu_migrate_save` ではこれらのデバイスを認識できません。また、`clu_migrate_configure` では、以前のスタイルのデバイス名が存在しないため、以前のデバイス名を Tru64 UNIX バージョン 5.0A のインストール時に割り当てられる新しいスタイルのデバイス名にマップすることができません。

13. `clu_create` を実行して、シングル・メンバ・クラスタを作成します。

14. Tru64 UNIX バージョン 5.0A システムを停止し、シングル・メンバ・クラスタとして起動する前に、クラスタのブート・ディスクに対するブート・パスを複数設定します。

```
P00>>> set bootdef_dev dgm1.1001.0.7.7,dgm1.1002.0.7.7,\
dgn1.1003.0.10.7,dgn1.1004.0.10.7
      ⋮
P00>>> init
```

15. `clu_add_member` を実行し、4 つのメンバの TruCluster Server バージョン 5.0A クラスタの作成を終了します。先行メンバと同様に、各メンバをクラスタとして起動する前にそれぞれについて複数のブート・パスを設定します。
16. Oracle バージョン 8.1.6 のバイナリを実行し、バージョン 1.6 の Production Server クラスタで作成されたテスト・データベースに Oracle からアクセスできるかどうかをテストします。
テストが成功しました。

移行は完了です。

8.9 TruCluster Memory Channel Software クラスタのアップグレード

この節では、TruCluster Memory Channel Software クラスタを TruCluster Server バージョン 5.1B へアップグレードするための一般的な方法を説明します。

この節の説明には、次のような前提条件を設けています。

- かなり低いコストで TruCluster Server バージョン 5.1B へアップグレードすることを目指す。したがって、ストレージを追加する場合は、HSZ70 RAID アレイ・コントローラや Fibre Channel と HSG80 の組み合わせではなく、可能な限り、SCSI アダプタ、ケーブル、および UltraSCSI BA356 などのローエンドのストレージ・コンテナを使用する。
- Memory Channel クラスタには、現在、TruCluster Server バージョン 5.1B のインストールに必要な共用ストレージが構成されていない。おそらく、共用ストレージはほとんど（またはまったく）存在せず、メンバは Memory Channel だけで接続されている。メンバが必要とするストレージはすべて、そのメンバの内部か、またはプライベート・パス

上にある。ハードウェアの主な変更は、TruCluster Server バージョン 5.1B のクラスタを作成するために必要な SCSI アダプタ、ケーブル、ストレージ・コンテナ、およびディスクの追加だけである。既存の Memory Channel インターコネクトと、すべての外部ネットワーク接続は変更の必要がない。

- ほとんどの Memory Channel クラスタは可用性よりも性能に重点を置いて設計されているため、アップグレードして NSPOF (no-single-point-of-failure) の構成にする必要がない。バージョン 5.1B クラスタを作成するためにハードウェアを追加して準備する場合は、ニーズに最も合った冗長構成を採用することができる。

すべてのメンバにそれぞれオペレーティング・システムがあるため、Memory Channel クラスタには、オペレーティング・システムのレベルである程度の冗長性が備わっています (1 つのメンバでオペレーティング・システム・ディスクに障害が発生しても、クラスタ全体はダウンしません)。アップグレードすると、TruCluster Server クラスタ・メンバは同じルート (/), /usr, /var ファイル・システムを共用するようになります。このため、ディスクが 1 台使用できなくなっただけでクラスタが利用できなくならないように、何らかの形でソフトウェアまたはハードウェアの RAID が必要となります。アップグレードしたクラスタが使用するストレージはローエンドであり、ハードウェアの RAID コントローラをサポートしないので、LSM を使用してルート (/), /usr, および /var ファイル・システムをミラー化します。

- アップグレード中のダウン時間は問題にならない。Memory Channel クラスタは、共用ストレージを追加して TruCluster Server バージョン 5.1B ソフトウェアをインストールするために、シャットダウンされる。このダウン時間を許容できない環境であれば、TruCluster Server バージョン 5.1B クラスタを別に作成する必要がある。

図 8-1 に、8 個のノードで構成するクラスタの基本ブロック図を示します。
『クラスタ・ハードウェア構成ガイド』に、この図の他にいくつかの図を加えて、詳細なケーブル接続とストレージのレイアウトが示されています。

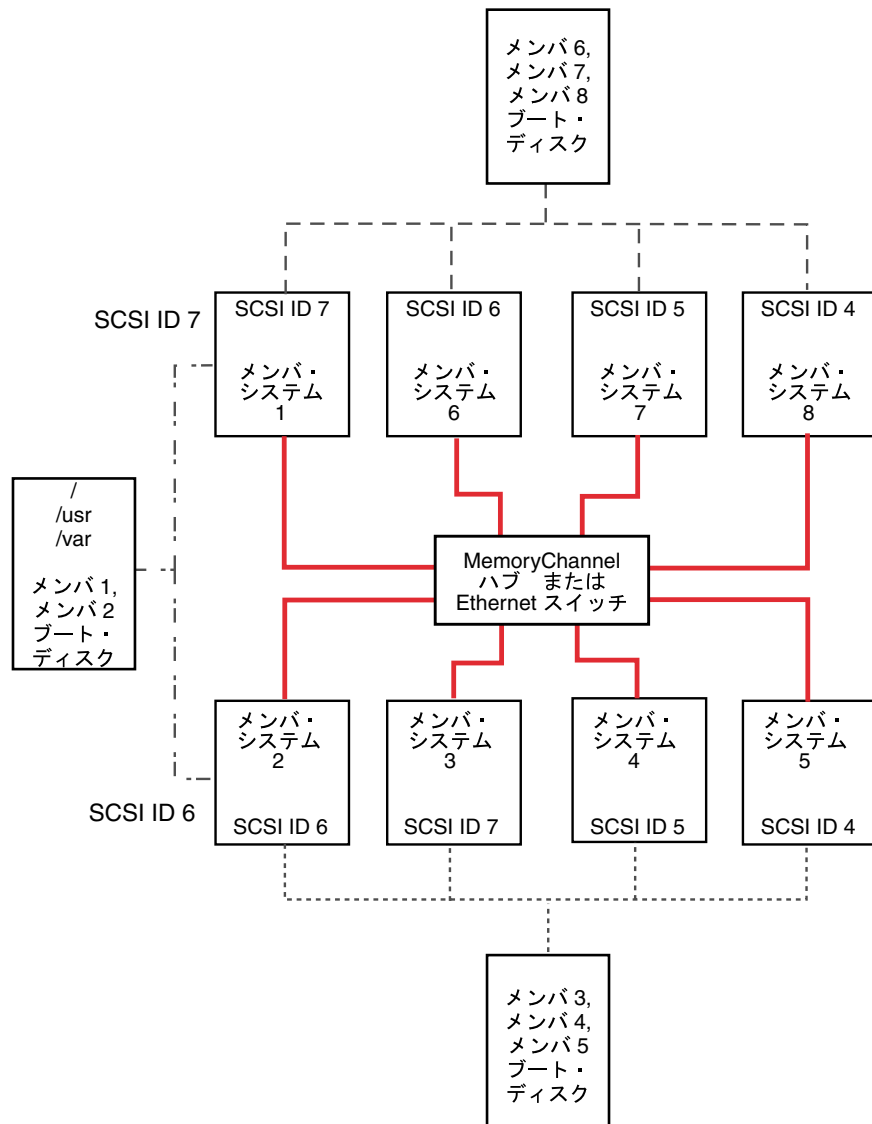
注意

『クラスタ・ハードウェア構成ガイド』には、「外部終端の共用 SCSI バスを使った 8 メンバ・クラスタの構成」という章があります。そこには、この図の他にも詳細な構成図がいくつか紹介され

ています。またその章には、ここで述べる手順と組み合わせて使用する詳細なハードウェア構成情報も記載されています。

このブロック図には、ルート (/)、/usr、および /var ファイル・システムの LSM ミラーが置かれているストレージは示してありません。『クラスタ・ハードウェア構成ガイド』の「概要」の章には、これらのファイル・システムを LSM でミラー化するために SCSI バスを二重化する方法が図で示されています。

図 8-1: 8 ノード・クラスタのブロック図



ZK-1847U-AI

Memory Channel クラスタの構成は多岐にわたるため、以下の手順では、クラスタのアップグレードに必要な手順をすべて記述しているわけではあ

りません。以下の手順は、場合に応じて必要なアップグレードを行うための出発点としてください。

1. Memory Channel クラスタ内のシステムが TruCluster Server バージョン 5.1B でサポートされていることを確認します。サポートされているシステムについては、TruCluster Server バージョン 5.1B の『*QuickSpecs*』を参照してください。『*QuickSpecs*』の最新版は、以下の URL から入手できます。

http://www.tru64unix.compaq.com/docs/pub_page/spds.html

注意

8.4 節に記載してある `clu_migrate_check` スクリプトは使用できません。このスクリプトは TruCluster Production Server および Available Server クラスタのためのものです。

2. 『クラスタ・ハードウェア構成ガイド』の「外部終端の共用 SCSI バスを使った 8 メンバ・クラスタの構成」という章を参照して、クラスタへ追加する必要があるストレージ・ハードウェアを決定します。Memory Channel クラスタのメンバの数に基づいて、アップグレードに必要な共用 SCSI バスの数を決定します (1 本の SCSI バスに 4 メンバまで接続できます)。次に、SCSI アダプタ、ケーブル、ターミネータ、ストレージ・シェルフ、およびディスクの必要な数を判断します。

さらに、本書の第 1 章と第 2 章を参照して、ディスク・スペースの割り当て方と、クォーラム・ディスクを使用するかどうかを決めます。付録 A のチェックリストにデータを記入します。

3. アップグレードに必要なハードウェア、ソフトウェア、およびライセンスを入手します。

注意

新バージョンのオペレーティング・システムおよびクラスタ・ソフトウェアには、通常、新バージョンの AlphaServer SRM ファームウェアが必要です。この時点で SRM ファームウェアをアップデートできますが、Memory Channel クラスタをシャットダウンした後で行ってもかまいません。ダウン時間を最短に抑えるには、SRM ファームウェアを一度に 1 シ

システムずつアップグレードし、その後で Memory Channel クラスタ全体をシャットダウンします。SRM ファームウェアについての詳細は、3.1 節を参照してください。

4. `sysconfig -q rm` を使用して `rm_rail_style` 属性の値を表示します。この値を記録しておきます (ほとんどの Memory Channel クラスタはマルチアクティブ・レール・スタイル (`rm_rail_style=0`) を使用しますが、TruCluster Server バージョン 5.1B の省略時のスタイルはフェイルオーバー・ペア (`rm_rail_style=1`) です)。
5. 新しいクラスタの最初のメンバになるシステムを決めます。このシステムは TruCluster Server の共用ルート (/), /usr, および /var ファイル・システムを置くストレージに直接接続していなければなりません。
6. 現在のオペレーティング・システムが置かれているディスクに Tru64 UNIX をインストールする場合は、現在のオペレーティング・システムのバックアップを取ってから作業を続けます。
7. 各システムをシャットダウンして、停止します。
8. 各システムのコンソールで、次のコンソール変数を設定します。

```
>>> set auto_action halt
>>> set bootdef_dev ""
>>> set boot_osflags A
>>> set boot_reset on
```

`bus_probe_algorithm` 変数をサポートしているシステムでは、次のように設定します。

```
>>> set bus_probe_algorithm new
```

コンソール変数についての詳細は、2.7 節を参照してください。

`bus_probe_algorithm` 変数を使用しないシステムでこの変数を設定しても問題はありません。この変数は、次の `init` または電源の再投入で、クリアされます。

9. 各システムの電源をオフにします。

注意

指示があるまで、これらのシステムの電源はオンにしないでください。

10. 『クラスタ・ハードウェア構成ガイド』の説明を参照して、TruCluster Server バージョン 5.1B クラスタの作成に必要なストレージを追加します。他のハードウェアを追加または再構成する場合は、この時点で実行します。SRM ファームウェアをアップデートしていない場合は、ここでアップデートします。
11. Memory Channel ハブを使用している場合は、ハブの電源がオンになっていることを確認します。
12. 新しいクラスタの最初のメンバになるシステムの電源をオンにします。コンソールのプロンプトに対して、`show config` コンソール・コマンドを使用して、コンソールおよびアダプタのファームウェア・リビジョンが Tru64 UNIX バージョン 5.1B と互換性があることを確認します。互換性がない場合は、必要に応じてファームウェアをアップデートします。
13. 第 3 章の説明に従って、Tru64 UNIX バージョン 5.1B のフル・インストールを実行します。

注意

Memory Channel クラスタの使用するオペレーティング・システムが置かれているディスクは、上書きしないことを強くお勧めします。ディスクを変更しなければ、後で問題が発生した場合に、このシステムを簡単に Memory Channel クラスタへ戻すことができます。

14. Tru64 UNIX バージョン 5.1B システムで、次の操作を実行します。
 - a. Tru64 UNIX オペレーティング・システムを完全に構成します。第 3 章の手順に従って作業してください (ベース・オペレーティング・システム上に LSM を構成する場合は、2.5.2 項を参照してください)。
 - b. アプリケーションをインストールします。
アプリケーションを、いつどのようにインストールするかは、アプリケーションの種類と Memory Channel クラスタの使い方によって異なります。TruCluster Server バージョン 5.1B クラスタのファイル・システムは同じネーム・スペースを共有しているので注意してください。TruCluster Server バージョン 5.1B クラスタでアプリ

ケーションを実行する方法についての詳細は、『クラスタ高可用性アプリケーション・ガイド』を参照してください。

- c. TruCluster Server バージョン 5.1B のライセンスとサブセットをインストールします。

注意

作業を続ける前にシステムのバックアップをとっておくことをお勧めします。この後の手順で不具合が生じた場合には、Tru64 UNIX をインストールおよび構成してからアプリケーションをインストールし、TruCluster Server サブセットをロードするよりも、速くこの時点まで復元することができます。

- 15. 第 4 章の手順に従い、clu_create コマンドを実行してシングル・メンバ・クラスタを作成します。
- 16. システムを停止し、それをシングル・メンバのクラスタとしてブートします。
- 17. Memory Channel クラスタ・システムの rm サブシステム属性が rm_rail_style=0 であった場合は、次のようにして、シングル・メンバの TruCluster Server バージョン 5.1B クラスタでそれを 0 に設定してから、システムをリブートします。
 - a. /etc/sysconfigtab ファイルを変更して、次のスタンザを含むようにします。

```
rm:
  rm_rail_style=0
```
 - b. シングル・メンバのクラスタをリブートします。

```
# shutdown -r now
```
- 18. LSM を使用して、ルート (/)、/usr、および /var ファイル・システムをミラー化します (2.5.2 項、volmigrate(8)、volencap(8)、および『クラスタ管理ガイド』を参照)。

19. その他のシステムを一度に 1 つずつクラスタに追加します。各システムで次の作業を行います。
 - a. システムの電源をオンにします。
 - b. `show config` コンソール・コマンドを使用して、コンソールおよびアダプタのファームウェア・リビジョンが Tru64 UNIX バージョン 5.1B と互換性があるかどうかを確認します。互換性がない場合は、必要に応じてファームウェアをアップデートします。
 - c. 第 5 章で説明している手順に従い、現在のクラスタ・メンバで `clu_add_member` コマンドを実行して、新しいメンバのブート・ディスクを作成します。
 - d. 新しいメンバをブートしてクラスタに追加します。

A

情報チェックリスト

表 A-1: Tru64 UNIX システム属性

属性	値
ホスト名	
ホスト名の IP アドレス	
Tru64 UNIX のルート・パーティション (たとえば, dsk0a)	
Tru64 UNIX のコンソールからのルート・デバイス (たとえば, dKA0)	
Tru64 UNIX の /usr のパーティション (たとえば, dsk0g)	
Tru64 UNIX の /var のパーティション (たとえば, dsk0h)	

表 A-2: クラスタ属性

属性	値
クラスタ名 (完全修飾名)	
省略時のクラスタ別名の IP アドレス	
クラスタ単位のルート (/) パーティション (たとえば, dsk1b)	
クラスタ単位のルート (/) のディスク・デバイス・シリアル番号 (たとえば, WWID)	
クラスタ単位の /usr のパーティション (たとえば, dsk2c)	
クラスタ単位の /usr のディスク・デバイス・シリアル番号 (たとえば, WWID)	
クラスタ単位の /var のパーティション (たとえば, dsk3c)	
クラスタ単位の /var のディスク・デバイス・シリアル番号 (たとえば, WWID)	
クォーラム・ディスクを使用している場合のディスク・デバイス (たとえば, dsk7)	

表 A-2: クラスタ属性 (続き)

属性	値
クォーラム・ディスクを使用している場合のディスク・デバイス・シリアル番号 (たとえば, WWID)	
クォーラム・ディスクを使用している場合にクォーラム・ディスクへ割り当てられたポート数	

表 A-3: メンバ属性

属性	第 1 メンバ	第 2 メンバ	第 3 メンバ
メンバ・ホスト名 (完全修飾名) ^a			
ホスト名の IP アドレス ^a			
メンバ ID ^b (memberid: 1 ~ 63)			
このメンバへ割り当てられたポート数 ^c			
ブート・ディスク (たとえば, dsk10)			
コンソールからのブート・デバイス (たとえば, DKC400)			
ブート・デバイスのシリアル番号 (たとえば, WWID)			
ブート・デバイスの物理的な位置 (バス/ターゲット/LUN)			
仮想クラスタ・インターコネクットの IP 名 ^d			
仮想クラスタ・インターコネクットの IP アドレス ^e			
物理クラスタ・インターコネクットのデバイス名 ^f			
物理クラスタ・インターコネクットの IP アドレス (LAN のみ) ^g			
追加ネットワーク・インタフェースの IP 名			
追加ネットワーク・インタフェースの IP アドレス			
属性	第 4 メンバ	第 5 メンバ	第 6 メンバ
メンバ・ホスト名 (完全修飾名) ^a			
ホスト名の IP アドレス ^a			

A-2 情報チェックリスト

表 A-3: メンバ属性 (続き)

属性	第 1 メンバ	第 2 メンバ	第 3 メンバ
メンバ ID ^b (memberid: 1 ~ 63)			
このメンバに割り当てられたポート数 ^c			
ブート・ディスク (たとえば, dsk10)			
コンソールからのブート・デバイス (たとえば, DKC400)			
ブート・デバイスのシリアル番号 (たとえば, WWID)			
ブート・デバイスの物理的な位置 (バス/ターゲット/LUN)			
仮想クラスタ・インターコネクットの IP 名 ^d			
仮想クラスタ・インターコネクットの IP アドレス ^e			
物理クラスタ・インターコネクットのデバイス名 ^f			
物理クラスタ・インターコネクットの IP アドレス (LAN のみ) ^g			
追加ネットワーク・インタフェースの IP 名			
追加ネットワーク・インタフェースの IP アドレス			

