

Tru64 UINX

AdvFS 管理ガイド

Part Number: AA-RK3TD-TE

2002 年 11 月

ソフトウェア・バージョン: Tru64 UINX V5.1B 以上

本書では、AdvFS ファイル・システムの理解、構成、トラブルシューティング、チューニングについての情報と手順について説明しています。本書はシステム管理者を対象としています。

© 2002 日本ヒューレット・パッカード株式会社

本書の著作権は日本ヒューレット・パッカード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・パッカード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・パッカードは、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。

原典: AdvFS Administration (AA-RH96D-TE)
Copyright ©2002 Hewlett-Packard Company

目次

まえがき

1 ファイル・システムの計画

1.1	AdvFS の概要	1-1
1.2	ライセンス登録	1-2
1.3	ファイル・システムの構造	1-2
1.3.1	ファイルセット, ドメイン, およびボリューム	1-3
1.3.2	トランザクション・ログ・ファイル	1-4
1.3.3	ファイル・ストレージの割り当て	1-5
1.4	AdvFS ファイル・システムの構成	1-7
1.4.1	ドメインとファイルセットの構成	1-7
1.4.2	ボリュームの構成	1-9
1.5	AdvFS のストレージの選択	1-10

2 ファイル・システムのセットアップ

2.1	概要	2-1
2.2	ボリューム	2-2
2.2.1	ボリューム属性	2-2
2.3	ドメイン	2-3
2.3.1	/etc/fdmns ディレクトリ	2-3
2.3.2	ドメイン情報の表示	2-4
2.3.3	ドメインの作成	2-5
2.3.3.1	ドメイン・バージョン番号	2-6
2.3.3.2	DVN4 のドメインの作成	2-7
2.3.3.3	DVN3 のドメインの作成	2-8

2.3.3.4	ドメインの DVN4 へのアップグレード	2-9
2.3.4	ドメイン・サイズの拡大	2-10
2.3.4.1	ボリュームを追加することによるドメインのストレージの拡大	2-10
2.3.4.2	ボリュームを交換することによるドメインのストレージの拡大	2-13
2.3.4.3	既存のボリュームを拡張することによるドメインのストレージの拡大	2-14
2.3.5	ドメイン・サイズの縮小	2-14
2.3.6	ドメインの削除	2-16
2.3.7	ドメイン名の変更	2-17
2.4	ファイルセット	2-18
2.4.1	/etc/fstab ファイルでのファイルセットの指定	2-19
2.4.2	ファイルセット情報の表示	2-20
2.4.3	ファイルセットの作成	2-22
2.4.4	ファイルセットのアップグレード	2-23
2.4.5	ファイルセットのマウント	2-23
2.4.6	ファイルセットのアンマウント	2-24
2.4.7	ファイルセットの削除	2-24
2.4.8	ファイルセット名の変更	2-25
2.4.9	ファイルセットの属性の変更	2-26
2.4.10	AdvFS ファイルセット・クローン	2-26
2.4.10.1	AdvFS ファイルセット・クローンの作成	2-29
2.4.10.2	AdvFS ファイルセット・クローンのマウントとアンマウント	2-29
2.4.10.3	AdvFS ファイルセット・クローンの削除	2-29
2.4.10.4	AdvFS ファイルセット・クローンの名前の変更	2-30
2.5	AdvFS ルート・ファイル・システムの構成	2-30
2.5.1	シングルユーザ・モードでのルート・ファイル・システムのマウント	2-31

2.5.2	ルート・ドメイン名の変更	2-31
2.5.3	ルート・ファイルセット名の変更	2-32
2.6	ファイル回復のためのゴミ箱の設定	2-34

3 クォータの管理

3.1	クォータの概要	3-1
3.1.1	クォータ限界値	3-2
3.1.2	猶予期間	3-4
3.1.3	クォータの機能の要約	3-5
3.2	ユーザおよびグループのクォータ	3-5
3.2.1	ユーザおよびグループのクォータ・ファイル	3-6
3.2.2	ユーザおよびグループのクォータ, および猶予期間の設定	3-7
3.2.2.1	ユーザ・クォータの設定	3-8
3.2.2.2	ユーザ猶予期間の設定	3-9
3.2.2.3	グループ・クォータの設定	3-10
3.2.2.4	グループ猶予期間の設定	3-11
3.2.2.5	一時的猶予期間の設定	3-13
3.2.3	複数のユーザおよびグループに対するユーザおよびグループ・クォータの設定	3-13
3.2.3.1	複数のユーザに対するクォータの設定	3-13
3.2.3.2	複数のグループに対するクォータの設定	3-14
3.2.4	システム・スタートアップ時のユーザおよびグループ・クォータの適用	3-15
3.2.5	ユーザおよびグループ・クォータの手動による適用	3-16
3.2.6	ユーザおよびグループ・クォータの適用解除	3-17
3.3	ファイルセット・クォータ	3-17
3.3.1	ファイルセット・クォータ・ファイル	3-17
3.3.2	ファイルセット・クォータおよび猶予期間の設定	3-18
3.3.2.1	ファイルセット・クォータの設定	3-18

3.3.2.2	猶予期間の設定	3-19
3.3.3	複数のファイルセットに対するクォータの設定	3-20
3.3.4	ファイルセット・クォータの適用	3-20
3.3.5	ファイルセット・クォータの適用解除	3-20
3.4	ファイルセットおよびディスク・スペース使用量の確認	3-20
3.4.1	ユーザおよびグループ・クォータの確認	3-20
3.4.1.1	各ファイルのタグとフルパス名の表示	3-21
3.4.1.2	ファイルセット所有者の要約	3-21
3.4.1.3	ディスク使用量と限界値の表示	3-22
3.4.1.4	ユーザおよびグループ・クォータの整合性の確認	3-23
3.4.1.5	ファイルセットごとのユーザおよびグループ・クォータの要約	3-24
3.4.2	ファイルセット・クォータの確認	3-25
3.4.2.1	ファイルセット・スペース情報の表示	3-25
3.4.2.2	ファイルセットの限界値の表示	3-27
3.4.2.3	ドメインの領域情報の表示	3-27
3.4.2.4	ドメインとファイルセットのディスク使用状況の表示	3-27

4 データのバックアップとリストア

4.1	データとメタデータの整合性	4-1
4.1.1	データ整合性の保証	4-1
4.1.1.1	アプリケーションの休止機能を使用したデータ整合性の保証	4-2
4.1.1.2	同期書き込み方式を使用したデータ整合性の保証	4-2
4.1.1.3	同期書き込み方式を用いないデータ整合性の保証	4-2
4.1.2	メタデータの整合性の保証	4-3
4.1.2.1	ドメインをフリーズすることによるメタデータ整合性の保証	4-3
4.1.2.2	ファイルセットをアンマウントすることによるメタデータ整合性の保証	4-4

4.2	バックアップ・ソースの作成	4-4
4.2.1	マウントしたオリジナル・ファイルセットのバックアップ・ソースとしての使用	4-5
4.2.1.1	長所と短所	4-5
4.2.1.2	データとメタデータ整合性の保証	4-5
4.2.1.3	バックアップ・ソースの準備	4-5
4.2.2	マウントしたファイルセット・クローンのバックアップ・ソースとしての使用	4-6
4.2.2.1	長所と短所	4-6
4.2.2.2	データとメタデータの整合性の保証	4-7
4.2.2.3	バックアップ・ソースの準備	4-7
4.2.3	LSM スプリット・ミラーからマウントしたファイルセットのバックアップ・ソースとしての使用	4-8
4.2.3.1	長所と短所	4-8
4.2.3.2	データとメタデータの整合性の保証	4-9
4.2.3.3	バックアップ・ソースの準備	4-9
4.2.4	コントローラ・ベースのクローンからマウントしたファイルセットのバックアップ・ソースとしての使用	4-10
4.2.4.1	長所と短所	4-10
4.2.4.2	データとメタデータの整合性の保証	4-11
4.2.4.3	バックアップ・ソースの準備	4-11
4.2.5	コントローラ・ベースのスナップショットからマウントしたファイルセットのバックアップ・ソースとしての使用	4-12
4.2.5.1	長所と短所	4-13
4.2.5.2	データとメタデータの整合性の保証	4-13
4.2.5.3	バックアップ・ソースの準備	4-13
4.3	バックアップ方式とツール	4-15
4.3.1	vdump および vrestore コマンドの概要	4-15
4.3.2	vdump コマンド固有の機能	4-16
4.3.3	vdump コマンドを使用したバックアップ・レベルの指定	4-17

4.3.4	vdump コマンドを使用したテープへのダンプ	4-18
4.3.5	vdump コマンドを使用した標準出力へのダンプ	4-18
4.3.6	vdump コマンドを使用したサブディレクトリのダンプ ...	4-19
4.3.7	ファイルやディスク・パーティションへのダンプ	4-19
4.3.8	セーブセットの圧縮	4-20
4.3.9	エラー保護機能付きのダンプ	4-21
4.3.10	vdump セーブセットに保存したファイルのリスト	4-21
4.3.11	clonefs コマンドと vdump コマンドを使用したシステム・ディスクのバックアップ(例)	4-22
4.3.12	リモートでのファイルのダンプおよびリストア	4-23
4.3.13	vrestore コマンド固有の機能	4-24
4.3.14	vrestore コマンドを使用したファイルのリストア	4-25
4.3.15	vrestore コマンドを使用したクォータのリストア	4-25
4.3.16	vrestore コマンドを使用した複数のセーブセットを含むテープからのリストア	4-26

5 AdvFS ファイル・システムの最適化とチューニング

5.1	性能の監視	5-2
5.2	フラグ・ファイルの無効化による性能改善	5-3
5.3	トランザクション・ログ・ファイルの性能改善	5-4
5.3.1	トランザクション・ログ・ファイルの移動	5-5
5.3.2	トランザクション・ログ・ファイルの分離	5-6
5.4	データ・キャッシュのチューニング	5-7
5.5	データの一貫性の向上	5-8
5.5.1	I/O の制御	5-8
5.5.1.1	非同期 I/O	5-9
5.5.1.2	同期 I/O	5-9
5.5.1.3	同期 I/O のオン/オフのチューニング	5-9
5.5.2	アトミック書き込みデータ・ロギング I/O の有効化	5-10

5.5.2.1	非同期アトミック書き込みデータ・ロギング I/O	5-11
5.5.2.2	同期アトミック書き込みデータ・ロギング I/O	5-11
5.5.2.3	永続的なアトミック書き込みデータ・ロギング	5-12
5.5.2.4	一時的なアトミック書き込みデータ・ロギング	5-13
5.5.2.5	アトミック書き込みデータ・ロギングのオン/オフの チューニング	5-14
5.6	ダイレクト I/O によるデータ転送速度の改善	5-15
5.7	属性の変更によるシステム性能の改善	5-16
5.8	vfast ユーティリティによるオペレーティング・システムのス ループットの改善	5-17
5.8.1	vfast ユーティリティの実行	5-21
5.8.2	vfast ユーティリティの例	5-21
5.9	ドメインの断片化の解消	5-23
5.10	ファイルの断片化解消	5-27
5.11	マルチボリューム・ドメインのバランシング	5-27
5.12	別のボリュームへのファイルの移動	5-29
5.13	ファイルのストライプ化	5-31
5.14	新しいボリュームへのドメインとそのファイルセットの移動	5-32
5.15	ドメイン・パニック情報の制御	5-33
6	トラブルシューティング	
6.1	ユーザ・ファイルの復旧	6-1
6.2	一般的な復旧手順	6-2
6.2.1	問題の報告のためのデータの保存	6-2
6.2.2	システム・メタデータのコピーの保存	6-2
6.2.3	ダメージを受けていないファイルの保存	6-3
6.2.4	ファイル・システムの整合性の確認	6-3
6.2.5	fixfdm ユーティリティを使用したディスク上のメタデータ 損傷への対処	6-5

6.2.6	損傷したドメインからのファイル・データの救済	6-6
6.2.6.1	ファイル・データの復元	6-7
6.2.6.2	破損したルート・ドメインからのデータの復元	6-9
6.2.6.3	ブロック単位でのデータの復元	6-9
6.3	問題の修復	6-10
6.3.1	ドメイン・パニックからの回復	6-10
6.3.2	読み取り専用でマウントされたファイルセットの復旧	6-12
6.3.3	/etc/fdmns ディレクトリのリストア	6-14
6.3.3.1	手作業による /etc/fdmns ディレクトリの再構成	6-15
6.3.3.2	advscan コマンドによる /etc/fdmns ディレクトリの再 構成	6-16
6.3.4	AdvFS ルート・ドメインのデータの損傷からの復旧	6-20
6.3.5	ハードウェア障害からの回復	6-25
6.3.6	AdvFS ディスクの別のマシンへの移動	6-26
6.3.7	マルチボリューム usr ドメインのリストア	6-28
6.3.8	mkfdmn コマンドまたは addvol コマンドの誤った使用か らの復旧	6-31
6.4	問題の予防	6-31
6.4.1	未使用スペースとディスク使用状況のチェック	6-32
6.4.2	ボリュームの再利用	6-33
6.4.3	ディスク障害	6-34
6.4.4	ディスク使用量の制御	6-34
6.4.5	クォータ限界値と猶予期間の上限の設定	6-34
6.4.5.1	クォータ限界値の超過	6-35
6.4.5.2	クォータ限界値までの保存	6-35
6.4.5.3	クォータ限界値の変更	6-35
6.4.5.4	猶予期間の無効化	6-35
6.4.6	ディスク構造の互換性の問題の回避	6-36
6.4.7	ユーティリティの互換性の問題の回避	6-36

6.4.8	ログ・ファイルの不整合の回避	6-37
6.4.9	AdvFS ルート・ドメインのサイズ拡張	6-37
6.4.9.1	ルート・ボリュームのサイズ拡張	6-38
6.4.9.2	ルート・ボリュームの大きなデバイスへの移動	6-38
6.4.10	メモリ・マッピング, ダイレクト I/O, およびデータ・ロギングの互換性の問題	6-40
6.4.11	古いデータへのアクセスの禁止	6-41
6.4.12	無効または破損したセーブセット・フォーマット	6-41
6.4.13	性能低下の対策	6-42
6.4.14	rmvol または migrate コマンドを起動できない	6-43

A SysMan Menu によるファイル・システム情報へのアクセス

B AdvFS コマンド

B.1	AdvFS ベース・システム・コマンド	B-1
B.2	AdvFS Utilities のコマンド	B-4
B.3	AdvFS コマンドと UFS コマンドの比較	B-5

C ファイル・システムの変換

C.1	/usr ファイル・システムの AdvFS への変換	C-1
C.1.1	バックアップ・テープの使用	C-1
C.1.2	中間ファイルの使用	C-3
C.1.3	別のディスクの使用	C-5
C.2	ルート・ファイル・システムの AdvFS への変換	C-7
C.3	ファイル・システムの AdvFS への変換	C-9
C.3.1	バックアップ・テープの使用	C-9
C.3.2	中間ファイルの使用	C-11
C.3.3	別のディスクの使用	C-12
C.3.4	別のシステムの使用	C-14
C.4	AdvFS から UFS への変換	C-15

C.4.1	ルート・ファイル・システムの UFS への変換	C-15
C.4.2	ファイルセットの UFS への変換	C-17

D データ管理アプリケーション・プログラミング・インタフェース (DMAPI)

E グラフィカル・ユーザ・インタフェース

E.1	AdvFS GUI のインストール	E-1
E.2	GUI の構成要素	E-2
E.2.1	GUI エージェント (advfsd)	E-2
E.2.2	AdvFS GUI (dtadvfs)	E-3
E.2.2.1	GUI セキュリティ	E-4
E.2.2.2	ディスクの無視	E-5
E.2.2.3	ログ・ファイル	E-5
E.2.2.4	ヘルプ	E-5
E.2.2.5	リモート・ファイル・システムの監視	E-6
E.2.2.6	LSM ボリュームの管理	E-6
E.3	ファイル・システム情報の表示	E-7
E.3.1	情報の表示単位の選択	E-8
E.3.2	メイン・ウィンドウ	E-9
E.3.3	「デバイスとボリューム」ウィンドウ	E-9
E.4	ファイル・システム操作の実行	E-10
E.4.1	ドメインに対する操作	E-10
E.4.1.1	ドメインの作成	E-10
E.4.1.2	ドメインの空き容量警告の設定	E-11
E.4.1.3	ドメインの拡大	E-12
E.4.1.4	ドメインの削除	E-12
E.4.1.5	ドメイン名の変更	E-13
E.4.2	ファイルセットに対する操作	E-14
E.4.2.1	ファイルセットの作成	E-14

E.4.2.2	ファイルセットのマウント	E-15
E.4.2.3	ファイルセットのアンマウント	E-15
E.4.2.4	ファイルセットの空き容量警告の設定	E-16
E.4.2.5	ファイルセット・クォータの設定	E-16
E.4.2.6	ファイルセットの削除	E-17
E.4.2.7	ファイルセット名の変更	E-17
E.4.3	AdvFS ファイルセット・クローンに対する操作	E-18
E.4.3.1	AdvFS ファイルセット・クローンの作成	E-18
E.4.3.2	クローンのマウント	E-19
E.4.3.3	クローンのアンマウント	E-19
E.4.3.4	クローンの削除	E-19
E.4.4	ボリュームに対する操作	E-20
E.4.4.1	ボリュームの空き容量警告の設定	E-20
E.4.4.2	ボリュームの追加	E-20
E.4.4.3	ボリュームの削除	E-20
E.4.5	AdvFS の保守	E-21
E.4.5.1	マルチボリューム・ドメインのバランシング	E-21
E.4.5.2	ドメインの断片化解消	E-22
E.5	GUI のトラブルシューティング	E-22

F アプリケーション・プログラミング・インタフェース (API)

用語集

索引

図

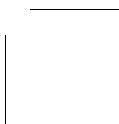
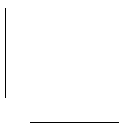
1-1	AdvFS ファイル・システムの構造	1-3
2-1	シングル・ボリューム・ドメイン	2-5
2-2	ドメインの拡大	2-11

2-3	ファイルセット・クローンの作成	2-27
5-1	ドメイン間のバランシング	5-20
5-2	ファイルの移動	5-29

表

1-1	ドメインとファイルセットの各種構成の比較	1-8
1-2	AdvFS ボリュームの各種構成の比較	1-10
2-1	ゴミ箱関連コマンド	2-35
3-1	クォータの限界サイズ	3-3
3-2	ユーザおよびグループ・クォータとファイルセット・クォータの比較	3-5
3-3	ユーザおよびグループの使用状況情報コマンド	3-21
3-4	ユーザおよびグループの使用量コマンド	3-25
4-1	マウントしたオリジナル・ファイルセットのバックアップ・ソースとしての使用	4-5
4-2	マウントしたファイルセット・クローンのバックアップ・ソースとしての使用	4-6
4-3	LSM スプリット・ミラーからマウントしたファイルセットのバックアップ・ソースとしての使用	4-8
4-4	コントローラ・ベースのクローンからマウントしたファイルセットのバックアップ・ソースとしての使用	4-10
4-5	コントローラ・ベースのスナップショットからマウントしたファイルセットのバックアップ・ソースとしての使用	4-13
5-1	アトミック書き込みデータ・ロギングの有効化のチューニング	5-14
5-2	アトミック書き込みデータ・ロギングの無効化のチューニング	5-15
6-1	リンク数の不一致	6-17
6-2	ドメイン・ボリューム・カウントの不一致	6-18
6-3	パーティション数の不一致	6-18

6-4	ディスク・スペースの使用状況に関するコマンド	6-32
B-1	AdvFS 構成コマンド	B-1
B-2	AdvFS 情報表示コマンド	B-2
B-3	AdvFS バックアップ・コマンド	B-2
B-4	AdvFS チェックおよび修復コマンド	B-2
B-5	AdvFS クォータ・コマンド	B-3
B-6	AdvFS ディスク構造コマンド	B-3
B-7	AdvFS Utilities コマンド	B-4
B-8	AdvFS コマンドと UFS コマンドの比較	B-5
E-1	AdvFS GUI サブセット	E-2
E-2	AdvFS GUI に必要なディスク・スペース	E-2
E-3	GUI のトラブルシューティング	E-22



まえがき

本書には、Advanced File System (AdvFS) に関するタスク・レベルの情報と、AdvFS を理解し、その構成とトラブルシューティング、およびチューニングを行うための方法を記載しています。AdvFS は、HP Tru64™ UNIX オペレーティング・システムの標準ファイル・システムです。

AdvFS Utilities は AdvFS のファイル管理機能を拡張し、グラフィカル・ユーザ・インタフェース (GUI) を通じてシステム管理を容易にするユーティリティ・パッケージです。AdvFS Utilities はオプションであり、Tru64 UNIX とは別のライセンスで提供されます。Logical Storage Manager Storage Administrator (LSMSA) は Java で実装された GUI を備えており、LSM オブジェクトと AdvFS ファイル・システムを視覚的に管理できます (『*Logical Storage Manager*』を参照)。AdvFS GUI は共通デスクトップ環境 (CDE) で動作し、AdvFS を視覚的に操作する環境を提供します (付録 E を参照)。

対象読者

本書は主にシステム管理者を対象として、AdvFS ファイル・システムと AdvFS Utilities について説明しています。他の UNIX システムに習熟しているシステム管理者やプログラム開発者は、本書を AdvFS ファイル・システムのガイドブックとして利用できます。

新機能と変更された機能

本書は、Version 5.1B で AdvFS に新たに追加された機能について説明しています。AdvFS の新機能は次のとおりです。

- 新しい `freezefs` および `thawfs` コマンドによるファイル・システム休止機能 (4.1.2.1 項)
- バックグラウンドで動作する最適化ツール `vfast` (5.8 節)
- `mount` コマンドのオプションを使用した動的ボリューム拡張 (2.3.4.3 項)

次の機能は、前回の AdvFS Version 5 リリースで追加されたものです。

- ドメイン復元ツール `/sbin/advfs/fixfdmn` によるメタデータのダメージチェック (6.2.5 項)
- ファイルの一時的なアトミック書き込みデータ・ロギングを行うための新しいマウント・オプション (5.5 節)
- オブジェクト・セーフティ (旧データの表面化防止) によるセキュリティ保護 (オプション)。ファイルセットに属するディスク上の各ページをゼロで充填し、ファイルで使用される前に強制的にディスクに書き出す機能 (2.4.9 項)
- `mkfset` と `chfsets` コマンドによるフラグ・ファイルのオン/オフ切り替え (5.2 節)
- UBC パラメータによる、AdvFS キャッシュのチューニング (5.4 節)
- ドメインによって管理可能なファイルの数を増やすためのディスク構造の改善 (2.3.3 項)
- ダメージを受けたディスクからブロック・レベルでデータを回復するためのディスク・サルベージ・ユーティリティ (6.2.6 項)
- ファイル作成およびファイル・アクセスの速度を向上するためのディレクトリ構造の改善 (2.3.3.1 項)
- `vdump` および `vrestore` コマンドの機能の強化 (第 4 章)
- リモート・デバイスでのバックアップおよびリストアのサポート (第 4 章)
- クォータ限界値の拡張 (第 3 章)
- バッファを使用しない同期 I/O を可能にするダイレクト I/O (5.6 節)
- 連続 I/O を促進する `smooth sync` オプション (5.7 節)
- ディスク使用量を表示するツール `/sbin/advfs/vdf` ユーティリティ (3.4.2.4 項)
- メタデータ表示ユーティリティ (6.2.2 項)
- データ管理 API (DMAPI) (付録 D)

本書の構成

本書の構成を次に示します。

- 第 1 章 AdvFS ファイル・システムを構成するボリューム，ドメイン，
 およびファイルセットについて説明します。
- 第 2 章 システムのセットアップ手順について説明します。
- 第 3 章 ユーザとグループのクォータ，および AdvFS ファイル・システム
 特有のファイルセットのクォータについて説明します。
- 第 4 章 バックアップ・ソースの作成，およびデータのバックアッ
 プとリストアについて説明します。
- 第 5 章 AdvFS ファイル・システムの最適化とチューニングの指針を示します。
- 第 6 章 AdvFS ファイル・システムのトラブルシューティング
 に役立つ情報を提供します。
- 付録 A SysMan Menu の使用方法を説明します。
- 付録 B AdvFS の全コマンドを示し，UFS のコマンドと比較します。
- 付録 C ファイル・システムを AdvFS に変換する方法を説明し，
 サンプル・コードを示します。
- 付録 D データ管理アプリケーション・プログラミング・インタ
 フェース (DMAPI) について説明します。
- 付録 E AdvFS GUI (Graphical User Interface) について説明します。
- 付録 F アプリケーション・プログラミング・インタフェース
 (API) コマンドを示します。

参考文献

Tru64 UNIX のその他の参考文献を次に示します。

- 『*AdvFS Quick Reference*』は，AdvFS の概要を示します。次の URL
で入手することができます。
<http://www.tru64unix.compaq.com/docs/techtips.html>
- 『システム管理ガイド』では，ワークステーションやサーバ上でのオペ
レーティング・システムの管理作業について説明しています。
- 『ハードウェア管理ガイド』では，Tru64 UNIX オペレーティング・シ
ステムを実行するハードウェアの管理作業について説明しています。
- 『インストール・ガイド』では，オペレーティング・システム・
ソフトウェアのインストール方法について説明しています。
- 『*Logical Storage Manager*』では，ストレージ管理に関する情報を記
載しています。

- 『プログラミング・ガイド』では、Tru64 UNIX オペレーティング・システムの開発環境について説明しています。
- 『Tru64 UNIX 概要』では、Tru64 UNIX とその構成要素の概要を紹介しています。
- 『Software License Management』では、LMF (License Management Facility) による Tru64 UNIX 上のソフトウェア・ライセンスの管理について説明しています。
- 『システムの構成とチューニング』では、システム性能のチューニング方法と、カーネル構成に関するより高度な情報を記載しています。
- 『クラスタ管理ガイド』では、TruCluster ソフトウェアを実行するシステムの管理方法を説明しています。
- Best Practices ドキュメントでは、一般的な管理作業の最適な実行方法が紹介されています。
- ストレージ・エリア・ネットワークに関する情報は、次の URL で得ることができます。http://www.compaq.com/storage/software_index.html
- その他、Tru64 UNIX と AdvFS Utilities の『System Limits』, 『リリース・ノート』, および『QuickSpecs』も用意しています。

『ドキュメント概要』には、Tru64 UNIX のドキュメント・セットに含まれる各マニュアルの情報が記載されています。

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>

これらのマニュアルは、ドキュメント CD-ROM にも含まれています。

本書の表記法

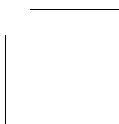
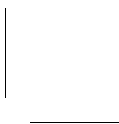
本書では、次の表記法を使用します。

%

\$

パーセント記号は、C シェルのシステム・プロンプトを表します。ドル記号は、Bourne シェル、Korn シェル、および POSIX シェルの場合のシステム・プロンプトを表します。

#	番号記号は <code>root</code> 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。
% <code>cat</code>	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。
[] { }	構文定義では、大カッコはオプションの項目を示し、中カッコは必須項目を示します。大カッコまたは中カッコの中の項目を縦線で区切っている場合は、そこに併記されている項目の中から1つの項目を選択することを示します。
...	構文定義では、水平の反復記号は、前の項目を1回以上繰り返して使用できることを示します。
<code>cat(1)</code>	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、 <code>cat(1)</code> は、 <code>cat</code> 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。
Ctrl/ <i>x</i>	この記号は、スラッシュの前に指定されているキーを押しながら、スラッシュの後のキーまたはマウス・ボタンを押すことを示します。例中では、このようなキーの組み合わせは、四角あるいは大カッコで囲まれて示されます(たとえば、 <code>Ctrl/C</code>)。



ファイル・システムの計画

Advanced File System (AdvFS) は HP Tru64 UNIX オペレーティング・システムの標準ファイル・システムです。この章では、次のトピックについて説明します。

- AdvFS と UFS の比較 (1.1 節)
- ライセンス登録 (1.2 節)
- AdvFS ファイル・システムの設計と構成要素 (1.3 節)
- ファイル・システムの構成 (1.4 節)
- AdvFS の各種ストレージ (1.5 節)

1.1 AdvFS の概要

AdvFS ファイル・システムは、従来の UNIX ファイル・システム (UFS) とは異なります。AdvFS では、システムをシャットダウンすることなく、いつでもシステム構成を変更できます。AdvFS は、AdvFS Utilities とともに使用して、マルチボリューム・ファイル・システムをサポートするので、システム条件の変化に応じて、ストレージを柔軟に追加/削除することが可能です。これに加えて、**LSM (Logical Storage Manager)** のボリュームとストレージ・エリア・ネットワーク (SAN) を AdvFS のストレージとして使用することができます。

これに比べると UFS モデルは柔軟性を欠きます。各ディスク、あるいはディスク・パーティションに 1 つのファイル・システムがあります。UFS のディレクトリ階層構造は、物理ストレージ層に密接に結び付いています。このため、ファイル・システムの空き容量が足りなくなっても、ファイルの絶対パス名を変更しない限り他のディスクへファイルを移動することはできません。論理ディレクトリをディレクトリ・サブツリーに分けてサブツリーを別のディスクにマッピングする作業は、慎重に考えなければなりません。周到に計画しても、UFS モデルではディレクトリ構造の調整には限界があります。

ユーザ側の視点では、AdvFS は従来の UNIX のファイル・システムとほとんど変わりありません。すなわち、各ユーザは `mkdir` コマンドでディレクトリを作成し、`cd` コマンドでディレクトリ間を移動し、`ls` コマンドでディレクトリの内容のリストを表示することができます。また、AdvFS の論理構造とクォータ管理、およびバックアップ機能は、従来のファイル・システムの設計に基づいています。ただし、AdvFS では `newfs`、`dump`、`restore`、`fsck` など、いくつかの標準的なコマンドがサポートされていないか、あるいは他のコマンドで置き換えられています。AdvFS のコマンドとユーティリティ、および UFS のコマンドとの比較についての詳細は、付録 B を参照してください。

システム管理者は、AdvFS システムをオンライン状態に維持したまま、ファイル・システムのバックアップや再構成、およびチューニングを行うことができます。各ユーザは、必要なファイルを誤って削除してしまった場合でも、定義済みのゴミ箱ディレクトリや AdvFS ファイルセット・クローンから、システム管理者の手を借りることなくファイルを復元できます。

1.2 ライセンス登録

AdvFS は Tru64 UNIX オペレーティング・システムの標準ファイル・システムですが、その機能を拡張する AdvFS Utilities は別ライセンスで提供されています。したがって、AdvFS Utilities を使用するには、事前にライセンス **PAK (Product Authorization Key)** を登録する必要があります。詳細については、最寄りの弊社代理店または各支店/営業所にお問い合わせください。

1.3 ファイル・システムの構造

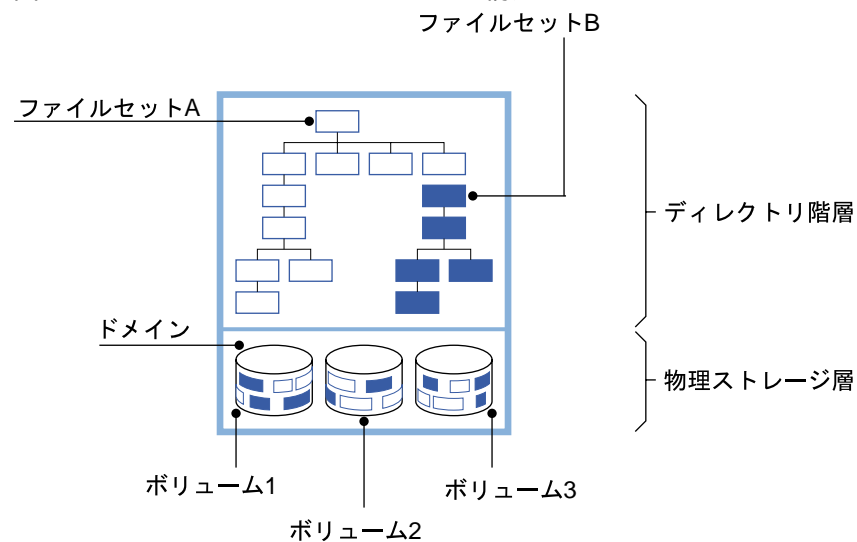
AdvFS ファイル・システムは、ディレクトリ階層レイヤと物理ストレージ・レイヤの 2 種類の階層で構成されます。ディレクトリ階層レイヤは、ファイル・ネーミング・スキームや、ファイルの作成およびオープン、ファイルの読み書きなどの POSIX に準拠した機能を実現します。物理ストレージ・レイヤは、先書きログ、キャッシュ機能、ファイル割り当て、物理ディスク入出力機能を実現します。

このように独立したファイル・システム構造により、ディレクトリ階層レイヤと物理ストレージ・レイヤを別々に管理することができます。ファイルのパス名を変えずに、定義された複数のディスク・ボリューム間でファイルを移動できます。パス名は変わらないため、物理レベルでのこの操作はエンド・ユーザには全く影響を与えません。

AdvFS は、クラスタ構成のファイル・システムです。クラスタの動作もまた、エンド・ユーザには全く影響を与えません。1つのクラスタ上で動作する AdvFS は、エンド・ユーザにとってはほとんど例外なく、1つのノード上で動作するのと全く変わらないように見えます。

AdvFS では、ファイルセットとドメインという独自の概念を実装しており、この2つの概念によって二重構造のファイル・システムが実現されています。図 1-1 は、AdvFS ファイル・システムの構造を示しています。

図 1-1: AdvFS ファイル・システムの構造



AdvFS が使用できる各種ストレージについては、1.5 節を参照してください。

1.3.1 ファイルセット，ドメイン，およびボリューム

ファイルセットとは、従来の UNIX ファイル・システムの論理構造に従ったディレクトリ名とファイル名の階層構造であり、これがシステムにマウントする対象になります。AdvFS は従来のファイル・システムより優れており、ドメインと呼ばれる共通のストレージ・プールを共用する複数のファイルセットを作成することができます。AdvFS ファイル・システムの構成についての詳細は、1.4.1 項を参照してください。ファイルセットの操作についての詳細は、2.4 節を参照してください。

ドメインは物理ストレージ層を表し、ディレクトリ構造とは別に管理されます。ドメインでは、ディレクトリ構造に影響を与えないでボリュームの追加/削除を行うことができます。ドメイン管理についての詳細は、2.3 節を参照してください。

ボリュームは、UNIX ブロック・デバイスと同様に動作するメカニズムです (1.4.2 項を参照)。最初に作成された時点では、どのドメインも単一ボリュームで構成されています。オプションの AdvFS Utilities を使うと、1 つ以上のボリュームを追加して、単一ボリューム・ドメインをマルチボリューム・ドメインに変換することができます。

1.3.2 トランザクション・ログ・ファイル

AdvFS ファイル・システムは、先書きログを採用してファイル・システムの完全性を確保する、ログ・ベースのファイル・システムです。先書きログでは、実際の変更をディスクに書き込む前に、メタデータ (ファイル・システム構造) の変更がトランザクション・ログ・ファイルにすべて書き込まれます。トランザクション・ログ・ファイルの内容は、一定間隔でディスクに書き込まれます。

ドメインを作成すると、AdvFS によって対応するトランザクション・ログ・ファイルが 1 つ作成されます。クラッシュから回復する際、AdvFS はトランザクション・ログ・ファイルを読み取り、ファイル・システム・トランザクションの状態を確認します。完了しているトランザクションはすべてディスクに書き込まれ、完了していないトランザクションは実行が取り消されます。回復速度を左右するのは、ファイル・システム内のデータ量ではなく、ディスクに書き込まれていないトランザクション・ログのレコードの数です。回復は通常、数秒で終わります。従来の UNIX ファイル・システムでは、fsck ユーティリティを使用してシステム障害から回復します。fsck ユーティリティでは、大容量のファイル・システムの検査と修復には数時間を要する場合があります。

省略時の設定では、ファイル・システム構造のみがロギングされます。ただし、ファイル・データをロギングするように選択し、システムがストレージに書き込みを行う方法を変更することもできます (5.5 節を参照)。データ・ロギングを有効にしておくと、システム・クラッシュが発生しても内部的にファイルの整合性が維持されますが、システムの性能が低下する可能性があります。

1.3.3 ファイル・ストレージの割り当て

ファイルは静的なものではなく、時間の経過とともに必要な容量が変化します。スペースを過度に割り当てず、ディスク上で連続的にファイルを配置できるように、AdvFS は独自のファイル・ストレージ割り当てスキームを使用します。

ファイル・ストレージ割り当てに際しては、以下のような機能が使用されます。

- エクステンツ

エクステンツは、AdvFS がファイルに割り当てるディスク・スペースの連続的な領域です。ファイルに対してストレージが追加されると、そのストレージはエクステンツとしてグループ化されます。ファイル・システムが存在する各ボリュームには、エクステンツ・マップが存在します。

エクステンツの数が少ないほどファイル I/O は効率的に行われます。ファイルが多数の小さなエクステンツから構成される場合は、同様のサイズで大きないくつかのエクステンツからなるファイルよりも、ファイルの読み取りおよび書き込みのための I/O 処理に時間がかかることになります。

ファイル・システムには動的な性質があるため、ファイル・ストレージの割り当てが常に連続するページになり、大きなエクステンツができるとは限りません。次のような要因が、ストレージの割り当てに影響を与えます。

- ディスク断片化 (ディスク・フラグメンテーション)

ディスクが過度に断片化すると、多数の小さな空きスペースが存在することになります。この場合、AdvFS は連続するページを利用できないため、データを非連続の物理ページに書き込みます。その結果、ファイルは多数のエクステンツに細分化されることになります。

- 複数のユーザ

システム上に多くのユーザがいる場合は、必要なディスク容量が増加し、ファイルに対して連続領域を割り当てるのが難しくなります。

多数のエクステンツから構成されるファイルがドメインに存在する場合には、`vfast` ユーティリティを使用してファイルの配置を最適化することができます (5.8 節を参照)。

- ブリアロケーション (事前割り当て)

ファイルが拡張されるたびに、AdvFS は、ファイル・サイズの 1/4 (最高 16 ページまで) を事前に割り当てて、ファイルにページを追加します。余分に割り当てられた事前割り当てスペースは、ファイルのクローズ時に解放されます。

マルチボリューム・ドメインでは、新しいファイルは各ボリュームに順番に割り当てられます。86% 以上使用された (割り当てられた) ボリュームは、すべてのボリュームが 86% 以上使用された状態にならない限り、新しいファイルへの割り当てには使用されません。既存のファイルにデータが追加されたときには、ボリュームが一杯にならない限り、ストレージはファイルが最初に割り当てられたボリューム上に割り当てられます。

- フラグメント

AdvFS は、ページ単位でファイルをディスクに書き込みます。最後のページの未使用領域が、割り当て済みスペースの 5 % 以上を浪費することになるファイルでは、ファイル・フラグメントが作成されます。フラグメントは、ファイルセットのフラグ・ファイルから割り当てられます。フラグ・ファイルは、未使用の無駄なディスク・スペースの量を大幅に減らす手法として効果的ですが、その一方でファイル・システムの性能に影響を与える可能性もあります。ディスク・スペース使用状況がそれほど問題ではない場合は、フラグ・ファイルを無効にすることができます (5.2 節を参照)。

- スパース・ファイル

スパース・ファイルとは、ファイル内のページの一部に対してディスク・ストレージが確保されていないファイルのことです。ストレージが確保されていないページでスパース・ファイルの読み取りを行うと、ゼロが返されます。ストレージが確保されていないページに書き込みを行うと、ディスク・ストレージのページが割り当てられます。

スパース・ファイルは、`ftruncate(2)` コマンドでファイルの終端を設定するか、`lseek(2)` コマンドと `write(1)` コマンドを使用してファイル内の書き込まれていない部分をスキップして書き込めば作成できます。

`core` ファイルはスパース・ファイルです。`core` ファイルは、情報が含まれていない大きな領域を持ち、データが存在しない場所にはディスク・ブロックは割り当てられていません。また、クォータ・ファイルは、

ユーザ ID によってインデックスされるためスパース・ファイルです。各ユーザの ID が連続していない場合、クォータ・ファイルにはデータが存在しない部分が含まれる可能性があります。

一方、データベース・アプリケーションでは通常、データが存在しない場合でもファイル全体のストレージが確保されます。一般に、データベース・アプリケーションは、データが存在しないページにゼロを書き込みます。データベース・アプリケーションでは、データを順番に書き込むことによって、連続する多数のページを含み、かつエクステント数が少ないデータベース・ファイルを作成します。

ディスク・ストレージを持たないページも含めたスパース・ファイルの大きさを調べるには、`ls` コマンドを `-l` オプションを指定して使用します。`ls` コマンドを `-s` オプションを指定して使用すると、実際にファイルが使用しているストレージの総量が表示されます。6.4.1 項に、ディスク使用状況を表示する、何通りかの方法を詳しく説明していますので、参照してください。

1.4 AdvFS ファイル・システムの構成

構成の計画段階で、`/` (ルート)、`/usr`、および `/var` ファイル・システムを AdvFS に設定することを検討してください。ルートと `/usr` に AdvFS を使用すれば、構成の柔軟性が高まるとともに、システム障害時のダウン時間を大幅に短縮させることができます。

従来の UFS 構成と同じように、ドメインごとにパーティション (ボリューム) が 1 つで、各ドメインにはファイルセットが 1 つあるという形式で AdvFS を設定できます。オプションの AdvFS Utilities を利用できる場合は、スペースが必要になった時点でボリュームを追加して (ボリュームが 1 つに制限されているローカルのルート・ファイル・システムを除く)、既存のドメインのサイズを大きくすることができます。その際、既存の構成を変更する必要はありません。

1.4.1 ドメインとファイルセットの構成

複数のファイルセットからなる 1 つのドメインを作成することもできますし、それぞれのファイルセットについてドメインを作成することもできます。また、これら 2 つ方法を取り混ぜてシステムを構築することもできます。AdvFS Utilities を利用できる場合、ドメインを複数のボリュームに分散させることもできます。

十分な数のドライブを利用できる場合、1つのドメインに1つのディスクを使用してください。ドメインには、同じディスク上のいくつかのパーティション(たとえば、a、b、g、hなど)を追加するよりも、1つのパーティション(通常パーティション c)を追加することをお勧めします。

パーティションを異なるドメインに分割しないでください。たとえば、パーティション a をあるドメインに追加し、パーティション b を別のドメインに追加しないでください。

同じタイプおよび同じ速度のディスクを複数使用する場合には、一般的にこれらの各ディスクにドメインを分散させた方が、効率化を図ることができます。たとえば、独立した3つのディスク・ボリュームを持つドメインは、1つのディスク上に3つのパーティションを持つドメインよりも効率的です。これは、後者の構成では、I/O パスが1つしかないためです。

複数のドメインを作成すると、物理リソースをより自由に制御できるようになります。ドメインを、組織内で意味のあるまとまり、たとえば、特定のプロジェクト、ユーザのグループ、部門、または事業所ごとに使用するように作成することができます。たとえばドメインを、組織内のエンジニアリング、財務、人事などの部門ごとに作成することができます。

データ管理アプリケーション・プログラミング・インタフェース (DMAPI) を使用可能にしたドメイン (付録 D) では、1つのファイルセットしか使用できません。

表 1-1 に、各種構成の長所と短所を示します。

表 1-1: ドメインとファイルセットの各種構成の比較

構成	長所	短所
ディスク全体を1つのドメインに使用。小さなパーティションが複数の物理デバイスに分散していない。	効率がよい。デバイスの I/O ストリームは共有されない。	大きなボリューム上に小さなドメインがある場合、スペースを無駄に使う。
ドメインが、1つのボリュームにではなく複数のボリュームにわたって構成されている (Advanced Utilities が必要)。	I/O 負荷を分散。スループットが向上。	ボリュームの1つに障害が発生すると、ドメイン全体にアクセスできなくなる。

表 1-1: ドメインとファイルセットの各種構成の比較 (続き)

構成	長所	短所
同じ数のボリューム上に、1つの大きなドメインではなく、小さなドメインが複数存在。	復旧が速い。スループットが向上。いくつかの管理作業を省略可能。	管理上のオーバーヘッドが増える可能性がある。
ドメイン上に1つの大きなファイルセットではなく、多数のファイルセットが存在。	個々のファイルセットに対して限界値を設けることができ、資源をより細かに制御できる。	管理上のオーバーヘッドが増える。あるユーティリティを実行するのにすべてのファイルセットをマウントしなければならない場合がある。
高速ボリュームまたは輻輳のないボリューム上にトランザクション・ログがある。	ログにより I/O のボトルネックが発生しない。	ハードウェアのコストがかかる。
ルート・ファイルセットを AdvFS として構成。	システム・クラッシュの後の復旧が速い。	なし

どのファイル・システムを使用しているかは、次の方法で調べることができます。

- `showfsets` を使用して、ドメイン内のファイルセットの数とそれぞれのファイルセットの大きさを表示します。
- `showfile` コマンドを使用して、ファイルが存在するボリュームを調べます。頻繁に使用する複数のファイルが、同じボリューム上に存在しないようにします。

第 5 章 に、ファイル・システムを最適化する方法について詳しく説明しています。『システムの構成とチューニング』に、ファイル・システムの計画と構成についての詳細なガイドラインがあります。グラフィカル・ユーザ・インタフェースを使って、ドメインを構成する方法は、付録 E を参照してください。

1.4.2 ボリュームの構成

ボリュームは、ブロック特殊デバイスとなるエンティティです。1つのディスク・パーティション、ディスク全体、LSM により作成されたボリュームの集合体、ハードウェア RAID、または SAN などがこのエンティティに相当します。ボリュームはそれぞれ、1つのドメインにしか割り当てできま

せん。そのドメインには、ドメイン **ID** によって関連付けられます。この ID は、そのボリュームの AdvFS メタデータに自動的に保存されます。オプションの AdvFS Utilities を使用する場合、1 つのドメインに複数のボリュームを追加することができます。

AdvFS のドメインのそれぞれのボリュームには、ビットファイル・メタデータ・テーブル (**BMT**) とストレージ・ビットマップが存在します。BMT にはファイルの構造に関する情報 (メタデータ) が格納されます。ストレージ・ビットマップには、ディスクの空き状況と割り当て状況の追跡結果が記録されています。これに加えて、各ドメインの個々のボリュームには、トランザクション・ログ・ファイルが格納されます。これには、ディスクに書き込まれるまでのすべてのメタデータの変更が格納されます。

次の表には、AdvFS ボリューム上にドメインを構成するさまざまな方法の長所と短所を示します。

表 1-2: AdvFS ボリュームの各種構成の比較

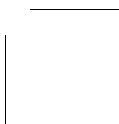
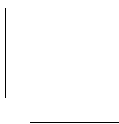
構成	長所	短所
ボリュームがディスク・パーティションに相当する。	ディスクの小さい領域も使用できる。	サイズの制約。他のパーティションとの I/O の競合が発生。
ボリュームが 1 つのディスクに相当する。	1 つのパーティションを使用するよりも効率的。	サイズの制約
ボリュームが 1 つの RAID ボリューム (LSM かハードウェア) または SAN に相当する。	性能と信頼性が向上する。利用可能な容量が増える。	コストがかかる。LSM ライセンスか、追加のハードウェアが必要。

1.5 AdvFS のストレージの選択

ドメインを作成すると、AdvFS はそのドメインに関連付けられたブロック特殊デバイス名を用いて、ストレージを識別できるようになります。ストレージには、直接接続されたディスク、ソフトウェア RAID ボリューム、ハードウェア RAID、およびストレージ・エリア・ネットワークなどを使用できます。

LSM、ハードウェア RAID、および SAN のボリュームを使用すると、高度なバックアップや動的ボリューム拡張などの機能を利用することができます。LSM の詳細は、『*Logical Storage Manager*』を参照してください。ハードウェア RAID と SAN の各種製品については、

<http://www.tru64unix.compaq.com/storage/index.html> を参照
してください。



ファイル・システムのセットアップ

ファイル・システムの構成を決定すると、これを構築する作業を行なうことができます。この章では、次のトピックについて説明します。

- AdvFS ファイル・システムのセットアップ (2.1 節)
- ボリューム (2.2 節)
- ドメイン (2.3 節)
- ファイルセット (2.4 節)
- AdvFS ルート・ファイル・システムの構成 (2.5 節)
- 削除したファイルを復元するためのシステムのセットアップ (2.6 節)

2.1 概要

AdvFS ファイル・システムを稼働させるのに必要な最小限の構成は、1 つのドメインと 1 つのマウント済みファイルセットです。

アクティブ・ドメインを作成する手順は、以下のとおりです。

1. ドメインを作成し、ボリュームを割り当てる (2.3.3 項)。
2. ファイルセットを作成する (2.4.3 項)。
3. マウント・ポイント・ディレクトリを作成する (2.4.5 項)。
4. ファイルセットをマウントする (2.4.5 項)。

ドメインごとに 1 つまたは複数のファイルセットを作成できます。ガイドラインについては、1.4.1 項を参照してください。

AdvFS は、`/etc/fstab` ファイルに完全に対応しており、システム・リブート時にファイルセットを自動的にマウントすることができます (2.4.1 項を参照)。この操作は、グラフィカル・インタフェース (付録 E) または SysMan Menu (付録 A) を使用して行なうこともできます。

AdvFS コマンドの一覧は、付録 B を参照してください。AdvFS の構成についての詳細は、『システムの構成とチューニング』を参照してください。

2.2 ボリューム

AdvFS ファイル・システムでは、raw ディスク・パーティション、ディスク全体、LSM (Logical Storage Manager) の集合ボリューム、あるいはハードウェア/ソフトウェア RAID (Redundant Array of Independent Disk) ストレージなどをボリュームとして設定することができます。

オプションの AdvFS Utilities を使用すれば、ボリュームを追加してマルチボリューム・ドメインを作成することができます。マルチボリューム・ドメインは、ファイルセットで使用可能なストレージ容量を増加させるとともに、予防的なディスク管理を可能にします。ボリュームの追加は、ドメイン作成直後でも、ファイルセットの作成およびマウント前でも可能です。予防的なディスク管理を行うためには、新しいボリュームをドメインに追加し、既存のファイルを新しいボリュームへ移行し、それらのファイルを古いボリュームから削除します。

ボリュームをドメインに関連付ける方法については、2.3 節を参照してください。

2.2.1 ボリューム属性

AdvFS ボリュームは、データの読み込み、キャッシュ、書き込み、および連結方法を決定する属性とともに構成できます。ドメインの初期作成やボリュームの追加によって AdvFS ボリュームがドメインに組み込まれる際に、省略時のボリューム属性が設定されます。これらの省略時の属性を変更すると、システム構成によっては性能が向上する場合があります。

現在のボリューム属性を表示または変更するには、SysMan Menu の「AdvFS ドメインの管理」ユーティリティを使用するか (付録 A 参照)、コマンド行で次のように `chvol` コマンドを実行します。

```
chvol device_name domain_name
```

`chvol` コマンドは、LSM のボリューム名の省略記法をサポートしています。以下の 2 つは同じ意味になります。

```
# chvol dom1 testdg.vol1
# chvol dom1 /dev/vol/testdg/vol1
```

2-2 ファイル・システムのセットアップ

アクティブ・ドメインのボリュームの属性は、システムを停止することなく、いつでも変更することができます。システム性能を改善するための属性の変更については、第 5 章および『システムの構成とチューニング』を参照してください。

2.3 ドメイン

ドメインとは、AdvFS ファイル・システムの物理ストレージ層のことです。ドメインは、1 つあるいは複数のボリュームを物理ストレージのプールとして定義したものです。このストレージはディレクトリ構造とは別に管理されるため (2.3.1 項)、ボリュームを追加/削除することにより、ドメインの大きさを拡大/縮小することができます。またファイルのパス名を変えずに、ドメイン内のボリューム間でファイルを移動できます。ドメインの名前を変更しても、ドメイン ID に影響はありません。

ドメインには、重複しない名前を付ける必要があります。ドメイン名には、スラッシュ (/)、シャープ記号 (#)、コロン (:)、アスタリスク (*)、疑問符 (?), タブ, 改行, フォームフィード, キャリッジ・リターン, 垂直タブ, および空白を含めることはできません。

2.3.1 /etc/fdmns ディレクトリ

/etc/fdmns ディレクトリでは、作成するドメインごとにサブディレクトリを用意することにより、ドメインを定義します。ドメイン名は、ディレクトリ・エントリとして定義されます。ドメインを作成すると、そのディレクトリ・エントリが作成されます。このサブディレクトリには、ドメイン内の各ボリュームへのシンボリック・リンクが含まれます。

このディレクトリは、標準の AdvFS コマンドを使用すると、AdvFS によって自動的に作成/管理されます。ドメインでファイルセットをマウントするためには、有効な /etc/fdmns ディレクトリが必要になります。

ドメインを作成すると、/etc/fdmns ディレクトリのドメイン・エントリからブロック・デバイスへのソフト・リンクが作成されます。このディレクトリ内にリンクを作成するだけでは、ドメインは作成できません。

/etc/fdmns ディレクトリは、定期的にバックアップを取ってください。このディレクトリの内容が壊れた場合、またはディレクトリ自体を削除してしまった場合は、最も新しいバックアップからディレクトリをリストアしてください。このディレクトリにダメージを受けると、そのドメインに

はアクセスできなくなります。これは、ドメインとファイルセットを含む物理ボリュームとの対応に関する情報が正しくなくなるためです。ファイルセットには影響はありません。

AdvFS が正しく機能するためには、同じドメイン ID を持つドメイン内のボリュームの数が、メタデータ情報に保存されたボリュームの数と等しくなければなりません。さらに、各ドメインは、`/etc/fdmns` ディレクトリのエントリに定義します(2.3.1 項を参照)。このディレクトリには、最新情報が定義されている必要があります。つまり、ドメイン・エントリはその対応するボリュームを正しく参照していなければなりません。`/etc/fdmns` ディレクトリのボリュームへのリンクの数は、ボリュームの数に等しくなければなりません。

ダメージを受けたディレクトリを持つドメインからファイルセットをマウントしようとする、次のようなメッセージが表示されます。これはドメイン `accounts` についてのものです。

```
Volume count mismatch for domain accounts.  
accounts expects 2 volumes, /etc/fdmns/accounts has 1 links.
```

`/etc/fdmns` ディレクトリの内容が破損したり、ドメインのボリュームが別のシステムに移された場合は、`/sbin/advfs/advscan` ユーティリティを使用してボリュームを検索し、ディレクトリを再構築します(6.3.3 項を参照)。

2.3.2 ドメイン情報の表示

ドメインが使用状態である場合、つまり 1 つ以上のファイルセットがマウントされている場合には、`showfdmn` コマンドを使用して、ドメインとそれに含まれるボリュームの詳しい情報を表示することができます。

```
showfdmn domain_name
```

たとえば、`domain_1` ドメインの情報を表示するには次のようにします。

```
# showfdmn domain_1
```

	Id	Date Created	LogPgs	Version	Domain Name
	2bb0c594.00008570	Fri Mar 22 12:33	2002 512		4 domain_1

Vol	512-Blks	Free	% Used	Cmode	Rblks	Wblks	Vol Name
1L	8325	79210	90%	on	128	128	/dev/disk/dsk1c
2	832527	1684	98%	on	128	128	/dev/disk/dsk2c

	1665054	80894	94%				

ディスク使用状況とクォータ値を調べるには、3.4 節を参照してください。

2-4 ファイル・システムのセットアップ

2.3.3 ドメインの作成

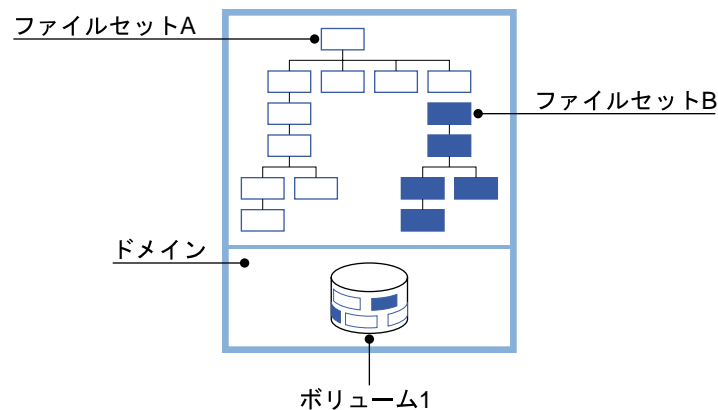
AdvFS ファイル・システムの設定の最初のステップは、ドメインを作成し、そのドメインに初期ボリュームを割り当てることです。ただし、ドメインは、マウント可能な完全なファイル・システムではありません。AdvFS ファイル・システムをマウントするためには、ドメインに1つ以上のファイル・セットが含まれている必要があります。ファイルセットをマウントすることにより、ファイルにアクセスできるようになります(2.4 節を参照)。ドメインとファイルセットの各種構成の長所短所については、1.4.1 項を参照してください。

UFS ファイル・システムを AdvFS に変換する方法、またはルート・ドメインを AdvFS として構成する方法は、付録 C を参照してください。

単一のファイルセットでシングル・ボリューム・ドメインを作成することは、従来の UFS ファイル・システムを作成するのと同じことです。図 2-1 に示すようなアクティブなシングル・ボリューム・ファイル・システムを設定するには、SysMan Menu の「新しい AdvFS ドメインの作成」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E 参照) を使用するか、あるいはコマンド行で次のように `mkfdmn` コマンドを実行します。

```
mkfdmn volume_name domain_name
```

図 2-1: シングル・ボリューム・ドメイン



あるパーティションにドメインを作成し、それが `/etc/fdmns` ディレクトリにエントリがあるパーティションとオーバーラップする場合、つまり、そのパーティションがあるドメインと既に関連付けられていた場合、エラー・

メッセージが表示されます。このエラー・メッセージを無視することもできますし、mkfdomn コマンドを -F オプションをつけて実行し、このテストをバイパスすることもできます。

ディスク・ラベルに使用中であることが表示されているボリュームに、ドメインを作成しようとしたときに、そのボリュームがマウントされているかまたは現在のスワップ・パーティションの場合、mkfdomn コマンドは失敗します。ボリュームがマウントされていない場合は、警告メッセージが表示されます。この警告を無視し、そのドメインを作成することもできます。たとえば、/dev/disk/dsk3g が使用中で、ドメイン usr_domain を作成しようとした場合、次のメッセージが表示されます。

```
# mkfdomn /dev/disk/dsk3g usr_domain
Warning: /dev/rdisk/dsk3g is marked in use for 4.2BSD.
If you continue with the operation you can
possibly destroy existing data.
CONTINUE? [y/n] <y>
```

mkfdomn コマンドは、LSM のボリューム名の省略記法をサポートしています。以下の 2 つは同じ意味になります。

```
# mkfdomn dom1 testdg.vol1
# mkfdomn dom1 /dev/vol/testdg/vol1
```