^a第 1 メンバは Tru64 UNIX システムからホスト名と IP アドレスを継承します。

^b第 1 メンバの省略時のメンバ ID は 1 です。それ以降のメンバの省略時のメンバ ID は 1 ずつ増えます。

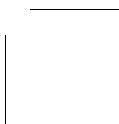
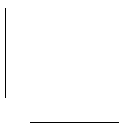
^cクォーラム・ディスクを構成している場合、省略時のポート数は 1 です。

^d省略時の設定では、インストール・プログラムにより、メンバのホスト名の短縮形の後に -ics0 を付けた IP 名が提供されます。

^e省略時の設定では、インストール・プログラムにより、サブネット 10.0.0 上の IP アドレスが、そのホスト部分にメンバ ID を設定して提供されます。

^fMemory Channel または LAN。

^gLAN クラスタ・インターコネクットにのみ必要。省略時の設定では、インストール・プログラムにより、10.1.0 サブネット上の IP アドレスが、そのホスト部分にメンバ ID を設定して提供されます。このアドレスに対応する IP 名は、インストール・プログラムにより自動的に membern-icstcp0 に設定されます (n はメンバ ID)。



B

クラスタ・インストレーションのトラブルシューティング

ここでは、インストレーション中に起こる問題について説明し、その解決方法を示します。

B.1 LAN インターコネクトのトラブルシューティング

この節では、LAN インターコネクトを誤って構成したときに起こる問題について説明し、その解決方法を示します。

B.1.1 省略時の物理クラスタ・インターコネクト IP 名との競合

LAN インターコネクトのあるクラスタでは、省略時の物理クラスタ・インターコネクト IP 名は `membermemberID-icstcp0` の形式となります。

`clu_create` および `clu_add_member` コマンドは `ping` を使用して、省略時の名前がすでにネットで使用されているかどうかを確認します。すでにホストが省略時の IP 名を使用していることがわかると、以下のように入力を求められます。

```
Enter the physical cluster interconnect interface device name []
```

このメッセージが表示された後、コマンドは失敗します。実行していたコマンドによって、以下のメッセージが表示されます。

```
Error: clu_create: Bad configuration
```

```
Error: clu_add_member: Bad configuration
```

上記のいずれかのメッセージが表示された場合、`/cluster/admin/clu_create.log` または `/cluster/admin/clu_add_member.log` を調べると、以下のようなメッセージが記されています。

```
Error: A system with the name 'membermemberID-icstcp0' is  
currently running on your network.
```

このメッセージが記録されている場合は、ネットワーク管理者に問い合わせ、省略時の IP 名を使用している非クラスタ・システムのホスト名を変えて

もらう必要があります。clu_create コマンドおよび clu_add_member コマンドでは、省略時の物理クラスタ・インターコネクト IP 名を変更することはできません。

B.1.2 メンバをブートすると、そのメンバはクラスタに加わるが、マルチユーザ・モードになる前にハングしているように見える

新しいメンバがブート時にクラスタに加わった後しばらくしてハングしているように見える場合は、そのメンバが使用している LAN インターコネクト・アダプタの速度または動作モードが、LAN インターコネクトの速度または動作モードに一致していないと考えられます。この現象は、ハードウェアの設定ミスやイーサネット・ハードウェアの障害によって、アダプタが自動折衝に失敗して起こることがあります。そうした問題の有無を確認するために、問題の起こっているブート中のメンバのコンソール・メッセージを注意深く調べてください。調べるメッセージは次のような形式になっています。

```
ee0: Parallel Detection, 10 Mbps half duplex
ee0: Autonegotiated, 100 Mbps full duplex
```

100 Mb/秒の全二重モードで動作しているクラスタ・インターコネクトで最初に示すメッセージが出ていれば、問題の原因はここにありそうです。2 番目のメッセージが出ていれば、自動折衝が正しく完了しているので、問題の原因はここではありません。

以下に示す注意事項を考慮していなければ、インターコネクトに構成されているイーサネット・アダプタとスイッチの自動折衝の内容によっては、ブート時に予期しないハングを招く可能性があります。

- 自動折衝の設定は、ケーブルの両端で同一にしなければなりません。つまり、イーサネット・アダプタを自動折衝用に構成している場合は、そのアダプタが接続しているスイッチ・ポートも自動折衝用に構成する必要があります。同様に、自動折衝を構成したアダプタが別のメンバのアダプタとクロス・ケーブルで接続されているときは、その別のメンバのアダプタも自動折衝に設定する必要があります。この規則を守らなければ(たとえば、一端を 100 Mb/秒の全二重に設定し、もう一端を自動折衝に設定する)、自動折衝に設定したメンバがブート時に自分自身を半二重モードに設定するので、クラスタのトランザクションが遅くなります。
- AlphaServer システムがサポートしている 100 Mb/秒のイーサネット・ネットワーク・アダプタでは、ee と tu の 2 種類のドライバが使えます。

(ew x0 という形式のコンソール名を持つ) DE50x ファミリのネットワーク・アダプタは、DECchip 21140, 21142, および 21143 チップセットをベースにしており、tu ドライバを使います。tu ドライバを使うネットワーク・アダプタには、自動折衝をサポートするものと、しないものがあります。

注意

DE500-XA アダプタは自動折衝をサポートしていません。多くの場合、DE500-BA と DE504 アダプタの方が DE500-AA アダプタより適切に自動折衝を行えます。

自動折衝を使う場合は、ewx0_mode コンソール変数を auto に設定し、ネットワーク・アダプタに接続しているスイッチのポートを自動折衝に設定します。

tu ドライバを使うネットワーク・アダプタでは、強制的に 100 Mb/秒の全二重モードで使わせるほうが簡単です。アダプタを強制的に 100 Mb/秒の全二重モードに設定する場合は、ewx0_mode 変数を FastFD に設定します。この場合、自動折衝を無効にできるスイッチを使うとともに、そのスイッチでネットワーク・アダプタとの接続に使用するポートを、100 Mb/秒の全二重に設定する必要があります。詳細は、tu(7) およびそのスイッチのマニュアルを参照してください。

(ei x0 という形式のコンソール名を持つ) DE60x ファミリのネットワーク・アダプタは、ee ドライバを使います。ee ドライバを使っているネットワーク・アダプタは速度設定の決定に IEEE 802.3u の自動折衝を使います。ネットワーク・アダプタが接続されているスイッチ上のポートが自動折衝に設定されていることを確認してください。詳細は、ee(7) およびそのスイッチのマニュアルを参照してください。

- DEGPA-xx ファミリでサポートされている 1000 Mb/秒のイーサネット・ネットワーク・アダプタは alt ドライバを使います。省略時の設定では、alt ドライバを使っているネットワーク・アダプタは速度設定の決定に IEEE 802.3u の自動折衝を使います。ネットワーク・アダプタが接続されているスイッチ上のポートが自動折衝に設定されていることを確認してください。詳細は、alt(7) およびそのスイッチのマニュアルを参照してください

B.1.3 メンバをブートすると、クラスタに加わるときにハングする

新しいメンバがブート中にクラスタへ加わるうとしてハングする場合は、その新しいメンバがクラスタ・インターコネクトから切り離されている可能性があります。切断には、以下の原因が考えられます。

- ケーブルが外れている。
- `clu_add_member` を実行するときに既存のイーサネット・アダプタを物理クラスタ・インターコネクト・インタフェースとして指定したが、そのアダプタが他のメンバに接続されていない(おそらくクライアント・ネットワークのように、アダプタが LAN インターコネクト以外の目的で使われている)。
- クラスタ・インターコネクト物理デバイスのアドレスとして、他のクラスタ・メンバとは異なるサブネット上のアドレスを指定した。たとえば、メンバのクラスタ・インターコネクト物理デバイスに対して、クラスタ・インターコネクト仮想サブネット (`ics0`) 上のアドレスを指定したなど。
- このメンバに他のメンバとは異なるインターコネクト・タイプを指定した(たとえば、このメンバの `clubase` カーネル・サブシステムにある `cluster_interconnect` 属性が `mct` になっているのにもかかわらず、他のクラスタ・メンバでは `tcp` になっているなど)。

以下のメッセージのいずれかがコンソールに表示されます。

```
CNX MGR: cannot form: quorum disk is in use. Unable to establish contact
         with members using disk.
```

```
CNX MGR: Node pepperoni id 2 incarn 0xa3a71 attempting to form or join cluster deli
```

この問題を解決するには、以下の手順を実行します。

1. 問題の起こったブート中のメンバを停止します。
2. アダプタが LAN インターコネクトに正しく接続されていることを確認します。
3. 新しいメンバのブート・パーティションを他のメンバにマウントします。たとえば次のようなコマンドを実行します。

```
# mount root2_domain#root /mnt
```

B-4 クラスタ・インストレーションのトラブルシューティング

4. /mnt/etc/sysconfigtab ファイルを調べます。表 C-2 に示されている属性が、メンバの LAN インターコネクト・インタフェースを反映するように正しく設定されていなければなりません。
5. /mnt/etc/sysconfigtab を適切に編集します。
6. メンバのブート・パーティションをアンマウントします。

```
# umount /mnt
```
7. メンバをリブートします。

B.1.4 メンバをブートすると、"ics_ll_tcp" メッセージが出てパニックが発生する

新しいメンバをブートしてクラスタへ加えようとしたときに "ics_ll_tcp: Unable to configure cluster interconnect network interface" というメッセージが出てパニックが発生した場合は、clu_add_member で、存在しないデバイスをメンバの物理クラスタ・インターコネクト・インタフェースとして指定したか、またはブート・カーネルにクラスタ・インターコネクト・デバイスをサポートするデバイス・ドライバが含まれていない可能性があります。

この問題を解決するには、以下の手順を実行してください。

1. 問題の起こったブート中のメンバを停止させます。
2. 新しいメンバのブート・パーティションを他のメンバにマウントします。たとえば、次のようなコマンドを実行します。

```
# mount root2_domain#root /mnt
```
3. /mnt/etc/sysconfigtab ファイルを調べます。表 C-2 に示されている ics_ll_tcp 属性が、メンバの LAN インターコネクト・インタフェースを反映するように正しく設定されていなければなりません。

インタフェースが存在しない場合は、以下を実行します。

1. /mnt/etc/sysconfigtab を適切に編集します。
2. メンバのブート・パーティションをアンマウントします。

```
# umount /mnt
```
3. メンバをリブートします。

インタフェース名が正しい場合は、`vmunix` カーネルに LAN インターコネクト・デバイスのデバイス・ドライバが含まれていない可能性があります。この問題を解決するには、以下を実行します。

1. メンバを `genvmunix` カーネルでブートします。
2. `/sys/conf/HOSTNAME` ファイルを編集して、必要なドライバを追加します。
3. `doconfig` コマンドで `vmunix` カーネルを再構築します。
4. 新しいカーネルをルート (`/`) ディレクトリにコピーします。
5. メンバを `vmunix` カーネルからリブートします。

B.1.5 メンバをブートすると、**"ics_ll_tcp: ERROR: Could not create a NetRAIN set with the specified members"** というメッセージが表示される

新しいメンバをブートしてクラスタに加えようとしたときに `"ics_ll_tcp: ERROR: Could not create a NetRAIN set with the specified members"` というメッセージがインストール作業開始直後に表示された場合は、クラスタ・インターコネクトで使われている NetRAIN 仮想インタフェースが誤って構成されていると考えられます。また、NetRAIN セットのメンバが誤って構成されている場合にも、このメッセージが表示されます。

おそらく、`/etc/rc.config` ファイルを間違えて編集し、LAN インターコネクトに従来の NetRAIN 管理を適用するようになってしまったものと考えられます。この場合、`/etc/rc.config` ファイルにある NetRAIN 構成は無視され、`/etc/sysconfigtab` で定義されている NetRAIN インタフェースがクラスタ・インターコネクトとして使われます。

クラスタ・インターコネクトとして使う NetRAIN セットは `/etc/rc.config` ファイルの中に構成してはなりません。クラスタ・インターコネクトの NetRAIN デバイスは `/etc/rc.config` ではなく、`/etc/sysconfigtab` ファイルの `ics_ll_tcp` カーネル・サブシステム中に構成されています。

この問題を解決するには、以下の手順を実行します。

1. `rcmgr delete` コマンドを使って、新しくブートするメンバの `/cluster/members/{memb}/etc/rc.config` ファイルを編集し、

そのデバイスに対応する NRDEV_X, NRCONFIG_X, NETDEV_X, および IFCONFIG_X の各変数を削除します。

2. rcmgr set コマンドを使って、クラスタ・インターコネクト NetRAIN デバイスを二重に定義している NR_DEVICES 変数と NUM_NETCONFIG 変数の値を減らします。
3. メンバをリブートします。

B.2 その他の問題への対処

B.2.1 クラスタ・ライセンスなしに新しいメンバをブートすると ATTENTION メッセージが表示される

新しく追加したメンバをブートすると、clu_check_config ユーティリティが一連の構成を検査します。TruCluster Server ライセンスの TCS-UA PAK (product authorization key) をメンバにインストールしていない場合、ブート中に以下のメッセージが表示されます。

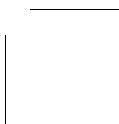
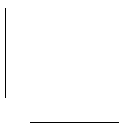
```
Starting Cluster Configuration Check...
The boottime cluster check found a potential problem.
For details search for !!!!!ATTENTION!!!!!! in /cluster/admin/clu_check_log_hostname
check_cdsl_config : Boot Mode : Running /usr/sbin/cdslinvcchk in the background
check_cdsl_config : Results can be found in : /var/adm/cdsl_check_list
clu_check_config : no configuration errors or warnings were detected
```

以下のメッセージが /cluster/admin/clu_check_log_hostname ファイルに格納されます。

```
/usr/sbin/caad is NOT_RUNNING !!!!!ATTENTION!!!!!!
```

TruCluster Server ライセンスがメンバで設定されていないと、CAA (cluster application availability) デーモン (caad) は自動的に起動されません。これは通常の期待される動作です。

新しいメンバを追加した際に、(第 5 章で説明しているように)、clu_add_member でライセンスを設定しなかった場合、その後 lmf register コマンドを使って設定することができます。ライセンスのインストール後、そのメンバに対して /usr/sbin/caad コマンドを使って CAA デーモンを開始することができます。



システム・ファイルの変更

TruCluster Server バージョン 5.1B ソフトウェアをインストールすると、次のシステム構成ファイルとインストール関連ファイルが作成または変更されます。

`/.rhosts`

このファイルが存在しない場合は、`clu_create` が作成します。
`clu_create` コマンドを実行すると、クラスタ別名に関連付けられている完全修飾ホスト名が追加されます。`clu_create` または `clu_add_member` コマンドを実行すると、各クラスタ・インターコネクト・インタフェースに関連付けられている IP 名 (LAN インターコネクトが複数の場合は複数の IP 名) が追加されます。

`/cluster/admin/.membern.cfg`

これらのファイルは、インストール構成ファイルであり、メンバーごとに 1 つずつあります。このファイルは `clu_create` または `clu_add_member` によって作成および追加されます。各ファイルには、そのメンバーに対して `clu_create` または `clu_add_member` を実行したときの選択内容を示した、名前と値のペアがリストされます。

これらのファイルは、`clu_create` および `clu_add_member` によって保守されます。このファイルは編集しないでください。

`/cluster/admin/.member.list`

このテキスト・ファイルは、割り当てられているクラスタ・メンバー ID を追跡するために、`clu_create` および `clu_add_member` が使用します。

このファイルは `clu_create` および `clu_add_member` によって保守されます。このファイルは編集しないでください。

`/cluster/admin/clu_check_config.log`

このログ・ファイルは、`clu_check_config` によって保守されます。

/cluster/admin/clu_create.log

このログ・ファイルは、clu_create によって保守されます。
clu_create を実行するたびに、このファイルにセッションのログ
が追加されます。

/cluster/admin/clu_add_member.log

clu_add_member によって保守されるログ・ファイル。
clu_add_member を実行するたびに、このファイルにセッション
のログが追加されます。

/cluster/admin/clu_upgrade

clu_upgrade は、完了したアップグレードのログ・ファイルをこ
のディレクトリに保存します。

/cluster/admin/clu_upgrade.log

このログ・ファイルは、clu_upgrade によって保
守されます。clu_upgrade は、ローリング・アッ
プグレードが完了すると、このログ・ファイルを
/cluster/admin/clu_upgrade/history/release_version ディ
レクトリに移動します。

/etc/fdmns

clu_create コマンドは、クラスタ単位のルート (/), /usr ,
/var , およびオプションの i18n の各ファイル・システム、および
第 1 クラスタ・メンバ用のドメイン・ディレクトリを作成します。
clu_add_member コマンドは、追加された各メンバ用のドメイン・
ディレクトリを作成します。

/etc/gated.conf

Tru64 UNIX システム上で gated を構成した場合は、このファイルが
第 1 メンバにコピーされ、残りのメンバにも継承されます。gated が
構成されていない場合は、インストール・スクリプトによって各
メンバ用のものが作成されます。クラスタ内では、/etc/gated.conf
は、/cluster/members/{memb}/etc/gated.conf をターゲット・
ファイルとする CDSL です。

注意

aliasd デーモンにより、それぞれのクラスタ・メンバ用に `/etc/gated.conf.membern` が 1 つずつ作成されます。このデーモンは、メンバの `/cluster/members/{memb}/etc/gated.conf` ファイルではなく、このファイルを `gated` の構成ファイルとして使用して、`gated` を起動します。

`/etc/hosts`

`clu_create` および `clu_add_member` コマンドを実行すると、各メンバのクラスタ・インターコネクト・インタフェースに関連付けられている IP アドレスおよびインタフェース名 (LAN インターコネクトが複数あれば複数のアドレスおよび名前) が追加されます。`clu_create` コマンドの場合は、省略時のクラスタ別名に対する IP アドレス、完全修飾ホスト名、および未修飾ホスト名も追加されます。

`/etc/ifaccess.conf`

`clu_create` および `clu_add_member` は、クラスタ・メンバ上にあるすべての非クラスタ・インターコネクト・ネットワーク・インタフェースのための `deny` エントリを追加します。`deny` エントリは、クラスタ外部から入ってくる IP パケットが、それらのインタフェースを介してクラスタ・インターコネクトに送信されないようにします。

`/etc/inittab`

`clu_create` と `clu_add_member` はどちらも、`cms` (クラスタ・マウント・サービス) 用の `sysconfig` コマンド行を追加します。

`/etc/rc.config`

このファイルは、`clu_create` により、`Tru64 UNIX` を実行しているシステムから第 1 クラスタ・メンバにコピーされます。その他のすべてのメンバについては、`clu_add_member` によって `.new.rc.config` から作成されます。インストール時に設定される構成変数のリストは、表 C-1 に示します。

/etc/sysconfigtab

インストール・スクリプトは、sysconfigtab 属性をいくつか初期化します。スクリプトによって情報の入力を求められる属性と、暗黙的に値が設定される属性があります (インストール・プロシージャが初期化する構成変数についての詳細は、表 C-2 を参照してください)。

/etc/sysconfigtab.cluster

clu_create および clu_add_member コマンドは、cluster_expected_votes の値を設定します。

表 C-3 に、インストール・プロシージャによって初期化される構成変数のリストを示しています。

/sys/conf/HOSTNAME.list

このファイルは、インストール・プロシージャによって doconfig プログラムが呼び出される前に、クラスタ・ソフトウェアをカーネルに登録します。HOSTNAME は、システムのホスト名です。