2.3.3.1 ドメイン・バージョン番号

ドメイン・バージョン番号 (**DVN**) は、ドメインに対応付けられています。新しいドメインの DVN は、Version 5.0 オペレーティング・システムで初めて 4 になりました。これより前のドメインの DVN は 3 です。

Version 5.0 より前のオペレーティング・システムで作成されたドメインは、いずれもこれ以降のバージョンで認識されますが、自動的に新しい構造にアップグレードされることはありません。

DVN3 のファイルセットを Version 5.0 以降が動作しているシステムにマウントすることはできます。Version 5.0 以降で作成された DVN4 ドメインのファイルセットを、Version 4 オペレーティング・システムが動作する環境にマウントすることはできません。以前のバージョンのオペレーティング・システムを動作させ、DVN4 ファイルセットにアクセスする必要がある場合は、Version 5.0 以降のソフトウェアが動作するサーバからそれを NFS マウントします。したがって、アプリケーションで旧バージョンとの互換性が必要な場合は、ドメインをアップグレードしないでください。

DVN4 ドメインではディスク構造が改善されており、2T バイトを超えるクォータ値をサポートし、何千ものファイルを含むディレクトリの性能が向上しています。ディレクトリが 1 ページ (約 200 ファイル) を超えると、DVN4 のドメインには自動的にインデックスが作成されます。これは 5000 を超えるファイルがあるディレクトリで特に有効的に機能します。ディレクトリにインデックスが付いているかどうかを調べるには、`showfile` コマンドを使用します。同コマンドを `-i` オプション付きで実行すると、インデックスについての情報が表示されます。詳細については、`showfile(8)` を参照してください。

2.3.3.2 DVN4 のドメインの作成

Version 5.0 以降のオペレーティング・システムで作成されたすべてのドメインは DVN 4 です。したがって、完全なインストールの場合、その過程で作成されたすべてのドメインは新しい構造を持つことになります。

Version 4 のオペレーティング・システムが動作するシステムからのアップデート・インストールの場合は、すべての既存のドメインは DVN 3 のままです。したがって、`/root`、`/usr`、および `/var` もまた DVN 3 になります。変換のユーティリティはありません。データを DVN4 に移行する場合は、2.3.3.4 項を参照してください。

アクティブな、DVN4 のシングル・ボリューム・ドメインを作成する手順は以下のとおりです。

1. `mkfdmn` コマンドを使用して、そのボリュームに関連するシングル・ボリューム・ドメインを作成します。
2. `mkfset` コマンドを使用して、1 つ以上のファイルセットを作成します。
3. `mkdir` コマンドを使用して、マウント・ポイントとなるディレクトリを作成します。このディレクトリには、マウントするファイルセットと同じ名前を付けるのが慣例です。たとえば、`tmp` という名前のファイルセットをマウントするディレクトリには、`/tmp` という名前を付けます。
4. `mount` コマンドを使用して、各ファイルセットをマウントします。

注意

保存しておきたいデータのあるボリュームには、`mkfdmn` コマンドは使用しないでください。このコマンドを使用すると、その

ボリュームのデータが壊れます。誤って `mkfdmn` コマンドを使用した場合、壊れたボリュームが AdvFS ボリュームであれば、`/sbin/advfs/salvage` ユーティリティを使用してデータの一部を復元できる場合もあります (6.2.6 項を参照)。

次の例では、DVN4 のシングル・ボリューム・ドメイン `domain_2` を作成し、そのドメインに 2 つのファイルセット `fileset_a` および `fileset_b` を作成しています。このドメインのボリュームは 1 つだけなので、それぞれのファイルセット上のファイルは、物理的には 1 つのボリューム上に存在します。ドメイン名とファイルセット名の間のシャープ記号 (#) は構文の一部であり、コメントを示すものではありません。

```
# mkfdmn /dev/disk/dsk2c domain_2
# mkfset domain_2 fileset_a
# mkfset domain_2 fileset_b
# mkdir /fileset_a
# mkdir /fileset_b
# mount domain_2#fileset_a /fileset_a
# mount domain_2#fileset_b /fileset_b
```

ドメインはさまざまな種類のストレージでセットアップできます。詳細については、1.5 節を参照してください。AdvFS Utilities を利用できる場合、ボリュームを追加することによってドメインのサイズを変更することができます。クラスタ化されていないルート・ドメインの場合を除いて、シングル・ボリューム・ドメインは、マルチボリューム・ドメインに編成し直すことができます。SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A)、AdvFS GUI (付録 E)、および 2.3.4 項を参照してください。

2.3.3.3 DVN3 のドメインの作成

Version 5 システムで DVN3 ドメインを作成することができます。 `mkfdmn` コマンドに `-v3` オプションを指定して実行します。これは旧バージョンとの互換性が必要な場合に便利です。

DVN3 ドメイン作成についての詳細は、`mkfdmn(8)` を参照してください。作成した DVN3 ドメインをマウントする方法については、2.4.5 項を参照してください。

2.3.3.4 ドメインの DVN4 へのアップグレード

ドメインをアップグレードするには、Version 5.0以降のソフトウェアが動作しているシステムで新しいドメインを作成し、古いドメインから新しいドメインへすべての情報をコピーします。ドメインの作成に精通していない場合、まず 2.3.3 項を参照してください。ドメインのアップグレードの手順は以下のとおりです。

1. `vdump` コマンドを使用して、ドメインのファイルセットのバックアップをテープに取ります。セーブセット・エラーを防ぐために `-x` オプションを使用することをお勧めします。新しいドメインを作成した後にドメインのデータをリストアできない場合は、それらのすべてのデータを失うことになります。
2. 古いドメインをアンマウントし、`rmfdmn` コマンドを使用して削除します。
3. `mkfdmn` コマンドを使用して新しいドメインを作成します。新しいドメインの DVN は 4 になります。`mkfdmn` コマンドを実行するときは、`-x` と `-p` オプションを使用する必要はありません。DVN4 ドメインでは、さらに BMT 割り当てを行なう必要はありません。
4. `mkfset` コマンドを使用して新しいファイルセットを作成し、それらをマウントします。
5. `vrestore` コマンドを使用して、ファイルセットを新しいドメインにリストアします。

たとえば、ファイルセット `fileset_p` と `fileset_m` のある `/dev/disk/dsk1c` 上のドメイン `domain_p` を、同じ名前および同じボリュームでアップグレードするには、次のコマンドを実行します。

```
# vdump -0 -N -x 8 /fileset_p
# vdump -0 -N -x 8 /fileset_m
# umount /fileset_p
# umount /fileset_m
# rmfdmn domain_p
# mkfdmn /dev/disk/dsk1c domain_p
# mkfset domain_p fileset_p
# mkfset domain_p fileset_m
# mount domain_p#fileset_p /fileset_p
# mount domain_p#fileset_m /fileset_m
# vrestore -x -D /fileset_p
# mt fsf 1
# vrestore -x -D /fileset_m
```

2.3.4 ドメイン・サイズの拡大

次のいずれかの方法でドメインを拡大することができます。

- ドメインに別のボリュームを追加する (AdvFS Utilities が必要)。
- ドメイン内のボリュームをより容量の大きなストレージ・デバイスに置き換える。
- ドメインが置かれているストレージの大きさを増やし、それをドメインに知らせる。

サイズを拡大/縮小してもディレクトリ構造には影響ありません。ファイルへのすべてのパス名はそのままです。また、処理中もファイル・システムを稼働させておくことができます。

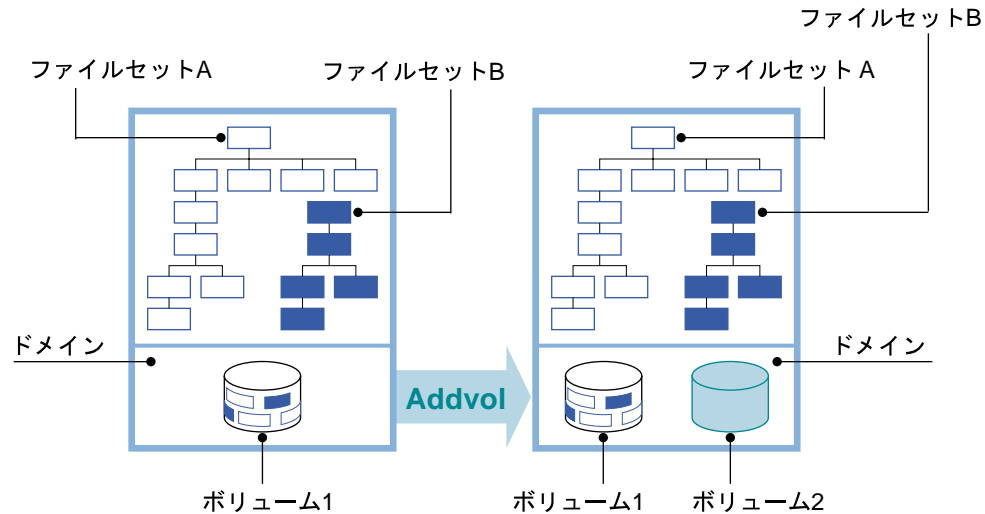
クラスタ環境の場合を除いて、ルート・ドメインは 1 つのボリュームに制限されます。ルート・ドメインのサイズを拡張する必要がある場合、ボリュームの状況によって、対応するストレージを拡張してそれをドメインに伝えるか、6.4.9 項の手順を行なってルートが置かれているデバイスを交換するかができます。

クォータが設定されている場合を除き、ドメインのすべてのファイルセットは利用可能なストレージにアクセスできます (クォータについては、第 3 章 を参照)。

2.3.4.1 ボリュームを追加することによるドメインのストレージの拡大

新しく作成したドメインは、ディスク、ディスク・パーティション、または論理ボリュームなどの 1 つのボリュームからなります。AdvFS Utilities を利用する場合、ドメインのサイズは 1 つ以上のボリュームを追加することによって拡張することができます。図 2-2 に、ドメインにボリュームを追加する様子を示しています。ボリュームを最初に追加した状態では、その中にデータは何も含まれません。ボリュームを追加した後は、ファイルをドメインに追加することで情報がその中に保存されます。また、vfast ユーティリティの実行 (5.8 節を参照)、ファイルの移動 (5.12 節 を参照)、およびドメインの負荷分散 (5.11 節を参照) によっても情報が保存されます。

図 2-2: ドメインの拡大



ボリュームを追加するには、SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E 参照) を使用するか、コマンド行で次のように `addvol` を実行してください。

```
addvol device_name domain_name
```

たとえば、ドメイン `resources` にボリューム `dsk3c` を追加するには次のコマンドを入力します。

```
# addvol /dev/disk/dsk3c resources
```

`addvol` コマンドは、LSM のボリューム名の省略記法をサポートしています。以下の 2 つは同じ意味になります。

```
# addvol dom1 testdg.vol1
# addvol dom1 /dev/vol/testdg/vol1
```

ボリュームの追加は、ドメイン作成直後でも、あるいは、ドメインで追加スペースが必要となった時点でも可能です。また、ボリュームの追加は、ファイルセットがマウントされ使用中のアクティブなドメインに対しても行うことができます。

DVN3 ドメインのサイズを増やすより、新しいファイル構造 (2.3.3.4 項) にアップデートするほうが、性能が向上する場合があります。

クラスタ環境でない場合は、ルート・ドメインにボリュームを追加することはできません (『クラスタ管理ガイド』を参照)。クラスタ構成で実行している場合は、ルート・ドメインに他のドメインと同じ要領でボリュームを追加できます。クラスタを使用していない環境でルート・ドメインのサイズを拡張するには、6.4.9 項を参照してください。

注意

`addvol` コマンドを使用して、保存しておきたいデータのあるボリュームを追加しないでください。このようにするとそのボリューム上のデータが壊れます。誤って `addvol` コマンドを使用した場合には、`/sbin/advfs/salvage` ユーティリティを使用してデータの一部を復元できる場合もあります (6.2.6 項を参照)。

ドメインにボリュームを追加するには、次の手順を行ないます。

1. `showfdmn` コマンドを使用して、ドメインの内容と各ボリュームの現在のディスク容量を表示します。この操作はオプションです。
2. 新しいボリュームをドメインに追加します。

パーティションを追加し、それが `/etc/fdmns` ディレクトリにエントリのあるパーティションとオーバーラップすると、つまり、そのパーティションがあるドメインとすでに関連付けられていると、エラー・メッセージが表示されます。このエラー・メッセージは無視することができますし、`addvol` コマンドに `-F` オプションを使用して、このテストをバイパスすることもできます。詳細については、`addvol(8)` を参照してください。

注意

`-F` オプションを使用するには細心の注意が必要です。オーバーラップ・チェックを無効にすると、広範囲のデータが失われることがあります (`addvol(8)` を参照)。

3. `vfast` ユーティリティを実行しない場合 (5.8 節を参照)、`balance` ユーティリティを使用してボリューム間のファイルの分散を均等化することをお勧めします。

次の例では、1つのディスクのボリューム /dev/disk/dsk3c を domain_1 に追加しています。

```
# showfdmn domain_1
      Id          Date Created      LogPgs Version Domain Name
2bb0c594.00008570 Fri Mar 22 12:33 2002 512          4 domain_1

Vol 512-Blks  Free % Used Cmode Rblks Wblks  Vol Name
1L  832527 79210   90%  on   128   128  /dev/disk/dsk1c
2    832527  1684   98%  on   128   128  /dev/disk/dsk2c
-----
      1665054 80894   94%
# addvol /dev/disk/dsk3c domain_1
```

2.3.4.2 ボリュームを交換することによるドメインのストレージの拡大

Advanced Utilities を利用できない環境でドメインのサイズを拡張するには、次の手順に従います。

1. より容量が大きいストレージ・デバイス上に新規ドメインを作成します。既存ドメインとは異なるドメイン名を使用してください。
2. 既存ドメインのファイルセットと同じ名前のファイルセットを作成します。
3. 作成したファイルセットごとに、一時的なマウント・ポイントを作成します。
4. 作成した一時マウント・ポイントに各ファイルセットをマウントします。
5. 適切なユーティリティ (vdump/vrestore, cpio, cp -R, tar など) を使用して、マウントした各ファイルセットに既存デバイスから対応するファイルセットをコピーします。
6. 新旧の各ファイルセットをアンマウントします。
7. 新しいドメインの名前を、必要に応じて旧ドメインの名前に変更します。ドメインとファイルセットの名前を変更しない場合には、/etc/fstab ファイルを編集する必要はありません。名前を変更した場合には、/etc/fstab ファイルにも変更を反映してください。
8. 新しい各ファイルセットを、旧ファイルセットのマウント・ポイントにマウントします。この結果、ディレクトリ構造の一貫性が保持されます。作成した一時マウント・ポイントのディレクトリはすべて削除します。

2.3.4.3 既存のボリュームを拡張することによるドメインのストレージの拡大

既存のボリュームのサイズを増やすことによって、利用可能なストレージの容量を拡大することができます。たとえば、LSM とハードウェア RAID のコントローラは、ボリュームの動的なサイズ拡張をサポートします。LSM またはハードウェア RAID のボリュームのサイズ拡張は、AdvFS とは独立して行なわれるので、ボリュームのサイズを変更したときには、それをドメインに知らせる必要があります。これは、`mount` コマンドに `-o extend` オプションを指定して行ないます。拡張のためにこのオプションを使用するのは、1 つのファイルセットのマウントに対してだけでよいです。

ファイルセットをまだマウントしていない場合は、次のように入力します。

```
mount -o extend domain#fileset /mount_point
```

ファイルセットをすでにマウントしている場合は、次のように入力します。

```
mount -u -o extend domain#fileset /mount_point
```

`-o extend` オプションを使用すると、追加したすべてのストレージを、ドメインのすべてのファイルセットで使用できます。

注意

ボリュームを拡張することによりドメインのサイズを拡大するという方向のみ可能です。AdvFS はボリュームの縮小をサポートしません。

2.3.5 ドメイン・サイズの縮小

残りのボリュームに十分な空き容量がある場合は、ユーザの操作を中断せず、またドメイン内のファイルセットの論理構造に影響を与えずに、ドメインからボリュームを削除することができます。ファイル・システムは、選択したボリュームの内容をドメイン内の他のボリュームへ自動的に移行します。ドメインからボリュームを削除する前に、そのドメインのすべてのファイルセットをマウントする必要があります。マウントされていないファイルセットがある状態でドメインからボリュームを削除しようとすると、エラーが発生します。

切り離そうとするボリュームからすべてのファイルを移すだけの十分な空き容量がそのドメインの他のボリュームにない場合には、可能な限りのファイルが他のボリュームへ移されます。その後、十分なスペースがないことを示すメッセージが出力されます。この場合、ドメインはダメージを受けません。

AdvFS コマンドでできるのは、ドメインのサイズを縮小することだけです。mount コマンドの `-o extend` オプションでドメインの拡張を通知するように、ドメインにストレージの縮小を知らせる操作はありません。

注意

ドメインが LSM ボリューム上にある場合、サイズを縮小するのに LSM の `shrink` オプションは使用しないでください。

ボリュームを削除するには、SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E 参照) を使用するか、コマンド行で次のように `rmvol` コマンドを実行します。

```
rmvol device_name domain_name
```

たとえば、ドメイン `inventory` から `dsk3c` を削除する場合、次のコマンドを入力します。

```
# rmvol /dev/disk/dsk3c inventory
```

`rmvol` コマンドは、LSM のボリューム名の省略記法をサポートしています。以下の 2 つは同じ意味になります。

```
# rmvol dom1 testdg.voll  
# rmvol dom1 /dev/vol/testdg/voll
```

`Ctrl/C` あるいは `kill -term` コマンドを使用すると、ドメインにダメージを与えないで `rmvol` 処理 (`rmvol(8)` を参照) を中断させることができます。既に削除したファイルは新しい場所にあります。`kill -KILL` コマンドは使用しないでください。

状況によっては、`rmvol` コマンドを `kill` コマンドで中断するとボリュームにアクセス (書き込み) できない状態になる可能性があります。この状態に陥ったボリュームは、`showfdmn` コマンドの出力で確認できます。該当するボリュームには、“`data unavailable`” というメッセージが表示されます。`rmvol` コマンドの中断後、ボリュームへの書き込みができなくなった

場合は、`chvol` コマンドに `-A` オプションを使用してボリュームを再びアクティブにしてください。

ドメインのサイズを縮小する手順は以下のとおりです。

1. `showfdmn` コマンドで、ドメインの内容と、各ボリュームの現在のディスク容量を表示します。この操作はオプションです。
2. ボリュームを削除します。
3. `vfast` ユーティリティを実行しない場合、`balance` ユーティリティを使用してボリューム間でのファイル分散を均等化することをお勧めします。この操作はオプションです。

バランシングや断片化の解消を行っているドメイン、または `vfast` を実行してバランシングや断片化の解消を行っているドメインから、ボリュームを削除することはできません。

次の例では、`domain_2` ドメインのディスク `/dev/disk/dsk2c` を削除しています。

```
# showfdmn domain_2
  Id              Date Created      LogPgs Version Domain Name
2bb0c594.00008570 Fri Apr 26 10:23 2002 512          4 domain_2

Vol 512-Blks      Free % Used Cmode Rblks Wblks  Vol Name
  1L   832527   386984    54%   on   128   128  /dev/disk/dsk1c
    2   832527   647681    22%   on   128   128  /dev/disk/dsk2c
    3   832527   568894    32%   on   128   128  /dev/disk/dsk3c
-----
      249758 1603559    36%
# rmvol /dev/disk/dsk2c domain_2
```

ストライプ・セグメントを含む AdvFS ボリュームを削除する場合、`rmvol` ユーティリティは、そのセグメントを、同じファイルのストライプ・セグメントをまだ含んでいない他のボリュームへ移動します。ドメイン内のすべてのボリュームに渡ってファイルがストライプ化されている場合、ボリューム削除の際に確認が必要になります。削除処理を続行する場合は、残りのボリューム上に1つ以上のストライプ・セグメントが置かれます。ファイルのストライプ化についての詳細は、5.13 節を参照してください。

2.3.6 ドメインの削除

ドメインの削除は、ドメイン内のすべてのファイルセットをアンマウントした後に可能になります。ドメインを削除すると、ドメインを定義していた

/etc/fdmns ディレクトリのエントリが削除され、ファイルセットがマウントできなくなります。削除されたドメインに割り当てられていたボリュームは未使用として再表示され、再使用できるようになります。

rmfdmn コマンドを使用してドメインを削除して、各ファイルセットを無効化することはできますが、ボリューム上のデータは変更されないためセキュリティ・ホールになる可能性があります。/sbin/advfs/salvage コマンド (6.2.6 項を参照) を使用してデータにアクセスできることがあるためです。rmfset コマンドを使用して、個々のファイルセットを削除する方が賢明です (2.4.7 項を参照)。

ドメインを削除するには、GUI (付録 E を参照) を使用するか、またはコマンド行ですべてのファイルセットとファイルセット・クローンをアンマウントした後、次のように rmfdmn コマンドを入力します。

```
rmfdmn domain_name
```

次に示すのは、ドメイン promotions を削除する例です。

```
# rmfdmn promotions
rmfdmn: remove domain promotions? [y/n]y
rmfdmn: domain promotions removed
```

マウントされたファイルセットがある状態でこのコマンドを使用すると、システムはエラー・メッセージを表示します。AdvFS は、アクティブ状態のドメインは削除しません。

2.3.7 ドメイン名の変更

既存のドメインには、そのドメイン ID を変更することなく、新しい名前を割り当てることができます。ドメイン名を変更する場合、そのドメイン内のすべてのファイルセットに対する、/etc/fstab (2.4.1 項を参照) のエントリをアップデートする必要があります。ドメイン名を変更するには、AdvFS GUI (付録 E を参照) を使用するか、コマンド行で次の手順を実行します。

1. すべてのファイルセットおよび関連するクローンをアンマウントします。
2. /etc/fdmns ディレクトリで、古いドメイン名を新しいドメイン名に変更します。

```
# mv /etc/fdmns/old_domain_name /etc/fdmns/new_domain_name
```

3. /etc/fstab ファイルを編集して、新しいドメイン名を入力し、古いドメイン名を削除します。

4. 名前を変更したドメインのファイルセットをマウントします。

ドメイン名 `marketing` を `advertising` に変更する例を次に示します。ここでは、ファイルセット `fset` が `/fset` にマウントされていると想定しています。また、ファイルの編集には `vi` エディタを使用します。

```
# umount /fset
# mv /etc/fdmns/marketing /etc/fdmns/advertising
# vi /etc/fstab
```

`/etc/fstab` ファイルで、次の行を探します。

```
marketing#fset /fset advfs rw,userquota,groupquota 0 2
```

この行を次のように編集します。

```
advertising#fset /fset advfs rw,userquota,groupquota 0 2
```

ファイルセットをマウントします。

```
# mount /fset
```

2.4 ファイルセット

ファイルセットは、ファイル・システムのディレクトリ階層のマウント可能部分を表しています。ファイルセットと従来の UNIX ファイル・システムは、多くの点で共通しています。すなわち、AdvFS ファイルセットはマウントできます。また、ファイルセットにはファイルが含まれ、ファイルセットはクォータ値が設定可能な単位であり、またデータをバックアップするための単位にもなっています。

ドメイン内のファイルセットは、ドメイン内のボリューム上の利用可能なスペースを共用し、同じドメイン・トランザクション・ログ・ファイルを使用します。各ファイルセットには、固有のディレクトリ構造、ルート・タグ・ディレクトリ、クォータ・ファイル、およびフラグ・ファイルがあります。1つのドメイン内の最適な AdvFS ファイルセット数は、主に、そのファイルセットを使用するアプリケーションの要件に依存します。ドメインとファイルセットのさまざまな構成の長所短所については、1.4.1 項を参照してください。

ファイルセットはその物理ストレージとは独立して管理されるため、各ファイルセットを個別にバックアップできます (第 4 章)。また、それぞれにクォータ値を割り当てることができます (第 3 章)。複数の小さいファイルセットを使用すると、大きいファイルセット 1 つの場合よりも速くバック

アップやリストアができます。vdump や vrestore コマンドは、複数のファイルセットに対して同時に実行できます。

従来のファイル・システムとは異なり、AdvFS のディレクトリ階層はストレージから独立しています。このため、ファイルセットの論理構造に影響を及ぼさずにファイル・レイアウトを変更できます。

ファイルセット名はドメイン内で一意でなければなりません。ファイルセット名には、スラッシュ (/)、シャープ記号 (#)、コロン (:)、アスタリスク (*)、疑問符 (?), タブ, 改行, フォームフィード, キャリッジ・リターン, 垂直タブ, および空白を含めることはできません。なお、ドメインが異なるファイルセットには、同じ名前をつけても構いません。

domain#fileset のように、ファイルセット名はドメイン名と関連付けることができます。シャープ記号 (#) はネーミング構文の一部であり、コメントを示すものではありません。

各ファイルセットには、ドメイン ID とファイルセット・タグで構成された一意のファイルセット **ID** があります。各ドメインには、独自の連続したファイルセット・タグがあります。

AdvFS では、ファイルセット・クローンもサポートしています。AdvFS のファイルセット・クローンは、データを瞬時に取り込むために作成される、既存のファイルセットの読み取り専用のコピーです。ファイルセット・クローンの詳細は、2.4.10 項を参照してください。

2.4.1 /etc/fstab ファイルでのファイルセットの指定

AdvFS ファイルセットは、ファイル・システムの種類として advfs を指定して /etc/fstab ファイルに登録します (fstab(4) を参照)。登録方法は、他のファイル・システムを追加する場合と同様です。/etc/fstab ファイルに記述されている AdvFS ファイルセットは、システム・リブートのたびにマウントされます。

ファイルセット・エントリには、ドメイン名、ファイルセット名、マウント・ポイント、ファイル・システム・タイプ、およびマウント・ポイント・オプションが含まれます。ユーザおよびグループのクォータ(3.2 節参照)を設定する場合には、次に示すように、userquota オプションと groupquota オプション、およびパス・フィールド番号もエントリに含めます。コンマ

で区切ったオプションのリスト, つまり `rw` から `groupquota` の間にスペースを含めることはできません。

```
domain#fileset /mount_point advfs rw,userquota,groupquota 0 2
```

たとえば, `crdt` ファイルセットをリブート時にマウントするには, `/etc/fstab` ファイルに次の行を追加します (ファイルセットのマウント・ポイントは既に存在するものと仮定します)。

```
acct_124#crdt /crdt advfs rw,userquota,groupquota 0 2
```

たとえば, グループ・クォータだけを追加したい場合, ユーザ・クォータを指定しません。この場合, ファイルセット・エントリは次のようになります。

```
acct_124#crdt /crdt advfs rw,,groupquota 0 2
```

`userquota` オプションと `groupquota` オプションは, クォータ関連のコマンドで処理できるマウント済みファイル・システムを示します。クォータ関連コマンドの多くは, `-a` (all) オプション付きで実行することによって, `userquota` や `groupquota` が指定されたファイル・システムのみ処理対象を限定することが可能です。

`quota.user` ファイルと `quota.group` ファイルをそのファイルセットのサブディレクトリに再配置することができます。しかし, 他のファイルセットに再配置したり, 削除することはできません。ファイルを再配置した場合, `/etc/fstab` ファイルのエントリを更新し, `userquota` と `groupquota` パラメータが再配置したファイルのパスと名前を含むようにする必要があります。

たとえば, `quota.user` ファイルをファイルセット `crdt` の `d4` サブディレクトリに再配置し, `nq` に名前変更するには, `/etc/fstab` の対応するエントリを次のように変更します。

```
acct_124#crdt /crdt advfs rw,userquota=/d4/nq,groupquota 0 2
```

この例では, グループ・クォータ・ファイルは移動されません。

2.4.2 ファイルセット情報の表示

どのシステム・ユーザも, マウントされているファイルセットおよびクローンに関する詳しい情報を表示することができます。ドメインがアクティブな状態でない場合 (ファイルセットがマウントされていない場合) のみ, `root` ユーザ特権が必要です。ファイルセット情報を調べるには, 次のコマンドを入力します。

```
showfsets domain_name
```

次の例ではドメイン `zso_domain` の情報が表示されます。このドメインには、4 つのファイルセットがあります。

```
# showfsets zso_domain
staff1_fs
  Id      : 2cb9d009.000419f4.1.8001
  Files   : 18554, SLim= 0, HLim= 0
  Blocks(512) : 712230, SLim= 0, HLim= 0
  Quota Status : user=on group=on
  Object Safety: off
  Fragging : on
  DMAPI    : off
guest_fs
  Id      : 2cb9d009.000419f4.2.8001
  Files   : 4765, SLim= 0, HLim= 0
  Blocks(512) : 388698, SLim= 0, HLim= 0
  Quota Status : user=on group=on
  Object Safety: off
  Fragging : on
  DMAPI    : off
staff2_fs
  Id      : 2cb9d009.000419f4.3.8001
  Files   : 12987, SLim= 0, HLim= 0
  Blocks(512) : 842862, SLim= 0, HLim= 0
  Quota Status : user=on group=on
  Object Safety: off
  Fragging:  on
  DMAPI    : off
staff3_fs
  Id      : 2cb9d009.000419f4.4.8001
  Files   : 48202, SLim= 0, HLim= 0
  Blocks(512) : 1341436, SLim= 0, HLim= 0
  Quota Status : user=on group=on
  Object Safety: off
  Fragging : on
  DMAPI    : off
```

次の例ではドメイン `domain_2` が表示されます。このドメインには、1 つのファイルセットと 1 つのファイルセット・クローンがあります。

```
# showfsets domain_2
test_fs
  Id      : 3003f44f.0008ac95.4.8001
  Clone is : clone_test
  Files   : 7456, SLim= 0, HLim= 0
  Blocks (512) : 388698, SLim= 0, HLim= 0
  Quota Status : user=on group=on
  Object Safety: off
```

```
    Fragging      :   on
    DMAPi         :   off
Clone_test
    Id            :   3003f44f.0008ac95.5.8001
    Clone of      :   test_fs
    Revision      :   2
```

グラフィカル・ユーザ・インタフェースを使用しても同様な情報が表示されます (付録 E を参照)。

2.4.3 ファイルセットの作成

ドメインには、マウントされたファイルセットが少なくとも 1 つ含まれていなければなりません (2.3.3 項参照)。1 つのドメインに複数のファイルセットを作成し、それらのファイルセット間でドメインに設定されたストレージ・プールを共用することも可能です。どのファイルセットも、そのドメインで利用可能なストレージをすべて使用することができます。

各ファイルセットは、ドメイン内の他のファイルセットとは独立してマウントあるいはアンマウントすることができます。ファイルセット・クォータを設定すれば、ファイルセットで利用できるストレージ・サイズを制限することが可能です (第 3 章を参照)。割り当て済みスペースの 5 % 以上を浪費するファイルにはフラグ・ファイルを作成し、ディスク・スペースの浪費を抑制できます (5.2 節を参照)。ファイルセットには、作成時に `mkfset` コマンドで属性を設定できるほか、後で `chfsets` コマンドを使用して設定済みの属性を変更することもできます (2.4.9 項を参照)。

ドメインにファイルセットを作成するには、SysMan Menu の「新しい AdvFS ファイルセットの作成」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E を参照) を使用するか、あるいはコマンド行で次のように `mkfset` コマンドを実行します。

```
mkfset domain_name fileset_name
```

たとえば、ファイルセット `coupons` をドメイン `advertising` に作成するには次のように入力します。

```
# mkfset advertising coupons
```

ファイルセットをマウントするには、マウント・ポイントとなるディレクトリを作成する必要があります (2.4.5 項を参照)。

ドメインについては、2.3.3 項を参照してください。

2.4.4 ファイルセットのアップグレード

Version 5.0 以降の Tru64 UINX で作成されたドメイン (DVN4) のファイルセットは、大きなクォータ値をサポートし、非常に大きなディレクトリでの性能が良くなっています。以前に作成されたドメイン (DVN3) のファイルセットでは、このような改善は行われていません。ファイルセットを新しいバージョンにアップグレードするためには、ドメインをアップグレード (すなわち新たに再作成) して、ファイルセットをリストアする必要があります。リストア後のファイルセットには新しいクォータが設定され、性能が向上します (2.3.3 項を参照)。

Version 5.0 以降の Tru64 UINX が稼働しているサーバから NFS マウントする場合を除いて、Version 5.0 より前の Tru64 UINX では、DVN4 のファイルセットをマウントすることはできません。互換性の問題を回避するための情報は (6.4.6 項を参照してください)。

2.4.5 ファイルセットのマウント

従来の UNIX ファイル・システムの場合と同様に、AdvFS ファイルセットにアクセスするためにはマウントする必要があります。マウント・ポイントが存在しない場合には、マウント・ポイントとなるディレクトリを作成します。マウントするファイルセットは、その作成時のオペレーティング・システムに互換性がある必要があります (6.4.6 項を参照)。

ファイルセットをマウントするには、SysMan Menu の「一般的なファイルシステム・ユーティリティ - ファイルシステムのマウント」ユーティリティ (付録 A を参照) または AdvFS GUI (付録 E を参照) を使用するか、あるいはコマンド行で次のように `mount` コマンドを入力します。

```
mount domain_name#fileset_name mount_point
```

たとえば、マウント・ポイント・ディレクトリを作成し、ドメイン `advertising` のファイルセット `coupons` をマウントするには、次のように入力します。

```
# mkdir /coupons
# mount advertising#coupons /coupons
```

ファイルセットがリブート時に自動的にマウントされるようにするには、`/etc/fstab` ファイルに 2.4.1 項の説明に従って、エントリを作成します。

AdvFS は、ファイルセットをマウントする前に、ドメインのすべてのボリュームのすべてのデータにアクセスできることを確認します。問題が検出されるとマウントに失敗するか、ファイルセットが読み取り専用でマウントされることがあります (6.3.2 項を参照)。

間違ったボリューム数でファイルセットをマウントしようとした場合は、マウント操作は失敗します。詳細については、6.3.3.2 項 および `advscan(8)` を参照してください。

Version 5.0 以降の Tru64 UNIX で作成したファイルセットを Version 4.0 あるいはそれ以前のバージョンの Tru64 UNIX が稼働しているシステムにマウントしようとする、エラー・メッセージが出力されます (6.4.6 項を参照)。

マウント操作で一時的アトミック書き込みデータ・ロギングを指定することができます (5.5 節を参照)。

2.4.6 ファイルセットのアンマウント

ファイルセットをアンマウントすると、そのファイルセットはドメイン内に残りますが、アクセスできなくなります。そのファイルセットをマウントすると、再度利用できるようになります。

AdvFS ファイルセットをアンマウントするには、SysMan Menu の「一般的なファイルシステム・ユーティリティ — ファイルシステムのアンマウント」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E を参照) を使用する、あるいはコマンド行で次のように `umount` コマンドを入力します。

```
umount mount_point
```

たとえば、2.4.5 項でマウントしたドメイン `advertising` のファイルセット `coupons` をアンマウントする場合、次のコマンドを入力します。

```
# umount /coupons
```

2.4.7 ファイルセットの削除

ファイルセットを削除すると、そのファイルセットを再度マウントすることはできません。削除されたファイルセットは、ドメインから消えます。

ファイルセットを削除するには、あらかじめアンマウントし、そのファイルセットのクローンも削除しておく必要があります。ファイルセットにゴミ

箱ディレクトリ (2.6 節を参照) が設定されている場合は、このディレクトリも削除されます。

ファイルセットを削除するには、SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E を参照) を使用するか、あるいはコマンド行で次のように `rmfset` コマンドを入力します。

```
rmfset domain_name fileset_name
```

たとえば、ドメイン `domain_1` のファイルセット `tmp_1` を削除する場合は、次のように入力します。

```
# rmfset domain_1 tmp_1
rmfset: remove fileset tmp_1? [Y/N] y
```

ドメインのすべてのファイルセットを削除する最も確実な方法は、`rmfset` コマンドを使用することです。このユーティリティを使用すると、ファイルセット内の全ファイルの、メタデータを指すポインタが削除されます。したがって、このコマンドで削除したファイルセットは、`/sbin/advfs/salvage` ユーティリティを使用しても復元することはできません (6.2.6 項を参照)。ドメインのすべてのファイルセットを削除する効率的な方法は、`rmfdmn` コマンドによるドメイン自体の削除です。ただし、この方法では、一部のデータにアクセスされる可能性があるためセキュリティ・リスクが伴います。`rmfdmn` コマンドは、`/etc/fdmns` ディレクトリのドメインの定義を削除し、ボリュームのラベルを書き換えますが、ボリューム内のデータはそのまま残ります。

2.4.8 ファイルセット名の変更

アンマウントされたファイルセットの名前は、変更できます。ファイルセット名は利用者が割り当てる属性の 1 つであり、その値はドメイン内で保持されます。ファイルセット名を変更しても、この属性が変化するだけであり、ファイルセット ID は変更されません。

ファイルセット名を変更するには、まず対象のファイルセットをアンマウントし、クローンが存在する場合には、それもアンマウントします。続いて、SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E を参照) を使用するか、あるいはコマンド行で次のコマンドを入力します。

```
renamefset domain_name old_fileset_name new_fileset_name
```

ファイルセット名を変更したら、`/etc/fstab` ファイルの対応するエントリを変更する必要があります (2.4.1 項を参照)。この処理を行わないと、変更したファイルセットがシステム・ブート時にマウントされません。

たとえば、`/mount_point` にマウントされているドメイン `dmn_1` のファイルセット `sad` の名前を `happy` に変更する手順は次のとおりです。この手順では、ファイルの編集に `vi` エディタを使用しています。

```
# umount /mount_point
# renamefsset dmn_1 sad happy
# vi /etc/fstab
```

`/etc/fstab` ファイル内で次の行を見つけます。

```
dmn_1#sad /mount_point advfs rw,userquota,groupquota 0 2
```

この行を次のように変更します。

```
dmn_1#happy /mount_point advfs rw,userquota,groupquota 0 2
```

ファイルセットをマウントします。

```
# mount /mount_point
```

注意

AdvFS のファイルセット・クローンの名前は変更できません。
既存のクローンを削除し、任意の名前で新しいクローンを作成する必要があります。

2.4.9 ファイルセットの属性の変更

`chfsets` コマンドを使用すると、ファイルセットの属性を変更できます。
`-F`、`-f`、`-B`、および `-b` オプションでファイルセットのクォータを変更できるほか (3.3.2 項を参照)、`-o` オプションでフラグ・ファイルのオン/オフを切り替えることができます (5.2 節を参照)。また、オブジェクト・セーフティのオン/オフ (6.4.11 項) やデータ管理 API (DMAPI) の有効/無効化 (付録 D) を切り替えることができます。

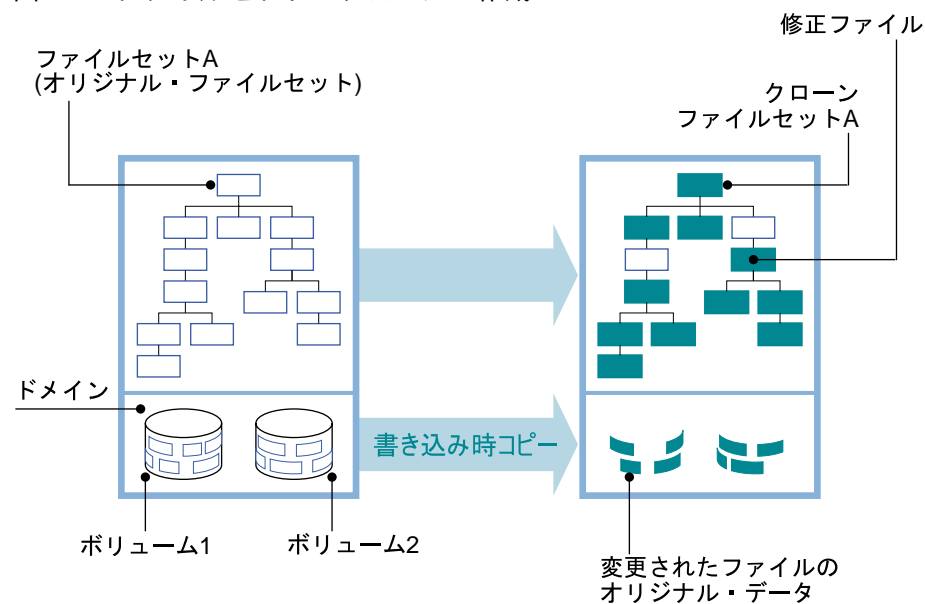
2.4.10 AdvFS ファイルセット・クローン

オプションの AdvFS Utilities を使用できる場合は、`root` ユーザでなくても AdvFS ファイルセット・クローンを使用してファイルをバックアップすることができます。ファイルセット・クローンは、ファイルセット構造情報 (メ

タデータ)の読み取り専用スナップショットです。ファイルセット・クローンを作成する場合、ユーティリティは、実際のデータではなくオリジナル・ファイルセットの構造だけをコピーします。ファイルの内容が変更されると、ファイル・システムは変更前のオリジナルのデータを AdvFS ファイルセット・クローンにコピーします。したがって、クローンを作成した時点でのシステムのコピーが、クローンが存在する限り保存されます。(この処理を書き込み時コピーと呼びます)。

ファイルセット・クローンに含まれるのは変更されたデータのコピーだけなので、ファイルセット・クローンはオリジナルのファイルセットよりも小さいのが普通です。図 2-3 に、オリジナル・ファイルセットとファイルセット・クローンの関係を図示します。

図 2-3: ファイルセット・クローンの作成



以下に示すように、ファイルセット・クローンを使用すると、データの可用性が向上します。

- 特定時点のシステムの保存

クローンは、(vdump コマンドなどによる) バックアップ処理の代わりにはなりません。しかし、ファイルが頻繁に変更されており、ある特定時点における情報を残しながらシステムのバックアップを取りたい場合、内部的な一貫性は保たれます。データのスナップショットを作成するた

めにクローンを使用し、そのクローンを `vdump` コマンドのバックアップ・ソースとして使用します (詳細は 第 4 章 を参照)。

- 不用意な削除や破損からのファイルの保護

アクセスや変更を予定している各ファイルセットは、クローンを作成しておくことをお勧めします。ファイルセット・クローンをオンライン状態にしておくで、誤って削除したファイルや破損したファイルを、バックアップ・テープを使わずにリストアすることができます。

ファイルセット・クローンのクローンは作成できません。クローンではデータ・キャッシングを使用できません。

注意

テキスト・ファイルをエディタで編集すると、オリジナルのファイル全体がクローンにコピーされる場合があります。これは、多くのエディタが、何が変更されたかには関係なくファイル全体を書き直すためです。この場合、AdvFS ファイルセット・クローンのサイズは非常に大きくなる可能性があります。この種の処理を AdvFS 側で変えることはできません。

クローン作成時に存在したファイルを削除した場合、(オリジナル・ファイルセットでは見えませんが) クローンにはそのまま残ります。そのファイルはクローンにはコピーされず、実際の削除はクローンが削除されるまで行われません。クローン内のファイルは、クローン作成時に存在したバージョンのファイルです。その後の更新は失われます。

AdvFS ファイルセット・クローンのサイズは、クローンが存在する間に発生するアップデートの量に依存します。ファイルセット・クローンはファイルがアップデートされると絶えず変更されるので、空きディスク・スペースに関する統計値を表示する `df` コマンドは、ファイルセット・クローンのサイズを正確には反映しません。

注意

ドメインがディスク・スペースを使い果たすと、ファイル・システムは AdvFS ファイルセット・クローン内のファイルの整合性を維持できなくなります。オリジナルのファイルセットは利用可能な状態ですが、ファイルセット・クローンは整合性を失い、使用

されなくなります。この場合、警告メッセージが、ユーザの端末とシステム・コンソールの両方に表示されます。

DMAPI を有効にしたファイルセットのクローンは作成できません。

システム・ディスクのクローンを作成し、他のシステムでブートする方法については、Best Practices ドキュメントを参照してください。この方法は、AdvFS のファイルセット・クローンを作成する方法とは異なります。

2.4.10.1 AdvFS ファイルセット・クローンの作成

AdvFS ファイルセットのクローン作成はユーザ側では意識されず、システム性能への影響もわずかに過ぎません。クローンの作成には root 権限が必要です。ファイルセット・クローンを作成するには、SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A 参照) または AdvFS GUI (付録 E を参照) を使用するか、あるいはコマンド行で次のように `clonefs` コマンドを入力します。

```
clonefs domain_name fileset_name clone_name
```

たとえば、`transactions` ドメインの `day300` ファイルセットに対するクローン `clone_day300` を作成する場合は、次のコマンドを使用します。

```
# clonefs transactions day300 clone_day300
```

コマンド行で、バックアップのソースとしてクローンを使用する方法は、4.3.11 項を参照してください。AdvFS GUI でのクローン作成の方法は E.4.3 項を参照してください。

2.4.10.2 AdvFS ファイルセット・クローンのマウントとアンマウント

クローンは、通常のファイルセットと同じ方法でマウント/アンマウントされます (2.4.5 項および 2.4.6 項を参照)。

2.4.10.3 AdvFS ファイルセット・クローンの削除

クローンは、通常のファイルセットと同じ方法で削除します (2.4.7 項を参照)。

2.4.10.4 AdvFS ファイルセット・クローンの名前の変更

ファイルセット・クローンの名前は変更できません。クローンに新しい名前を割り当てるには、古いクローンを削除して、そのファイルセットの新しいクローンを作成する必要があります。この新しいクローンは、削除したファイルセット・クローンより後の時点のスナップショットであることに注意してください。

2.5 AdvFS ルート・ファイル・システムの構成

AdvFS をルート・ファイル・システムに構成すると、次のような利点があります。

- クラッシュ後すぐに再起動できます。 `fsck` ユーティリティを実行する必要はありません。
- 1 組のツールを使用してすべてのローカル・ファイル・システムを管理できます。 `addvol` コマンドと `rmvol` コマンドを除く AdvFS のすべての機能が、ルート・ファイル・システムを管理するために利用できます。ただし、マルチボリュームのルート・ドメインが使用できるクラスタ構成では上記コマンドを使用できます。
- ルート・ファイル・システムの保護のためにミラー化した AdvFS を使用できます。これにより、メディア障害が発生してもルート・ファイル・システムが利用可能です。

AdvFS ルート・ファイル・システムには、次のような制限事項が存在します。

- クラスタ環境で実行している場合 (『クラスタ管理ガイド』を参照) を除き、ルート・ドメインには、1 つのボリュームしか含めることができません。また、ルート・ドメインにボリュームを追加することはできません。
- ボリュームは、物理デバイスの先頭 (a または c パーティション) から開始しなければなりません。
- ルート・ドメインで最初に作成するファイルセットは、ルート・ファイルセットでなければなりません。
- ルート・ドメインおよびファイルセットにはどんな名前でも割り当てることができますが、`/etc/fstab` ファイルに割り当てた名前を記述しなければなりません。

ルート・ドメインには、ルート・ファイルセットのみを含めてください。
/usr ファイルセットや /var ファイルセットを含めることはお勧めできません。システムで利用できるストレージ・デバイスが 1 台だけの場合には、ルート・ドメインとそれ以外のドメインを、異なるパーティションに配置してください。

基本オペレーティング・システムの初期インストール時にルート・ディレクトリを AdvFS ボリューム上に配置することも、あるいはインストール後に既存のルート・ファイル・システムを AdvFS に変換することもできます。初期インストール時にルート・ディレクトリを AdvFS ファイル・システムでインストールする場合には、ルート・ディレクトリは特に指定しなければ a パーティション上に作成されます。

既存の UFS ルート・ファイル・システムを AdvFS へ変換する方法については、C.2 節を参照してください。オペレーティング・システムの初期インストール時にルート・ファイル・システムとして AdvFS を使用する方については、『インストール・ガイド』を参照してください。

2.5.1 シングルユーザ・モードでのルート・ファイル・システムのマウント

システムがシングルユーザ・モードにブートされる際に、ルート・ファイル・システムは自動的に読み取り専用でマウントされます。mount コマンドに -u オプションを使用すると、ルート・ファイルセットのマウント状態を read-only から read-write に変更することができます。

```
# mount -u /
```

ルート構成に変更を加える必要がある場合には、この手順を使用します。たとえば、/etc/fstab ファイルの変更が必要な場合に使用します。詳細については mount(8)を参照してください。

2.5.2 ルート・ドメイン名の変更

ルート・ドメインの名前は、他のドメインと同じ方法で変更できます (2.3.7 項を参照)。ルート・ドメインの名前は、/etc/fdmns ディレクトリ内のディレクトリ名として、また /etc/fstab ファイルのルート・エントリとして格納されています。ルート・ドメインの名前を変更した場合、両方をアップデートする必要があります。

2.5.3 ルート・ファイルセット名の変更

ルート・ファイルセット名の変更は、その他のファイルセットの名前を変更する手順と同様に行います (2.4.8 項を参照)。ただし、複雑化する要因が 2 つあります。

- `renamefs` コマンドを実行するにはファイルセットがアンマウントされている必要がありますが、ユーザがルート・ファイルセットをアンマウントすることはできません。
- ルート・ファイルセット名を変更するには、`/etc/fstab` ファイルの編集が必要です。このためには、ルート・ファイルセットを書き込み可能にする必要があります。ただし、`/etc/fstab` ファイル内のエントリが正しくないと、ルート・ファイルセットを書き込み可能でマウントすることはできません。

このため、代わりのブート可能パーティションを使用して、変更しようとしているルート・ファイルセットを操作し、変更を行った後で、変更したファイルセットをルート・ファイルセットとしてリブートします。

ルート・ファイルセット名の変更手順は、以下のとおりです。

1. 変更するもの以外のパーティションをブートします (UFS でも構いません)。
2. 名前を変更しようとしているファイルセット用に、ブートしたパーティションの新しいエントリを `/etc/fdmns` ディレクトリに作成します。
3. 新しいディレクトリへ移動して、元のファイルセットを保持するデバイスへのシンボリック・リンクを作成します。
4. `renamefs` コマンドでルート・ファイルセット名を変更します。
5. `/etc/fstab` ファイルをアップデートするために、名前を変更したルート・ファイルセットを適当なディレクトリに一時的にマウントします。
6. `fstab` エントリを変更して、新しいルート・ファイルセット名を反映させます。
7. システムをシャットダウンします。
8. 元の AdvFS システムでリブートします。

次の例では、ルート・ファイルセットの名前を `root_fs` から `new_root` に変更しています。ルート・ファイルセットは、`/dev/disk/dsk2a` の `root_domain` ドメインにあるものとします。

1. 変更するもの以外のデバイスをブートします。
2. `tmp_root_domain` のエントリを `/etc/fdmns` ディレクトリに作成します。

```
# mkdir /etc/fdmns/tmp_root_domain
```
3. 新しいディレクトリに移動して、`tmp_root_domain` に対するシンボリック・リンクを作成します。

```
# cd /etc/fdmns/tmp_root_domain
# ln -s /dev/disk/dsk2a
```
4. ファイルセット名を `root_fs` から `new_root` に変更します。

```
# renamefset tmp_root_domain root_fs new_root
```
5. 変更したルート・ファイルセットをマウントして `/etc/fstab` ファイルをアップデートします。

```
# mount tmp_root_domain#new_root /mnt
```
6. `/mnt/etc/fstab` 内の `tmp_root_domain` のエントリを `vi` エディタを使って編集します。

```
# cd /mnt/etc
# vi fstab
```

`/etc/fstab` ファイルで、次の行を探します。

```
root_domain#root_fs / advfs rw,userquota,groupquota 0 2
```

この行を次のように変更します。

```
root_domain#new_root / advfs rw,userquota,groupquota 0 2
```
7. システムをシャットダウンします。

```
# shutdown -h now
```
8. AdvFS システムをリブートします。

注意

ルート・ドメインおよびファイルセットの名前を変更した後
`/etc/fstab` ファイルのエントリの変更を忘れると、ブート時に

マルチユーザ・モードに入れなくなります。この場合、シングルユーザ・モードで `/etc/fstab` ファイルを編集してください。

2.6 ファイル回復のためのゴミ箱の設定

オプションの AdvFS Utilities を使用できる環境では、各エンドユーザは必要に応じて、削除したファイルのコピーを保存するようにシステムを設定することが可能です。すなわち、ファイルセット内の 1 つ以上のディレクトリに、ゴミ箱ディレクトリを付加することができます。こうしてゴミ箱を付加したディレクトリから削除したすべてのファイルは、対応するゴミ箱ディレクトリに自動的に移動されます。その結果、ゴミ箱付きのディレクトリから削除したファイルの最も新しいバージョンを、`mv` コマンドで元のディレクトリに戻すことができます。

ただし、ゴミ箱ディレクトリには欠点もあります。バックアップ媒体を使用せずにファイルを復元できる反面、ファイル削除時のディスクへの書き込み処理が増えます。

ゴミ箱ディレクトリの設定には `root` 権限は不要ですが、次の点に留意する必要があります。

- 復元できるファイルは、最後に削除したバージョンだけです。
- 複数のディレクトリに同じゴミ箱ディレクトリを設定できますが、これらのディレクトリから同じ名前のファイルを削除すると、ゴミ箱には最後に削除したファイルだけが残ります。
- ゴミ箱に移動されるのは、ファイル単位で削除したファイルだけです。`rmfset` コマンドを使用してファイルセット全体を削除した場合には、そのファイルセットに含まれるファイルはゴミ箱に移動されず、ゴミ箱も削除されます。
- クォータのチェックでは、ゴミ箱内の削除済みファイルもカウントされます。
- ゴミ箱ディレクトリから削除したファイルは復元できません。

表 2-1 は、ゴミ箱の設定と管理に使用するコマンドと、その機能を示しています。

表 2-1: ゴミ箱関連コマンド

コマンド	説明
mktrashcan	ゴミ箱を作成する。
shtrashcan	ゴミ箱の内容を表示する。
rmtrashcan	ゴミ箱ディレクトリを削除する。

たとえば、booklist ディレクトリにゴミ箱ディレクトリ keeper を設定するには、次のコマンドを実行します。

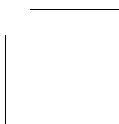
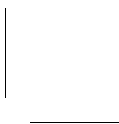
```
# mkdir keeper
# mktrashcan keeper /booklist
  'keeper' attached to '/booklist'
```

ファイルを削除し、ゴミ箱の内容を確認する手順は次のとおりです。

```
# rm old_titles
# shtrashcan /booklist
  '//keeper' attached to '/booklist'
# cd keeper
# ls
  old_titles
```

ゴミ箱とディレクトリの関係設定を解除するには、次のコマンドを実行します。

```
# rmtrashcan /booklist
  '/booklist' detached
```



クォータの管理

AdvFS ファイル・システムでクォータを使用すれば、ユーザ、グループ、あるいはファイルセットの単位で、使用する物理ストレージの量を追跡および管理することができます。クォータの設定および編集には、root ユーザ特権が必要です。root ユーザは、クォータによる使用ストレージ量の制限を受けません。クォータが適用されるのは、root 以外のユーザだけです。

この章では、次のトピックについて説明します。

- クォータと猶予期間 (3.1 節)
- ユーザ・クォータとグループ・クォータ (3.2 節)
- ファイルセット・クォータ (3.3 節)
- 使用状況を確認する方法 (3.4 節)

3.1 クォータの概要

AdvFS のクォータ体系は、UFS のクォータと互換性があります。ただし、AdvFS のクォータ体系は、次の 2 つの点で UFS のクォータと異なります。

- AdvFS は、クォータの管理と適用を区別します。クォータ情報は常時管理されますが、クォータの適用については有効/無効を切り替えることができます。
- AdvFS はファイルセット・クォータをサポートします。つまり、ドメインのファイルセットのサイズを制御するためにクォータ制限を設定することができます。

AdvFS ファイル・システムではクォータ・ファイルが作成され、それによってクォータ、猶予期間、およびファイルセットの利用状況が追跡されます。クォータ・ファイルはファイルセット内に保持されますが、UFS の場合とは異なり、ユーザ側でクォータ・ファイルを削除したり、作成することはできません。クォータを設定していない場合も、ファイルセットにはクォータ・ファイルが存在します。

クォータは、ディスクの容量とファイル数に対して設定できます。さらに、次に示す 2 種類のクォータが設定可能です。

- ユーザおよびグループ・クォータ

AdvFS のユーザおよびグループ・クォータは、UFS クォータに良く似ています。各ファイルセットの各ユーザまたはグループごとに、別々のクォータを設定することができます。クォータ・ファイルは、特に指定しなければファイルセットのルート・ディレクトリに保存されます。

- ファイルセット・クォータ

ファイルセット・クォータは AdvFS 独自の機能であり、ファイルセットそのものに使用できる容量を制限できます。ファイルセット・クォータは、ドメインが複数のファイルセットを含む場合に役立ちます。ファイルセット・クォータが設定されていないと、いずれかのファイルセットでドメイン内のすべてのディスク・スペースを使用してしまふような事態が発生します。

たとえば、`/usr/users` などのホーム・ディレクトリを含むファイルセットは急速に増大することがあるので、このようなファイルセットにはクォータを設定しておく有効です。これに対して、`/tmp` ファイルセットはサイズが上下に変動するため、このファイルセットにクォータ値を設定するのはお勧めできません。

クォータ・ファイルとファイルセット・クォータは、`root` ユーザがローカル・システムで完全なバックアップを指定したときに保存されます。バックアップの手順については、第 4 章を参照してください。

3.1.1 クォータ限界値

限界値は、ディスク使用量 (ブロック数) およびファイル数 (i ノード数) の両方に設定することができます。表 3-1 に、両方のタイプのクォータ値の限界サイズを示します。

表 3-1: クォータの限界サイズ

	ユーザおよびグループ・クォータ	ファイルセット・クォータ
ディスク使用量	80 億 T バイト*	40 億 T バイト
ファイル数	40 億	40 億

* Version 5.0 より前のオペレーティング・システムでは、ユーザおよびグループのクォータの限界値は 2T バイトでした。Version 5.0 以降のオペレーティング・システム上で、Version 5.0 より前のバージョンで作成したドメインを使用している場合、新しい限界値を利用するためにはドメインをアップグレードする必要があります (2.3.3.4 項を参照)。

クォータの限界値には、ハード限界値とソフト限界値の 2 種類があります。ハード限界値は、超過することができません。したがって、それ以上のスペースを割り当てたりファイルを作成したりできません。一方ソフト限界値 (3.1.2 項 参照) は、一定期間、ハード限界値を超えない範囲で値を超えることができます。

root ユーザは、いつでも必要に応じてハード限界値とソフト限界値を設定/変更できます。これらの限界値の変更は、クォータが有効なファイルセットにただちに反映されます。ファイルセットのマウントやアンマウントは、クォータの限界値には影響を与えません。ハードおよびソフト限界値は、ユーザ、グループ、およびファイルセットに対して設定することができます。省略時の設定は、クォータ制限なしです。また、次のように設定することもできます。