/sys/HOSTNAME/vmunix

新しいカーネルは、/sys/HOSTNAME/vmunix に構築され、その中に置かれます。HOSTNAME は、インストール時に指定した構成ファイルの名前です。clu_create および clu_add_member コマンドは、各メンバの vmunix ファイルを、そのメンバのブート・パーティションにコピーします。

表 C-1 に、clu_create または clu_add_member が作成または変更する /etc/rc.config 変数を示します。

表 C-1: /etc/rc.config の変数

変数	コメント
BASE_VERSION	TruCluster Server がインストールされている Tru64 UNIX オペレーティング・システムのバージョン。現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。
CLU_BOOT_FILESYSTEM	このメンバのブート・ディスクのドメインおよびファイル・セット。clu_create および clu_add_member の両方で設定される。

C-4 システム・ファイルの変更

表 C-1: /etc/rc.config の変数 (続き)

変数	コメント
CLU_NEW_MEMBER	インストール時には 1 に設定される。メンバを初めてクラスタにブートした後に 0 に設定される。
CLU_VERSION	インストールされた TruCluster Server 製品のバージョン番号。現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。
CLUSTER_NET	システムの仮想クラスタ・インターコネクト・インタフェースの IP 名。インストール時に CLUSTER_NET エントリが追加され、その値には、ユーザが指定したインタフェース名が設定される。
GATED	Tru64 UNIX システム上で GATED および ROUTED のいずれも yes に設定されていない場合、GATED が clu_create により yes に設定される。clu_add_member によって現在のメンバの値に設定される。
GATED_FLAGS	clu_add_member によって現在のメンバの値に設定される。
HOSTNAME	メンバを追加した場合に、ユーザが指定した値に設定される。
IFCONFIG_n	クラスタの仮想インターコネクト・インタフェースの IP アドレスとネットマスク。インストール・プロシージャが、IFCONFIG_n エントリを追加する。
IMC_AUTO_INIT	1 に設定すると、システムのブート時に必ず libimc が自動的に初期化される。この初期化時には、約 4.5 MB がライブラリ用に予約される。初期化時における省略時の値は 0 で、ライブラリは初期化されない。現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。
IMC_MAX_ALLOC	クラスタ内での libimc による使用のために割り当てる Memory Channel アドレス空間の最大総容量を、MB 単位で指定する。この変数の値がクラスタのメンバ間で異なる場合は、指定した最大値が使用される。インストール時の省略時の値は 10。現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。

表 C-1: `/etc/rc.config` の変数 (続き)

変数	コメント
IMC_MAX_RECV	libimc が Memory Channel アドレス空間の読み込みのためにマップできる物理メモリの最大総容量を、MB 単位で指定する。この上限はノードによって異なる。インストール時の省略時の値は 10。現在のメンバに対して設定される場合、 <code>clu_add_member</code> で現在のメンバの値に設定される。
MAX_NETDEVS	設定されていないと、 <code>clu_add_member</code> によって 24 に設定される。
NETDEV_n	システムの仮想クラスター・インターコネクト・インタフェースのネットワーク・デバイス名 (<code>ics0</code>)。インストール・プロシージャにより <code>NETDEV_n</code> エントリが追加される。LAN インターコネクトで既存の NetRAIN セットを指定すると、インストール・プロシージャは、そのセットに関連付けられている <code>NETDEV_n</code> エントリを使用する。
NRCONFIG_n	NetRAIN インタフェース・セットのインタフェース。メンバが LAN インターコネクトに対して既存の NetRAIN セットを使用する場合は、 <code>clu_create</code> によってクリアされる。クラスター用の NetRAIN は <code>/etc/rc.config</code> ではなく <code>/etc/sysconfigtab</code> で定義されている。
NRDEV_n	NetRAIN インタフェース名。メンバが LAN インターコネクトに対して既存の NetRAIN セットを使用する場合は、 <code>clu_create</code> によってクリアされる。クラスター用の NetRAIN は <code>/etc/rc.config</code> ではなく <code>/etc/sysconfigtab</code> で定義されている。
NR_DEVICESn	NetRAIN デバイスの数。メンバが LAN インターコネクトに対して既存の NetRAIN セットを使用する場合は、 <code>clu_create</code> によってクリアされる。クラスター用の NetRAIN は <code>/etc/rc.config</code> ではなく <code>/etc/sysconfigtab</code> で定義されている。
NUM_NETCONFIG	構成されたネットワーク・デバイスの数。クラスター・インターコネクトに対して既存の NetRAIN セットを指定しなければ、 <code>NUM_NETCONFIG</code> の値は、 <code>clu_create</code> コマンドにより増分される。 <code>clu_add_member</code> コマンドは <code>NUM_NETCONFIG</code> の値を 1 に設定する。
PAGERAW	<code>clu_add_member</code> によって 1 に設定される。
ROUTER	現在のメンバに対して設定される場合、 <code>clu_add_member</code> によって現在の値に設定される。

表 C-1: /etc/rc.config の変数 (続き)

変数	コメント
TCR_INSTALL	この値が TCR であれば、インストールが成功したことを示す。この値が BAD であれば、インストールが成功しなかったことを示す。現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。
TCR_PACKAGE	TCR に設定される。現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。
XLOGIN	現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。
XNTPD_CONF	現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。現在のメンバに対して設定されない場合、YES に設定される。
XNTPD_OPTS	現在のメンバに対して設定される場合、clu_add_member によって現在のメンバの値に設定される。現在のメンバに対して設定されない場合、-g に設定される。
XNTP_SERVn	クラスタのメンバはすべてお互いにピアとして構成される。各メンバの ntp.conf ファイルには fudge コマンドが置かれ、構成されるサーバが 3 台より少ない場合は、サーバ 127.127.1.0 が各メンバの rc.config ファイルに追加される。

表 C-2 に、clu_create または clu_add_member が作成または変更する /etc/sysconfigtab 変数を示します。

表 C-2: `/etc/sysconfigtab` の属性

サブシステム	属性	コメント
clubase	<code>cluster_interconnect</code>	クラスタのノード間通信に使用するインターコネクトの種類。LAN インターコネクトを使用する場合は、 <code>clu_create</code> および <code>clu_add_member</code> コマンドにより、この属性が <code>tcp</code> に設定される。Memory Channel を使用する場合は <code>mct</code> に設定される。
	<code>cluster_name</code>	未修飾のクラスタ名。 <code>clu_create</code> および <code>clu_add_member</code> の両方により設定される。
	<code>cluster_expected_votes</code>	クラスタ・メンバが保持するすべてのポートの合計数に、(構成されていれば) クォーラム・ディスクに割り当てられたポート数を加算したもの。
	<code>cluster_node_inter_name</code>	このメンバのクラスタ・インターコネクト・インタフェースに関連付けられている IP 名。たとえば、 <code>polishham-ics0</code> 。インストレーション・プロシージャにより、この情報を入力するプロンプトが表示される。
	<code>cluster_node_name</code>	このメンバの未修飾ホスト名。
	<code>cluster_node_votes</code>	1 つのノードがクォーラムに占める固定ポート数。ゼロ以外の値を与えられたノードは、投票メンバとみなされる。
	<code>cluster_qdisk_major</code>	クォーラム・ディスク上の <code>h</code> パーティションの主デバイス番号。
	<code>cluster_qdisk_minor</code>	クォーラム・ディスク上の <code>h</code> パーティションの副デバイス番号。
	<code>cluster_qdisk_votes</code>	クォーラム・ディスクがクォーラムに占める固定ポート数。
	<code>cluster_seqdisk_major</code>	メンバのブート・ディスク上にある <code>h</code> パーティションの主デバイス番号。
	<code>cluster_seqdisk_minor</code>	メンバのブート・ディスク上にある <code>h</code> パーティションの副デバイス番号。

表 C-2: /etc/sysconfigtab の属性 (続き)

サブシステム	属性	コメント
drd	drd_nopr_list	デバイス要求ディスパッチャ・サブシステムが持続的な予約を実行しないデバイスに対するハードウェア ID リストの最初のエントリ。現在のメンバに対して設定される場合は、clu_add_member によって現在のメンバの値に設定される。
generic	act_vers_high	クラスタ内での変更の同期をとるため、アップグレード時に使用されるバージョン番号。変更しないこと。
	act_vers_low	クラスタ内での変更の同期をとるため、アップグレード時に使用されるバージョン番号。変更しないこと。
	memberid	このメンバのクラスタ・メンバ ID (memberid)。各メンバには固有の memberid (1 - 63) を付けなければならない。インストール・プロシージャにより、この属性の値を入力するプロンプトが表示される。
	msgbuf_size	16384 より小さい場合は、clu_create によって 16384 に設定される。clu_add_member によって現在のメンバの値に設定される。
	new_vers_high	クラスタ内での変更の同期をとるため、アップグレード時に使用されるバージョン番号。変更しないこと。
	new_vers_low	クラスタ内での変更の同期をとるため、アップグレード時に使用されるバージョン番号。変更しないこと。

表 C-2: /etc/sysconfigtab の属性 (続き)

サブシステム	属性	コメント
	rolls_ver_lookup	clu_create および clu_add_member によって、0 に 設定される。ローリング・アッ プグレードまたはパッチ中にタ グ付きファイルを使用するメン バについては、clu_upgrade コマンドによって 1 に設定さ れる。メンバがロールされると clu_upgrade によって 0 に再設 定される。変更しないこと。
ics_ll_tcp	ics_tcp_inetaddr0	clu_create と clu_add mem- ber によって物理クラスタ・イン ターコネクト・デバイスの IP ア ドレス (10.1.0.1 など) に設定され る (LAN インターコネクトのみ)。
	ics_tcp_netmask0	クラスタ・インターコネクトのサ ブネット・マスク。clu_create と clu_add_member によつて 255.255.255.0 に設定される (LAN インターコネクトのみ)。
	ics_tcp_adapter0	物理クラスタ・インターコ ネクト・デバイス名 (tu0 , nr0 など)。clu_create およ び clu_add_member によつ て設定される (LAN インター コネクトのみ)。
	ics_tcp_nr0	ics_tcp_adapter0 属性が NetRAIN セットを示している場 合、この属性は、そのセットを構 成するネットワーク・アダプタ のデバイス名を示す配列になる (LAN インターコネクトのみ)。
lsm	lsm_rootdev_is_volume	clu_create および clu_add_member によつて 0 に設定される。
sec	acl_mode	設定する場合は、現在のシス テムまたはメンバの値に設定す る。設定しない場合は、disable を設定する。clu_create お よび clu_add_member によつ て設定される。

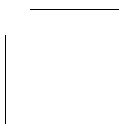
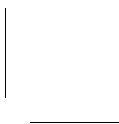
表 C-2: `/etc/sysconfigtab` の属性 (続き)

サブシステム	属性	コメント
vm	swapdevice	このメンバのスワップ・デバイス。インストール・プロセスにより、このメンバのブート・ディスクとしてユーザが指定したデバイスの b パーティションが、この属性の値として使用される。
	vm_page_free_reserved	値が 20 未満の場合、clu_create および clu_add_member によって 20 に設定される。
	vm_page_free_min	値が 30 未満の場合、clu_create および clu_add_member によって 30 に設定される。

表 C-3 に、clu_create または clu_add_member が作成または変更する `/etc/sysconfigtab.cluster` 変数を示します。

表 C-3: `/etc/sysconfigtab.cluster` の属性

サブシステム	属性	コメント
clubase	cluster_expected_votes	クラスタ・メンバが保持するすべてのポートの合計数に、(構成されていれば) クォーラム・ディスクに割り当てられたポート数を加算したもの。clu_create および clu_add_member により設定される。



D

インストールの例

この章では、次のコマンドにより書き込まれたログの例を示します。

- `clu_create` (D.1 節)
- `clu_add_member` (D.2 節)
- `clu_upgrade` (D.3 節)

D.1 `clu_create` のログ

`clu_create` を実行するたびに、`/cluster/admin/clu_create.log` にログ・メッセージが書き込まれます。例 D-1 に、Memory Channel インターコネクトを使うクラスタの `clu_create` のログ・ファイルの例を示します。例 D-2 に、LAN インターコネクトを使うクラスタの `clu_create` のログ・ファイルの例を示します。

例 D-1: Memory Channel インターコネクトに対する `clu_create` のログ・ファイルの例

```
Do you want to continue creating the cluster? [yes]: Return

Each cluster has a unique cluster name, which is a hostname
used to identify the entire cluster.

Enter a fully-qualified cluster name []: deli.zk3.dec.com
Checking cluster name: deli.zk3.dec.com.

You entered 'deli.zk3.dec.com' as your cluster name.
Is this correct? [yes]: Return

The cluster alias IP address is the IP address associated with the
default cluster alias. (192.168.168.1 is an example of an IP address.)

Enter the cluster alias IP address []: 16.140.160.124
Checking cluster alias IP address: 16.140.160.124

You entered '16.140.160.124' as the IP address for the default cluster alias.
Is this correct? [yes]: Return

The cluster root partition is the disk partition (for example, dsk4b)
that will hold the clusterwide root (/) file system.

Note: The default 'a' partition on most disks is not large
enough to hold the clusterwide root AdvFS domain.
```

例 D-1: Memory Channel インターコネクトに対する `clu_create` のログ・ファイルの例 (続き)

Enter the device name of the cluster root partition []:**dsk2g**
Checking the cluster root partition: dsk2g.

You entered 'dsk2g' as the device name of the cluster root partition.
Is this correct? [yes]: **Return**

The cluster usr partition is the disk partition (for example, dsk4g)
that will contain the clusterwide usr (/usr) file system.

Note: The default 'g' partition on most disks is usually
large enough to hold the clusterwide usr AdvFS domain.

Enter the device name of the cluster usr partition []:**dsk6c**
Checking the cluster usr partition: dsk6c.

You entered 'dsk6c' as the device name of the cluster usr partition.
Is this correct? [yes]: **Return**

The cluster var device is the disk partition (for example, dsk4h)
that will hold the clusterwide var (/var) file system.

Note: The default 'h' partition on most disks is usually
large enough to hold the clusterwide var AdvFS domain.

Enter the device name of the cluster var partition []:**dsk2h**
Checking the cluster var partition: dsk2h.

You entered 'dsk2h' as the device name of the cluster var partition.
Is this correct? [yes]: **Return**

Do you want to define a quorum disk device at this time? [yes]: **Return**
The quorum disk device is the name of the disk (for example, 'dsk5')
that will be used as this cluster quorum disk.

Enter the device name of the quorum disk []:**dsk3**
Checking the quorum disk device: dsk3.

You entered 'dsk3' as the device name of the quorum disk device.
Is this correct? [yes]: **Return**

By default the quorum disk is assigned '1' vote(s).
To use this default value, press Return at the prompt.

The number of votes for the quorum disk is an integer, usually 0 or 1.
If you select 0 votes then the quorum disk will not contribute votes to the
cluster. If you select 1 vote then the quorum disk must be accessible to
boot and run a single member cluster.

Enter the number of votes for the quorum disk [1]: **Return**
Checking number of votes for the quorum disk: 1.

You entered '1' as the number votes for the quorum disk.
Is this correct? [yes]: **Return**

The default member ID for the first cluster member is '1'.
To use this default value, press Return at the prompt.

A member ID is used to identify each member in a cluster.

D-2 インストレーションの例

例 D-1: Memory Channel インターコネクトに対する `clu_create` のログ・ファイルの例 (続き)

Each member must have a unique member ID, which is an integer in the range 1-63, inclusive.

Enter a cluster member ID [1]: `Return`
Checking cluster member ID: 1.

You entered '1' as the member ID.
Is this correct? [yes]: `Return`

By default the 1st member of a cluster is assigned '1' vote(s).
Checking number of votes for this member: 1.

Each member has its own boot disk, which has an associated device name; for example, 'dsk5'.

Enter the device name of the member boot disk []:`dsk4`
Checking the member boot disk: dsk4.

You entered 'dsk4' as the device name of this member's boot disk.
Is this correct? [yes]: `Return`

Device 'ics0' is the default virtual cluster interconnect device.
Checking virtual cluster interconnect device: ics0.

The virtual cluster interconnect IP name 'swiss-ics0' was formed by appending '-ics0' to the system's hostname.
To use this default value, press Return at the prompt.

Each virtual cluster interconnect interface has a unique IP name (a hostname) associated with it.

Enter the IP name for the virtual cluster interconnect [swiss-ics0]: `Return`
Checking virtual cluster interconnect IP name: swiss-ics0.

You entered 'swiss-ics0' as the IP name for the virtual cluster interconnect.
Is this name correct? [yes]: `Return`

The virtual cluster interconnect IP address '10.0.0.1' was created by replacing the last byte of the default virtual cluster interconnect network address '10.0.0.0' with the previously chosen member ID '1'.
To use this default value, press Return at the prompt.

The virtual cluster interconnect IP address is the IP address associated with the virtual cluster interconnect IP name. (192.168.168.1 is an example of an IP address.)

Enter the IP address for the virtual cluster interconnect [10.0.0.1]: `Return`
Checking virtual cluster interconnect IP address: 10.0.0.1.

You entered '10.0.0.1' as the IP address for the virtual cluster interconnect.
Is this address correct? [yes]: `Return`

What type of cluster interconnect will you be using?