- ハード限界値を 0 ブロックあるいは 0 ファイルに設定する。クォータ制限は設定されません。
- ハード限界値を 1 ブロックあるいは 1 ファイルに設定する。ディスク・スペースの割り当てはできなくなります。
- ソフト限界値を 1 ブロックあるいは 1 ファイルに設定し、ハード限界値を 0 ブロックあるいは 0 ファイルに設定する。一時的なディスク・スペースの割り当てが可能になります。

設定したクォータ限界値は、root ユーザによって変更されるまで有効です。

クォータ限界値を超えて書き込みをする必要がある場合は、6.4.5 項を参照してください。

3.1.2 猶予期間

ソフト限界値には猶予期間が設定されます。猶予期間とは、ソフト限界値を超えることが可能な期間のことです。猶予期間までの時間計時は、使用しているディスク容量がクォータのソフト限界値を下回ると、いったんリセットされます。

猶予期間が終了すると、当該ユーザやグループは、ソフト限界値を下回る水準までファイルを削除しない限り、新しいファイルの作成や、ディスク・スペースの割り当てができなくなります。このような場合に既存のファイルを更新すると、データが失われる可能性があります。

ソフト限界値を超えると、猶予期間のタイマが起動されます。ソフト限界値を下回ると、そのたびにタイマが停止し、リセットされます。ソフト限界値を超えてから猶予期間を変更した場合は、使用量が限界値を下回るまでは古い猶予期間が有効です。

猶予期間は次のものに対して設定できます。

- ユーザ

ファイルセットごとに、すべてのユーザ用の猶予期間を 1 つだけ設定することができます (3.2.2 項を参照)。ブロック数の猶予期間とファイル数の猶予期間は、同じである必要はありません。

- グループ

ファイルセットごとに、すべてのグループ用の猶予期間を 1 つだけ設定することができます (3.2.2 項を参照)。ユーザ猶予期間の方が長い場合でも、グループ内のすべてのユーザは、そのグループに設定された猶予期間によって制限されます。

- ファイルセット

ファイルセット猶予期間はグループ猶予期間と同じです (3.3.2 項を参照)。どのユーザあるいはグループもファイルセットのソフト限界値を超え、ファイルセット猶予期間のタイマが起動する原因となり得ます。また、どのユーザあるいはグループも (限界値を超える原因となったユーザあるいはグループでなくても) ファイルセットがソフト限界値を下回るように、そのファイルを削除することができます。

AdvFS は、省略時では猶予期間を 7 日間に設定します。猶予期間は、日、時間、分、秒単位で指定できます。この期間は変更することができます。猶

予期間を 0 日に設定すると、省略時の設定が強制されます。猶予期間を 1 秒に設定すると、猶予期間は無効になります。

3.1.3 クォータの機能の要約

表 3-2 では、ユーザおよびグループ・クォータと、ファイルセット・クォータを比較しています。

表 3-2: ユーザおよびグループ・クォータとファイルセット・クォータの比較

	ユーザおよびグループ	ファイルセット
クォータ・ファイル	quota.user および quota.group という名のファイルに定義する (3.2.1 項)。	ファイルセットのメタデータの一部。見ることはできない。
クォータ限界値の設定	/etc/fstab ファイル内のマウント・オプションを変更し、ファイルセットをマウントし、edquota コマンドを実行する (3.2.2 項)。	chfsets コマンドを実行 (3.3.2 項)。
猶予期間の設定	edquota コマンドに -ut オプションまたは -gt オプションをつけて実行する (3.2.2 項)。	グループの猶予期間と同じ (3.3.2 項)。
クォータの有効化	quotaon コマンドを実行するか、リブート時にクォータを適用するために rcmgr コマンドを実行する (3.2.4 項および 3.2.5 項)。	クォータ限界値を設定すると有効になる (3.3.4 項)。
クォータの無効化	quotaoff コマンドを使用するか、/etc/rc.config.common ファイルの QUOTA_CONFIG="yes" 行を削除する (3.2.6 項)。	chfsets コマンドを使用してクォータ限界値を 0 に設定する (3.3.5 項)。
クォータ限界値のチェック	edquota コマンドに -u または -g オプションをつけて実行する。	chfsets コマンドまたは showfsets コマンドを実行する。

3.2 ユーザおよびグループのクォータ

以下の項では、クォータ・ファイルについて説明し、ユーザおよびグループに対してクォータおよび猶予期間を設定する方法について説明します。

3.2.1 ユーザおよびグループのクォータ・ファイル

AdvFS ファイル・システムは、ユーザおよびグループのクォータ情報をファイルセット内のルート・ディレクトリの `quota.user` ファイルと `quota.group` ファイルに保存します。これらのファイルは、ファイルセットの作成時に作成され、ユーザ ID とグループ ID でインデックスされます。クォータを適用する必要がない場合も、これらのファイルは削除できません。各クォータ・ファイル・エントリには、次の情報が含まれています。

- ハード・ブロック限界値
- ソフト・ブロック限界値
- ブロック使用量
- ハード・ファイル限界値
- ソフト・ファイル限界値
- ファイル使用量
- ブロック猶予期間
- ファイル猶予期間

クォータ・ファイルはスパース・ファイルです (1.3.3 項を参照)。つまり、ユーザ ID やグループ ID が存在しない部分には、ディスク・スペースは割り当てられていません。 `ls -l` コマンドを使用すれば、`quota.user` ファイルあるいは `quota.group` ファイルが関係するファイルの容量を調べることができますが、それらは実際のディスク使用量を反映していません。たとえば、`quota.user` ファイルを表示するには、次のコマンドを実行します。

```
# ls -l quota.user
-rw-r----- 1 root      operator  294912 Jul 20 08:50 quota.user
```

`ls -s` コマンドまたは `du -k` コマンドを使用すれば、ファイルによるディスク使用量を 1K バイト単位のブロック数で表示できます。たとえば、`quota.user` ファイルのブロック数を表示するには、次のコマンドを実行します。

```
# ls -s quota.user
16  quota.user

# du -k quota.user
16      quota.user
```

3.2.2 ユーザおよびグループのクォータ，および猶予期間の設定

ユーザ・クォータおよびグループ・クォータの両方を設定する必要は必ずしもありません。グループ・クォータを指定すると，そのクォータ値は当該グループに属する全ユーザに適用されます。そのため，グループ・クォータよりも大きなユーザ・クォータを指定しても，指定したユーザ・クォータ値に達する前にグループ・クォータによって制限されるため，ユーザ・クォータは無視されます。

ユーザとグループのクォータを設定するには，`edquota` コマンドを使用します。また，その猶予期間を変更するのもこのコマンドを使用します。猶予期間は，各ファイルセットごとに，すべてのユーザまたはすべてのグループに対して 1 つしか設定できません。ただし，ブロック数とファイル数に対して同じ値の猶予期間を設定する必要はありません。設定したグループ猶予期間は，ファイルセット猶予期間としても設定されます（詳細は 3.3.2.2 項を参照）。

次のような手順に従って，ユーザおよびグループのクォータと猶予期間を設定します。

1. `/etc/fstab` ファイル (2.4.1 項) にクォータ・ファイルのマウント・ポイント・オプション (`userquota` および `groupquota`) を追加します。ファイルセットをマウントしていない場合はマウントします。
2. ユーザおよびグループ・クォータ，または猶予期間を設定するために，`edquota` コマンドに適切なオプションを指定して実行します。クォータ・ファイルの内容が，`EDITOR` 環境変数に指定されているエディタによって ASCII 形式で表示されます。`EDITOR` 環境変数が設定されていない場合は，`vi` エディタが使用されます。クォータを設定することのできるファイルセットだけを見ることができます。ファイルセットがわからない場合，ステップ 1 からやり直してください。
3. ユーザまたはグループのクォータ情報が表示されたら，必要に応じてハードおよびソフト限界値フィールドの数値を変更します。その後，エディタを終了して変更内容を保存します。
4. ユーザまたはグループの猶予期間を設定するには，ユーザの場合は `-ut` オプションを指定し，グループの場合は `-gt` オプションを指定して，`edquota` コマンドを実行します。編集することのできる ASCII 形式のファイルが表示されます。

5. 猶予期間情報が表示されたら、必要に応じて猶予期間を変更します。その後、エディタを終了して変更内容を保存します。

クォータの設定と猶予期間の詳細については、`edquota(8)` を参照してください。

ファイルセットに設定したクォータを適用するには、`quotaon` コマンドで有効化するか、システムを再起動する必要があります (3.2.4 項 および 3.2.5 項を参照)。ファイルセットに対してすでにクォータ機能が有効になっている場合は、クォータ値を変更すると新しい限界値がすぐに有効になります。

1 人のユーザに対してクォータを設定した後に、`edquota` コマンドに `-p` オプションをつけて使用して、そのユーザのエントリをプロトタイプとして他のユーザに適用することができます (3.2.3 項を参照)。`-p` オプションを使用せずに `edquota` コマンドを複数のユーザ名を指定して実行した場合、指定したすべてのユーザについて、エディタ・ウィンドウが表示されます。

3.2.2.1 ユーザ・クォータの設定

下記の手順では、ユーザ `user5` のクォータを設定しています。

1. クォータを設定しようとしているファイルセットのマウント・ポイントが存在しない場合、`/etc/fstab` ファイル (2.4.1 項) にクォータ・マウント・ポイント・オプションを追加します。そして、ファイルセットをマウントしていない場合はマウントします。

```
domain_1#geb1 /geb1 advfs rw,userquota,groupquota 0 2
domain_2#geb3 /geb3 advfs rw,userquota,groupquota 0 2
domain_4#geb4 /geb4 advfs rw,userquota,groupquota 0 2
```

2. クォータ設定ファイルを表示するために、`edquota` コマンドに `-u` オプションをつけてユーザ名を指定して実行します。`edquota` の出力に、そのファイルセットがない場合、ステップ 1 から繰り返します。

```
# edquota -u user5
Quotas for user user5:
/geb1: blocks in use: 0, limits (soft = 0, hard = 0)
      inodes in use: 0, limits (soft = 0, hard = 0)
/geb3: blocks in use: 0, limits (soft = 0, hard = 0)
      inodes in use: 0, limits (soft = 0, hard = 0)
/geb4: blocks in use: 0, limits (soft = 0, hard = 0)
      inodes in use: 0, limits (soft = 0, hard = 0)
```

`blocks in use` と `inodes in use` は、それぞれ各ファイルセットで使用されている 1K バイトのブロック数とファイル数を示しています。

これらの値は変更できません。この例では、ファイルセットが新しく作成されています。ソフトおよびハード限界値がゼロになっているので、限界値は設定されていません。

3. `soft` および `hard` フィールドに設定された、指定ユーザの限界値を変更します。複数のユーザ名を指定した場合、値はすべてのユーザ名に反映されます。

```
/geb1: blocks in use: 0, limits (soft = 0, hard = 0)
      inodes in use: 0, limits (soft = 0, hard = 0)
/geb3: blocks in use: 0, limits (soft=5000, hard=10000)
      inodes in use: 0 limits (soft= 100, hard= 200)
/geb4: blocks in use: 0, limits (soft = 0, hard = 0)
      inodes in use: 0, limits (soft = 0, hard = 0)
```

4. エディタを終了して変更を保存します。

ファイルセット `geb3` のクォータが既に有効になっている場合は、新しい限界値が直ちに有効になります。ファイルセットのクォータがまだ有効になっていない場合は、クォータが有効にされると同時に新しい限界値が有効になります。クォータを手動で有効にする方法は、3.2.5 項を参照してください。

ユーザとグループの両方に対してクォータが設定されている場合、小さい方の値が適用されます。

3.2.2.2 ユーザ猶予期間の設定

ファイルセットに対してソフト限界値を適用する場合には、そのファイルセットに猶予期間を設定することができます。猶予期間を指定しない場合、AdvFS の省略時の設定である 7 日が猶予期間として使用されます。猶予期間を無効にするには、値を 1 秒に設定します。ブロック数とファイル数に対しては、異なる猶予期間を設定することができます。グループ猶予期間がユーザ猶予期間よりも短い場合、ユーザはグループ猶予期間によって制限されます。

次の手順では、ファイルセット `geb3` のすべてのユーザに対する猶予期間を設定しています。

1. クォータを設定しようとしているファイルセットのマウント・ポイントが存在しない場合、`/etc/fstab` ファイル (2.4.1 項) にクォータ・マウント・ポイント・オプションを追加します。そして、ファイルセットをマウントしていない場合はマウントします。

```
domain_1#geb1 /geb1 advfs rw,userquota,groupquota 0 2
domain_2#geb3 /geb3 advfs rw,userquota,groupquota 0 2
domain_4#geb4 /geb4 advfs rw,userquota,groupquota 0 2
```

2. 猶予期間の設定ファイルを表示するために、edquota コマンドに -ut オプションをつけて実行します。edquota の出力に、そのファイルセットがない場合、ステップ 1 から繰り返します。

```
# edquota -ut
Time units may be: days, hours, minutes, or seconds
Grace period before enforcing soft limits for users:
/geb1: block grace period: 7 days, file grace period: 7 days
/geb3: block grace period: 7 days, file grace period: 7 days
/geb4: block grace period: 7 days, file grace period: 7 days
```

3. geb3 ファイルセットのユーザ猶予期間を設定するために、block grace period および file grace period フィールドの値を変更します。

```
Time units may be: days, hours, minutes, or seconds
Grace period before enforcing soft limits for users:
/geb1: block grace period: 7 days, file grace period: 7 days
/geb3: block grace period: 2 days, file grace period: 3 days
/geb4: block grace period: 7 days, file grace period: 7 days
```

4. エディタを終了して変更を保存します。

ファイルセット geb3 のクォータが既に有効になっている場合は、新しい猶予期間が直ちに有効になります。すでにソフト限界値を超えている場合は、ソフト限界値を下回るまで新しい猶予期間は有効になりません。ファイルセットのクォータがまだ有効になっていない場合は、クォータが有効にされるとすぐに新しい限界値が有効になります。クォータを手動で有効にする方法は、3.2.5 項を参照してください。

3.2.2.3 グループ・クォータの設定

次の手順では、グループ rsgusers に対して geb3 にクォータを設定しています。

1. クォータを設定しようとしているファイルセットのマウント・ポイントが存在しない場合、/etc/fstab ファイル (2.4.1 項) にクォータ・マウント・ポイント・オプションを追加します。そして、ファイルセットをマウントしていない場合はマウントします。

```
domain_1#geb1 /geb1 advfs rw,userquota,groupquota 0 2
domain_2#geb3 /geb3 advfs rw,userquota,groupquota 0 2
domain_4#geb4 /geb4 advfs rw,userquota,groupquota 0 2
```


2. `edquota` コマンドに `-g` オプションをつけて実行します。複数のグループ名を指定した場合、値はすべてのグループに反映されます。
`edquota` の出力に、そのファイルセットがない場合、ステップ 1 から繰り返します。

```
# edquota -g rsgusers
Quotas for group rsgusers:
/geb1: blocks in use: 0, limits (soft=0, hard=0)
       inodes in use: 0, limits (soft=0, hard=0)
/geb3: blocks in use: 0, limits (soft=0, hard=0)
       inodes in use: 0, limits (soft=0, hard=0)
/geb4: blocks in use: 0, limits (soft=0, hard=0)
       inodes in use: 0, limits (soft=0, hard=0)
```

`blocks in use` と `inodes in use` は、それぞれ各ファイルセットで使用されている 1K バイトのブロック数とファイル数を示しています。これらの値は変更できません。ソフトおよびハード限界値がゼロになっているので、限界値は設定されていません。

3. `soft` および `hard` フィールドに、指定したグループの限界値を設定します。

```
/geb1: blocks in use: 0, limits (soft=0, hard=0)
       inodes in use: 0, limits (soft=0, hard=0)
/geb3: blocks in use:0, limits(soft=60000, hard=80000)
       inodes in use:0, limits(soft= 6000, hard= 8000)
/geb4: blocks in use: 0, limits (soft=0, hard=0)
       inodes in use: 0, limits (soft=0, hard=0)
```

4. エディタを終了して変更を保存します。

ファイルセット `geb3` のクォータが既に有効になっている場合は、新しい限界値が直ちに有効になります。ファイルセット `geb3` のクォータがまだ有効になっていない場合は、クォータが有効にされると同時に新しい限界値が有効になります。クォータを手動で有効にする方法は、3.2.5 項を参照してください。ユーザとグループの両方のクォータが設定されている場合は、両方のうちに小さい方の値が使用されます。

3.2.2.4 グループ猶予期間の設定

ソフト限界値を適用する場合、すべてのグループに対して、ファイルセットごとに 1 つの猶予期間を設定することができます。グループ猶予期間はファイルセット猶予期間と同じです (3.3.2.2 項参照)。猶予期間を指定しない場合は、AdvFS の省略時の設定である 7 日間が猶予期間として使用されます。猶予期間を無効にするには、値を 1 秒に設定します。ブロック数とファイル

数に対しては、異なる猶予期間を設定することができます。グループ猶予期間は、すべてのどのユーザ猶予期間よりも優先されます。

次の手順では、ファイルセット geb3 のすべてのグループに対して猶予期間を設定しています。

1. クォータを設定しようとしているファイルセットのマウント・ポイントが存在しない場合、`/etc/fstab` ファイル (2.4.1 項) にクォータ・マウント・ポイント・オプションを追加します。そして、ファイルセットをマウントしていない場合はマウントします。

```
domain_1#geb1 /geb1 advfs rw,userquota,groupquota 0 2
domain_2#geb3 /geb3 advfs rw,userquota,groupquota 0 2
domain_4#geb4 /geb4 advfs rw,userquota,groupquota 0 2
```

2. 猶予期間の設定ファイルを表示するために、`edquota` コマンドに `-gt` オプションをつけて実行します。`edquota` の出力に、そのファイルセットがない場合、ステップ 1 から繰り返します。

```
# edquota -gt
Time units may be: days, hours, minutes, or seconds
Grace period before enforcing soft limits for groups:
/geb1: block grace period: 7 days, file grace period:7 days
/geb3: block grace period: 7 days, file grace period:7 days
/geb4: block grace period: 7 days, file grace period:7 days
```

3. geb3 ファイルセットのグループ猶予期間を設定するために、`block grace period` および `file grace period` フィールドの値を変更します。

```
Time units may be: days, hours, minutes, or seconds
Grace period before enforcing soft limits for groups:
/geb1: block grace period: 7 days, file grace period:7 days
/geb3: block grace period:12hours, file grace period:5 days
/geb4: block grace period: 7 days, file grace period:7 days
```

4. エディタを終了して変更を保存します。

すでにクォータが有効になっている場合、グループが geb3 のソフト限界値を超えていない限り、新しい猶予期間が直ちに有効になります。すでにソフト限界値を超えている場合は、そのグループの使用量がソフト限界値を下回るまで、新しい猶予期間は有効になりません。クォータがまだ適用されていない場合は、グループ猶予期間はクォータが適用されるとすぐに有効になります。クォータを手動で有効にする方法については、3.2.5 項を参照してください。

3.2.2.5 一時的猶予期間の設定

ファイルセットのファイルの寿命を制限することができます。一時的猶予期間を設定すると、指定した期間中、制限なしにファイルを割り当てることができます。その期間を過ぎると、猶予期間に作成されたすべてのファイルが削除されるまで、それ以上ファイルを割り当てることができません。この一時的猶予期間の設定は、ファイルが短い期間しか必要のない場合に役立ちます。

一時的猶予期間を設定するには、`edquota` コマンドに `-t` を指定して実行します。ブロック数または `i` ノードのソフト限界値は 1 に設定し、ブロック数または `i` ノードのハード限界値は 0 に設定します。

3.2.3 複数のユーザおよびグループに対するユーザおよびグループ・クォータの設定

AdvFS ファイル・システムでは、単一コマンドで一連のユーザあるいはグループのクォータを変更することができ、個々のクォータ値を個別に変更する必要はありません。ファイルセットごとに 1 つの猶予期間がすべてのユーザやすべてのグループに適用されるため、各ユーザおよびグループ・クォータに対して複数の猶予期間を適用する必要はありません。

すべてのユーザに同じクォータ値を設定するには、次の手順を行ないます。

1. ユーザまたはグループにクォータを設定するために、`edquota` コマンドに `-u` オプションまたは `-g` オプションをつけて実行します。
2. 最初のユーザのクォータ設定を残りのユーザに適用するために、`edquota` コマンドに `-p` オプションをつけて実行します。

詳細は、`edquota(8)` を参照してください。

3.2.3.1 複数のユーザに対するクォータの設定

次の手順では、後で他のユーザのクォータの変更に利用するためのプロトタイプ・ユーザ・クォータを設定しています。

1. ユーザ `user5` に対してクォータを設定します (3.2.2.1 項を参照)。

```
# edquota -u user5
Quotas for user user5:
/geb1:blocks in use:0, limits(soft=    0,hard=    0)
      inodes in use:0, limits(soft=    0,hard=    0)
/geb3:blocks in use:1 ,limits(soft= 5000,hard= 10000)
      inodes in use:4, limits(soft=   100,hard=   200)
```

```
/geb4:blocks in use:2, limits(soft=    0,hard=    0)
      inodes in use:1, limits(soft=    0,hard=    0)
```

2. ユーザ `user5` のクォータをプロトタイプとして使用して、新しいユーザ `user7` , `user8` , および `user9` のクォータを設定します。

```
# edquota -p user5 -u user7 user8 user9
```

3. クォータが設定されたかどうかチェックするために、`user7` に対して `edquota` コマンドを実行します。

```
# edquota -u user7
Quotas for user user7:
/geb1:blocks in use:0, limits(soft=    0,hard=    0)
      inodes in use:0, limits(soft=    0,hard=    0)
/geb3:blocks in use:1 ,limits(soft= 5000,hard= 10000)
      inodes in use:4, limits(soft=   100,hard=   200)
/geb4:blocks in use:2, limits(soft=    0,hard=    0)
      inodes in use:1, limits(soft=    0,hard=    0)
```

3.2.3.2 複数のグループに対するクォータの設定

次の手順では、後で他のグループのクォータの変更に利用するためのプロトタイプ・グループ・クォータを設定しています。

1. グループ `rsgusers` のクォータを設定します (3.2.2.3 項を参照)。

```
# edquota -g rsgusers
Quotas for group rsgusers:
/geb1:blocks in use:0, limits(soft=    0,hard=    0)
      inodes in use:0, limits(soft=    0,hard=    0)
/geb3:blocks in use:0, limits(soft= 60000,hard= 80000)
      inodes in use:0, limits(soft=   6000,hard=   8000)
/geb4:blocks in use:0, limits(soft=    0,hard=    0)
      inodes in use:0, limits(soft=    0,hard=    0)
```

2. グループ `rsgusers` のクォータをプロトタイプとして使用して、新しいグループ `rsgstudents` のクォータを設定します。

```
# edquota -p rsgusers -g rsgstudents
```

3. クォータが設定されたかどうかチェックするために、`rsgstudents` に対して `edquota` コマンドを実行します。

```
# edquota -g rsgstudents
Quotas for group rsgstudents:
/geb1:blocks in use:0, limits(soft=    0,hard=    0)
      inodes in use:0, limits(soft=    0,hard=    0)
/geb3:blocks in use:0, limits(soft= 60000,hard= 80000)
      inodes in use:0, limits(soft=   6000,hard=   8000)
/geb4:blocks in use:0, limits(soft=    0,hard=    0)
```

```
inodes in use:0, limits(soft=      0,hard=      0)
```

3.2.4 システム・スタートアップ時のユーザおよびグループ・クォータの適用

システムのスタートアップ時に、ユーザ・クォータおよびグループ・クォータを自動的に適用するには、次の手順を実行します。

1. `/etc/fstab` ファイルのエントリを編集して、クォータ・ファイルのマウント・ポイント・オプションを追加します。マウント・ポイント・オプションについての詳細は、2.4.1 項を参照してください。
2. `rcmgr` コマンドを使用して、`QUOTA_CONFIG` オプションを `/etc/rc.config.common` ファイルに追加します。

```
/usr/sbin/rcmgr -c set QUOTA_CONFIG "yes"
```

このエントリによって、`/sbin/init.d` クォータ・スクリプトが、`/etc/fstab` ファイル内で `userquota` オプションまたは `groupquota` オプションが指定されているファイル・システムに対して、`quotaon` コマンドを実行します。これにより、次回以降のリブートで、マウントしたファイルセットに対して、クォータが適用されるようになります。

注意

クォータが適用されている状態でファイルセットをアンマウントした場合、ファイルセットを再マウントする時点で `quotaon` コマンドによりクォータの適用を明示的に有効にしなければなりません。これは、`/etc/rc.config.common` ファイルに `QUOTA_CONFIG="yes"` エントリが記述されている場合でも必要です。

`QUOTA_CONFIG` オプションを “yes” に設定すると、`quotacheck` コマンドも実行されます。このコマンドは、`/etc/fstab` ファイル内でクォータ・オプションが設定されている UFS ファイル・システムに対して実行され、ファイル・システムのクォータの整合性をチェックします。AdvFS ファイル・システムでは、AdvFS メタデータ・トランザクション・ログによってストレージ割り当てとクォータの同期が維持されるため、特に指定しない限り、`quotacheck` コマンドは AdvFS には実行されません。

/etc/fstab ファイル内でクォータ・オプションが指定されている AdvFS ファイル・システムに対して、スタートアップ時に quotacheck コマンドを実行するためには、rcmgr コマンドを使用して次のオプションを /etc/rc.config.common ファイルに追加します。

```
/usr/sbin/rcmgr -c set QUOTACHECK_CONFIG -a
```

quotacheck コマンドの処理対象を省略時の状態 (UFS のみ) に戻すには、このオプションを次のように変更します。

```
/usr/sbin/rcmgr -c set QUOTACHECK_CONFIG " "
```

3.2.5 ユーザおよびグループ・クォータの手動による適用

マウントしたファイルセットの /etc/fstab ファイル・エントリに userquota と groupquota を追加し、限界値と猶予期間を設定するために edquota コマンドを実行したら、クォータを有効化するために quotaon コマンドを実行します。

システムのスタートアップ時にクォータを初期化するようにシステムが設定されている場合には (3.2.4 項を参照)、ファイルセットをアンマウントしない限り、quotaon コマンドを再度実行する必要はありません。スタートアップ時にクォータが適用されるように設定していない場合 (3.2.6 項を参照) には、リブートのたびに quotaon コマンドを実行する必要があります。

次の例では、前の各項でクォータ値が設定されたファイルセットに関して、クォータを適用しています。

```
# quotaon -av
/geb1:  group quotas turned on
/geb1:  user quotas turned on
/geb3:  group quotas turned on
/geb3:  user quotas turned on
/geb4:  group quotas turned on
/geb4:  user quotas turned on
```

省略時の設定では、quotaon コマンドは、ユーザおよびグループ・クォータの両方に作用します。対象にユーザ (-u オプション) あるいはグループ (-g オプション) を指定して、どちらか一方に対してクォータを適用することも可能です。ファイルセットを指定してユーザまたはグループ・クォータを適用することもできます。詳細は、quotaon(8) を参照してください。

3.2.6 ユーザおよびグループ・クォータの適用解除

クォータは、一時的にまたは永続的に適用を解除することができます。ファイルおよびディスク・スペースの使用量に関する情報は、クォータを現在適用しているかどうかには関係なく取得できます。

`quotaoff` コマンドは、クォータの設定を無効にします。無効にしたクォータは、`quotaon` コマンドが手動かシステム初期化時に実行され、クォータが再び有効になるまで無効の状態になります。クォータの適用は、対象にユーザ (`-u` オプション) かグループ (`-g` オプション) かを指定して解除できます。ユーザまたはグループ・クォータを適用しているファイルセットを指定することもできます。詳細は、`quotaoff(8)` を参照してください。

`umount` コマンドを実行するとクォータの適用が解除され、ファイルセットがアンマウントされます。アンマウントしたファイルセットのクォータは、有効にできません。ファイルセットを再マウントした場合に、そのファイルセットのユーザおよびグループ・クォータを有効にするには `quotaon` コマンドを実行する必要があります。

ユーザまたはグループのクォータの適用を永続的に停止したい場合には、次の手順を行ないます。

1. `quotaoff` コマンドを実行し、ファイルセットに対してクォータ機能が適用されないようにします。
2. `edquota` コマンドを使用してクォータ限界値を 0 に設定します。
3. `/etc/fstab` ファイルからそのファイルセットの `userquota` および `groupquota` エントリを削除します。

3.3 ファイルセット・クォータ

ここでは、ファイルセット・クォータ・ファイルについてと、コマンド行で、ファイルセットに対してクォータと猶予期間を設定する方法について説明します。AdvFS GUI を使用して、ファイルセット・クォータを設定する方法は、付録 E を参照してください。

3.3.1 ファイルセット・クォータ・ファイル

AdvFS は、ファイルセットに関する構造情報に、ファイルセット・ソフト限界値およびハード限界値を保存します。このファイルに直接アクセスす

ることはできません。このファイルには、ユーザおよびグループ・クォータ・ファイルに含まれているのと同じ情報、つまり、ブロック数のハードおよびソフト限界値、ファイル数のハードおよびソフト限界値が含まれています。あるファイルセットについてみると、ファイルセット猶予期間はグループ猶予期間と同じです。

3.3.2 ファイルセット・クォータおよび猶予期間の設定

ファイルセット・クォータは、ファイルセットが使用できるファイル数とディスク・スペース量を制限します。ファイルセットにはソフト限界値とハード限界値の両方を設定できます。ファイルセット・クォータが設定されていない場合、どのファイルセットも、ドメイン内のすべてのディスク・スペースにアクセスできます。ファイルセット・クォータは `chfsets` コマンドで設定されます。ファイルセット・クォータが設定されている場合、それらのクォータはファイルセットをマウントするたびに有効になります。

ファイルセット・クォータには猶予期間があります (3.3.2.2 項を参照)。猶予期間を設定していない場合、AdvFS の省略時の猶予期間である 7 日間が使用されます。猶予期間を適用解除するには、1 秒に設定します。

ファイルセット・クォータ値の定義には `chfsets` コマンドを使用します。このコマンドには、次のオプションを指定できます。

- F ファイル数のソフト限界値を設定
- f ファイル数のハード限界値を設定
- B ブロック使用量のソフト限界値を設定
- b ブロック使用量のハード限界値を設定

このコマンドは古い限界値と新しい限界値を両方表示します。

3.3.2.1 ファイルセット・クォータの設定

次の手順では、`year1` ドメインの `student_files` ファイルセットのファイルセット・クォータを設定しています。その他のクォータ・コマンドとは異なり、`showfsets` コマンドは、512 バイト・ブロック単位でブロック

使用量を表示します。K バイト値を表示したい場合には、`-k` オプションを使用します。

1. 既存のファイルセット・クォータを表示するために、`showfsets` コマンドを使用します。

```
# showfsets -k year1 student_files
student_files
  Id           : 2feff762.00034e3f.1.8001
  Clone is     : stufiles_clone
  Files        :      7,   SLim=    0,   HLim=    0
  Blocks (1k)  :    118,   SLim=    0,   HLim=    0
  Quota Status : user=on  group=on
  Object Safety: off
  Fraggings    : on
  DMAPI        : off
```

SLim は、ファイル数とブロック使用量に対するソフト限界値、HLim はハード限界値です。

2. `chfsets` コマンドを使用してクォータを設定します。

```
# chfsets -F 1000 -f 2000 -B 25000 -b 50000
year1 student_files
```

3. 新しいファイルセット・クォータをチェックするには、`showfsets` コマンドをもう一度実行します。

```
# showfsets -k year1 student_files
student_files
  Id           : 2feff762.00034e3f.1.8001
  Clone is     : stufiles_clone
  Files        :      7,   SLim=  1000,   HLim=   2000
  Blocks (1k)  :    118,   SLim= 25000,   HLim=  50000
  Quota Status : user=on  group=on
  Object Safety: off
  Fraggings    : on
  DMAPI        : off
```

3.3.2.2 猶予期間の設定

ファイルセット猶予期間はグループ猶予期間と同じもので、独立して変更することはできません。このため、グループがソフト限界値を超えることのできる猶予期間を、`edquota` コマンドに `-gt` オプションを使用して変更する場合は、ファイルセット猶予期間も変更することになります。グループ猶予期間とファイルセット猶予期間の設定の詳細は、3.2.2.4 項を参照してください。

3.3.3 複数のファイルセットに対するクォータの設定

`chfsets` コマンドの実行時に複数のファイルセット名を指定することにより、ドメイン内の複数のファイルセットに対してクォータ限界値を設定できます。

たとえば、`domain_2` の `data` および `data2` ファイルセットのハード限界値を変更するには、これらの両方のファイルセット名を指定して `chfsets` コマンドを実行します。

```
# chfsets -b 1000 -f 200 domain_2 data data2
```

3.3.4 ファイルセット・クォータの適用

`chfsets` コマンドを実行すると、ファイルセット・クォータは、すぐに適用されます。他の操作は必要ありません。これによって、ファイルセット・クォータは、そのファイルセットをマウントすれば有効になります。

3.3.5 ファイルセット・クォータの適用解除

`chfsets` コマンドを、ハード限界値とソフト限界値をそれぞれ 0 に設定して実行することによって、ファイルセットに対するクォータの適用を、一時的または永続的に解除することができます。ファイルおよびディスク・スペースの使用量に関する情報は、クォータを現在適用しているかどうかには関係なく取得できます。

3.4 ファイルセットおよびディスク・スペース使用量の確認

ファイルおよびディスク・スペース使用量を、`df`、`showfdmn`、`/sbin/advfs/vdf`、および `showfsets` コマンドを使用してモニタすることができます。ディスク使用量を表示する一般的な方法については、または表示されたディスク使用量情報が読めない場合は、6.4.1 項を参照してください。ファイルのアドレス範囲を表示する `ls` などのコマンドでは、スパーズ・ファイルであるクォータ・ファイルは、実際より大きいサイズに表示されます。

3.4.1 ユーザおよびグループ・クォータの確認

ユーザおよびグループ・クォータを適用している場合、定期的にクォータの設定を確認することができます。root ユーザでない場合にできるのは、自分のファイルの情報を表示することだけです。root ユーザは、すべての

ファイルセットのすべてのユーザとグループのクォータ情報を表示することができます。すべてのディスク・クォータ値は、1K バイト・ブロック単位で表示されます。

表 3-3 に示すコマンドは、ユーザおよびグループ・クォータが適用されているファイルセットのディスク・スペースおよびファイルの使用状況を調べるために利用できます。

表 3-3: ユーザおよびグループの使用状況情報コマンド

コマンド	説明
ncheck	ファイルセット内の各ファイルのタグおよび絶対パス名を表示します。
quot	ファイルセットの所有者の要約を表示します。
quota	ディスク使用状況と限界値をユーザまたはグループ別に表示します。
quotacheck	ファイルセット内でのクォータの一貫性をチェックします。
repquota	ファイルセットのユーザおよびグループ・クォータの要約を表示します。

3.4.1.1 各ファイルのタグとフルパス名の表示

ncheck コマンドは、ファイルをタグ (iノードと同じ) 番号別にリストします。このコマンドの出力をパイプ処理して sort コマンドに渡すことによって、ソートされた出力を quot コマンドの入力として使用し、すべてのファイルとその所有者をリストすることができます。このようなリストを表示するには、次のように入力します。

```
ncheck domain#fileset | sort +0n | quot -n domain#fileset
```

3.4.1.2 ファイルセット所有者の要約

quot コマンドは、各ユーザが所有する、マウントされたファイルセットのブロック使用量とファイル数を表示します。ファイルセットを指定しない場合、このコマンドは /etc/fstab ファイルの ro, rw, および rq マウント・オプションを含むすべてのファイルセットを対象にします。

quot コマンドをオプションを指定せずに入力すると、各ユーザのブロック数のみが表示されます。

```
# quot domain_1#set_1
domain_1#set_1:
34128      root
816        user5
```

quot コマンドを -f オプションを指定して入力すると、各ユーザについて、ファイルのサイズとファイル数の両方が表示されます。

```
# quot -f domain_1#set_1
domain_1#set_1:
34128      125      root
816        9        user5
```

3.4.1.3 ディスク使用量と限界値の表示

quota コマンドは、/etc/fstabファイルに userquota および groupquota エントリのあるマウントされたファイルセットの、ブロック使用量、ファイル数、およびユーザ・クォータまたはグループ・クォータを表示します。クォータ値を超えて使用しているすべてのファイルセットに関して、あるいはクォータが適用されているかどうかには関係なくマウントされているすべてのファイルセットに関して、ユーザ・クォータ情報を表示するか、グループ・クォータ情報を表示するかを選択できます。

quota コマンドは、各ユーザの、ファイルセットのブロック使用量、ソフト限界値 (quota)、ハード限界値 (limit)、猶予期間、および使用ファイル数を表示します。欄内のアスタリスク (*) は、ソフト限界値を超えていることを表しています。ソフト限界値を超えていなければ、猶予期間は表示されません。現在のクォータ値を表示するには、edquota コマンドを使用します。

次の例はユーザ runner1 のクォータ情報を表示しています。

```
# quota -u runner1
Disk quotas for user runner1 (uid 446):
  Filesystem  blocks  quota   limit grace files quota limit  grace
    /           60    100     150      3     10    20
  /usr       5071*  5000   10000 24:40     2     20    40
  /mile1       816  20000   30000      9    350   500
  /mile2     22032  50000  200000      2   2000  4000
  /mile3       2344  10000   15000     370   1000  2000
  /mile4     18023* 10000   20000 7days     3    100   150
  /mile5     32012* 20000   50000 7days     0   2000  3000
```

次の例はグループ kingco のクォータ情報を表示しています。

```
# quota -g kingco
Disk quotas for group kingco (gid 15):
  Filesystem  blocks  quota   limit grace files quota limit  grace
```

/	118	200	300		2	20	40
/usr	13184*	10000	20000	7days	2	40	80
/mile1	36136	100000	200000		124	10000	20000
/mile2	44064	200000	400000		4	2000	4000
/mile3	3587	30000	60000		628	3000	5000
/mile4	51071	150000	300000		6	1050	1800
/mile5	61044	100000	200000		3	10000	20000

3.4.1.4 ユーザおよびグループ・クォータの整合性の確認

quotacheck コマンドは、実際のブロック使用量とファイル数が、設定された限界値と矛盾がないかをチェックします。このコマンドは、ユーザおよびグループ・ファイルを調べ、現在のディスク使用状況の表を作成し、この表とディスク・クォータ・ファイルに保管されている情報とを比較します。

quotacheck コマンドを使用するためには、ファイルセットがマウントされている必要があります。-v オプション (詳細モード) を指定すると、見つかった矛盾点とチェック・プロセスで実行された処理手順を表示します。

省略時の設定では、ユーザとグループの両方のクォータがチェックされますが、-u (ユーザ) または -g (グループ) のいずれかのオプションを選択してどちらか一方の情報のみをチェックすることもできます。

次の例では、詳細モードでファイルセット set_1 のチェックを行い、矛盾がないことが示されています。

```
# quotacheck -v domain_1#set_1
*** Checking user and group quotas for domain_1#set_1 (/set_1)
```

次の例は、/etc/fstab ファイルにクォータが定義されているすべてのファイルセットについてチェックを行なっています。この出力結果で inodes はファイルの数、blocks はブロック使用量を表します。quotacheck コマンドの結果、/usr ディレクトリの矛盾点が訂正されています。

```
# quotacheck -va
*** Checking user and group quotas for /dev/rdisk/dsk0g (/usr)
*** Checking user and group quotas for domain_1#set_1 (/set_1)
/usr:root    fixed group quota: inodes   3057 ->   3022
                                   blocks 100616 -> 123440
/usr:system  fixed group quota: inodes    2483 ->   2488
                                   blocks  91721 -> 114568
/usr:adm     fixed group quota: inodes     280 ->    240
                                   blocks   487 ->   464
```

3.4.1.5 ファイルセットごとのユーザおよびグループ・クォータの要約

repquota コマンドは、指定したファイルセットの実際のディスク使用量とクォータを表示します。出力される要約には、マウント済みで、`/etc/fstab` ファイルにクォータ・エントリがあるファイルセットのみが含まれます。特に指定しなければ、ユーザ・クォータが報告されますが、`-u` オプションでユーザ・クォータの報告、`-g` オプションでグループ・クォータの報告を、それぞれ明示的に指定することもできます。`-a` オプションを指定するとユーザとグループの両方のクォータが表示されます。

repquota コマンドは、ユーザまたはグループのそれぞれについて、現在のファイル数、使用中のスペースの量、edquota コマンドで設定したクォータ限界値が表示されます。

次の例では、`/set_1` にマウントした 1 つのファイルセットのクォータの要約を出力しています。

```
# repquota -v /set_1
*** Report for user quotas on /set_1 (domain_1#set_1)
              Block limits              File limits
User      used  soft  hard  grace  used  soft  hard  grace
root  --  34088      0      0      123      0      0
user5  --    816 20000 30000      9    350    500
```

次の例では、`/etc/fstab` ファイルにクォータを定義しているすべてのマウントされたファイルセットについて、ユーザおよびグループ・クォータの情報を表示しています。この例では、UFS と AdvFS の両方のファイルが表示されています。

```
# repquota -va
*** Report for group quotas on /usr (/dev/disk/dsk0g)
              Block limits              File limits
Group      used  soft  hard  grace  used  soft  hard  grace
system  -- 114568      0      0      2488      0      0
daemon  --   144      0      0        1      0      0
uucp    --   801      0      0        8      0      0
mem      --  1096      0      0       10      0      0
bin      -- 108989      0      0     3219      0      0
mail     --   209      0      0        2      0      0
terminal --    56      0      0        2      0      0
adm      --   464      0      0     240      0      0
operator --   392      0      0        3      0      0
211      --  6937      0      0       33      0      0
*** Report for user quotas on /usr (/dev/disk/dsk0g)
              Block limits              File limits
User      used  soft  hard  grace  used  soft  hard  grace
```

```

root      -- 123440      0      0      3022      0      0
bi        -- 102534      0      0      2940      0      0
uucp      -- 729        0      0      7         0      0
adm       -- 1          0      0      1         0      0
user5     -- 15         18     24      1         0      0
kraetsch  -- 6937       0      0      35        0      0
*** Report for group quotas on /set_1 (domain_1#set_1)
          Block limits          File limits
Group      used  soft  hard  grace  used  soft  hard  grace
system    -- 22816      0      0      50      0      0
daemon    -- 12088      0      0      82      0      0
*** Report for user quotas on /set_1 (domain_1#set_1)
          Block limits          File limits
User       used  soft  hard  grace  used  soft  hard  grace
root      -- 34088      0      0      123      0      0
user5     -- 816 20000 30000      9    350    500
*** Report for group quotas on /set_3 (domain_2#set_3)
          Block limits          File limits
Group      used  soft  hard  grace  used  soft  hard  grace
system    -- 1593      0      0      6         0      0
*** Report for user quotas on /set_3 (domain_2#set_3)
          Block limits          File limits
User       used  soft  hard  grace  used  soft  hard  grace
root      -- 1593      0      0      6         0      0

```

3.4.2 ファイルセット・クォータの確認

ファイルセット・クォータ情報は、ファイルセットをチェックする一般的なコマンドで表示することができます。表 3-4 に示すのは、この目的のために使用できる便利なコマンドです。

表 3-4: ユーザおよびグループの使用量コマンド

コマンド	説明
df	ファイルセットごとにディスク使用量を表示します。
showfsets	-q オプションを指定すると、ファイルセット・クォータ限界値を表示します。
showfdmn	ドメイン情報を表示します。
/sbin/advfs/vdf	他のファイルセット・ディスク使用量コマンドの出力フォーマットを加工できます。

3.4.2.1 ファイルセット・スペース情報の表示

df コマンドを使用して、ファイルセット内の使用済みディスク・スペースおよび空きディスク・スペースを表示することができます。このコマンド

は、ハードあるいはソフトのいずれか低い方の限界値を使用して利用可能ディスク容量を計算します。

- ファイルセット・クォータが設定されている場合、このコマンドは、クォータ制限に到達するまでに利用可能なスペースの量を表示します。
- ソフトおよびハード限界値が設定されている場合、このコマンドは、低い方の限界値を使用して利用可能なディスク・スペースを計算します。
- ファイルセット・クォータで許容されるスペースより少ないスペースしかドメイン内にない場合、このコマンドは、そのドメインで利用可能な実際のスペースを表示します。
- ファイルセット・クォータが設定されていない場合、このコマンドは、ドメイン・サイズを表示します。未使用スペースはどのファイルセットにも利用できます。

df コマンドの出力結果には、メタデータやクォータ・ファイルが使用するスペースは含まれません。

AdvFS は、各ファイルセットの容量をそれぞれ個別に計算します。ドメインに複数のファイルセットがある場合は、ファイルセット・クォータによって制限されない限り、すべての未使用スペースが各ファイルセットで利用できます。そのため、ドメイン内の各ファイルセットの総容量が 100% 超えた値で表示される場合もあります。次の例では、fileset_1 で利用可能なスペースの量を表示しています。

```
# df /fileset_1
Filesystem      512-blocks  Used Avail  Capacity  Mounted on
domain_1#fileset_1      1500  1750    0    117%  /fileset_1
```

この例では、使用量が限界値を超えているため、容量は実際の使用スペース (1750/1500) で計算され、結果は 100% を超えています。使用量が限界値未満の場合は、容量は (Used)/(Used + Avail) で計算されます。

次の例では、ファイルセット jan および mar のそれぞれが、credit ボリュームで利用可能なすべてのディスク・スペースを使用できることを示しています。

```
# df
Filesystem      512-blocks  Used    Avail Capacity  Mounted on
credit#jan      2000000    390820   98864      80%  /jan
credit#mar      2000000    271580   98864      73%  /mar
```


3.4.2.2 ファイルセットの限界値の表示

showfsets コマンドに `-q` オプションを指定すると、指定したドメイン内のファイルセットに関する、ファイルの使用状況、ハード限界値、ソフト限界値、および猶予期間が表示されます。表示される数値には、使用中のブロック数、ブロック数の限界値、使用中のファイル数、およびファイル数の限界値が含まれます。正しい情報を得るには、ファイルセットをマウントする必要があります。

次の例は、ドメイン `domain_1` のファイルセット情報を表示しています。

```
# showfsets -q domain_1
          Block (512) Limits      File Limits
Fileset  BF  used soft hard grace  used soft hard grace
fileset1 +- 1750 1500 2000 11:32   35  300  400
```

この例で `BF` フィールドに表示されているプラス記号 (+) は、使用中のブロック数がソフト限界値を超えていることを示しています。ハード限界値に達すると、アスタリスク (*) が表示されます。この場合は、ファイル数の限界にはまだ達していません。

3.4.2.3 ドメインの領域情報の表示

showfdmn コマンドで、ファイルセットとクォータに関する意思決定の材料として利用できるドメインの統計情報が得られます。このコマンドはドメインの属性と、(ボリュームの総容量、空きブロックの総数、および割り当て済みボリューム領域の割合 (パーセント) などの) ドメイン内の各ボリュームの情報を表示します。

```
# showfdmn usr_domain
      Id          Date Created LogPgs Version Domain Name
39cbf2d6.0002691e Sun Jan 20 17:01 2002   512         4 usr_domain

Vol  512-Blks    Free % Used Cmode Rblks Wblks Vol Name
1L  10605520 7757728   27%   on   256   256 /dev/disk/dsk8d
```

3.4.2.4 ドメインとファイルセットのディスク使用状況の表示

`/sbin/advfs/vdf` ユーティリティは、showfdmn、showfsets、および `df` コマンドの出力を再フォーマットして、AdvFS ドメインおよびファイルセットのディスク使用量に関する情報を表示します。このコマンドは、ドメインのディスク使用量とファイルセットのディスク使用量との間の関係を明らかにします。

このユーティリティには、次の制限事項があります。

- すべてのファイルセットをマウントしておく必要があります。
- AdvFS ファイルセット・クローンが使用するディスク・スペースは計算されません。
- NFS マウントされているファイルセットに関しては有効な結果を生成しません。

vdf コマンドは、アクティブでないシステム上で実行するこのをお勧めします。このコマンドは、一連のユーティリティを実行するため、システムが実際にファイルの追加、削除、修正などを行なっていると vdf コマンドの出力に矛盾が生じる可能性があるからです。

vdf コマンドでは、ドメインおよびファイルセット名のどちらでも指定できます。ファイルセット名のみを指定した場合、df コマンドを実行した場合と同じ出力結果となります。ドメインを指定した場合、このユーティリティはメタデータのディスク・ブロック数也表示します。-l オプションでドメインあるいはファイルセットを指定した場合、ドメインおよびファイルセットの両方の情報が表示されます。表示されるドメイン・メタデータは、ドメイン内のすべてのファイルセットで共用される総メタデータです。

次の例では、ドメイン `usr_domain` に関する要約情報を表示しています。

```
# /sbin/advfs/vdf usr_domain
Domain      512-blocks  Metadata    Used    Available  Capacity
usr_domain   65536         11219      47549    6768       89%
```

次の例では、2つのファイルセットを含む jba ドメインについて調べています。それぞれ 60,000 ブロックのクォータが設定されています。

```
# /sbin/advfs/vdf -l jba

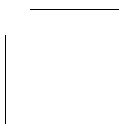
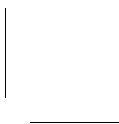
Domain      512-blocks  Metadata    Used    Available  Capacity
jba          266240      5824       29128    231288     13%

Fileset     QuotaLimit          Used    Available  Capacity
jbfsa       60000             20800    39200     35%
jbfs2a      60000             8328     51672     14%
```

次の例では、クォータが設定されていない2つのファイルセットを含む jbb ドメインに関する情報を表示しています。この例では、各ファイルセットで使用可能な総容量は、ドメインの総容量と同じです。

```
# /sbin/advfs/vdf -l jbb
Domain      512-blocks  Metadata    Used    Available  Capacity
```

jbb	266240	5824	29128	231288	13%
Fileset	QuotaLimit		Used	Available	Capacity
jbfsb	-		20800	231288	35%
jbfs2b	-		8328	231288	14%



データのバックアップとリストア

ファイル・システムのデータのバックアップは、アプリケーション障害、システム障害、およびさまざまな予期しない事態に備えた包括的な計画の重要な要素の1つです。適切なバックアップ計画をたてることで、バックアップしたデータが必要なときに確実に使用できるようになります。

この章では、バックアップ計画をたてる際に考慮する必要のある、次のような項目について説明します。

- データとメタデータの整合性の確認 (4.1 節)
- バックアップ・ソースの作成 (4.2 節)
- バックアップを行なうために使用できるメソッドとツール (4.3 節)

4.1 データとメタデータの整合性

データをバックアップする前に、バックアップするソース、つまり、ファイルのデータと AdvFS メタデータ (ファイル構造情報) が整合性のとれた状態にあり、バックアップの間に変わってしまわないことを確認します。

4.1.1 データ整合性の保証

バックアップは、収集したデータの整合性がとれていて、はじめて役に立ちます。バックアップを実行する前に、データ・ファイルをそのような状態にする必要があります。これには、次の2つの手順を行ないます。

1. アプリケーションを停止または休止し、データ・ファイルの現在の状態がアプリケーションの観点から矛盾がないようにします。
2. アプリケーションのデータ・ファイルへの更新が固定記憶装置に書き込まれ、アプリケーションまたはファイル・システムによってキャッシュされていないことを確認します。

これらの手順を踏まないと、バックアップしたデータがアプリケーションの観点から矛盾のあるものになり、バックアップとして使えなくなります。

Oracle のようなアプリケーションは、休止しデータをディスクにフラッシュする明示的な方法を用意しています。アプリケーションによっては、このような機能は用意していなくても、同期書き込み方式により、アプリケーションが停止するとデータがディスクに書き込まれるものがあります。しかし、休止・フラッシュ機能も同期書き込み技術も使用していないアプリケーションもあります。

4.1.1.1 アプリケーションの休止機能を使用したデータ整合性の保証

アプリケーションによっては、バックアップに備えて休止し、ディスクへのフラッシュを行なうものがあります。たとえば、Oracle では、これを Hot Backup Mode と呼んでいます。これを使用するには、次のような Oracle SQL コマンドを実行します。

```
ALTER TABLESPACE name BEGIN BACKUP
```

バックアップのソースが作成されると (4.2 節)、次の Oracle SQL コマンドを使用してテーブル・スペースを Hot Backup Mode から解放します。

```
ALTER TABLESPACE name END BACKUP
```

4.1.1.2 同期書き込み方式を使用したデータ整合性の保証

AdvFS では次の方法で、アプリケーションが書き込んだデータが、アプリケーションまたはファイル・システムのバッファ・キャッシュにではなく、確実に固定記憶装置に書き込まれることを保証します。

- `fsync()` システム・コールを使用してキャッシュされたデータをフラッシュします。
- ファイルを `O_SYNC` または `O_DSYNC` フラグをつけてオープンすると、すべてのキャッシュされたデータが同期をとってストレージに書き込まれます。

これらの技術を使用するアプリケーションを適切にシャットダウンすると、すべてのキャッシュされたデータは固定記憶装置に確実に書き込まれることになります。

4.1.1.3 同期書き込み方式を用いないデータ整合性の保証

アプリケーションが休止機能を持っていず、同期書き込みを呼び出さない場合、次に示す方法のいずれかを採ることで、アプリケーションがデータ・

4-2 データのバックアップとリストア

ファイルに書き込んだすべてのデータが、バックアップを開始する前に、固定記憶装置に書き込まれていることを保証できます。

- ファイルセットを `-o sync` オプションを指定してマウントします。これによってファイルセットのファイルへの更新が、同期をとってストレージに書き込まれるようになります。このオプションを使用すると、そのファイルセットを使用するすべてのアプリケーションの性能が著しく低下します。そこで、この他の方法のいずれかで、ストレージにフラッシュする必要があるデータ部分だけをフラッシュする方法をアプリケーション側で行なわせるようにする方が賢明です。
- `chfile` コマンドに `-l on` オプションを指定して実行すると、そのファイルを使用するアプリケーションの要求する I/O モードが何であっても、ファイルへの更新が同期をとってストレージにフラッシュされるようになります。I/O モードについては、5.5.1 項を参照してください。
- アプリケーションを適切な段取りを踏んで停止します。これにより、オペレーティング・システムの `smoothsync` デーモンがキャッシュされたデータをすべてディスクにフラッシュします。アプリケーションが停止された後、`smoothsync_age` カーネルのチューニング・パラメータに指定した秒数の 2 倍以上待ってから、バックアップを開始、またはバックアップ・ソースを作成してください。`smoothsync_age` チューニング・パラメータの値は、次のコマンドで表示することができます。

```
# sysconfig -q vfs smoothsync_age
```

4.1.2 メタデータの整合性の保証

データ・ファイルがアプリケーションの観点で矛盾がないことを確認したら、次は AdvFS メタデータの整合性を確認します。これを適切に行なわないと、AdvFS ドメイン・パニック、システム・パニック、またはユーザ・データの損傷が生じる可能性があります。バックアップ・ソース内の AdvFS メタデータの整合性を保証するには、次のいずれかの手順を実行します。

- `freezefs` コマンドを使用してドメインをフリーズします。(4.1.2.1 項)
- ドメインのすべてのファイルセットをアンマウントします(4.1.2.2 項)。

4.1.2.1 ドメインをフリーズすることによるメタデータ整合性の保証

`freezefs` コマンドによって、ドメインはメタデータの整合を保った状態になり、指定した凍結時間が過ぎるまで、または `thawfs` コマンドで明示的に

解凍されるまで、その状態を保ちます。ドメイン内のすべてのファイルセットが凍結されます。すべてのメタデータは、複数のボリュームまたは論理ユニット (LUN) に分散されて配置されていることもありますが、すべてディスクにフラッシュされ、凍結期間中は変更されません。

ファイル・システムをフリーズすると、すべての処理中のファイル・システムの操作は処理を完了することができます。メタデータのアップデートが不要なファイル・システム操作 (読み取り操作など) は、ファイル・システムがフリーズしても正常に機能します。

ファイル・システムはフリーズすると、次のいずれかの動作で解凍されるまでメタデータの整合が取れている状態を保ちます。

- タイムアウト
- `thawfs` コマンドの実行
- クラスタ環境では、フリーズしたファイル・システムのあるノードのシャットダウン、またはクラスタ・メンバのいずれかでの障害の発生。

`freezefs` コマンドは、省略時設定でファイル・システムを 60 秒間フリーズします。タイムアウト値を、これより大きく、あるいは小さくするには `-t` オプションを使用し、秒単位で指定します。またこのオプションで、`thawfs` コマンドで解凍されるまでフリーズしたままになるように指定することもできます。詳細については、`freezefs(8)` および『クラスタ管理ガイド』を参照してください。

4.1.2.2 ファイルセットをアンマウントすることによるメタデータ整合性の保証

ドメインのすべてのファイルセットをアンマウントすれば、AdvFS メタデータが変更されないことが保証されます。メタデータの整合性を保ちたい場合は、アンマウントしたドメインに対して実行できる AdvFS Utilities は使用しないでください。

4.2 バックアップ・ソースの作成

マウントしたファイルセットからバックアップ・ソースを作成すると、バックアップを行なうことができます。ソースは次のいずれかをもとにして作成します。

- オリジナルのファイルセット (4.2.1 項)
- ファイルセット・クローン (4.2.2 項)

4-4 データのバックアップとリストア

- Logical Storage Manager (LSM) スプリット・ミラー (4.2.3 項)
- コントローラ・ベースのクローン (4.2.4 項)
- コントローラ・ベースのスナップショット (4.2.5 項)

4.2.1 マウントしたオリジナル・ファイルセットのバックアップ・ソースとしての使用

状況によっては、マウントしたオリジナル・ファイルセットをバックアップ・ソースとして使用することができます。

4.2.1.1 長所と短所

表 4-1 に、オリジナル・ファイルセットをバックアップ・ソースとして使用することの長所と短所を示します。

表 4-1: マウントしたオリジナル・ファイルセットのバックアップ・ソースとしての使用

長所	短所
簡単さ。新しくマウント・ポイントを作成する必要はありません。	バックアップ中はマウント・ポイントにあるファイルをアップデートできない。アップデートすると、データに矛盾が生まれる恐れがあります。
	バックアップ処理を他のホストに任せることはできない。

4.2.1.2 データとメタデータ整合性の保証

ファイルセットのデータの整合性を保つには 4.1 節の手順に従います。

ファイル・システムに POSIX インタフェースを介してアクセスするバックアップ・アプリケーション (vdump ユーティリティなど) では、メタデータの整合性を保証するのに特に何もする必要はありません。