Selection	Type of Interconnect

1	Memory Channel
2	Local Area Network
3	None of the above

例 D-1: Memory Channel インターコネクトに対する clu_create のログ・ファイルの例 (続き)

```

4      Help
5      Display all options again
-----
Enter your choice [1]:4
A cluster must have a dedicated cluster interconnect to which all
members are connected. The cluster interconnect serves as the primary
communications channel between cluster members. For hardware, the cluster
interconnect can use either Memory Channel or a private LAN. For
more information about cluster interconnect hardware, see the
Cluster Hardware Configuration manual.

What type of cluster interconnect will you be using?

      Selection  Type of Interconnect
-----
1      Memory Channel
2      Local Area Network
3      None of the above
4      Help
5      Display all options again
-----
Enter your choice [1]: Return
You selected option '1' for the cluster interconnect.
Is that correct? (y/n) [y]: Return

Device 'mc0' is the default physical cluster interconnect interface device.
Checking physical cluster interconnect interface device name(s): mc0.

You entered the following information:

Cluster name:                                deli.zk3.dec.com
Cluster alias IP Address:                    16.140.160.124
Clusterwide root partition:                  dsk2g
Clusterwide usr partition:                   dsk6c
Clusterwide var partition:                   dsk2h
Clusterwide il8n partition:                  Directory-In-/usr
Quorum disk device:                          dsk3
Number of votes assigned to the quorum disk: 1
First member's member ID:                    1
Number of votes assigned to this member:     1
First member's boot disk:                    dsk4
First member's virtual cluster interconnect device name: ics0
First member's virtual cluster interconnect IP name:      swiss-ics0
First member's virtual cluster interconnect IP address:   10.0.0.1
First member's physical cluster interconnect devices:     mc0
First member's NetRAIN device name:                     Not-Applicable
First member's physical cluster interconnect IP address:  Not-Applicable

If you want to change any of the above information, answer 'n' to the
following prompt. You will then be given an opportunity to change your
selections.
Do you want to continue to create the cluster? [yes]: Return

Creating required disk labels.
Creating disk label on member disk: dsk4.
Initializing cnx partition on member disk: dsk4h.
Creating disk label on quorum disk: dsk3.
```

D-4 インストレーションの例

例 D-1: Memory Channel インターコネクトに対する clu_create のログ・ファイルの例 (続き)

```
Initializing cnx partition on quorum disk: dsk3h.

Creating AdvFS domains:
  Creating AdvFS domain 'root1_domain#root' on partition '/dev/disk/dsk4a'.
  Creating AdvFS domain 'cluster_root#root' on partition '/dev/disk/dsk2g'.
  Creating AdvFS domain 'cluster_usr#usr' on partition '/dev/disk/dsk6c'.
  Creating AdvFS domain 'cluster_var#var' on partition '/dev/disk/dsk2h'.

Populating clusterwide root, usr, and var file systems:
  Copying root file system to 'cluster_root#root'.
  ....
  Copying usr file system to 'cluster_usr#usr'.
  .....
  Copying var file system to 'cluster_var#var'.
  ...

Creating Context Dependent Symbolic Links (CDSLs) for file systems:
  Creating CDSLs in root file system.
  Creating CDSLs in usr file system.
  Creating CDSLs in var file system.
  Creating links between clusterwide file systems.

Populating member's root file system.

Modifying configuration files required for cluster operation:
  Creating /etc/fstab file.
  Configuring cluster alias.
  Updating /etc/hosts - adding IP address '16.140.160.124' and hostname 'deli.zk3.dec.com'.
  Updating member-specific /etc/inittab file with 'cms' entry.
  Updating /etc/hosts - adding IP address '10.0.0.1' and hostname 'swiss-ics0'.
  Updating /etc/rc.config file.
  Updating /etc/sysconfigtab file.
  Retrieving cluster_root major and minor device numbers.
Warning: The following sysconfig variables are being set:
kmem_debug=0xe
kmem_audit_count=5000
lockmode=4
rt_preempt_opt=1
  Creating cluster device file CDSLs.
  Updating /.rhosts - adding hostname 'deli.zk3.dec.com'.
  Updating /etc/hosts.equiv - adding hostname 'deli.zk3.dec.com'.
  Updating /.rhosts - adding hostname 'swiss-ics0'.
  Updating /etc/hosts.equiv - adding hostname 'swiss-ics0'.
  Updating /etc/ifaccess.conf - adding deny entry for 'sl0'.
  Updating /etc/ifaccess.conf - adding deny entry for 'tu0'.
  Updating /etc/ifaccess.conf - adding deny entry for 'tun0'.
  Updating /etc/ifaccess.conf - adding deny entry for 'tun624'.
  Updating /etc/cfgmgr.auth - adding hostname 'swiss.zk3.dec.com'.
  Finished updating member1-specific area.

Building a kernel for this member.
  Saving kernel build configuration.
  The kernel will now be configured using the doconfig program.

*** Warning ***
  File in /usr/sys/BINARY found as a file, expected symlink: GENERIC.mod.

*** Warning ***
  File in /usr/sys/BINARY found as a file, expected symlink: GENERIC_EXTRAS.mod.
```

例 D-1: Memory Channel インターコネクトに対する clu_create のログ・ファイルの例 (続き)

```
*** KERNEL CONFIGURATION AND BUILD PROCEDURE ***

Saving /sys/conf/SWISS as /sys/conf/SWISS.bck

*** PERFORMING KERNEL BUILD ***
Working...Wed Feb 20 14:37:35 EST 2002
Working...Wed Feb 20 14:39:40 EST 2002

The new kernel is /sys/SWISS/vmunix
Finished running the doconfig program.

The kernel build was successful and the new kernel
has been copied to this member's boot disk.
Restoring kernel build configuration.

Updating console variables.
Setting console variable 'bootdef_dev' to dsk4.
Setting console variable 'boot_dev' to dsk4.
Setting console variable 'boot_reset' to ON.
Saving console variables to non-volatile storage.

clu_create: Cluster created successfully.

To run this system as a single member cluster it must be rebooted.
If you answer yes to the following question clu_create will reboot the
system for you now. If you answer no, you must manually reboot the
system after clu_create exits.
Would you like clu_create to reboot this system now? [yes]: 
Shutdown at 14:46 (in 0 minutes) [pid 26664]
```

例 D-2: LAN インターコネクトに対する clu_create のログ・ファイルの例

```
Do you want to continue creating the cluster? [yes]: 

*** Info ***
Memory Channel hardware not found in system; continuing ...

Each cluster has a unique cluster name, which is a hostname
used to identify the entire cluster.

Enter a fully-qualified cluster name []: deli.zk3.dec.com
Checking cluster name: deli.zk3.dec.com.

You entered 'deli.zk3.dec.com' as your cluster name.
Is this correct? [yes]: 

The cluster alias IP address is the IP address associated with the
default cluster alias. (192.168.168.1 is an example of an IP address.)

Enter the cluster alias IP address []: 16.140.112.209
Checking cluster alias IP address: 16.140.112.209

You entered '16.140.112.209' as the IP address for the default cluster alias.
```

D-6 インストレーションの例

例 D-2: LAN インターコネクトに対する `clu_create` のログ・ファイルの例 (続き)

Is this correct? [yes]:

The cluster root partition is the disk partition (for example, dsk4b) that will hold the clusterwide root (/) file system.

The cluster root partition is the disk partition (for example, dsk4b) that will hold the clusterwide root (/) file system.

Note: The default 'a' partition on most disks is not large enough to hold the clusterwide root AdvFS domain.

Enter the device name of the cluster root partition []:**dsk7b**
Checking the cluster root partition: dsk7b.

You entered 'dsk7b' as the device name of the cluster root partition.
Is this correct? [yes]:

The cluster usr partition is the disk partition (for example, dsk4g) that will contain the clusterwide usr (/usr) file system.

Note: The default 'g' partition on most disks is usually large enough to hold the clusterwide usr AdvFS domain.

Enter the device name of the cluster usr partition []:**dsk7g**
Checking the cluster usr partition: dsk7g.

You entered 'dsk7g' as the device name of the cluster usr partition.
Is this correct? [yes]:

The cluster var device is the disk partition (for example, dsk4h) that will hold the clusterwide var (/var) file system.

Note: The default 'h' partition on most disks is usually large enough to hold the clusterwide var AdvFS domain.

Enter the device name of the cluster var partition []:**dsk7h**
Checking the cluster var partition: dsk7h.

You entered 'dsk7h' as the device name of the cluster var partition.
Is this correct? [yes]:

Do you want to define a quorum disk device at this time? [yes]:
The quorum disk device is the name of the disk (for example, 'dsk5') that will be used as this cluster quorum disk.

Enter the device name of the quorum disk []:**dsk6**
Checking the quorum disk device: dsk6.

You entered 'dsk6' as the device name of the quorum disk device.
Is this correct? [yes]:

By default the quorum disk is assigned '1' vote(s).
To use this default value, press Return at the prompt.

The number of votes for the quorum disk is an integer, usually 0 or 1.
If you select 0 votes then the quorum disk will not contribute votes to the cluster. If you select 1 vote then the quorum disk must be accessible to boot and run a single member cluster.

例 D-2: LAN インターコネクトに対する `clu_create` のログ・ファイルの例 (続き)

```
Enter the number of votes for the quorum disk [1]: 
Checking number of votes for the quorum disk: 1.

You entered '1' as the number votes for the quorum disk.
Is this correct? [yes]: 

The default member ID for the first cluster member is '1'.
To use this default value, press Return at the prompt.

A member ID is used to identify each member in a cluster.
Each member must have a unique member ID, which is an integer in
the range 1-63, inclusive.

Enter a cluster member ID [10]: 
Checking cluster member ID: 10.

You entered '10' as the member ID.
Is this correct? [yes]: 

By default the 1st member of a cluster is assigned '1' vote(s).
Checking number of votes for this member: 1.

Each member has its own boot disk, which has an associated
device name; for example, 'dsk5'.

Enter the device name of the member boot disk []: dsk2
Checking the member boot disk: dsk2.

You entered 'dsk2' as the device name of this member's boot disk.
Is this correct? [yes]: 

Device 'ics0' is the default virtual cluster interconnect device.
Checking virtual cluster interconnect device: ics0.

The virtual cluster interconnect IP name 'pepicelli-ics0' was formed by
appending '-ics0' to the system's hostname.
To use this default value, press Return at the prompt.

Each virtual cluster interconnect interface has a unique IP name (a
hostname) associated with it.

Enter the IP name for the virtual cluster interconnect [pepicelli-ics0]: 
Checking virtual cluster interconnect IP name: pepicelli-ics0.

You entered 'pepicelli-ics0' as the IP name for the virtual cluster interconnect.
Is this name correct? [yes]: 

The virtual cluster interconnect IP address '10.0.0.10' was created by
replacing the last byte of the default virtual cluster interconnect network
address '10.0.0.0' with the previously chosen member ID '10'.
To use this default value, press Return at the prompt.

The virtual cluster interconnect IP address is the IP address
associated with the virtual cluster interconnect IP name. (192.168.168.1
is an example of an IP address.)

Enter the IP address for the virtual cluster interconnect [10.0.0.10]: 
Checking virtual cluster interconnect IP address: 10.0.0.10.
```

D-8 インストレーションの例

例 D-2: LAN インターコネクトに対する `clu_create` のログ・ファイルの例 (続き)

You entered '10.0.0.10' as the IP address for the virtual cluster interconnect.
Is this address correct? [yes]: **Return**

The physical cluster interconnect interface device is the name of the physical device(s) that will be used for low level cluster node communications. Examples of the physical cluster interconnect interface device name are: tu0, ee0, and nr0.

Enter the physical cluster interconnect device name(s) []:**alt0**
Would you like to place this Ethernet device into a NetRAIN set? [yes]:**n**
Checking physical cluster interconnect interface device name(s): alt0.

You entered 'alt0' as your physical cluster interconnect interface device name(s). Is this correct? [yes]: **Return**

The physical cluster interconnect IP name 'member10-icstcp0' was formed by appending '-icstcp0' to the word 'member' and the member ID.
Checking physical cluster interconnect IP name: member10-icstcp0.

The physical cluster interconnect IP address '10.1.0.10' was created by replacing the last byte of the default cluster interconnect network address '10.1.0.0' with the previously chosen member ID '10'.
To use this default value, press Return at the prompt.

The cluster physical interconnect IP address is the IP address associated with the physical cluster interconnect IP name. (192.168.168.1 is an example of an IP address.)

Enter the IP address for the physical cluster interconnect [10.1.0.10]: **Return**
Checking physical cluster interconnect IP address: 10.1.0.10.

You entered '10.1.0.10' as the IP address for the physical cluster interconnect.
Is this address correct? [yes]: **Return**

You entered the following information:

Cluster name:	deli.zk3.dec.com
Cluster alias IP Address:	16.140.112.209
Clusterwide root partition:	dsk7b
Clusterwide usr partition:	dsk7g
Clusterwide var partition:	dsk7h
Clusterwide il8n partition:	Directory-In-/usr
Quorum disk device:	dsk6
Number of votes assigned to the quorum disk:	1
First member's member ID:	10
Number of votes assigned to this member:	1
First member's boot disk:	dsk2
First member's virtual cluster interconnect device name:	ics0
First member's virtual cluster interconnect IP name:	pepicelli-ics0
First member's virtual cluster interconnect IP address:	10.0.0.10
First member's physical cluster interconnect devices:	alt0
First member's NetRAIN device name:	Not-Applicable
First member's physical cluster interconnect IP address:	10.1.0.10

If you want to change any of the above information, answer 'n' to the following prompt. You will then be given an opportunity to change your selections.

Do you want to continue to create the cluster? [yes]: **Return**

例 D-2: LAN インターコネクトに対する clu_create のログ・ファイルの例 (続き)

```
*** Info ***
Beginning configuration of initial cluster member.

Configuring Disks:
Creating required disk labels.
  Creating disk label on member disk: dsk2.
  Initializing cnx partition on member disk: dsk2h.
  Creating disk label on quorum disk: dsk6.
  Initializing cnx partition on quorum disk: dsk6h.
Configuring AdvFS Domains:
Creating AdvFS domains:
  Creating AdvFS domain 'root10_domain#root' on partition '/dev/disk/dsk2a'.
  Creating AdvFS domain 'cluster_root#root' on partition '/dev/disk/dsk7b'.
  Creating AdvFS domain 'cluster_usr#usr' on partition '/dev/disk/dsk7g'.
  Creating AdvFS domain 'cluster_var#var' on partition '/dev/disk/dsk7h'.
Creating Clusterwide File System:
Populating clusterwide root, usr, and var file systems:
  Copying root file system to 'cluster_root#root'.
  Copying usr file system to 'cluster_usr#usr'.
  Copying var file system to 'cluster_var#var'.
Setting Up Member Area:
Creating Context Dependent Symbolic Links (CDSLs) for file systems:
  Creating CDSLs in root file system.
  Creating CDSLs in usr file system.
  Creating CDSLs in var file system.
  Creating links between clusterwide file systems.
Creating Member's Boot Disk:
Populating member's root file system.
Setting up cluster configuration files:
Modifying configuration files required for cluster operation:
  Creating /etc/fstab file.
  Configuring cluster alias.
  Updating /etc/hosts - adding IP address '16.140.112.209' and hostname 'deli.zk3.dec.com'.
  Updating member-specific /etc/inittab file with 'cms' entry.
  Updating /etc/hosts - adding IP address '10.0.0.10' and hostname 'pepicelli-ics0'.
  Updating /etc/hosts - adding IP address '10.1.0.10' and hostname 'member10-icstcp0'.
  Updating /etc/rc.config file.
  Updating /etc/sysconfigtab file.
  Retrieving cluster_root major and minor device numbers.
Warning: The following sysconfig variables are being set:
kmem_debug=0xe
kmem_audit_count=5000
lockmode=4
rt_preempt_opt=1
Creating cluster device file CDSLs.
Updating /.rhosts - adding hostname 'deli.zk3.dec.com'.
Updating /etc/hosts.equiv - adding hostname 'deli.zk3.dec.com'.
Updating /.rhosts - adding hostname 'pepicelli-ics0'.
Updating /etc/hosts.equiv - adding hostname 'pepicelli-ics0'.
Updating /.rhosts - adding hostname 'member10-icstcp0'.
Updating /etc/hosts.equiv - adding hostname 'member10-icstcp0'.
Updating /etc/ifaccess.conf - adding deny entry for 'ee0'.
Updating /etc/ifaccess.conf - adding deny entry for 'sl0'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu0'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu1'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu2'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu3'.
Updating /etc/ifaccess.conf - adding deny entry for 'tun0'.
```

例 D-2: LAN インターコネクトに対する `clu_create` のログ・ファイルの例 (続き)

```
Updating /etc/ifaccess.conf - adding deny entry for 'tun624'.
Updating /etc/ifaccess.conf - adding deny entry for 'ee0'.
Updating /etc/ifaccess.conf - adding deny entry for 'sl0'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu0'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu1'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu2'.
Updating /etc/ifaccess.conf - adding deny entry for 'tu3'.
Updating /etc/ifaccess.conf - adding deny entry for 'tun0'.
Updating /etc/ifaccess.conf - adding deny entry for 'tun624'.
Finished updating member10-specific area.

Building a kernel for this member.
Saving kernel build configuration.
The kernel will now be configured using the doconfig program.

*** KERNEL CONFIGURATION AND BUILD PROCEDURE ***

Saving /sys/conf/PEPICELLI as /sys/conf/PEPICELLI.bck

*** PERFORMING KERNEL BUILD ***
Working...Fri Feb 15 10:51:37 EST 2002

The new kernel is /sys/PEPICELLI/vmunix
Finished running the doconfig program.

The kernel build was successful and the new kernel
has been copied to this member's boot disk.
Restoring kernel build configuration.

Updating console variables.
Setting console variable 'bootdef_dev' to dsk2.
Setting console variable 'boot_dev' to dsk2.
Setting console variable 'boot_reset' to ON.
Saving console variables to non-volatile storage.

Cluster created successfully.
Cluster log created in /cluster/admin/clu_create.log

To run this system as a single member cluster it must be rebooted.
If you answer yes to the following question clu_create will reboot the
system for you now. If you answer no, you must manually reboot the
system after clu_create exits.
Would you like clu_create to reboot this system now? [yes]: 
Shutdown at 10:52 (in 0 minutes) [pid 6011]
```

D.2 `clu_add_member` の口グ

`clu_add_member` を実行するたびに、`/cluster/admin/clu_add_member.log` にログ・メッセージが書き込まれます。例 D-3 に、`clu_add_member` のログ・ファイルの例を示します。

例 D-3: clu_add_member のログ・ファイルの例

```
Do you want to continue adding this member? [yes]: Return

Each cluster member has a hostname, which is assigned to the HOSTNAME
variable in /etc/rc.config.

Enter the new member's fully qualified hostname []: polishham.zk3.dec.com
Checking member's hostname: polishham.zk3.dec.com

You entered 'polishham.zk3.dec.com' as this member's hostname.
Is this name correct? [yes]: Return

The next available member ID for a cluster member is '2'.
To use this default value, press Return at the prompt.

A member ID is used to identify each member in a cluster.
Each member must have a unique member ID, which is an integer in
the range 1-63, inclusive.

Enter a cluster member ID [2]: Return
Checking cluster member ID: 2

You entered '2' as the member ID.
Is this correct? [yes]: Return

By default, when the current cluster's expected votes are greater than 1,
each added member is assigned 1 vote(s). Otherwise, each added member is
assigned 0 (zero) votes.
To use this default value, press Return at the prompt.

The number of votes for a member is an integer usually 0 or 1
Enter the number of votes for this member [1]: Return
Checking number of votes for this member: 1

You entered '1' as the number votes for this member.
Is this correct? [yes]: Return

Each member has its own boot disk, which has an associated
device name; for example, 'dsk5'.

Enter the device name of the member boot disk []: dsk12
Checking the member boot disk: dsk12

You entered 'dsk12' as the device name of this member's boot disk.
Is this correct? [yes]: Return

Device 'ics0' is the default virtual cluster interconnect device
Checking virtual cluster interconnect device: ics0

The virtual cluster interconnect IP name 'polishham-ics0' was formed by
appending '-ics0' to the system's hostname.
To use this default value, press Return at the prompt.

Each virtual cluster interconnect interface has a unique IP name (a
hostname) associated with it.

Enter the IP name for the virtual cluster interconnect [polishham-ics0]: Return
Checking virtual cluster interconnect IP name: polishham-ics0

You entered 'polishham-ics0' as the IP name for the virtual cluster interconnect.
Is this name correct? [yes]: Return

The virtual cluster interconnect IP address '10.0.0.2' was created by
```

D-12 インストレーションの例

例 D-3: clu_add_member のログ・ファイルの例 (続き)

replacing the last byte of the virtual cluster interconnect network address '10.0.0.0' with the previously chosen member ID '2'.
To use this default value, press Return at the prompt.