4.2.1.3 バックアップ・ソースの準備

マウントしたオリジナル・ファイルセットを使ってバックアップ・ソースを準備する場合、データとメタデータの整合性を保つために特に何もする必要はありません。

4.2.2 マウントしたファイルセット・クローンのバックアップ・ソースとしての使用

ファイルセット・クローン (2.4.10 項を参照) は、既存のファイルセット・データの読み取り専用スナップショットです。ファイルセット・クローンには、作成時のすべてのデータが入っている訳ではありません。オリジナル・ファイルのデータを変更すると、AdvFS はオリジナルにあるデータをページ単位でファイルセット・クローンに保存します。

ファイルセット・クローンを作成するには、オプションの AdvFS Utilities が必要です。ファイルセットのクローンを作成できるのは、root ユーザだけです。1 つのファイルセットにつき、1 つのクローンしか作成できません。

4.2.2.1 長所と短所

表 4-2 に、ファイルセット・クローンをバックアップ・ソースとして使用することの長所と短所を示します。

表 4-2: マウントしたファイルセット・クローンのバックアップ・ソースとしての使用

長所	短所
追加のストレージ・ハードウェアは必要ない。ドメインにクローン用のスペースがある。	ファイルセット・クローンは、同じドメイン内のファイルセットとして作成される。ストレージに障害が発生するとオリジナルとクローンの両方のファイルセットが影響を受けることになります。
簡単に使用できる。 ^a	バックアップ処理の負荷を他のマシンに移行できない。ファイルセット・クローンは、オリジナルのファイルセットと同じホストにマウントされる必要があります。

表 4-2: マウントしたファイルセット・クローンのバックアップ・ソースとしての使用 (続き)

長所	短所
オリジナルのファイルセットのオフライン時間はファイルセット・クローンを作成するのにかかる時間だけでよい (通常数秒)。	クローンへの書き込みはファイルセットの I/O に影響を与える。 オリジナルのファイルセットにつき、1 つのファイルセット・クローンしか作成できない。データが複数のファイルセットにまたがっている場合、クローン作成の際は、一貫性を保証するためにアプリケーションを停止する必要がある場合があります。

^aファイルセット・クローンを作成するには、コマンド行か、SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A 参照)、または AdvFS GUI (付録 E を参照) を使用します。また、ファイルセット・クローンを処理するプログラムを書くのに、AdvFS API を使用することができます。詳細は付録 F を参照してください。

4.2.2.2 データとメタデータの整合性の保証

ファイルセットのデータの整合性を保つには 4.1 節の手順に従います。データが複数のファイルセットにまたがっている場合、すべての関連するファイルセットの整合性を確認する必要があります。

ファイルセット・クローン作成はすべて、AdvFS 内で行なわれるので、AdvFS メタデータの整合性を保証するために特に何もする必要はありません。

4.2.2.3 バックアップ・ソースの準備

ファイルセット・クローンを作成するために、オリジナル・ファイルセットのデータが整合性のとれた状態であることを確認します。そして、clonefset コマンドでファイルセット・クローンを作成します。

次の例では、ドメイン domain1 内のファイルセット pssm のクローンを作成しています。新しいファイルセット・クローンの名前は pssm_clone です。

- 1. クローンを作成しマウントします。

```
# clonefset domain1 pssm pssm_clone
# mkdir /pssm_clone
# mount -t advfs domain1#pssm_clone /pssm_clone
```

- 2. pssm_clone を使用してバックアップ・プロセスを実行します。詳細は 4.3.11 項 を参照してください。
- 3. クローンを削除します。

```
# umount /pssm_clone
# rmfset domain1 pssm_clone
```

4.2.3 LSM スプリット・ミラーからマウントしたファイルセットのバックアップ・ソースとしての使用

バックアップしたいデータが、LSM ミラー・ボリュームに構築されたドメイン上にある場合、LSM ミラーをスプリットし(ミラーを切り離し)、マウントし、バックアップ・ソースとして使用することができます。この操作を行なうには、root ユーザの権限が必要です。ミラーを切り離す方法については、『Logical Storage Manager』を参照してください。

4.2.3.1 長所と短所

表 4-3 に、LSM スプリット・ミラーをバックアップ・ソースとして使うことの長所と短所を示します。

表 4-3: LSM スプリット・ミラーからマウントしたファイルセットのバックアップ・ソースとしての使用

長所	短所
データの完全な物理コピーが作成される。オリジナル・ファイルセットが置かれるストレージのハードウェアが故障しても、スプリット・ミラーには影響しません。	LSM ライセンスが必要。
オリジナルのファイルセットのオフライン時間は、LSM ミラーをスプリットするのにかかる時間だけですむ(通常数秒)。	少なくともデータの2つのコピーを保持するのに十分な物理ストレージが必要。
データが複数のシングル・ドメインのファイルセットにまたがっている場合、LSM スプリット・ミラーはそのすべてのバックアップとなる。	LSM と AdvFS の両方のコマンドが必要なので複雑。
複数の LSM スプリット・ミラーを、同時に1つのドメインについて作成できる。	バックアップ処理の負荷を他のマシンに移行できない。LSM スプリット・ミラー・ファイルセットは、オリジナルのファイルセットと同じホストにマウントされる必要があります。

4.2.3.2 データとメタデータの整合性の保証

ファイルセットのデータの整合性を保つには 4.1 節の手順に従います。

ストレージが 1 つの LSM ボリュームからなるドメインの場合、AdvFS メタデータの整合性を保証するために特に何もする必要はありません。ストレージが複数の LSM ボリュームからなる場合は、4.1.2 項を参照してください。

4.2.3.3 バックアップ・ソースの準備

ドメインが複数の LSM ボリュームからなる場合だけ、凍結/解凍の手順が必要です。複数のボリュームからなるかわからない場合は、とりあえず、これらのコマンドを試してみます。ドメインに影響はありません。単一ファイルセットの凍結/解凍の手順で、ドメイン全体を凍結/解凍できます。

次の例では、オリジナル・ドメインが 2 つの LSM ボリュームからなり、それぞれからミラーが切り離されていると仮定します。データは、/data にマウントされているファイルセット data のあるドメインにあります。2 つのミラー・ボリュームは、/dev/vol/mirrorvol1 と /dev/vol/mirrorvol2 です。

1. ドメインをフリーズします。

```
# /usr/sbin/freezefs /data
```

2. ミラーを切り離し、新しい LSM ボリュームとして再構築するために LSM コマンドを実行します。手順は、『*Logical Storage Manager*』を参照してください。

3. LSM スプリット・ミラーを作成したら、ドメインのメタデータを解凍します。

```
# /usr/sbin/thawfs /data
```

アプリケーションは引き続き、オリジナル・ドメインで動作できます。

4. 2 つの新しく作成した LSM ボリュームを使用して、新しいドメインのディレクトリを作成します。

```
# mkdir /etc/fdmns/mirror_domain
# ln -fs /dev/vol/mirrorvol1 /etc/fdmns/mirror_domain
# ln -fs /dev/vol/mirrorvol2 /etc/fdmns/mirror_domain
```

新しいディレクトリを作成するのに `mkfdmn` コマンドは使用しないでください。ミラーにアクセスできなくなります。

5. マウント・ポイント・ディレクトリを作成し、mirror_domain のスプリット・ミラーから作成したファイルセット data をマウントします。

```
# mkdir /backup
# mount -o dual mirror_domain#data /backup
```

バックアップは /backup ディレクトリから行なえるようになります。

4.2.4 コントローラ・ベースのクローンからマウントしたファイルセットのバックアップ・ソースとしての使用

Compaq HSG80 や HSV110 などのストレージ・コントローラは、コントローラ・ベースのコマンドを使用してハードウェア・ミラーをスプリットできます。これは、概念的にはスプリット・ミラーと同じ原理です。すなわち、データの 2 つ以上の完全な物理コピーが存在し、1 つがバックアップのために切り離されている状態です。

4.2.4.1 長所と短所

表 4-4に、コントローラ・ベースのクローンをバックアップ・ソースとして使用することの長所と短所を示します。

表 4-4: コントローラ・ベースのクローンからマウントしたファイルセットのバックアップ・ソースとしての使用

長所	短所
データの完全な物理コピーが作成される。オリジナル・ファイルセットが置かれるストレージのハードウェアが故障しても、コピーには影響しません。	少なくとも 2 つのデータのコピーを収容するのに十分な物理ストレージが必要。
オリジナルのファイルセットのオフライン時間は、コントローラ・ベースのクローンを作成するのにかかる時間だけですむ (通常数秒)。	コントローラと AdvFS の両方のコマンドが必要なので複雑。
バックアップ処理の負荷を他のマシンに移行できる。コントローラ・ベースのクローンは、多くの場合、異なるホストにマウントできます。	
ドメインのすべてのファイルセットに対し単一のバックアップ・ソースとして機能できる。	
複数のコントローラ・ベースのクローンを、同時に同じドメインに作成できる。	

4.2.4.2 データとメタデータの整合性の保証

ファイルセットのデータの整合性を保つには 4.1 節の手順に従います。

ストレージが 1 つのハードウェア LUN からなるドメインの場合、AdvFS メタデータの整合性を保証するために特に何もする必要はありません。ストレージが複数のハードウェア LUN からなる場合は、4.1.2 項を参照してください。

4.2.4.3 バックアップ・ソースの準備

ドメインが複数の LUN からなる場合だけ、凍結/解凍の手順が必要です。複数のボリュームからなるかわからない場合は、とりあえず、これらのコマンドを試してみます。ドメインに影響はありません。単一ファイルセットの凍結/解凍の手順で、ドメイン全体を凍結/解凍できます。

次の例では、データは、ファイルセット `data` があるドメインに置かれていて、そのドメインは `/data` にマウントされていると仮定します。作成されるクローンは、`/dev/disk/dsk25c` と `/dev/disk/dsk26c` です。

1. ドメインをフリーズします。

```
# /usr/sbin/freezefs /data
```

2. コントローラ・ベースのクローンを作成するのに必要なコマンドを実行します。たとえば、Compaq HSG80 コントローラを使用する場合の、これらのコマンドの例は、Best Practice ドキュメントの『*Using StorageWorks HSG80 Controller-Based Cloning and Snapshotting*』を参照してください。

3. ドメインのメタデータを解凍します。

```
# /usr/sbin/thawfs /data
```

アプリケーションは引き続き、オリジナル・ドメインで動作できます。

4. 新しくクローンを作成した LUN のブロック型デバイス特殊ファイルを作成します。これは、新しいバックアップ・ソースをマウントするホスト上で行ないます。`/sbin/hwmgr` コマンドは非同期に実行されるため、実際にブロック型デバイス特殊ファイルが作成される前に終了している場合があります。

スタンドアロン・システムの場合、次のコマンドを実行します。

```
# /sbin/hwmgr -scan scsi
```

クラスタの場合、次のコマンドを実行します。

```
# /sbin/hwmgrr scan component -category scsi_bus -cluster
```

新しい LUN が作成されるたびに、新しいデバイス特殊ファイルが作成されます。

5. ブロック型デバイス特殊ファイルを探します。

```
# /sbin/hwmgrr -view devices
```

6. /dev/disk/dsk25c と /dev/disk/dsk26c の新しいドメインにアクセスするための、ディレクトリとシンボリック・リンクを作成します。

```
# mkdir /etc/fdmns/clone_domain
# ln -fs /dev/disk/dsk25c /etc/fdmns/clone_domain
# ln -fs /dev/disk/dsk26c /etc/fdmns/clone_domain
```

新しいディレクトリを作成するのに mkfdmn コマンドは使用しないでください。クローン・データにアクセスできなくなります。

7. ディレクトリを作成し、clone_domain のファイルセット data をマウントします。

```
# mkdir /backup
```

クローン・ドメインとファイルセットを元のホストにマウントしていない場合、次のコマンドを入力します。

```
# mount clone_domain#data /backup
```

クローン・ドメインとファイルセットを元のホストにマウントしている場合、次のコマンドを入力します。

```
# mount -o dual clone_domain#data /backup
```

バックアップは /backup ディレクトリから行なえるようになります。

4.2.5 コントローラ・ベースのスナップショットからマウントしたファイルセットのバックアップ・ソースとしての使用

Compaq HSG80 や HSV110 などのストレージ・コントローラでは、データセットの重要な RAID メタデータのコピーであるスナップショットを別のメディアに作成できます。この機能では、書き込み時コピー技術を使用して、オリジナル・データ・セットの変更に即したコピーを維持します。これは、概念的にはファイルセット・クローンと同じ原理です。

4.2.5.1 長所と短所

表 4-5に、コントローラ・ベースのスナップショットをバックアップ・ソースとして使用することの長所と短所を示します。

表 4-5: コントローラ・ベースのスナップショットからマウントしたファイルセットのバックアップ・ソースとしての使用

長所	短所
ドメインのすべてのファイルセットのバックアップ・ソースとして機能できる。	少なくとも2つのデータのコピーを収容するのに十分な物理ストレージが必要。
オリジナルのファイルセットのオフライン時間は、コントローラ・ベースのスナップショットを作成するのにかかる時間だけですむ (通常数秒)。	コントローラと AdvFS の両方のコマンドが必要なので複雑。

4.2.5.2 データとメタデータの整合性の保証

ファイルセットのデータの整合性を保つには 4.1 節の手順に従います。

ストレージが1つのハードウェア LUN からなるドメインの場合、AdvFS メタデータの整合性を保証するために特に何もする必要はありません。ストレージが複数のハードウェア LUN からなる場合は、4.1.2 項を参照してください。

4.2.5.3 バックアップ・ソースの準備

ドメインが複数の LUN からなる場合だけ、凍結/解凍の手順が必要です。複数のボリュームからなっているかわからない場合は、とりあえず、これらのコマンドを試してみます。ドメインに影響はありません。単一ファイルセットの凍結/解凍の手順で、ドメイン全体を凍結/解凍できます。

次の例では、データが、ファイルセット `data` があるドメインに置かれていて、そのドメインは `/data` にマウントされていると仮定します。作成されるスナップショットは、`/dev/disk/dsk25c` と `/dev/disk/dsk26c` です。

1. ドメインをフリーズします。省略時の凍結時間は 60 秒です。

```
# /usr/sbin/freezefs /data
```
2. コントローラ・ベースのスナップショットを作成するのに必要なコマンドを実行します。たとえば、Compaq HSG80 コントローラを使用する場合の、これらのコマンドの例は、Best Practice ドキュメントの『Using

StorageWorks HSG80 Controller-Based Cloning and Snapshotting』を参照してください。

3. ドメインのメタデータを解凍します。

```
# /usr/sbin/thawfs /data
```

アプリケーションは引き続き、オリジナル・ドメインで動作できます。

4. 新しくスナップショットを作成した LUN のブロック型デバイス特殊ファイルを作成します。これは、新しいバックアップ・ソースをマウントするホスト上で行ないます。/sbin/hwmgrr コマンドは非同期に実行されるため、実際にブロック型デバイス特殊ファイルが作成される前に終了している場合があります。

スタンドアロン・システムの場合、次のコマンドを実行します。

```
# /sbin/hwmgrr -scan scsi
```

クラスタの場合、次のコマンドを実行します。

```
# /sbin/hwmgrr scan component -category scsi_bus -cluster
```

新しい LUN が作成されるたびに、新しいデバイス特殊ファイルが作成されます。

5. ブロック型デバイス特殊ファイルのファイルを探します。

```
# /sbin/hwmgrr -view devices
```

6. /dev/disk/dsk25c と /dev/disk/dsk26c の新しいドメインにアクセスするための、ディレクトリとシンボリック・リンクを作成します。

```
# mkdir /etc/fdmns/snap_domain
# ln -fs /dev/disk/dsk25c /etc/fdmns/snap_domain
# ln -fs /dev/disk/dsk26c /etc/fdmns/snap_domain
```

新しいディレクトリを作成するのに mkfdmn コマンドは使用しないでください。スナップショット・データにアクセスできなくなります。

7. ディレクトリを作成し、snap_domain のファイルセット data をマウントします。

```
# mkdir /backup
```

スナップショット・ドメインとファイルセットを元のホストにマウントしていない場合、次のコマンドを入力します。

```
# mount snap_domain#data /backup
```

スナップショット・ドメインとファイルセットを元のホストにマウントしている場合、次のコマンドを入力します。

```
# mount -o dual snap_domain#data /backup
```

バックアップは /backup ディレクトリから行なえるようになります。

4.3 バックアップ方式とツール

データとメタデータの整合性が確認されたバックアップ・ソースを作成したら、データをバックアップすることができます。これには、いくつかの方法があります。

- バックアップを行なうのにオリジナル・データと同じホスト上にバックアップ・ソースをマウントする。バックアップ・ソースがオリジナル・ファイルセットまたはファイルセット・クローンのときに必要です。
- バックアップを行なうのにオリジナル・データとは異なるホスト上にバックアップ・ソースをマウントする。これによって元のホストのバックアップ処理による負荷を軽減することができます。
- raw LUN の内容をディスクから直接テープ、または他のメディアにバックアップする。

バックアップとリストアには、多くのコマンドを使用することができます。AdvFS では、vdump および vrestore コマンドと、リモートから実行する rvdump および rvrestore コマンドを利用できます。この節では、これらのコマンドの使用法を説明します。

4.3.1 vdump および vrestore コマンドの概要

バックアップ・ソースを準備したら vdump コマンドを使用してバックアップ作業を行なうことができます。vdump コマンドはファイル・レベルで作用します。このコマンドは、各ディレクトリを走査し、正規の POSIX ファイル・システム・コールを使用してディレクトリおよびファイルにアクセスします。詳細については、vdump(8) および vrestore(8)を参照してください。

dump および restore コマンドは、vdump および vrestore コマンドと違った機能をします。これらは、i ノード・レベルで作用するので、UFS ファイルだけを扱うことができます。

この項では、vdump および vrestore コマンドについてのみ説明しますが、リモート操作の場合は rvdump および rvrestore コマンドに置き換えてお読みいただけます。

注意

`vdump` コマンドや `vrestore` コマンドは、`root` 以外のユーザでも実行できます。ただし、ファイルのリストア先ディレクトリへの書き込み許可が必要になります。AdvFS のユーザとグループの、クォータ・ファイルとファイルセット・クォータの保存およびリストアができるのは `root` ユーザだけです。

注意

Version 4.0 より前のオペレーティング・システムの `vrestore` コマンドでは、Version 4.0 以降の `vdump` コマンドで保管された情報を正しくリストアすることができません。`vdump` と `vrestore` のダンプ・ファイルの形式は、Version 4 と Version 5 のオペレーティング・システムで互換性があります。

`vdump` コマンドは、新しいファイルや一定の日付以降に変更したファイルを、省略時のストレージ・デバイスまたは指定のデバイスにコピーする際に、セーブセットと呼ばれる固定長ブロックの集合を作成します。

セーブセットは、複数のテープにまたがってバックアップすることができ、また同じテープに複数のセーブセットを収めることもできます。テープ上の各セーブセットはファイル・マークによって区切られます。ファイル・マークは、`vdump` コマンドがセーブセットをクローズするときに書き込まれます。

`vdump` コマンドの構文を次に示します。

```
vdump options mount_point
```

次の例では、`/psm` にマウントされたファイルセットをテープにダンプしています。

```
# vdump -0 -f /dev/tape/tape0_d1 /psm
```

4.3.2 `vdump` コマンド固有の機能

`vdump` コマンドは、次に示すように、UFS `dump` コマンドが持っていない多数の機能を備えています。`vdump` コマンドを使用して次の作業を行うことができます。

- マウントされたファイルセットを保存する。
- バックアップしたいサブディレクトリを選択する。必ずしもファイルセット全体をダンプする必要はありません。
- ファイルを圧縮して、セーブセット・サイズを最小限に抑える。
- メモリ内のバッファ数を指定する。ストレージ・デバイスに適したバッファ数を指定することにより、スループットを最大限に高めることができます。
- 現在の `vdump` バージョン番号を表示する。
- ダンプ・プロセス中にヘルプ情報を表示する。
- 警告メッセージは表示しないで、エラー・メッセージのみを表示するように指定する。
- バックアップ時にファイルの名前を表示する。
- エラー保護システムによって出力を構成することにより、リストア時に読み取りエラーが生じた場合でもデータをリストアする。
- ゼロで満たさない AdvFS および UFS のスパーズ・ファイルを処理する。

4.3.3 `vdump` コマンドを使用したバックアップ・レベルの指定

`vdump` コマンドでは、増分バックアップのレベル (ダンプ・レベル) を指定することができます。0 を指定すると、ファイルセットの完全なバックアップが作成されます。ダンプ・レベルの値が大きいくほど、バックアップされるデータ量は小さくなります。`vdump(8)` のリファレンス・ページには、ダンプ・レベルをサイクリックに増減して、2 本のテープに完全なバックアップを確保する方法が示されています。

ユーザおよびグループの、クォータ・ファイルとファイルセット・クォータは、`root` ユーザがダンプ・レベル 0 でバックアップした場合のみ保存されます。バックアップできるクォータは、ローカルでマウントしているシステムのクォータのみです。

`vdump` コマンドは、ファイルの更新日時を基にバックアップを行います。ところが、この方法では増分バックアップが意図したとおりに実行されない可能性があります。ファイルの名前や場所を変更しても、ファイルの内容が変わらない限り、更新日時は変更されないからです。この問題を回避するに

は、バックアップ後にファイルの名前や場所を変更した場合に、touch コマンドを使って該当する各ファイルの更新日時を変更してください。

```
touch filename
```

4.3.4 vdump コマンドを使用したテープへのダンプ

vdump コマンドで、1 本のテープに複数のセーブセットを格納 (ダンプ) することができます。-N オプションを設定して、巻き戻しを行わないように指定するか、あるいは /dev/ntape/tape0 などの巻き戻しなしのデバイスを指定してください。これにより、vdump コマンド終了時のテープの巻き戻しを回避できます。次の vdump コマンドの実行時には、セーブセットが現在のテープ位置から格納されます。

セーブセットをダンプするのに複数のテープが必要な場合、vdump コマンドはその時点で別のテープをマウントするよう指示します。

複数のセーブセットを含むテープからデータをリストアする方法については、4.3.16 項を参照してください。

システムのテープ・デバイスの情報を知るには、/sbin/hwmgrr コマンドに -view devices オプションを指定します。

詳細は hwmgrr(8) および『ハードウェア管理ガイド』を参照してください。

4.3.5 vdump コマンドを使用した標準出力へのダンプ

バックアップのセーブセット・デバイスとしてハイフン (-) を指定すると、vdump コマンドは標準出力に書き出します。したがって、vdump コマンドと vrestore コマンドをパイプで連結すれば、ファイルセットを別のファイルセットにコピーすることができます。一般的なコマンド例を以下に示します。この 2 例は同等の動作を実行します。

```
# vdump -0f - /usr | vrestore -xf - -D /mnt
```

```
# vdump -0 -f - /usr | (cd /mnt; vrestore -x -f -)
```

rvdump および rvrestore コマンドには、ハイフン (-) は使用できません。出力デバイスを指定する必要があります。

4.3.6 vdump コマンドを使用したサブディレクトリのダンプ

vdump コマンドに `-D` オプションを使用して、サブディレクトリを指定すれば、ファイルセット内の指定したサブディレクトリのみをバックアップできます。`-D` オプションを使用せずに、ファイルセットの代わりにサブディレクトリを指定すると、vdump コマンドは、そのサブディレクトリを含むファイルセット全体をバックアップします。`-D` オプションを指定すると、バックアップは常にレベル 0 で実行されます。

4.3.7 ファイルやディスク・パーティションへのダンプ

セーブセットはファイルとしてディスク上に保存することが可能です。たとえば、完全バックアップを週一回の頻度でテープに格納して保管しておく一方、増分バックアップをディスク上に保存すれば、増分バックアップを手軽に利用できます。ディスクはテープ・デバイスより高速なので、ディスクに保存すれば、バックアップやリストアをより速やかに実行できます。

月曜日に行った作業の増分バックアップを、その日の夕方にファイルとして保存する例を次に示します。この例では、その日のファイルをファイルセット `projects.Monday` としてバックアップします。このファイルセットは `/projects` にマウントされています。

```
# vdump -9f /backup/projects.Monday /projects
```

月曜日にディスク上にバックアップした `revenue` という名前のファイルをリストアするには、次のコマンドを実行します。

```
# vrestore -xf /backup/projects.Monday -D /projects revenue
```

セーブセットは空のパーティションに格納することも可能です。しかしその場合、格納したセーブセットはファイル・システムの制御下では利用できず、パーティションの再割り当てを行うと失われてしまう可能性があります。`/projects` にマウントされているファイルセットをパーティション `/dev/disk/dsk2g` にダンプする例を次に示します。

```
# vdump -f /dev/disk/dsk2g -D /projects
```

システムのデバイスを調べるには、`/sbin/hwmgr` コマンドに `-view devices` オプションを指定します。

パーティション `a` やパーティション `c` は、バックアップの保存先として指定しないでください。これらのパーティションにはブロック 0 が含まれており、その中にはディスク・ラベルが格納されているためです。デバイ

ス・ドライバは、ディスク・ラベルを上書きしないので、データの一部が消滅します。パーティション a やパーティション c へのバックアップを試みると、文字型デバイス (つまり raw デバイス) を使用している場合のみエラー・メッセージが表示されます。ブロック型特殊デバイスを使用している場合には、エラーは返されません。ディスクの空きスペースに余裕があるときには、ブロック 0 を含まない他のディスク・パーティションをバックアップ先として使用してください。

パーティション a やパーティション c をバックアップ先として使用する場合には、ディスクの残りの領域にデータが存在しないこと、および `disklabel` コマンドに `-z` オプションを使用してあらかじめディスク・ラベルをクリアすることが条件になります。ディスク・ラベルをクリアすると、そのディスクに格納されていたデータはすべて失われます。ディスク・ラベルをクリアしない場合でも、`vdump` コマンドではセーブセットが正しく作成されたように見える場合があります。ところが、`vrestore` コマンドを実行すると、ディスク・ラベルがセーブセットの一部として解釈され、次のようなメッセージが表示されます。

```
vrestore: unable to use save-set; invalid or corrupt format
```

`rvdump` や `rvrestore` コマンドでも、同じ結果になります。

詳細は、`disklabel(8)`、`vdump(8)`、`vrestore(8)`、および『システム管理ガイド』を参照してください。

ディスク全体をコピーする方法については、『システム管理ガイド』を参照してください。

4.3.8 セーブセットの圧縮

バックアップ時には、セーブセットを圧縮できます。これによりバックアップに必要なストレージの量が減り、書き込まれるデータが少なくなるため、低速デバイス上でもより高速にダンプを実行できるようになります。圧縮を行うには `vdump` コマンドに `-c` オプションを指定して実行します。圧縮率を指定することはできません。圧縮率はダンプの内容によって決まります。

注意

ハードウェア圧縮を自動的に行うテープ・ドライブを使用している場合は、圧縮を指定して `vdump` コマンドを使用しても、期待したよりも大きいセーブセットになることがあります。場合によ

ては、圧縮アルゴリズムが原因で、圧縮済みのデータをさらに圧縮しようとする、データが大きくなることがあります。

4.3.9 エラー保護機能付きのダンプ

`vdump` コマンドに `-x` オプションを使用すれば、破損したブロックを `vrestore` コマンドでリストアできるように、テープ上にチェックサム・ブロックを置くことができます。 `vdump` コマンドは、ユーザが指定する n ブロックごとに、これらのブロックを作成します。 n の有効範囲は 2 ~ 32 です。省略時の値は 8 です。 `vrestore` コマンドがブロックの不良で読み取りエラーを検出すると、他のブロックおよびチェックサム・ブロックを使用して不良ブロックを再作成します。

エラー保護付きのダンプでは、 n ブロックごとに余計な 1 ブロックを保管する必要があります。この機能では、ブロックのリストア時に、一連の n ブロックの次の 1 ブロックしか修正できません。この機能では、保護機能を上げるほど、ストレージが余計に必要になります。

- テープでは信頼性が低いと思われる場合、あるいは正確なバックアップが必要でバックアップ用に多数のテープを利用できる場合には、`-x` オプションの値を 2 に設定します。これにより、保存される 2 ブロックごとに不良ブロックが 1 つあるようなエラーの訂正を行うことが可能になります。ただし、2 ダンプ・ブロックごとにチェックサム・ブロックが書き込まれるため、必要なテープの量は 50% 増加します。
- テープの信頼性は高いと考えられるが、稀に発生する不良ブロックを訂正できるようにするためには、`-x` オプションの値を 32 に設定します。この場合には、32 ブロック書き込むごとにブロックが 1 つ追加されるため、3% の余分なテープが必要になります。これで、32 ダンプ・ブロックごとに 1 つの不良ブロックであれば、正しくリストアできるようになります。

4.3.10 `vdump` セーブセットに保存したファイルのリスト

セーブセットを調べることで、必要なファイルがバックアップできているかどうかを確認することができます。バックアップが完了したら、`vrestore` コマンドに `-t` オプションを指定して実行し、保存したファイルを表示します。このコマンドでは、リストア処理は行われません。

4.3.11 clonefset コマンドと vdump コマンドを使用したシステム・ディスクのバックアップ(例)

この例では、clonefset コマンドでバックアップ情報のソースを作成したディスクの、レベル0のバックアップを示します。ここで、バックアップ用ディスク dsk6 は、バックアップ対象ディスク dsk0 と同じサイズであるとしています。この例では、ルート・ドメイン root_domain は /dev/disk/dsk0a 上にあります。ユーザ・ドメイン usr_domain は /dev/disk/dsk0g 上にあり、スワップ・パーティションは /dev/disk/dsk0b 上にあります。/var ファイルセットは usr_domain にあります。