The virtual cluster interconnect IP address is the IP address associated with the virtual cluster interconnect IP name. (192.168.168.1 is an example of an IP address.)

Enter the IP address for the virtual cluster interconnect [10.0.0.2]: **Return**
Checking virtual cluster interconnect IP address: 10.0.0.2

You entered '10.0.0.2' as the IP address for the virtual cluster interconnect.
Is this address correct? [yes]: **Return**

Device 'mc0' is the default physical cluster interconnect interface device
To use this default value, press Return at the prompt.

The physical cluster interconnect interface device is the name of the physical device(s) which will be used for low level cluster node communications. Examples of the physical cluster interconnect interface device name are: tu0, ee0, and nr0.

Enter the physical cluster interconnect device name(s) [mc0]: **Return**
Checking physical cluster interconnect interface device name(s): mc0

You entered 'mc0' as your physical cluster interconnect interface device name(s). Is this correct? [yes]: **Return**

Each cluster member must have its own registered TruCluster Server license. The data required to register a new member is typically located on the License PAK certificate or it may have been previously placed on your system as a partial or complete license data file. If you are prepared to enter this license data at this time, clu_add_member can configure the new member to use this license data. If you do not have the license data at this time you can enter this data on the new member when it is up and running. Do you wish to register the TruCluster Server license for this new member at this time? [yes]: **no**

You entered the following information:

Member's hostname:	polishham.zk3.dec.com
Member's ID:	2
Number of votes assigned to this member:	1
Member's boot disk:	dsk12
Member's virtual cluster interconnect devices:	ics0
Member's virtual cluster interconnect IP name:	polishham-ics0
Member's virtual cluster interconnect IP address:	10.0.0.2
Member's physical cluster interconnect devices:	mc0
Member's NetRAIN device name:	Not-Applicable
Member's physical cluster interconnect IP address:	Not-Applicable
Member's cluster license:	Not Entered

If you want to change any of the above information answers 'n' to the following prompt. You will then be given an opportunity to change your selections.

Do you want to continue to add this member? [yes]: **Return**

Creating required disk labels.

Creating disk label on member disk : dsk12

Initializing cnx partition on member disk : dsk12h

例 D-3: clu_add_member のログ・ファイルの例 (続き)

```
Creating AdvFS domains:
  Creating AdvFS domain 'root2_domain#root' on partition '/dev/disk/dsk12a'.

Creating cluster member-specific files:
  Creating new member's root member-specific files
  Creating new member's usr  member-specific files
  Creating new member's var  member-specific files
  Creating new member's boot member-specific files

Modifying configuration files required for new member operation:
  Updating /etc/hosts - adding IP address '10.0.0.2' and hostname 'polishham-ics0'
  Updating /etc/rc.config
  Updating /etc/sysconfigtab
  Updating member-specific /etc/inittab file with 'cms' entry.
  Updating /etc/securettys - adding ptys entry
  Updating /.rhosts - adding hostname 'polishham-ics0'
  Updating /etc/hosts.equiv - adding hostname 'polishham-ics0'
  Updating /etc/cfgmgr.auth - adding hostname 'polishham.zk3.dec.com'
  Configuring cluster alias.
  Configuring Network Time Protocol for new member
  Adding interface 'pepicelli-ics0' as an NTP peer to member 'polishham.zk3.dec.com'
  Adding interface 'polishham-ics0' as an NTP peer to member 'pepicelli.zk3.dec.com'

Configuring automatic subset configuration and kernel build.

clu_add_member: Initial member 2 configuration completed successfully.
From the newly added member's console, perform the following steps to
complete the newly added member's configuration:

1. Set the console variable 'boot_osflags' to 'A'.
2. Identify the console name of the newly added member's boots device.

>>> show device

The newly added member's boot device has the following properties:

Manufacturer: DEC
Model: HSG80
Target: IDENTIFIER=4
Lun: UNKNOWN
Serial Number: SCSI-WWID:01000010:6000-1fe1-0006-3f10-0009-0270-0619-0005

Note: The SCSI bus number may differ when viewed from different members.

3. Boot the newly added member using genvmunix:

>>> boot -file genvmunix <new-member-boot-device>

During this initial boot the newly added member will:

o Configure each installed subset.

o Attempt to build and install a new kernel. If the system cannot
  build a kernel, it starts a shell where you can attempt to build
  a kernel manually. If the build succeeds, copy the new kernel to
  /vmunix. When you are finished exit the shell using ^D or 'exit'.

o The newly added member will attempt to set boot related console
  variables and continue to boot to multi-user mode.
```

例 D-3: clu_add_member のログ・ファイルの例 (続き)

- o After the newly added member boots you should setup your system default network interface using the appropriate system management command.

D.3 clu_upgrade のログ

ローリング・アップグレードを実行するたびに、clu_upgrade は /cluster/admin/clu_upgrade.log にログ・メッセージを書き込みます。ローリング・アップグレードが完了すると、clu_upgrade はログ・ファイルを /cluster/admin/clu_upgrade/history/release_version ディレクトリに移動します。例 D-4 に、バージョン 5.1A からバージョン 5.1B ヘクラストをローリング・アップグレードする場合の clu_upgrade ログ・ファイルの例を示します (このログはフォーマットを少し直して読みやすくしています)。

例 D-4: clu_upgrade のログ・ファイルの例

```
#####
clu_upgrade Command: clu_upgrade upgrade setup
On Host: pepicelli2.zk3.dec.com Invoked at: Tue Apr 30 08:41:15 EDT 2002
-----

This is the cluster upgrade program.
You have indicated that you want to perform the 'setup' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return

What type of rolling upgrade will be performed?

      Selection   Type of Upgrade
-----
      1           An upgrade using the installupdate command
      2           A patch using the dupatch command
      3           A new hardware delivery using the nhd_install command
      4           All of the above
      5           None of the above
      6           Help
      7           Display all options again
-----

Enter your Choices (for example, 1 2 2-3):1
You selected the following rolling upgrade options: 1
Is that correct? (y/n) [y]: Return

Enter the full pathname of the cluster kit mount \
point ['???']: /cdrom/TruCluster/kit
/
```

```
/cdrom/TruCluster/kit
```

'Tru64 UNIX TruCluster(TM) Server Software T5.1B-4 (Rev 639)'

```
Checking inventory and available disk space.
Marking stage 'setup' as 'started'.
Copying cluster kit '/cdrom/TruCluster/kit' to '/var/adm/update/TruClusterKit/'.
```

..... \

..... \

..... \

.....

The 'setup' stage of the upgrade has completed successfully.

```
This is the cluster upgrade program.
You have indicated that you want to perform the 'preinstall' stage of the
upgrade.
```

```

Checking tagged files.
.....\
.....\
.....\
.....\
.....\
.....\
Marking stage 'preinstall' as 'started'.

```

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

```
Backing up member-specific data for member: 1
.....
Marking stage 'preinstall' as 'completed'.
The cluster upgrade 'preinstall' stage has completed successfully.
On the lead member, perform the following steps before running
the installupdate command:

# shutdown -h now
>>> boot -fl s

When the system reaches single-user mode run the following commands:

# init s
# bcheckrc
# lmf reset

See the Tru64 UNIX Installation Guide for detailed information on using the
installupdate command.

The 'preinstall' stage of the upgrade has completed successfully.
-----
clu_upgrade Command: clu_upgrade upgrade preinstall
On Host: pepicelli2.zk3.dec.com Exited at: Tue Apr 30 11:41:15 EDT 2002
#####
clu_upgrade Command: clu_upgrade check install
On Host: pepicelli2.zk3.dec.com Invoked at: Tue Apr 30 12:33:47 EDT 2002
-----
Checking install...
The 'install' stage of cluster upgrade is ready to be run.
-----
clu_upgrade Command: clu_upgrade check install
On Host: pepicelli2.zk3.dec.com Exited at: Tue Apr 30 12:33:48 EDT 2002
#####
clu_upgrade Command: clu_upgrade boot
On Host: pepicelli2.zk3.dec.com Invoked at: Tue Apr 30 14:51:51 EDT 2002
-----
clu_upgrade Command: clu_upgrade boot
On Host: pepicelli2.zk3.dec.com Exited at: Tue Apr 30 14:51:51 EDT 2002
#####
clu_upgrade Command: clu_upgrade boot
On Host: pepicelli2.zk3.dec.com Invoked at: Tue Apr 30 15:12:43 EDT 2002
-----
clu_upgrade Command: clu_upgrade boot
On Host: pepicelli2.zk3.dec.com Exited at: Tue Apr 30 15:12:44 EDT 2002
#####
clu_upgrade Command: clu_upgrade upgrade postinstall
On Host: pepicelli2.zk3.dec.com Invoked at: Tue Apr 30 15:13:47 EDT 2002
-----

This is the cluster upgrade program.
You have indicated that you want to perform the 'postinstall' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
Marking stage 'postinstall' as 'started'.
Marking stage 'postinstall' as 'completed'.
```

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

```
The 'postinstall' stage of the upgrade has completed successfully.
-----
clu_upgrade Command: clu_upgrade upgrade postinstall
  On Host: pepicelli2.zk3.dec.com Exited at: Tue Apr 30 15:14:00 EDT 2002

USER SETTINGS:
-----
OPTION_KERNEL=y

*** END UPDATE INSTALLATION (Tue Apr 30 15:46:51 EDT 2002) ***

#####
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli1.zk3.dec.com Invoked at: Tue Apr 30 16:13:28 EDT 2002
-----
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli1.zk3.dec.com Exited at: Tue Apr 30 16:13:29 EDT 2002

USER SETTINGS:
-----
OPTION_KERNEL=y

*** END UPDATE INSTALLATION (Tue Apr 30 16:45:18 EDT 2002) ***

#####
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli3.zk3.dec.com Invoked at: Tue Apr 30 17:08:57 EDT 2002
-----
clu_upgrade Command: clu_upgrade upgrade roll
  On Host: pepicelli3.zk3.dec.com Invoked at: Tue Apr 30 16:28:03 EDT 2002
-----

This is the cluster upgrade program.
You have indicated that you want to perform the 'roll' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
Marking stage 'roll' as 'started'.

*** Info ***
This is the last member requiring a roll.

Backing up member-specific data for member: 3
.....

*** START UPDATE INSTALLATION (Tue Apr 30 16:32:50 EDT 2002) ***
  FLAGS:

Checking for installed supplemental hardware support...

Completed check for installed supplemental hardware support
```

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

```
Checking for retired hardware...done.

Initializing new version information (OSF)...done
Initializing new version information (TCR)...done
Initializing the list of member specific files for member3...done


Update Installation has detected the following update installable
products on your system:

    Tru64 UNIX V5.1A Operating System ( Rev 1885 )
    Tru64 UNIX TruCluster(TM) Server Software V5.1A (Rev 1312)


These products will be updated to the following versions:

    Tru64 UNIX T5.1B-4 Operating System ( Rev 459 )
    Tru64 UNIX TruCluster(TM) Server Software T5.1B-4 (Rev 639)


It is recommended that you update your system firmware and perform a
complete system backup before proceeding.  A log of this update
installation can be found at /var/adm/smlogs/update.log.


Do you want to continue the Update Installation? (y/n) []: y
Do you want to select optional kernel components? (y/n) [n]: Return
Do you want to archive obsolete files? (y/n) [n]: Return


*** Checking for conflicting software ***


The following software may require reinstallation after the Update
Installation is completed:

    Advanced Server for UNIX V5.1A ECO3
    Legato NetWorker


Do you want to continue the Update Installation? (y/n) [y]: Return
*** Determining installed Operating System software ***


Working...Tue Apr 30 16:36:34 EDT 2002


*** Determining installed Tru64 UNIX TruCluster(TM) Server \
    Software V5.1A (Rev 1312) software ***


*** Determining kernel components ***


*** KERNEL OPTION SELECTION ***


    Selection    Kernel Option
-----
1 System V Devices
2 NTP V3 Kernel Phase Lock Loop (NTP_TIME)
3 Kernel Breakpoint Debugger (KDEBUG)
4 Packetfilter driver (PACKETFILTER)
```

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

```
5 IP Gateway Screening Facility (GWSCREEN)
6 IP-in-IP Tunneling (IPTUNNEL)
7 IP Version 6 (IPV6)
8 Point-to-Point Protocol (PPP)
9 STREAMS pckt module (PCKT)
10 X/Open Transport Interface (XTISO, TIMOD, TIRDWR)
11 Digital Versatile Disk File System (DVDFS)
12 ISO 9660 Compact Disc File System (CDFS)
13 Audit Subsystem
14 ATM UNI 3.0/3.1 ILMI (ATMILMI3X)
15 IP Switching over ATM (ATMIFMP)
16 LAN Emulation over ATM (LANE)
17 Classical IP over ATM (ATMIP)
--- MORE TO FOLLOW ---
Enter your choices or press <Return>
to display the next screen.
```

```
Choices (for example, 1 2 4-6):
 18 ATM UNI 3.0/3.1 Signalling for SVCs (UNI3X)
 19 Asynchronous Transfer Mode (ATM)
 20 All of the above
 21 None of the above
 22 Help
 23 Display all options again
-----
```

Enter your choices.

```
Choices (for example, 1 2 4-6) [21]:20
You selected the following kernel options:
System V Devices
NTP V3 Kernel Phase Lock Loop (NTP_TIME)
Kernel Breakpoint Debugger (KDEBUG)
Packetfilter driver (PACKETFILTER)
IP Gateway Screening Facility (GWSCREEN)
IP-in-IP Tunneling (IPTUNNEL)
IP Version 6 (IPV6)
Point-to-Point Protocol (PPP)
STREAMS pckt module (PCKT)
X/Open Transport Interface (XTISO, TIMOD, TIRDWR)
Digital Versatile Disk File System (DVDFS)
ISO 9660 Compact Disc File System (CDFS)
Audit Subsystem
ATM UNI 3.0/3.1 ILMI (ATMILMI3X)
IP Switching over ATM (ATMIFMP)
LAN Emulation over ATM (LANE)
Classical IP over ATM (ATMIP)
ATM UNI 3.0/3.1 Signalling for SVCs (UNI3X)
Asynchronous Transfer Mode (ATM)
```

```
Is that correct? (y/n) [y]: Return
*** Checking for file type conflicts ***
```

```
Working...Tue Apr 30 16:38:16 EDT 2002
```

```
*** Checking for obsolete files ***
```

```
*** Checking file system space ***
```


例 D-4: clu_upgrade のログ・ファイルの例 (続き)

Update Installation is now ready to begin modifying the files necessary to reboot the cluster member off of the new OS. Please check the /var/adm/smlogs/update.log and /var/adm/smlogs/it.log files for errors after the installation is complete.

Do you want to continue the Update Installation? (y/n) [n]:

*** Starting configuration merges for Update Install ***

*** Merging new file ./etc/.new..sysconfigtab into
existing ./etc/./cluster/members/member3/boot_partition/etc/sysconfigtab

Merging ./etc/./cluster/members/member3/boot_partition/etc/sysconfigtab
Merge completed successfully.

The critical files needed for reboot have been moved into place. The system will now reboot with the generic kernel for Compaq Computer Corporation Tru64 UNIX T5.1B-4 and complete the rolling upgrade for this member (member3).

clubase: Entry not found in /cluster/admin/tmp/stanza.stdin.1585508

The 'roll' stage has completed successfully. This member must be rebooted in order to run with the newly installed software. Do you want to reboot this member at this time? []:yes
You indicated that you want to reboot this member at this time. Is that correct? [yes]:

The 'roll' stage of the upgrade has completed successfully.
kill: 1573021: no such process

clu_upgrade Command: clu_upgrade upgrade roll
On Host: pepicelli3.zk3.dec.com Exited at: Tue Apr 30 16:45:52 EDT 2002

clu_upgrade Command: clu_upgrade upgrade roll
On Host: pepicelli1.zk3.dec.com Invoked at: Tue Apr 30 15:27:32 EDT 2002

This is the cluster upgrade program.
You have indicated that you want to perform the 'roll' stage of the upgrade.

Do you want to continue to upgrade the cluster? [yes]:

*** Warning ***

The cluster upgrade command was unable to find or verify the configuration file used to build this member's kernel. clu_upgrade attempts to make a backup copy of the configuration file which it would restore as required during a clu_upgrade undo command. To use the default configuration file or to continue without backing up a configuration file hit return.
Enter the name of the configuration file for this member [WAVY1]:
Marking stage 'roll' as 'started'.
Marking stage 'roll' as 'started'.

*** Info ***

The current quorum conditions indicate that beginning a roll of another member at this time may result in the loss of quorum.

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

Mostly likely the limiting factor is the number of currently rolling members that also contribute member votes.

You may attempt to run the clu_upgrade roll command on a member that is not contributing member votes to the cluster.

You may also use the clu_quorum command to change to zero the votes of members that have not rolled.

If this problem persists use, the clu_upgrade verbose command to identify the members that are currently DOWN or rolling. Use the clu_quorum command to identify and change the current quorum vote configuration of the cluster.

Backing up member-specific data for member: 2
.....

*** START UPDATE INSTALLATION (Tue Apr 30 15:34:05 EDT 2002) ***
FLAGS:

Checking for installed supplemental hardware support...

Completed check for installed supplemental hardware support
Checking for retired hardware...done.

Initializing new version information (OSF)...done
Initializing new version information (TCR)...done
Initializing the list of member specific files for member2...done

Update Installation has detected the following update installable products on your system:

Tru64 UNIX V5.1A Operating System (Rev 1885)
Tru64 UNIX TruCluster(TM) Server Software V5.1A (Rev 1312)

These products will be updated to the following versions:

Tru64 UNIX T5.1B-4 Operating System (Rev 459)
Tru64 UNIX TruCluster(TM) Server Software T5.1B-4 (Rev 639)

It is recommended that you update your system firmware and perform a complete system backup before proceeding. A log of this update installation can be found at /var/adm/smlogs/update.log.

Do you want to continue the Update Installation? (y/n) []: **y**
Do you want to select optional kernel components? (y/n) [n]: **Return**
Do you want to archive obsolete files? (y/n) [n]: **Return**

*** Checking for conflicting software ***

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

The following software may require reinstallation after the Update Installation is completed:

Advanced Server for UNIX V5.1A ECO3
Legato NetWorker

Do you want to continue the Update Installation? (y/n) [y]: Return

*** Determining installed Operating System software ***

Working...Tue Apr 30 15:37:34 EDT 2002

*** Determining installed Tru64 UNIX TruCluster(TM) Server \ Software V5.1A (Rev 1312) software ***

*** Determining kernel components ***

*** KERNEL OPTION SELECTION ***

Selection	Kernel Option
-----------	---------------

- | | |
|----|---|
| 1 | System V Devices |
| 2 | NTP V3 Kernel Phase Lock Loop (NTP_TIME) |
| 3 | Kernel Breakpoint Debugger (KDEBUG) |
| 4 | Packetfilter driver (PACKETFILTER) |
| 5 | IP Gateway Screening Facility (GWSCREEN) |
| 6 | IP-in-IP Tunneling (IPTUNNEL) |
| 7 | IP Version 6 (IPV6) |
| 8 | Point-to-Point Protocol (PPP) |
| 9 | STREAMS pkt module (PCKT) |
| 10 | X/Open Transport Interface (XTISO, TIMOD, TIRDWR) |
| 11 | Digital Versatile Disk File System (DVDFS) |
| 12 | ISO 9660 Compact Disc File System (CDFS) |
| 13 | Audit Subsystem |
| 14 | ATM UNI 3.0/3.1 ILMI (ATMILMI3X) |
| 15 | IP Switching over ATM (ATMIFMP) |
| 16 | LAN Emulation over ATM (LANE) |
| 17 | Classical IP over ATM (ATMIP) |

--- MORE TO FOLLOW ---

Enter your choices or press <Return> to display the next screen.

Choices (for example, 1 2 4-6):

- 18 ATM UNI 3.0/3.1 Signalling for SVCs (UNI3X)
- 19 Asynchronous Transfer Mode (ATM)
- 20 All of the above
- 21 None of the above
- 22 Help
- 23 Display all options again

Enter your choices.

Choices (for example, 1 2 4-6) [21]:20

You selected the following kernel options:

System V Devices
NTP V3 Kernel Phase Lock Loop (NTP_TIME)
Kernel Breakpoint Debugger (KDEBUG)

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

```
Packetfilter driver (PACKETFILTER)
IP Gateway Screening Facility (GWSCREEN)
IP-in-IP Tunneling (IPTUNNEL)
IP Version 6 (IPV6)
Point-to-Point Protocol (PPP)
STREAMS pckt module (PCKT)
X/Open Transport Interface (XTISO, TIMOD, TIRDWR)
Digital Versatile Disk File System (DVDFS)
ISO 9660 Compact Disc File System (CDFS)
Audit Subsystem
ATM UNI 3.0/3.1 ILMI (ATMILMI3X)
IP Switching over ATM (ATMIFMP)
LAN Emulation over ATM (LANE)
Classical IP over ATM (ATMIP)
ATM UNI 3.0/3.1 Signalling for SVCs (UNI3X)
Asynchronous Transfer Mode (ATM)

Is that correct? (y/n) [y]: Return
*** Checking for file type conflicts ***

Working....Tue Apr 30 15:39:23 EDT 2002

*** Checking for obsolete files ***

*** Checking file system space ***

Update Installation is now ready to begin modifying the files necessary
to reboot the cluster member off of the new OS. Please check the
/var/adm/smlogs/update.log and /var/adm/smlogs/it.log files for errors
after the installation is complete.

Do you want to continue the Update Installation? (y/n) [n]: Return

*** Starting configuration merges for Update Install ***

*** Merging new file ./etc/.new..sysconfigtab into
existing ./etc/./cluster/members/member2/boot_partition/etc/sysconfigtab

Merging /etc/./cluster/members/member2/boot_partition/etc/sysconfigtab
Merge completed successfully.

The critical files needed for reboot have been moved into place. The
system will now reboot with the generic kernel for Compaq Computer
Corporation Tru64 UNIX T5.1B-4 and complete the rolling upgrade for
this member (member2).

clubase: Entry not found in /cluster/admin/tmp/stanza.stdin.1061256

The 'roll' stage has completed successfully. This
member must be rebooted in order to run with the newly installed software.
Do you want to reboot this member at this time? []:y
You indicated that you want to reboot this member at this time.
Is that correct? [yes]: Return

The 'roll' stage of the upgrade has completed successfully.
```

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

```
kill: 1048734: no such process
-----
clu_upgrade Command: clu_upgrade upgrade roll
  On Host: pepicelli1.zk3.dec.com Exited at: Tue Apr 30 15:48:19 EDT 2002
Marking stage 'roll' as 'completed'.
-----
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli3.zk3.dec.com Exited at: Tue Apr 30 17:08:59 EDT 2002
#####
clu_upgrade Command: clu_upgrade upgrade switch
  On Host: pepicelli3.zk3.dec.com Invoked at: Tue Apr 30 17:09:37 EDT 2002
-----

This is the cluster upgrade program.
You have indicated that you want to perform the 'switch' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
Initiating version switch on cluster members
.Marking stage 'switch' as 'started'.
.Successful switch of the version identifiers

Marking stage 'switch' as 'completed'.
The cluster upgrade 'switch' stage has completed successfully.
All cluster members must be rebooted before running the 'clean' command.

The 'switch' stage of the upgrade has completed successfully.
-----
clu_upgrade Command: clu_upgrade upgrade switch
  On Host: pepicelli3.zk3.dec.com Exited at: Tue Apr 30 17:10:18 EDT 2002
#####
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli3.zk3.dec.com Invoked at: Tue Apr 30 17:20:33 EDT 2002
-----
Marking stage 'switch' as 'completed'.
-----
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli3.zk3.dec.com Exited at: Tue Apr 30 17:20:34 EDT 2002
#####
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli1.zk3.dec.com Invoked at: Tue Apr 30 17:29:08 EDT 2002
-----
Marking stage 'switch' as 'completed'.
-----
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli1.zk3.dec.com Exited at: Tue Apr 30 17:29:08 EDT 2002
#####
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli2.zk3.dec.com Invoked at: Tue Apr 30 17:42:44 EDT 2002
-----
Marking stage 'switch' as 'completed'.
-----
clu_upgrade Command: clu_upgrade boot
  On Host: pepicelli2.zk3.dec.com Exited at: Tue Apr 30 17:42:44 EDT 2002
#####
clu_upgrade Command: clu_upgrade upgrade clean
  On Host: pepicelli2.zk3.dec.com Invoked at: Wed May  1 05:30:02 EDT 2002
-----

This is the cluster upgrade program.
```

例 D-4: clu_upgrade のログ・ファイルの例 (続き)

You have indicated that you want to perform the 'clean' stage of the upgrade.

Do you want to continue to upgrade the cluster? [yes]:
.Marking stage 'clean' as 'started'.

Deleting tagged files.
.....\
.....\
.....\
.....Removing back-up and kit files
.....

The Update Administration Utility is typically run after an update installation to manage the files that are saved during an update installation.

Do you want to run the Update Administration Utility at this time? [yes]:

The Update Installation Cleanup utility is used to clean up backup files created by Update Installation. Update Installation can create two types of files: .PreUPD and .PreMRG. The .PreUPD files are copies of unprotected customized system files as they existed prior to running Update Installation. The .PreMRG files are copies of protected system files as they existed prior to running Update Installation.

Please make a selection from the following menu.

```
Update Installation Cleanup Main Menu
-----
c) Unprotected Customized File Administration (.PreUPD)
p) Pre-Merge File Administration (.PreMRG)
x) Exit This Utility
```

Enter your choice:*

Exiting /usr/sbin/updadmin...
Marking stage 'clean' as 'completed'.

The 'clean' stage of the upgrade has completed successfully.

clu_upgrade Command: clu_upgrade upgrade clean
On Host: pepicelli2.zk3.dec.com Exited at: Wed May 1 06:11:49 EDT 2002

E

システム・ログ・ファイルのクラスタ関連 メッセージ

以降の各節では、`/var/adm/syslog.dated/date` ディレクトリに格納される、次の `kern.log` ファイルの抜粋を示します。

- `clu_create` 実行後のスタートアップ・メッセージ (E.1 節)
- `clu_add_member` 実行後のスタートアップ・メッセージ (E.2 節)

これらのメッセージは、通常のクラスタ・スタートアップ操作を追跡したものです。クラスタの形成と回復の操作が正常に進行しているときのおおよその例として提供しています。また、クラスタに関連した問題を解決する際の出発点にもなります。

E.1 `clu_create` 実行後のスタートアップ・メッセージ

例 E-1 は、`clu_create` を実行した後、第 1 クラスタ・メンバ・システムをリブートしたときに表示されるスタートアップ・メッセージの一部分をコピーしたものです。この情報は、`/var/adm/syslog.dated/date/kern.log` にも記録されます。

例 E-1: `clu_create` 実行後のスタートアップ・メッセージ

```
Alpha boot: available memory from 0x1710000 to 0x7f58000
Compaq Tru64 UNIX T5.1B-4 (Rev. 2350); Tue Apr 23 16:10:43 EDT 2002
physical memory = 128.00 megabytes.
available memory = 104.27 megabytes.
using 413 buffers containing 3.22 megabytes of memory
Firmware revision: 5.8-10
PALcode: UNIX version 1.79-63
AlphaServer DS20 500 MHz
pci1 (primary bus:1) at nexus
pci2 (primary bus:1 subordinate bus:2) at pci1 slot 7
itpsa0 at pci2 slot 0
IntraServer ROM Version V2.0 (c)1998
scsi0 at itpsa0 slot 0 rad 0
itpsa1 at pci2 slot 1
IntraServer ROM Version V2.0 (c)1998
scsi1 at itpsa1 slot 0 rad 0
tu0: DECchip 21140: Revision: 2.2
tu0: auto negotiation capable device
tu0 at pci2 slot 2
```

例 E-1: clu_create 実行後のスタートアップ・メッセージ (続き)

```
tu0: DEC TULIP (10/100) Ethernet Interface, hardware address: 00-06-2B-00-FC-99
tu0: auto negotiation off: selecting 10BaseT (UTP) port: half duplex
pci0 (primary bus:0) at nexus
isa0 at pci0
gpc0 at isa0
gpc1 not probed
ace0 at isa0
ace1 at isa0
lp0 at isa0
fdi0 at isa0
fd0 at fdi0 unit 0
ata0 at pci0 slot 105 (slot 5, function 1)
ata0: CYPRESS 82C693
scsi2 at ata0 slot 0 rad 0
ata1 at pci0 slot 205 (slot 5, function 2)
ata1: CYPRESS 82C693
scsi3 at ata1 slot 0 rad 0
usb0 at pci0 slot 305 (slot 5, function 3)
comet0: Card type 'Elsa GLoria' with 8MB framebuffer memory.
comet0 at pci0 slot 7
emx0 at pci0 slot 8
KGPSA-CA : Driver Rev 2.03 : F/W Rev 3.03X2(1.11) : wwn 1000-0000-c924-4b7b
emx0: Using console topology setting of : Fabric
scsi4 at emx0 slot 0 rad 0
mchan0: Module revision = 34
mchan0: jumpered as HUB configuration
mchan0 at pci0 slot 9
Created FRU table binary error log packet
kernel console: comet0
i2c: Server Management Hardware Present
dli: configured
NetRAIN configured.
Random number generator configured.
ATM Subsystem configured with 1 restart threads
ATMUNI: configured
ATMSIG: 3.x (module=uni3x) configured
ILMI: 3.x (module=ilmi) configured
ATM IP: configured
ATM LANE: configured.
ATM IFMP: configured
TruCluster Server T5.1B-4 (Rev. 639); 04/11/02 04:20
Warning: subsystem xpc is neither new nor old format, not configured.
clubase: configured
Configuring RDG to use Memory Channel
ics_hl: Configuring memory channel as transport.
icsnet: configured
drd configured 0
kch: configured
dlm: configured
Starting CFS daemons
Registering CFS Services
Initializing CFSREC ICS Service
Registering CFSMSFS remote syscall interface
Registering CMS Services
TNC kproc_creator_daemon: Initialized and Ready
rm slave: mchan0, hubslot = 4, phys_rail 0 (size 512 MB)
rm slave: log_rail 0 (size 512 MB), phys_rail 0 (mchan0)
ics_mct: icsinfo set for node 1
ics_mct: Declaring this node up 1
CNX MGR: insufficient votes to form cluster: have 1 need 2
```

E-2 システム・ログ・ファイルのクラスタ関連メッセージ

例 E-1: clu_create 実行後のスタートアップ・メッセージ (続き)

```
ics_mct: icsinfo set for node 2
CNX MGR: Cluster deli incarnation 0x9e2b4 has been formed
CNX MGR: Founding node id is 2 csid is 0x10001
CNX MGR: membership configuration index: 1 (1 additions, 0 removals)
CNX MGR: quorum (re)gained, (re)starting cluster operations.
Joining versw kch set.
CNX MGR: Node swiss 1 incarn 0xe6e30 csid 0x10002 has been added to the cluster
ics_mct: Declaring this node up 2
CNX MGR: Node rye 2 incarn 0xcac0e csid 0x10001 has been added to the cluster
dlm: resuming lock activity
kch: resuming activity
scsil: SCSI Bus was reset
cam_logger: SCSI event packet
cam_logger: bus 1 target 255 lun 255
itpsa SCSI HBA
SCSI Bus was reset

scsil: SCSI Bus was reset
cam_logger: SCSI event packet
cam_logger: bus 1 target 255 lun 255
itpsa SCSI HBA
SCSI Bus was reset

scsil: SCSI Bus was reset
cam_logger: SCSI event packet
cam_logger: bus 1 target 255 lun 255
itpsa SCSI HBA
SCSI Bus was reset

scsil: SCSI Bus was reset
cam_logger: SCSI event packet
cam_logger: bus 1 target 255 lun 255
itpsa SCSI HBA
SCSI Bus was reset

clsm: checking for peer configurations
clsm: initialized
Waiting for cluster mount to complete
clsm: sent CNX partition data

vm_swap_init: swap is set to eager allocation mode
CMS: Joining deferred filesystem sets
CNX QDISK: Successfully claimed quorum disk, adding 1 vote.
cluster alias subsystem enabled
Environmental Monitoring Subsystem Configured.
```

TruCluster Server によって生成された情報メッセージとエラー・メッセージについては、『クラスタ管理ガイド』を参照してください。

E.2 clu_add_member 実行後のスタートアップ・メッセージ

例 E-2 は、clu_add_member を実行した後、第 2 クラスタ・メンバー・システムのブートを行ったときに表示されるスタートアップ

ブ・メッセージの一部分をコピーしたものです。この情報は、
/var/adm/syslog.dated/date/kern.log にも記録されます。

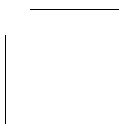
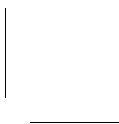
例 E-2: clu_add_member 実行後のスタートアップ・メッセージ

```
Alpha boot: available memory from 0x1ab8000 to 0xff58000
Compaq Tru64 UNIX T5.1B-4 (Rev. 2350); Tue Apr 23 16:10:07 EDT 2002
physical memory = 256.00 megabytes.
available memory = 228.61 megabytes.
using 905 buffers containing 7.07 megabytes of memory
Firmware revision: 5.8-10
PALcode: UNIX version 1.79-63
AlphaServer DS20 500 MHz
pci1 (primary bus:1) at nexus
pci2 (primary bus:1 subordinate bus:2) at pci1 slot 7
itpsa0 at pci2 slot 0
IntraServer ROM Version V2.0 (c)1998
scsi0 at itpsa0 slot 0 rad 0
itpsa1 at pci2 slot 1
IntraServer ROM Version V2.0 (c)1998
scsi1 at itpsa1 slot 0 rad 0
tu0: DECchip 21140: Revision: 2.2
tu0: auto negotiation capable device
tu0 at pci2 slot 2
tu0: DEC TULIP (10/100) Ethernet Interface, hardware address: 00-06-2B-01-1F-5A
tu0: auto negotiation off: selecting 10BaseT (UTP) port: half duplex
mchan0: Module revision = 34
mchan0: jumpered as HUB configuration
mchan0 at pci1 slot 8
pci0 (primary bus:0) at nexus
isa0 at pci0
gpc0 at isa0
gpc1 not probed
ace0 at isa0
ace1 at isa0
lp0 at isa0
fdi0 at isa0
fd0 at fdi0 unit 0
ata0 at pci0 slot 105 (slot 5, function 1)
ata0: CYPRESS 82C693
scsi2 at ata0 slot 0 rad 0
ata1 at pci0 slot 205 (slot 5, function 2)
ata1: CYPRESS 82C693
scsi3 at ata1 slot 0 rad 0
usb0 at pci0 slot 305 (slot 5, function 3)
comet0: Card type 'Elsa GLoria' with 8MB framebuffer memory.
comet0 at pci0 slot 7
emx0 at pci0 slot 9
KGPSA-CA : Driver Rev 2.03 : F/W Rev 3.02A1(1.11) : wwn 1000-0000-c922-4aac
emx0: Using console topology setting of : Fabric
scsi4 at emx0 slot 0 rad 0
Created FRU table binary error log packet
kernel console: comet0
i2c: Server Management Hardware Present
dli: configured
NetRAIN configured.
Random number generator configured.
ATM Subsystem configured with 1 restart threads
ATMUNI: configured
ATMSIG: 3.x (module=uni3x) configured
ILMI: 3.x (module=ilmi) configured
ATM IP: configured
ATM LANE: configured.
ATM IFMP: configured
TruCluster Server T5.1B-4 (Rev. 639); 04/11/02 04:20
Warning: subsystem xpc is neither new nor old format, not configured.
clubase: configured
Configuring RDG to use Memory Channel
ics_hl: Configuring memory channel as transport.
icsnet: configured
drd configured 0
kch: configured
```

E-4 システム・ログ・ファイルのクラスタ関連メッセージ

例 E-2: clu_add_member 実行後のスタートアップ・メッセージ (続き)

```
dlm: configured
Starting CFS daemons
Registering CFS Services
Initializing CFSREC ICS Service
Registering CFSMSFS remote syscall interface
Registering CMS Services
TNC kproc_creator_daemon: Initialized and Ready
rm primary: mchan0, hubslot = 2, phys_rail 0 (size 512 MB)
rm primary: log_rail 0 (size 512 MB), phys_rail 0 (mchan0)
ics_mct: icsinfo set for node 2
ics_mct: Declaring this node up 2
ics_mct: icsinfo set for node 1
CNX MGR: Cluster deli incarnation 0x9e2b4 has been formed
CNX MGR: Founding node id is 2 csid is 0x10001
CNX MGR: membership configuration index: 1 (1 additions, 0 removals)
CNX MGR: quorum (re)gained, (re)starting cluster operations.
CNX MGR: Node rye 2 incarn 0xcac0e csid 0x10001 has been added to the cluster
Joining versw kch set.
ics_mct: Declaring this node up 1
CNX MGR: Node swiss 1 incarn 0xe6e30 csid 0x10002 has been added to the cluster
dlm: resuming lock activity
kch: resuming activity
clsm: incoming CNX data: '...a'
clsm: checking for peer configurations
clsm: initialized
Waiting for cluster mount to complete
vm_swap_init: swap is set to eager allocation mode
CMS: Joining deferred filesystem sets
CNX QDISK: Successfully claimed quorum disk, adding 1 vote.
cluster alias subsystem enabled
Environmental Monitoring Subsystem Configured.
```



F

アップグレード時の手動によるストレージの構成

この付録では、TruCluster Software バージョン 1.5 またはバージョン 1.6 から TruCluster Server バージョン 5.1B にアップグレードしようとする際に、第 8 章で説明されているオプション 2 あるいはオプション 3 のストレージ・マッピングおよび構成スクリプトを使用することができない、あるいは選択することができない場合のサイトについて、アップグレードを行うための情報を説明します。

一般的に、第 8 章で説明されているプロシージャならびにスクリプトを使用することができる場合は、これらを使用することを推奨します。ストレージ・トポロジ、システム構成、サイト・ポリシ等の理由によりこれらを使用することができない場合は、この付録で説明する ASE (Available Server Environment) での手動によるストレージ構成情報の収集方法、ならびに新規 Tru64 UNIX システムあるいはシングル・メンバ・クラスタ上での手動によるストレージ構成方法を使用してアップグレードを行ってください。旧スタイルのデバイス名 (rz*) から新スタイルのデバイス名 (dsk*) へのマッピングはユーザの責任で行います。

ユーザは独自のアップグレード・プロシージャを作成しなければなりません。第 8 章ならびにこの付録の内容を理解した後、いずれのアップグレード・オプションが最適であるかを判断してください。続いて、そのオプション・プロシージャを修正するようにします。

F.1 デバイスならびにストレージ構成情報の手動による収集

この節では、現在の ASE 構成ならびにストレージ環境に関する情報を収集するために `clu_migrate_save` スクリプトを使用するオプション 2 およびオプション 3 に置き換わる手順について説明します。

まず、8.3.2 項を参照し、現在の ASE に対する最新の構成マップを作成します。

第 8 章では、`clu_migrate_save` スクリプトにより LSM および AdvFS 構成情報を含む現在の共用ストレージ情報を収集します。`clu_migrate_configure` スクリプトは収集された情報を読み取り、新規 Tru64 UNIX システムあるいはシングル・メンバ・クラスタ上のストレージを構成します。ただし、物理デバイスの接続後に新規クラスタ上のストレージを手動により構成しようとしている場合、`clu_migrate_configure` は実行されません。このため、ASE のメンバのストレージ構成情報を手動により収集しなければなりません。

注意

手動による情報収集に加えて、現在の ASE メンバ上で `clu_migrate_save` を実行することを推奨します。ストレージ構成情報を収集するだけでなく、`clu_migrate_save` は各ディスクの `rz*` スペシャル・ファイル名をディスクのラベルの `label:` フィールドに書き込みます (このスクリプトは、第 8 章で説明されているように元のディスク・ラベルを保存します。アップグレード後に、このラベルを復元することができます)。

また、新規 Tru64 UNIX システムあるいは TruCluster Server クラスタ上で `clu_migrate_configure -x` を実行することを推奨します。`clu_migrate_configure -x` コマンドはストレージを構成しませんが、`-x` オプションを指定しないで起動された場合に実行するコマンドの一覧を表示し、旧スタイルのデバイス名 (`rz*`) から新スタイルのデバイス名 (`dsk*`) へのマッピングを表示します。ただし、`clu_migrate_configure -x` がこのマッピングを行うために、ASE メンバ上で `clu_migrate_save` を実行しておく必要があります。

旧スタイルのデバイス名から新スタイルのデバイス名へのマッピングの補助としてスクリプトを使用しない場合は、デバイス名を手動によりマッピングする際に十分注意してください。ユーザは各デバイスの物理的な位置を知っている必要があり、アップグレード時にこの知識と、`hwmgr` および `scu` などのユーティリティを使用して `dsk*` 名がいずれのデバイスに割り当てられているかを判定することができなければなりません。新規システムあるいはクラスタが `dsk*` を、以前 `rz*` として認識されていたいずれのデバイスに割り当てているかを判定することは、簡単な作業ではありません。これが、

スクリプトを提供している第一の理由です。たとえ、新規 Tru64 UNIX システムあるいは TruCluster Server クラスタ上のストレージを構成するために移行スクリプトを使用する必要がない場合であっても、`rz*` デバイスが現在いずれの `dsk*` として認識されているかを明確にするために、これらのスクリプトを使用することを推奨します。

ASE の各メンバ上で次の手順を行います。

- バージョン 114 以上の `sys_check -all` コマンドを実行して、システム情報を保存し、ストレージ・マップを作成します。次の例を参照してください。

```
# /usr/sbin/sys_check -all > file.html
```

注意

システムに `sys_check` ユーティリティがない場合は、次の URL から入手することができます。

ftp://ftp.digital.com/pub/DEC/IAS/sys_check

- `/var/ase/config/asecdb` データベースとそのテキスト・コピーの両方を保存します。次の例を参照してください。

```
# cp /var/ase/config/asecdb asecdb.copy
# asemgr -d -C > asecdb.txt
```

- AdvFS ドメインおよびファイル・セットに関する情報を保存します。たとえば、`/etc/fdmns` にディレクトリを移動し、次のコマンドの結果を保存します。
 - `ls -lR` コマンドを使用して、すべてのドメインおよび関連するデバイスの一覧を表示します。
 - `showfdmn *` コマンドを使用して、ファイル・ドメインおよびボリュームに関する情報を表示します。
 - `showfsets` コマンドを使用して、各ドメインに関するファイル・セット情報を表示します。
- LSM を使用している場合は、`volsave` コマンドを実行してすべてのディスク・グループに関する LSM 構成情報を保存します (`volsave` コマンドを実行する前に、すべての ASE サービスがオンラインで提供されている必要があります)。

```
# volsave -d volsave.output
```

- デバイス名のマッピングに `clu_migrate_save` および `clu_migrate_configure -x` コマンドを使用しない場合は、新規システム上のストレージを構成するために旧スタイルのデバイス名 (rz*) から対応する新スタイルのデバイス名 (dsk*) へのマッピングを手動で行わなければなりません。マッピングの補助として、`scu` コマンドを使用して旧スタイルのデバイス名とその属性の一覧を作成します。表示される属性は、ベンダ、シリアル番号、バス/ターゲット/LUN (SCSI バスの番号の変更に `ase_fix_config` が使用された場合は適用されません) などです。次の `scu` コマンドを使用してこれらの属性を表示し、結果をファイルに保存します。

```
# scu -f device show device
# scu -f device show inq page serial
# scu -f device show nexus
```

次の例を参照してください。

```
# scu -f /dev/rrz28g show device | grep -E "Vendor|Product|Firmware"
Vendor Identification: DEC
Product Identification: RZ26L      (C) DEC
Firmware Revision Level: 440C

# scu -f /dev/rrz28g show inq page serial | grep "Product Serial"
Product Serial Number: PCB=420240831056 (ZG40831056 ?); \
HDA=0000000042181869

# scu -f /dev/rrz28g show nexus
Device: RZ26L, Bus: 3, Target: 4, Lun: 0, Type: Direct Access
```

- 保存された構成情報を含むファイルを新規 Tru64 UNIX システムあるいはシングル・メンバ・クラスタにコピーします。

F.2 新規 Tru64 UNIX システムあるいは TruCluster Server クラスタ上のストレージの手動による構成

この節では、新規 Tru64 UNIX システムあるいはシングル・メンバ・クラスタ上のストレージを構成するために `clu_migrate_configure` スクリプトを使用するオプション 2 およびオプション 3 に置き換わる手順について説明します。

注意

ASE メンバ上で `clu_migrate_save` を実行していない場合、`clu_migrate_configure -x` を使用してデバイス名のマッピングを表示することはできません。次に進む前に、すべての旧ス

F-4 アップグレード時の手動によるストレージの構成

タイトルのデバイス名 (rz*) から新スタイルのデバイス名 (dsk*) へのマッピングを手動で行ってください。以降の手順では、`clu_migrate_configure -x` で得られるマッピング結果をユーザ独自のものに置き換えてください。

ASE メンバ、Tru64 UNIX システム、シングル・メンバ・クラスタ上で `clu_migrate_save` を実行している場合は、`clu_migrate_configure -x` を実行します。

```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_configure -x
```

`clu_migrate_configure -x` コマンドを実行すると、旧スタイルのデバイス名から新スタイルのデバイス名へのマッピングを表示します。ASE に制御されていたストレージを構成する場合は、この情報を使用します。

次の手順は、ストレージを構成する場合の例です。

1. デバイス名のマッピングに `clu_migrate_save` および

`clu_migrate_configure -x` コマンドを使用しない場合は、旧スタイルのデバイス名 (rz*) から対応する新スタイルのデバイス名 (dsk*) へのマッピングを手動で行わなければなりません。新デバイス名へのマッピングの補助として、新規システム上で `scu` コマンドを使用します。次の例を参照してください。

```
# scu -f /dev/rdisk/dsk5g show device | grep -E "Vendor|Product|Firmware"
Vendor Identification: DEC
Product Identification: RZ26L      (C) DEC
Firmware Revision Level: 440C

# scu -f /dev/rdisk/dsk5g show inq page serial | grep "Product Serial"
Product Serial Number: PCB=420240831056(ZG40831056 ?); \
HDA=0000000042181869

# scu -f /dev/rdisk/dsk5g show nexus
Device: RZ26L, Bus: 1, Target: 4, Lun: 0, Type: Direct Access
```

ASE メンバから収集した `scu` 情報を使用して、旧スタイルのデバイス名から新スタイルのデバイス名へのマッピングを作成します。デバイス名を手動によりマッピングする場合は、`hwmgr` コマンドも有用なツールです。

F.1 節の `scu` の例では、このデバイスが ASE で `rz28` として認識されています。`show nexus` の結果のバス番号が同一でないことに注意してください。ASE で `ase_fix_config` が実行されたため、両方のシス

テムのバス番号が異なっていますが、これは信頼できるマッピング・デバイス情報ではありません。

2. 保存された `asecdb` データベース, `sys_check -all` の結果, 手動あるいは `clu_migrate_configure -x` を使用して作成したデバイス・マッピングに基づく LSM 情報を使用して, 各デバイスおよび LSM ディスク・グループを構成します。

注意

ディスク上の LSM 構成情報のフォーマットは変更されています。この時点以降, ASE に戻す必要がある場合は, インポート時に ASE システム上に LSM 情報を復元しなければなりません。

各新規 LSM デバイスに対して, 次のコマンドを実行します。

```
# voldisk define device
# voldisk online device
```

各ディスク・グループに対して, 次のコマンドを実行します。

```
# voldg import disk_group
```

各ボリュームに対して, 次のコマンドを実行します。

```
# volume -g disk_group start volume
```

次の例を参照してください。

```
# voldisk define dsk5g
# voldisk online dsk5g
# voldg import toolsdg
lsm:voldg: WARNING: Volume vol01: \
Temporarily renumbered due to conflict
# volume -g toolsdg start vol01
# volume -g toolsdg start vol02
```

警告は無視することができます。

3. 次回のリブート後, LSM が名前を更新するようにする場合は, `lsmupdate_setup` コマンドを実行します。

```
# /sbin/lsm.d/bin/lsmupdate_setup
```

4. 保存された `asecdb` データベース, `sys_check -all` の結果, 手動あるいは `clu_migrate_configure -x` を使用して作成したデバイス・マッピングに基づく AdvFS 情報を使用して, 旧スタイルのデバイス名

(rz) から新スタイルのデバイス名 (dsk) へのマッピングを行い、適切な /etc/fdmns エントリを作成し、ASE 上にあった AdvFS ドメインを手動により再作成します。次の例を参照してください。

```
# mkdir /etc/fdmns/data1_domain
# cd /etc/fdmns/data1_domain
# ln -s /dev/disk/dsk6g

# mkdir /etc/fdmns/tools_dmn
# cd /etc/fdmns/tools_dmn
# ln -s /dev/vol/toolsdg/vol01 toolsdg.vol01
```

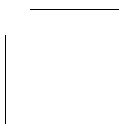
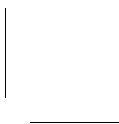
各ドメインに対して、`showfsets domain` コマンドを実行し、各ドメインに関するファイル・セットが正しいことを確認します。

5. ドメインをマウントします。各ドメインに対して `showfdmn domain` コマンドを実行します。
6. /etc/fstab ファイルにファイル・システムを追加します。また、/etc/exports などのストレージ情報を含むその他の構成ファイルを更新します。
7. システムをリブートし、新規デバイス名を使用して LSM を構成します。

```
# shutdown -r now
```

8. 次の LSM コマンドを実行して、LSM 構成を確認します。

```
# voldisk list
# volprint -thA
```



A

acl_mode 属性..... C-10t
act_vers_high 属性..... C-9t
act_vers_low 属性..... C-9t
Advanced File System
 (AdvFS を参照)
AdvFS
 Tru64 UNIX インストール時の使
 用 3-4
 ドメイン・バージョン番号... 8-12
aliasd デーモン 3-7
Alpha システム
 Firmware CD-ROM..... 3-3
ATM カーネル・オプション 3-5
auto_action コンソール変数.... 5-3
automount デーモン
 NFS ファイル・システムと
 clu_create..... 4-3n

B

BASE_VERSION 変数 C-4t
bcheckrc コマンド 7-12, 7-14
Berkeley Internet Name Domain
 (BIND を参照)
BIND
 クラスタをクライアントまたはサー
 バとして構成 3-8

boot_dev コンソール変数..... 2-28
boot_osflags コンソール変数 .. 2-26
boot_reset コンソール変数 2-26
bootdef_dev コンソール変数 . 2-26,
 5-3
 Fibre Channel デバイス用の設
 定 2-27n
 HS デバイスの設定 2-27
bus_probe_algorithm コンソール変
 数..... 2-28

C

CAA
 ローリング・アップグレード中の先
 行メンバのテスト 7-13
caa_stat コマンド 4-8
CDE
 ホスト名に関連付けられたネット
 ワーク・インタフェースが必
 要 5-7
CDSL 4-8
cfsmgr コマンド 7-13
clu_add_member コマンド ... 1-4,
 5-1, 5-3
 構成ファイル 6-3
 使用して LAN インターネットを構
 成する..... 5-3

ログ・ファイル 5-5
 ログ・ファイルの例..... D-11
CLU_BOOT_FILESYSTEM 変数..... C-4t
clu_check_config コマンド
 ATTENTION メッセージ B-7
 ログ・ファイル 4-8, 5-8
clu_create コマンド 1-4
 LAN インターコネクトの構成. 4-4
 NFS ファイル・システム, アクセス可能性の保証..... 4-3n
 構成ファイル 6-3
 実行前にベースおよびクラスタ・パッチを適用 4-3
 ログ・ファイル 4-6
 ログ・ファイルの例..... D-1
clu_create と **clu_add_member**
 省略時の物理クラスタ・インターコネクト IP 名 B-1
clu_delete_member コマンド.. 1-4
clu_get_info コマンド 4-8
clu_migrate_check スクリプト..... 8-14
clu_migrate_configure スクリプト..... 8-15
clu_migrate_recover スクリプト..... 8-16
clu_migrate_save スクリプト. 8-14
CLU_NEW_MEMBER 変数 ... C-5t
clu_quorum コマンド 4-8
clu_upgrade コマンド..... 1-5
 check roll [memberid] コマンド 7-17
 check setup コマンド . 7-9, 7-25t
 check コマンド..... 7-17
 clean コマンド 7-16, 7-35t
 completed コマンド 7-17
 postinstall コマンド.. 7-14, 7-32t
 preinstall コマンド .. 7-10, 7-30t
 roll コマンド 7-14, 7-33t
 setup コマンド..... 7-9, 7-28t
 started コマンド 7-17
 status コマンド 7-17
 switch コマンド..... 7-16, 7-34t
 tagged add コマンド 7-39
 tagged check コマンド 7-17, 7-38
 tagged disable コマンド..... 7-39
 tagged enable コマンド 7-39
 undo コマンド 7-18
 version コマンド 7-20
 ログ・ファイル 7-28, 7-35
 ログ・ファイルの例..... D-15
CLU_VERSION 変数 C-5t
CLUAMGR_ROUTE_ARGS 変数..... 2-5
cluamgr コマンド 4-8, 7-13
CLUIF ifconfig フラグ 2-9
cluster_expected_votes 属性 . 2-23
cluster_interconnect 属性 C-8t
cluster_map_create コマンド.. 8-9
cluster_name 属性 C-8t
CLUSTER_NET 変数 C-5t
cluster_node_inter_name 属性..... C-8t
cluster_node_name 属性 C-8t
cluster_node_votes 属性..... C-8t
cluster_qdisk_major 属性 C-8t
cluster_qdisk_minor 属性 C-8t
cluster_qdisk_votes 属性 C-8t
cluster_seqdisk_major 属性.. C-8t

cluster_seqdisk_minor 属性.. C-8t
cmon コマンド..... 8-9

D

DE50x アダプタ
自動折衝に設定 B-2
DE60x アダプタ
自動折衝の設定 B-2
DEC_VERSION_TAG プロパ
ティ 7-35
DEGPA-xx アダプタ
自動折衝の設定 B-3
DHCP
クライアント, ~としてサポートさ
れないクラスタ..... 3-9
サーバ, クラスタを~として構
成 3-9
diskconfig コマンド..... 3-12
disklabel コマンド 3-12
drd_nopr_list 属性..... C-9t
drdmgr コマンド 4-8
dupatch コマンド 7-12, 7-31t
ノーロール・パッチ..... 7-5
DVN 8-12
Dynamic Host Configuration
Protocol
(DHCP を参照)

E

/etc/gated.conf
インストールによるコピーま
たは作成 C-2

/etc/gated.conf.member<n> ... C-2
/etc/hosts
インストールによる更新 C-3
/etc/ifaccess.conf
インストールによる更新 C-3
/etc/inittab
インストールによる更新 C-3
/etc/rc.config
BASE_VERSION 変数 C-4t
CLU_BOOT_FILESYSTEM 変
数 C-4t
CLU_NEW_MEMBER 変数.. C-5t
CLU_VERSION 変数 C-5t
CLUSTER_NET 変数 C-5t
GATED_FLAGS 変数 C-5t
GATED 変数 C-5t
HOSTNAME 変数 C-5t
IFCONFIG 変数 2-7
IMC_AUTO_INIT 変数..... C-5t
IMC_MAX_ALLOC 変数 C-5t
IMC_MAX_RECV 変数..... C-6t
MAX_NETDEVS 変数 C-6t
NETDEV 変数 2-7, C-6t
NR_DEVICES 変数 C-6t
NRCONFIG 変数 C-6t
NRDEV 変数..... C-6t
NUM_NETCONFIG 変数 C-6t
PAGERAW 変数..... C-6t
ROUTER 変数 C-6t
TCR_INSTALL 変数 C-7t
TCR_PACKAGE 変数 C-7t
XLOGIN 変数..... C-7t
XNTP_SERV 変数 C-7t

XNTPD_CONF 変数 C-7t
 XNTPD_OPTS 変数 C-7t
 インストレーションによる更新 C-3
 コピーの保存 4-9, 5-9
/etc/rc.config.common
 CLUAMGR_ROUTE_ARGS 変
 数 2-5
 コピーの保存 4-9
/etc/sysconfigtab
 acl_mode 属性 C-10t
 act_vers_high 属性 C-9t
 act_vers_low 属性 C-9t
 cluster_expected_votes 属性. 2-23,
 C-8t, C-11t
 cluster_interconnect 属性 C-8t
 cluster_name 属性 C-8t
 cluster_node_inter_name 属
 性 C-8t
 cluster_node_name 属性 C-8t
 cluster_node_votes 属性 C-8t
 cluster_qdisk_major 属性 C-8t
 cluster_qdisk_minor 属性 C-8t
 cluster_qdisk_votes 属性 C-8t
 cluster_seqdisk_major 属性 .. C-8t
 cluster_seqdisk_minor 属性 .. C-8t
 drd_nopr_list 属性 C-9t
 ics_tcp_adapter0 属性 C-10t
 ics_tcp_inetaddr0 属性 C-10t
 ics_tcp_netmask0 属性 C-10t
 ics_tcp_nr0 属性 C-10t
 lsm_rootdev_is_volume 属性 C-10t
 memberid 属性 C-9t
 msgbuf_size 属性 C-9t
 new_vers_high 属性 C-9t
 new_vers_low 属性 C-9t

rolls_ver_lookup 属性 C-10t
 swapdevice 属性 C-11t
 vm_page_free_min 属性 C-11t
 vm_page_free_reserved 属性 C-11t
 インストレーションによる更新 C-3
 コピーの保存 4-9, 5-8
/etc/sysconfigtab.cluster C-4
 コピーの保存 4-9, 5-8

F

Fibre Channel

bootdef_dev コンソール変数 2-27n
 参照が必要 2-10n

G

GATED_FLAGS 変数 C-5t
gated デーモン
 クラスタでのルーティング・デーモ
 ンとしての使用 3-7
GATED 変数 C-5t
/genvmunix
 メンバのブート 5-5

H

HOSTNAME 変数 C-5t
hwmgr コマンド 3-14, 6-6

I

I18N サポート
 (WLS を参照)
ics_ll_tcp

ics_ll_tcp NetRAIN エラー・メッセージ..... B-6
 パニック・メッセージ B-5
ics_tcp_adapter0 属性 C-10t
ics_tcp_inetaddr0 属性 C-10t
ics_tcp_netmask0 属性 C-10t
ics_tcp_nr0 属性 C-10t
IFCONFIG 変数
 クラスタ・インターコネクト・アダプタの数との関連 2-7
IMC_AUTO_INIT 変数 C-5t
IMC_MAX_ALLOC 変数 C-5t
IMC_MAX_RECV 変数 C-6t
init コマンド..... 7-12, 7-14
installupdate コマンド 7-12, 7-31t
IPv6 2-5
IP 名と **IP** アドレス
 クラスタ・インターコネクト . 2-7
 省略時のクラスタ別名 2-4
 ネットワーク・インタフェース 2-6

L

LAN
 クラスタ・インターコネクト 2-3n
 クラスタ・インターコネクト・プライベート・サブネット..... 2-7
lmf reset コマンド..... 7-12, 7-14
Logical Storage Manager
 (LSM を参照)
LSM
 アップグレード時の手動による構成 F-6

 アップグレード時のボリュームのインポート 8-15
 クラスタ作成後にファイル・システムをミラー化する 2-17
 制約事項..... 2-10n
 物理デバイスの位置..... 2-14n
 ベース・オペレーティング・システム上にインストール 3-4
lsm_rootdev_is_volume 属性..... C-10t

M

MAX_NETDEVS 変数 C-6t
Memory Channel
 アダプタ..... 8-27
 クラスタ・インターコネクト 2-3n
mount -o server_only コマンド..... 8-11n
msgbuf_size 属性 C-9t

N

NETDEV 変数 C-6t
 クラスタ・インターコネクト・アダプタの数との関連 2-7
NetRAIN
 /etc/rc.config 構成変数..... C-6t
 /etc/sysconfigtab 属性 C-10t
 冗長ネットワーク構成の設定 3-12
 二重に定義されたデバイス.... B-6
Network Time Protocol
 (NTP を参照)
New Hardware Delivery

(NHD を参照)

new_vers_high 属性 C-9t

new_vers_low 属性 C-9t

NFS

クラスタをクライアントまたはサーバとして構成 3-8

ファイル・システムと

clu_create..... 4-3n

nfsmount stop コマンド 4-3n

NHD

ローリング・アップグレード中に
キットをインストールする . 7-4

nhd_install コマンド .. 7-12, 7-31t

NIS

クラスタをクライアントまたはサーバとして構成 3-9

NR_DEVICES 変数..... C-6t

NRCONFIG 変数 C-6t

NRDEV 変数 C-6t

NTP

クラスタをクライアントまたはサーバとして構成 3-7

NUM_NETCONFIG 変数..... C-6t

O

ogated デーモン 3-7

.Old.. ファイル 7-35

P

PAGERAW 変数 C-6t

PAK..... 4-2

Product Authorization Key

(PAK を参照)

R

RAID

共用ファイル・システムのミラー
化 2-18

クォーラム・ディスクのミラーリン
グ 2-10n

Redundant Array of Independent Disks

(RAID を参照)

Request for Comments

(RFC を参照)

RFC

1035, Network Time Protocol . 3-7

1123, Requirements for Internet
Hosts..... 2-4

1918, Address Allocation for Private
Internets..... 2-3, 2-5

952, DoD Internet Host Table
Specification..... 2-4

/.rhosts

アップグレード時に ASE へのアク
セスの許可 8-21

アップグレード時の ASE メンバに
よるアクセスの許可 8-28

インストレーションによる更新 C-1

RIS

省略時のクラスタ別名を登録 7-42

rolls_ver_lookup 属性 7-30,
7-36, C-10t

routed デーモン 3-7

ROUTER 変数..... C-6t

rsh

Secure Shell 接続を使用するための
構成 3-11

S

Secure Shell..... 3-11
setld コマンド
 TruCluster Server キットのロード
 4-2
shutdown コマンド 4-7
SRM ファームウェア
 Alpha Systems Firmware CD-ROM
 によるアップデート ... 3-3, 5-3
swapdevice 属性 C-11t
/sys/<HOSTNAME>/vmunix
 新しいカーネルの省略時の名前 C-4
sys_check コマンド 6-6n
 入手先 F-3n
/sys/conf/<HOSTNAME>.list .. C-4
SysMan Station..... 3-14
 SysMan によるインストレーションはクラスタではサポートしない 1-1
System Reference Manual
 (SRM ファームウェア を参照)

T

TCR_INSTALL 変数..... C-7t
TCR_PACKAGE 変数 C-7t

U

UFS
 クラスタ単位で読み取り専用 . 1-9
 パーティション化されたファイル・システムとしてマウント. 8-11n

メンバだけが読み取り/書き込み可能 1-9

UNIX ファイル・システム
(UFS を参照)

updadmin コマンド 7-35
/usr ファイル・システム
 クラスタ作成に必要な空き容量 4-3
 ローリング・アップグレードに必要なディスク容量 2-22
 ローリング・アップグレードに関するディスク容量の条件 7-27

V

/var ファイル・システム
 ローリング・アップグレードに必要なディスク容量 2-22
 ローリング・アップグレードに関するディスク容量の条件 7-27
versw コマンド 7-40
vm_page_free_min 属性 C-11t
vm_page_free_reserved 属性 C-11t
vMAC
 ハードウェア・アドレス作成のアルゴリズム 7-42
/vmunix
 CDSL と ~ 1-9
 その他のクラスタ・メンバ 5-6
volencap コマンド 2-17
volmigrate コマンド 2-17

W

WLS

- i18n ファイル・システムのローリング・アップグレードに必要な容量 2-22
- i18n ファイル・システムのローリング・アップグレードに関するディスク容量の条件 7-27
- 独立したファイル・システムへのインストール 3-4

X

- XLOGIN** 変数 C-7t
- XNTP_SERV** 変数 C-7t
- XNTPD_CONF** 変数 C-7t
- XNTPD_OPTS** 変数 C-7t
- xntpd** デーモン
 - 推奨されるタイム・サービス . 3-7

あ

- アダプタ
 - DE50x B-2
 - DE60x B-2
 - DEGPA-xx B-3
 - NetRAIN 3-19
 - イーサネット 3-19
- アップグレード・スクリプト ... 8-13
- アドレス
 - ブロードキャスト 2-8

い

- インストレーション

- 構成ファイル 6-3
- タイプ 1-3
- ディスクの必要条件 2-10
- 手順の概要 1-5
- インストール段階 7-31t
- インターコネクト
 - (クラスタ・インターコネクトを参照)

え

- エラー・メッセージ
 - ics_ll_tcp NetRAIN エラー B-6
- エンハンスト・セキュリティ
 - 構成 3-11

お

- オペレーティング・システム
 - インストール 3-3

か

- 各国語サポート
 - (WLS を参照)
- カーネル
 - ATM 構成オプション 3-5
 - 第 1 メンバ以外のメンバ用に構築 5-6

き

- 共通デスクトップ環境
 - (CDE を参照)

く

- クォーラム
 - sysconfigtab 属性 C-8t, C-11t
 - ポートの割り当て 2-23
- クォーラム・ディスク 2-16
- クラスタ・アプリケーションの可用性
(CAA を参照)
- クラスタ・インターコネクト ... 1-1,
2-7
- IP 名と IP アドレス 2-7
- LAN 2-3n
- Memory Channel 2-3n
- アダプタ 8-33
- クラスタ構成の確認
 - ATTENTION メッセージ B-7
- クラスタ単位のファイル・システム
 - 定義された 1-6
 - ディスクの割り当て 2-13
 - ローリング・アップグレードに必要な
空き容量 7-26
- クラスタの作成 4-1
- クラスタ別名
 - aliasd デーモン 3-7
 - IPv6 アドレス 2-5
 - RIS への登録 7-42
 - 省略時の別名 IP アドレス 2-4
 - ローリング・アップグレード中の先
行メンバのテスト 7-13
- クラスタ・メンバの再インストール
..... 6-1
- クラスタ・メンバの追加 5-1
- クリーンアップ段階 7-35t

け

- 警告
 - DHCP クライアントとしてのシステ
ムの構成は不可 3-9
 - /var/adm/update 内のファイルの削
除不可 7-29n
 - クラスタ再作成前のメンバの削
除 6-2n
 - クラスタ動作中はベース・オペレー
ティング・システムのブート不
可 2-12n
 - タグ付きファイルの作成 ... 7-30n
 - 適切なメンバ・ディスクのブー
ト 5-6n
 - ブロッキング・レイヤード・プロダ
クトとローリング・アップグレー
ド 7-38n

こ

- 構成ファイル 6-3
- 国際化
 - (WLS を参照)
- コンソール変数
 - boot_dev 2-28
 - boot_osflags 2-26
 - boot_reset 2-26
 - bootdef_dev 2-26
 - bus_probe_algorithm 2-28
- コンテキスト依存シンボリック・リ
ンク
 - (CDSL を参照)

さ

- サブセット
 - インストール 4-2
 - オペレーティング・システム . 1-8
 - サイズ 1-7
 - 内容 1-7
- サブネット
 - クラスタ・インターコネクト用プラ
イベート・サブネット 2-7
- サブネット・マスク 2-8n
- サポートされていない作業
 - ローリング・アップグレード中 7-7

し

- システム・ファイル
 - 変更 C-1
- 自動折衝 B-2
 - 設定 B-2
- 準備段階 7-25t
- 冗長ネットワーク構成
 - 設定 3-12
- 省略時のクラスタ別名
 - IP アドレス 2-4
 - ローリング・アップグレード中の先
行メンバのテスト 7-13
- 省略時の物理クラスタ・インターコ
ネクト **IP** 名
 - 名前の重複 B-1

す

- スイッチ段階 7-34t

せ

- セキュリティ
 - 構成 3-11
- セットアップ段階 7-28t
- 先行メンバ 7-25, 7-37

た

- タイム・サービス
 - NTP 3-7
 - 構成 3-7
- タグ付きファイル 7-35
 - OSF, TCR, および IOS 製品コー
ドに対して作成 7-37
 - クリーンアップ段階で削除... 7-35
 - 作成規則 7-37
 - セットアップ段階で作成 7-29
 - 存在確認 7-17
 - ブリ・インストール段階で検
査 7-30
- 段階
 - インストール 7-31t
 - クリーンアップ 7-35t
 - 実行の所要時間 7-8
 - 準備 7-25t
 - スイッチ 7-34t
 - ステータスの確認 7-17
 - セットアップ 7-28t
 - 取り消し 7-18
 - ブリ・インストール 7-30t
 - 並列ロール 7-15
 - ポスト・インストール 7-32t
 - ロール 7-33t

て

- ディスク
 - Fibre Channel..... 2-10n
 - インストールに必要な数 2-10
 - クラスタ用ディスクの構成... 3-12
 - 使用前のラベル付け..... 3-12
 - 推奨するディスク・スペース 2-19
 - スペシャル・ファイル名の物理デバイスへのマッピング 3-14
 - 必要な空き容量
 - クラスタ作成 4-3
 - ローリング・アップグレード..... 7-26
 - メンバ・ブート・ディスクのコピー不可 5-6n
 - ローリング・アップグレードのための設計..... 2-21
- ディスク・スペース
 - インストールに必要な 2-19
 - ローリング・アップグレードのための設計..... 2-21
- デバイス名
 - ネットワーク・アダプタ 3-18

と

- 同時 (並列) ロール 7-15
- ドメイン・バージョン番号
(DVN を参照)

ね

- ネットワーク・アダプタ
 - デバイス名..... 3-18
- ネットワーク・インタフェース
 - ics_ll_tcp パニック・メッセージ B-5
- ネットワーク・サービス
 - 構成 3-6
- ネットワーク情報サービス
(NIS を参照)
- ネットワーク・ファイル・システム
(NFS を参照)

の

- ノーロール・パッチ 7-5

は

- バックアップ
 - ローリング・アップグレード中の 7-25n
- パッチ
 - clu_create 実行前にベース・パッチとクラスタ・パッチを適用 . 4-3
 - ノーロール・パッチ..... 7-5
- パニック
 - ics_ll_tcp パニック・メッセージ B-5
- ハング
 - メンバのブート時..... B-2, B-4
- バージョン 5.1B へのアップグレード
 - ド 1-3
 - バージョン 5.1A から 7-1

バージョン 1.5 または 1.6 から 8-1
バージョン・スイッチ 7-39
パーティション化されたファイル・システム 8-11n
ハードウェアのミラーリング . 2-10n

ひ

非同期転送モード
(ATM カーネル・オプション を
参照)

ふ

ファイル・システムのミラーリン
グ 2-10n
ファイル・システムのレイアウト
クラスタ 2-18
ファームウェア
Alpha Systems Firmware CD-ROM
によるアップデート ... 3-3, 5-3
複数 (並列) ロール 7-15
複数バス・フェイルオーバ 2-27
プライベート・ネットワーク・アドレ
ス 2-3
プリ・インストール段階 7-30t
プリント・サーバ
クラスタを ~ として構成 3-10
フル・インストレーション 1-3
ブロッキング・レイヤード・プロダク
ト 7-41
ブロードキャスト・アドレス 2-8
ブート
ATTENTION メッセージ B-7
clu_add_member の実行後 5-5
clu_create の実行後 4-6

ブート中の **ATTENTION** メッセー
ジ B-7
ブート・ディスク
Fibre Channel 2-10n
ブート・パーティション
ファイルセットの追加 2-16
ブート・メッセージ
clu_add_member の実行後 E-4
clu_create の実行後 E-1

へ

並列ロール 7-15
別名
(クラスタ別名 を参照)

ほ

ポスト・インストール段階 7-32t
ホスト名
ホスト名の重複 B-1
命名規則 2-4

ま

マスク
サブネット 2-8n
マニュアル
インストレーションに必要な . 2-2

め

メッセージ
Error: Bad configuration B-1
ブート時のクラスタ・チェックによ
る ATTENTION B-7

メモリの必要条件 1-9
メンバ **ID** 2-2
 memberid 属性 C-9t
メンバ・ブート・ディスク 2-14
 clu_add_member で使用するための
 検索 3-14
 コピー不可 5-6n
 ローリング・アップグレードに関する
 ディスク容量の条件 7-27
 ローリング・アップグレードに必要な
 容量 2-22
メンバ・ブート・パーティション
 ファイルセットの追加 2-16
メール
 クラスタをサーバとして構成 3-9

よ

予約されているネットワーク・アドレス 2-3

ら

ライセンス
 TruCluster Server の登録 4-2
 クラスタの要件 1-6
 追加メンバのオペレーティング・システム 5-7
 メンバの追加 5-3

り

リモート・インストレーション・サービス

(RIS を参照)

リモート・シェル (**rsh**)

Secure Shell 接続を使用するための
構成 3-11

る

ルーティング情報プロトコル

(RIP を参照)

ルート・ファイル・システム

ローリング・アップグレードに必要な
ディスク容量 2-22

ローリング・アップグレードに関する
ディスク容量の条件 7-27

れ

レイヤード・プロダクト

ブロッキング 7-41

ローリング・アップグレードの一般
的なガイドライン 7-40

ろ

ログ・ファイル

clu_add_member コマンド ... 5-5,
D-11

clu_check_config コマンド ... 4-8,
5-8

clu_create コマンド 4-6, D-1

clu_upgrade コマンド 7-28, D-15

ローカル・エリア・ネットワーク

(LAN を参照)

ローリング・アップグレード ... 1-1, 7-1 (段階 も 参照) CAA を使用した先行メンバのテスト 7-13 NHD..... 7-4 Tru64 UNIX オペレーティング・システム・ディスクのアップデイト 7-4 クラスタ別名を使用した先行メンバのテスト 7-13 サポートされていない作業.... 7-7 ステータスの確認..... 7-17 先行メンバ..... 7-25, 7-37 タグ付きファイル..... 7-35	段階 7-24 段階の取り消し 7-18 ディスク容量の条件..... 7-26 手順 7-8 バックアップ 7-25n パッチ 7-4 バージョン・スイッチ 7-39 必要なディスク容量..... 2-21 並列ロール..... 7-15 ローリング・パッチ (ロールリング・アップグレード を参照) ロール段階 7-33t 並列ロール..... 7-15
---	--

マニュアルに対するご意見

TruCluster Server

クラスタ・インストール・ガイド

AA-RM86D-TE

弊社のマニュアルに関して、ご意見、ご要望、または内容の不明確な部分など、お気づきの点がございましたら、下記にご記入の上、弊社社員にお渡しくださるようお願い申し上げます。

マニュアルの採点：

	大変良い	良い	普通	良くない
正確さ (説明どおりに動作するか)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
情報量 (十分か)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
分かり易さ	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
マニュアルの構成	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
図 (役立つか)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
例 (役立つか)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
索引 (項目の検索性)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ページ・レイアウト (情報の検索性)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

内容の不明確な部分がありましたら、以下にご記入ください：

ペー ジ

その他お気づきの点がございましたら、以下にご記入ください：

ご使用のソフトウェアのバージョン： _____

貴社名/部課名 _____

御名前 _____

記入日 _____

(注) 当用紙を受け取った弊社社員は、すみやかに下記にお送りください。

ビジネスクリティカルシステム統括本部 **BCS** 技術本部 **Alpha** ソフトウェア技術部