ファイルセット・クローンを作成する間は、システムをシャットダウンし、シングルユーザ・モードでリブートすることをお勧めします。これにより、/、/usr、/var で他のユーザが作業していないことが保証されます。バックアップ作業はマルチ・ユーザ・モードでも可能ですが、システムの活動の少ない時期を選んで作業することを強くお勧めします。

1. 現在のシステム・ディスクから直接ディスク・ラベルをコピーし、新しいシステム・ディスクにリストアします。

```
# disklabel -r dsk0 > /tmp/backupdisklabel
# disklabel -R -r -t advfs dsk6 /tmp/backupdisklabel
# rm /tmp/backupdisklabel
```

2. バックアップ・ドメインとバックアップ・ファイルセットを作成し、マウントします。

```
# mkfdmn -F -r /dev/disk/dsk6a root_domain_backup
# mkfset root_domain_backup root
# mkfdmn -F /dev/disk/dsk6g usr_domain_backup
# mkfset usr_domain_backup usr
# mkfset usr_domain_backup var
# mkdir /root_backup /usr_backup /var_backup
# mount root_domain_backup#root /root_backup
# mount usr_domain_backup#usr /usr_backup
# mount usr_domain_backup#var /var_backup
```

3. ファイルセット・クローンを作成し、マウントします。

```
# clonefset root_domain root root_clone
# clonefset usr_domain usr usr_clone
# clonefset usr_domain var var_clone
# mkdir /clones /clones/root /clones/usr /clones/var
# mount root_domain#root_clone /clones/root
# mount usr_domain#usr_clone /clones/usr
# mount usr_domain#var_clone /clones/var
```

4. `vdump` コマンドを使用してファイルセット・クローンをダンプし、`vrestore` コマンドを使用してファイルセットをリストアします。


```
# vdump -0f - /clones/root | vrestore -xf - -D /root_backup
# vdump -0f - /clones/usr | vrestore -xf - -D /usr_backup
# vdump -0f - /clones/var | vrestore -xf - -D /var_backup
```
5. バックアップの `/etc/fdmns` ディレクトリを変更して正しいディスクを指すようにします。


```
# rm -rf /root_backup/etc/fdmns/root_domain
# mv /root_backup/etc/fdmns/root_domain_backup\
    /root_backup/etc/fdmns/root_domain
# rm -rf /root_backup/etc/fdmns/usr_domain
# mv /root_backup/etc/fdmns/usr_domain_backup\
    /root_backup/etc/fdmns/usr_domain
```
6. 次のようにして `/etc/sysconfigtab` ファイルを編集し、新しいスワップ・ファイルを指すようにします。
 - a. スワップ・デバイスを設定しているエントリを探します。


```
swapdevice=/dev/disk/dsk0b
```
 - b. 新しいスワップ・デバイス名と置き換えます。


```
swapdevice=/dev/disk/dsk6b
```
7. システムをシャットダウンし、新しく作成したバックアップを確認するため `dsk6` からリブートします。

`dsk6` をバックアップ用に作成していて、元のルート・ドメインに戻したいときは、シャットダウンして `dsk0` からリブートします。

4.3.12 リモートでのファイルのダンプおよびリストア

`rvdump` コマンドは、単一のマウント・ファイル・システムあるいは AdvFS ファイルセット・クローンからリモート・ストレージ・デバイスにファイルをバックアップします。このコマンドを使用する場合、ダンプしようとするリモート・ノードで `rsh` コマンドが実行可能であることが必要です。サーバおよびクライアント・アクセス規則については `rsh(8)` を参照してください。

`rvdump` コマンドは `vdump` コマンドと同じオプションを持ちますが、ファイルのバックアップを置きたいデバイスのノード名を指定する必要があります。次の例では、ファイルセット `sar` をノード `rachem` 上のテープにダンプしています。

```
# rvdump -0f rachem:/dev/tape/tape0 /sar
```

リモート・テープ・デバイスからファイルセットをリストアする場合は、次のように入力します。

```
# rvrestore -xf rachel:/dev/tape/tape0 -D /sar
```

リモート・バックアップ・コマンドでは、ユーザおよびグループのクォータ・ファイルとファイルセット・クォータはバックアップできません。

rvdump と rvrestore コマンドでは、標準出力を表すダッシュ (-) は認識しません。出力デバイスを指定する必要があります。

4.3.13 vrestore コマンド固有の機能

vrestore コマンドは、vdump コマンドで作成したセーブセットのブロックを処理することによってファイルを復元します。vrestore コマンドは、UFS dump コマンドで作成したセーブセットには機能しません。

vrestore コマンドを実行するのに root ユーザの権限は必要ありませんが、リストア先のディレクトリの書き込み権限を持っている必要があります。root ユーザだけが、クォータ・ファイルとファイルセット・クォータをリストアできます。詳細については、vrestore(8) を参照してください。

vrestore コマンドでは、UFS restore コマンドではサポートしない次のような処理をサポートします。

- vrestore の現在のバージョン番号を表示することができます。
- ソース・ディレクトリ・パスを表示することができます。
- セーブセット構造を表示することができます。これは、保存したファイルのリストです。
- 情報メッセージは表示せずにエラー・メッセージだけを表示することができます。
- 既存のファイルを見つけた場合の vrestore コマンドの動作を指定することができます。既存ファイルを常に上書きするか、既存ファイルは常に上書きしないか、あるいはそのたびにユーザに確認するかを選択することができます。

4.3.14 vrestore コマンドを使用したファイルのリストア

vrestore コマンドでは、リストアするファイルやディレクトリを選択することができます。このコマンドは、ファイル、パイプ、磁気テープ、またはディスクからデータをリストアすることができます。

vdump および vrestore コマンドは、同じバージョンのものを使用してください。使用しているバージョンの vrestore でセーブセットのフォーマットを読み取ることができない場合、エラー・メッセージが表示されます。

vrestore コマンドでリストアを実行する前に、このコマンドを `-t` オプション付きで実行すれば、セーブセットに含まれているすべてのファイルの名前とサイズを一括して表示できます。この場合リストア操作は行われません。

また、vrestore コマンドに `-i` (interactive) オプションを指定して実行すると、保存したファイルおよびディレクトリを表示することができます。これによって、リストアするファイルやディレクトリをリストから個々に選択することができます。

AdvFS ファイルセット・クローンからデータをリストアする作業は、その他のファイルセットからデータをリストアする場合と同様です。

ファイルセット全体をリストアする場合、まずフル・バックアップをリストアしてください。次にバックアップ以降に変更されたファイルのみが含まれる増分バックアップを使用してリストアしてください。この場合、フル・バックアップ実行後に削除されたファイルもリストアされることになります。そこで、これらのファイルは手作業で削除する必要があります。

4.3.15 vrestore コマンドを使用したクォータのリストア

AdvFS のユーザおよびグループのクォータ・ファイルは、AdvFS ファイルセット、UFS ファイル・システムのどちらにもリストアできます。AdvFS クォータ・ファイルを UFS ファイル・システムにリストアする場合、そのクォータは UFS ファイル・システム上で有効にしなければなりません。UFS には AdvFS ファイルセット・クォータに相当するものがないため、AdvFS ファイルセット・クォータを UFS ファイル・システムにリストアすることはできません。クォータのリストア操作は、root ユーザとして実行しなければなりません。

4.3.16 vrestore コマンドを使用した複数のセーブセットを含むテープからのリストア

複数のセーブセットを含んでいるテープから現在の作業ディレクトリに対してリストアを行うには、`mt fsf n` コマンド (n 個のセーブセットまたはファイルをスキップ) を実行して、リストアするセーブセットの位置を指定します。続いて、`vrestore` コマンドを実行します。

次の例では、テープ上の 4 番目のセーブセットを選択しリストアしています。

```
# mt fsf 3
# vrestore -xf /dev/ntape/tape0
```

リストアするセーブセット・ディレクトリの位置がわからない場合には、`vrestore` コマンドに `-i` オプションを指定して実行します。このコマンドで必要なセーブセットまで移動し、対話型インタフェースを使って取り込むファイルを指定します。

`vrestore` コマンドに `-x` オプションを指定し、ファイル名を指定して実行すると、セーブセットから選択的にファイルをリストアすることができます。リストアするファイルに対して、現在の作業ディレクトリ以外のリストア先パスを指定することもできます。

次の例では、ファイル `data_file` をセーブセット `/mnt/fdump` からリストアしています。このセーブセットは、`/mnt` ディレクトリに保存されています。

```
# vrestore -f /mnt/fdump -D /mnt -x data_file
vrestore: Date of the vdump save-set: Fri Apr 26 15:27:36 2002
```

リストア処理に複数のテープが必要な場合には、`vrestore` コマンドによって残りのテープのマウントが促されます。

AdvFS ファイル・システムの最適化とチューニング

AdvFS ファイル・システムは、さまざまな方法で設定し、その性能をチューニングすることができます。一部のチューニング機能は、AdvFS GUI (付録 E を参照) から実行できます。AdvFS のチューニング・パラメータについての詳細は、『システムの構成とチューニング』を参照してください。

この章では、次のトピックについて説明します。

- システム性能の監視方法 (5.1 節)
- フラグ・ファイルの無効化による性能改善 (5.2 節)
- トランザクション・ログ・ファイルの性能を向上させる方法 (5.3 節)
- データ・キャッシュのチューニング (5.4 節)
- システム障害の際のデータの一貫性を改善する方法 (5.5 節)
- データ転送速度を向上させる手段の 1 つとしてのダイレクト I/O (5.6 節)
- システム性能を向上させる変更可能なシステム属性 (5.7 節)
- システム・スループットを向上させる `vfast` ユーティリティ (5.8 節)
- ドメインの断片化解消 (5.9 節)
- ファイル単位での断片化解消 (5.10 節)
- データ配置の均等化 (5.11 節)
- ファイルを別のボリュームに移すことにより性能を向上させる方法 (5.12 節)
- AdvFS ファイルのストライプ化 (5.13 節)
- ファイルセットを移動してシステム・リソースへの負荷を軽減する方法 (5.14 節)
- ドメイン・パニックのレベルを制御する方法 (5.15 節)

ドメインとファイルセットの効果的な割り当て方法についての詳細は、『システムの構成とチューニング』および第 1 章を参照してください。Version 5 のディレクトリについては、2.3.3.1 項を参照してください。Version 5 ではディレクトリのインデックスが利用でき、ファイルにアクセスしやすくなっています。

5.1 性能の監視

システムを最適化する第一歩は、その動作を理解することです。性能情報は、多くの方法で得ることができます。

- `iostat` ユーティリティは、端末、ディスク、および CPU の I/O 統計情報を報告します。1 秒あたりの送信回数 (tps) と 1 秒あたりの送信 K バイト数 (bps) を表示します。

`iostat` の出力は I/O のボトルネックの所在を特定する材料になります。すなわち、スループットが高い状態を持続しているデバイスが存在する場合、そのデバイスは他のデバイスよりも使用頻度が高く、ボトルネックになっています。ボトルネックが見つければ、ファイルの移動や、より高速なボリュームの導入、ファイルのストライプ化など、スループットを向上させるための処置を講じることができます。I/O 統計情報は、SysMan Menu の「モニタリング/チューニング - 入出力 (I/O) 統計の参照」ユーティリティ (付録 A 参照) でも表示できます。コマンド行での表示方法については、`iostat(1)` を参照してください。

- `advfsstat` ユーティリティは、長期にわたるファイルセットとドメインの使用状況に関する詳しい情報を表示します。

また、バッファ・キャッシュの使用状況、ボリュームの読み書き、BMT (Bitfile Metadata Table) レコード、その他のパラメータを調べ、通常でないアクティビティがないか調べることもできます。詳細については、`advfsstat(8)` を参照してください。

- `Collect for Tru64 UNIX` は、メモリ、ディスク、テープ、ネットワーク、ファイル・システムといったサブシステムの情報を収集して表示します。

`Collect` は、サポートされている Tru64 UNIX の全リリースで動作します。詳細については、次の web サイトを参照してください。

<http://www.tru64unix.compaq.com/collect/>

5.2 フラグ・ファイルの無効化による性能改善

割り当てられたストレージ領域の 5% 以上が無駄になるファイルの、ディスク・スペースの割り当てを制御することができます。ファイルまたはファイルの終端部分がファイルセットのフラグ・ファイルに保存されます (1.3.3 項を参照)。フラグ処理、つまりフラグ・ファイルを作成する目的はファイルセットのストレージの無駄を最小化することです。ファイルセットのフラグ処理を無効にすると、入出力効率は上がりますが、ストレージの消費量が増加します。フラグ・ファイルを持つかどうかは、デフラグ操作 (5.9 節) には影響がありません。

永続的なアトミック書き込みデータ・ロギングを行なうには、ファイルのフラグ処理を無効にする必要があります。詳細は、5.5.2 項を参照してください。

省略時の設定では、ファイルセットにはフラグ処理が適用され、ファイルの断片がフラグ・ファイルに保存されます。フラグ処理は次の 2 通りの方法で無効化できます。

- ファイルセットの作成時に次の形式のコマンドを使用する。

```
mkfset -o nofrag domain_name fileset_name
```

- 既存のファイルセットに対して、次のコマンドを使用する。

```
chfsets -o nofrag domain_name fileset_name
```

showfsets コマンドを使用すると、ファイルセットのフラグ処理の状態を確認できます。次に例を示します。

```
# showfsets domain_1 fileset_3

Id           : 3a3a47cb.000b52a5.2.8006
Files        :      15,  SLim=      0,  HLim=      0
Blocks (512) :   13730,  SLim=      0,  HLim=      0
Quota Status : user=off group=off
Object Safety: off
Fragging     : on
DMPAPI       : off
```

ファイルセットのフラグ処理を有効化または無効化しても、既存のファイルには影響しません。既存のフラグメント(ファイル断片)は存在し続けます。既存ファイルの終端のフラグメントを削除するには、次の手順に従います。

1. **chfsets** コマンドを使用して、ファイルセットのフラグ状態を、そのファイルに設定したいフラグ状態と同じになるように変更します。

2. フラグメント付きファイルを、別のファイルとしてコピーします。
3. 元のファイルを削除し、コピーした新しいファイルに元のファイルの名前を付けます。
4. ファイルセットのフラグ状態を元に戻したい場合は、`chfsets` コマンドを使用して戻します。こうしておけば、そのファイルセットのファイルにデータを追加したときに、再びフラグメントが作成されるようになります。

たとえば、`states` ドメインの `arizona` ファイルセットのファイル `taxes` からフラグメントを削除するには、次のコマンドを実行します。

```
# chfsets -o nofrag states arizona
# cp taxes tmptaxes
# rm taxes
# mv tmptaxes taxes
```

ファイルセット内のファイルが再びフラグ・ファイルを使用できるようにするには、`chfsets` コマンドに `-o frag` オプションを指定して実行します。フラグ・ファイルを使用していたファイルは、再びこれを使用できるようになります。

詳細は、Best Practice ドキュメントの『*Improving AdvFS Performance by Disabling the Frag File*』を参照してください。

5.3 トランザクション・ログ・ファイルの性能改善

各ドメインには、ドメイン内のすべてのファイルセットのアクティビティを追跡するトランザクション・ログ・ファイル (1.3.2 項) が含まれています。これにより、このログ・ファイルへの大量の読み取り/書き込み動作が発生します。処理が集中するディスクあるいはバス上にこのログ・ファイルが存在する場合、あるいはドメインに多数のファイルセットが存在する場合、システムの性能は低下します。ログ・ファイルを置く場所を変えることで、そのアクティビティによってファイルを置いているデバイスの帯域幅が占有される事態を回避できます。

ドメイン内のボリュームの性能は、SysMan Menu の「モニタリング/チューニング - 入出力 (I/O) 統計の参照」ユーティリティ (付録 A を参照)、または `iostat` ユーティリティで監視できます。AdvFS Utilities が利用できる場合、ログ・ファイルが置かれているボリュームの負荷が高いようであれば、次のいずれかの方法で対処します。

- ドメインをいくつかの小さなドメインに分割します。各ドメインにそれぞれのトランザクション・ログ・ファイルが用意されるため、各ログ・ファイルが扱うファイルセットの数が少なくなります。
- トランザクション・ログ・ファイルを、より速く、あまり処理の集中していないボリュームへ移動します。
- トランザクション・ログ・ファイルを、専用のボリュームに独立させます。

5.3.1 トランザクション・ログ・ファイルの移動

トランザクション・ログ・ファイルをより速い、処理の集中しないボリュームに移動すると、スループットが向上します。LSM (Logical Storage manager) ストレージを使用している場合も、このファイルを移動することで効果があがります。ミラー化されているボリュームにトランザクション・ログ・ファイルを置いて信頼性を高めることができます。そのような構成にすれば、たとえばトランザクション・ログ・ファイルが格納されているディスクがクラッシュした場合でも、ミラー側のログを使用できます。

トランザクション・ログ・ファイルを別のボリュームへ移動する手順は以下のとおりです。

1. `showfdmn` コマンドを使用してログ・ファイルの位置を確認します。ボリューム番号の後に文字 `L` があるボリュームにログ・ファイルが存在します。
2. `/sbin/advfs/switchlog` コマンドを使用して他のボリュームへログ・ファイルを移動します。

たとえば、ドメイン `region1` のトランザクション・ログ・ファイルを移動する手順は、以下のとおりです。

```
# showfdmn region1
      Id              Date Created      LogPgs Version Domain Name
31bf51ba.0001be10 Wed Feb  6 16:24 2002  512          4 region1

Vol  512-Blks      Free % Used Cmode Rblks Wblks Vol Name
  1L   1787904   885168   52%   on   128   128 /dev/disk/dsk0g
    2    1790096 1403872   22%   on   128   128 /dev/disk/dsk0h
-----
      3578000 2259040   37%
```

```
# /sbin/advfs/switchlog region1 2
# showfdmn region1
```

Id	Date Created	LogPgs	Version	Domain Name
31bf51ba.0001be10	Wed Feb 6 16:24 2002	512		4 region1

Vol	512-Blks	Free %	Used	Cmode	Rblks	Wblks	Vol Name
1	1787904	885168	52%	on	128	128	/dev/disk/dsk0g
2L	1790096	1395680	22%	on	128	128	/dev/disk/dsk0h

	3578000	2250848	37%				

5.3.2 トランザクション・ログ・ファイルの分離

トランザクション・ログ・ファイルを独立させると、すべてのログ I/O を他のドメインの読み取りおよび書き込みと分離させることができます。このログ・ボリューム上では他の動作が発生しないため、ログ・ファイルの I/O が遅くなることも、他のドメインの I/O が遅くなることもありません。

トランザクション・ログ・ファイルを専用のボリュームに独立させるには、次の手順を実行します。

1. ドメインがアクティブになっていないことを確認します。アクティブの場合、ファイルへの書き込みはすべて、分離させようとするボリューム上のストレージに割り当てられます。

2. ログ・ファイルを独立させるドメインに、小さなパーティション (ボリューム) を追加します。この場所にログ・ファイルを置きます。

このデバイス上の他のパーティションの I/O 負荷は、ログ・ファイル・パーティションも含めディスク全体の性能に影響することに注意してください。残りのパーティションが他のドメインに割り当てられている場合は、同じデバイス上に複数のトランザクション・ログ・ファイルが存在することになります。この状態は半導体ディスクでは問題はありませんが、低速のデバイスではログ・ファイルを独立させた効果がなくなることがあります。

3. `switchlog` コマンドを使用して、ログ・ファイルを新しいボリュームに移動します。
4. `showfdmn` コマンドを使用して、ログ・ファイルのあるボリューム上の空きブロック数を調べます。
5. `showfdmn` コマンドから得られた情報を参考に、`dd` コマンドを使用して正しいサイズのダミー・ファイルを作成します。

6. ダミー・ファイルを、ログ・ファイルのあるボリュームに移動します。このファイルはボリュームを完全に埋めるため、他のファイル用のスペースは残りません。ユーザはこのファイルにはアクセスしないため、トランザクション・ログ・ファイルだけがこのボリューム上でアクティブになります。

sales ドメインのトランザクション・ログ・ファイルを独立させるには、次のようにコマンドを入力します。

```
# addvol /dev/disk/dsk9a sales
# /sbin/advfs/switchlog sales 2

# showfdmn sales
      Id                Date Created      LogPgs Version Domain Name
312387a9.000b049f Thu Mar 14 14:24 2002   512      4 sales

Vol  512-Blks      Free % Used Cmode Rblks Wblks Vol Name
  1    2050860 1908016    7%   on   128   128 /dev/disk/dsk10c
 2L    131072 122752    6%   on   128   128 /dev/disk/dsk9a
-----
      2181932 2030768    7%
```

ログ・ファイルが置かれているボリューム上のすべての空きブロックをダミー・ファイル /adv1/foo に割り当ててから、データをログ・ファイルと同じボリュームに移動します

```
# dd if=/dev/zero of=/adv1/foo count=122752
122752+0 records in
122752+0 records out
# migrate -d 2 /adv1/foo
```

5.4 データ・キャッシュのチューニング

データが頻繁に再利用される環境では、キャッシュを使用すると性能が向上します。AdvFS は、ファイル・メタデータとユーザ・データの管理に、ユニファイド・バッファ・キャッシュ (UBC) と呼ばれるダイナミック・メモリ・キャッシュを使用します。

キャッシュに UBC を使用することで、AdvFS ではメモリが利用可能な限り、ファイル・データをメモリ内に保持することができます。他のシステム・リソースのために、ファイル・システムのキャッシュとして使われているメモリが必要になると、UCB は、ファイル・システムで使用しているメモリの一部を取り返し、メモリを必要としているリソースに割り当てます。

AdvFS はキャッシュに UBC を使用するため、キャッシュのチューニングには UBC のチューニング・パラメータを使用します。チューニングできるパラメータは次のとおりです。

- UBC として一度に使用できるメモリ量の、物理メモリ全体に占める割合。
- UBC からディスクへの書き込みが開始される前に超過すべき、ダーティ・ページ数の全ページ数に対する割合
- 1 つのファイルのキャッシュとして使用できる UBC メモリの最大サイズ

これらのパラメータを変更する際の指針については、『システムの構成とチューニング』を参照してください。

データ・キャッシュは省略時の設定で有効になっており、ファイル・システムの性能が向上しますが、特定の状況下では、データ・キャッシュをバイパスする方が、アプリケーション側でより高いスループットが得られることもあります (5.6 節を参照)。

5.5 データの一貫性の向上

データをファイルへ書き込む方法によって、マシンに障害が発生した場合にデータがどの程度保護されるかが決まります。次の 2 つのことについて選択できます。これらの選択肢は互いに独立しています。

- I/O を同期させるかどうか
- メタデータの書き込みに加えて、ファイルのデータもトランザクション・ログ・ファイルに書き込むかどうか (アトミック書き込みデータ・ロギングのオン/オフ)

5.5.1 I/O の制御

特に指定しなければ、書き込み要求はキャッシュされます。すなわち、データはすぐにディスクに書き込まれるのではなくバッファ・キャッシュに書き込まれます。ファイルへの I/O 書き込みを同期する方法を選択することで、キャッシュによる性能向上とクラッシュ時のデータ一貫性とのバランスをとることができます。

5.5.1.1 非同期 I/O

非同期 **I/O** は、省略時設定であり、一般に最高のスループットが得られます。同じページに対する複数の書き込みをディスクに対する 1 回の物理的書き込みに結合します。この方法では、ディスクのトラフィックが減るだけでなく、複数のスレッドおよびプロセスで共通のデータへ同時にアクセスする機会が増えます。さらに、ディスクへの書き込みを遅らせるので、連続するページ同士を連結して単一の大きな物理的書き込みにできる可能性が高くなり、シーク時間と回転待ち時間による遅れが短くなります。

クラッシュが発生すると、ドメイン内のファイルセットが次にマウントされたときに、ディスクへのログ・トランザクションのうち、完了しているものが再度実行され、完了していないものは取り消されるため、ディスク上の元のメタデータが復元されます。これらのログ・トランザクションは、省略時はメタデータのみを保存し、ファイルへ書き込まれたデータは保存しません。すなわち、ディスク上のファイル・サイズと位置は一貫性が保たれますが、クラッシュが起きた時点が、ディスクへの書き込みの前であれば、最新の書き込みのユーザ・データが失われる可能性があります。これは、この方法を使用することで得られるスループットの向上とトレードオフの関係にあります。

5.5.1.2 同期 I/O

同期 **I/O** は非同期 I/O と類似していますが、書き込み要求の処理から呼び出し元のアプリケーションに戻る前に、データがキャッシュとディスクの両方に書き込まれます。すなわち、書き込みが正常に終了すると、データがディスクに正しく書き込まれたことが保証されます。

同期 I/O の欠点は、I/O が完了するまで書き込み処理から戻らないため、スループットが低くなることです。また、データをディスクにフラッシュしなければならないタイミングを決定するのがファイル・システムではなくアプリケーションなので、同期書き込み要求の処理量が小さい場合、I/O が結合される可能性が低くなります。

5.5.1.3 同期 I/O のオン/オフのチューニング

省略時のキャッシング方式は非同期 I/O です。同期 I/O のオン/オフをファイルごとに切り替えるには、`chfile` コマンドに `-l` オプションを使用するか、`open()` システム・コールに対して `O_SYNC` フラグまたは `O_DSYNC` フラグを使用します (『プログラミング・ガイド』および `open(2)` を参照)。 `chfile`

コマンドの `-l` オプションと `-L` オプションは同時に使用できないので、`chfile` コマンドを使用してアトミック書き込みデータ・ロギングを制御することを計画しているならば、同期 I/O をアクティブにするためには、システム・コールを使用する必要があります。アトミック書き込みデータ・ロギングのオン/オフのチューニングについては、5.5.2.5 項を参照してください。

ファイルセット内のすべてのファイルに同期 I/O を強制するには、`mount` コマンドに `-o sync` オプションを指定します。

```
mount -o sync filename
```

また、あるファイルにアクセスするすべてのアプリケーションに同期 I/O を強制することができます。この場合、`chfile` コマンドに `-l on` オプションを指定します。

```
chfile -l on filename
```

同期処理をオフにしてファイルを再び非同期 I/O 状態にするには、`chfile` コマンドに `-l off` オプションを指定して実行します。

```
chfile -l off filename
```

5.5.2 アトミック書き込みデータ・ロギング I/O の有効化

アトミック書き込みデータ・ロギングでは (通常ロギングされるメタデータに加えて) ユーザ・データをログ・ファイルに書き込み、システム障害のときにデータの一貫性が保たれるようにします。この機能のオン/オフでは、メタデータとファイル・データの両方がディスクに書き込まれるか、両方が書き込まれないかを切り替えることになります。アトミック書き込みデータ・ロギングが使用する I/O 方式は、I/O が非同期に設定されているか (5.5.1.1 項)、同期に設定されているか (5.5.1.2 項) で異なります。

アトミック書き込みデータ・ロギングには、永続的なロギング (5.5.2.3 項) と一時的なロギング (5.5.2.4 項) の 2 種類があります。永続的なデータ・ロギングでは、ファイルセットがマウント/アンマウントされても、ログは継続されます。一時的なデータ・ロギングは、ファイルセットがマウントされているときだけ有効です。ロギングのステータスは、オプションをつけない `chfile` コマンドで確認することができます。

この機能のデータ・ロギングでは、データとメタデータの両方をトランザクション・ログ・ファイルに書き込むので性能が低下します。ロギングのトラ

フィックが増加し、各ユーザの書き込みでの I/O の処理が倍になります。アトミック書き込みデータ・ロギングを計画しているファイルまたはファイルセットを、1 つまたはいくつかのドメインにグループ化することによって、性能重視の他のドメインの負荷を軽くすることができます。

5.5.2.1 非同期アトミック書き込みデータ・ロギング I/O

非同期アトミック書き込みデータ・ロギング I/O は、非同期 I/O と同じような方法ですが、それぞれの書き込み要求でバッファ・キャッシュに書き込まれたユーザ・データがログ・ファイルにも書き込まれるという点が異なります。これは、8K バイトの倍数のサイズで行われます。データをログ・ファイルにも書き込むため、クラッシュのときのデータの一貫性は保証されますが、非同期 I/O と比べるとスループットが低くなります。非同期 I/O を使用し、アトミック書き込みデータ・ロギングを有効にすると、I/O は非同期のアトミック書き込みデータ・ロギングとなります。

クラッシュが発生すると、ファイルセットが再度マウントされたときにログ・ファイルからデータが復元されます。非同期 I/O の場合と同じように、完了したログ・トランザクションはすべて再実行され、完了していないトランザクションは取り消されます。ただし、非同期 I/O の場合と異なり、ユーザのデータはログに書き込まれているので、メタデータとファイルに書き込もうとしていたデータの両方を復元できます。これにより、8K バイトの倍数のサイズの書き込みがアトミックであることが保証されます。つまり、データは完全にディスクに書き込まれたか書き込まれていないかのいずれかです。

完了した書き込み要求のみが処理されるため、クラッシュ時にシステムが書き込もうとしていた位置にある、古くなった、傷ついている可能性のあるデータはアクセスされません。ディスクへの書き込み順序が乱れて、クラッシュ時に一貫性が損なわれることはありません。

5.5.2.2 同期アトミック書き込みデータ・ロギング I/O

同期アトミック書き込みデータ・ロギング I/O は、非同期アトミック書き込みデータ・ロギング I/O と同じような方法ですが、書き込み要求が呼び出し側アプリケーションに戻る前に、ロギングされたデータがバッファ・キャッシュからディスクにフラッシュされるという点が異なります。ログ・フラッシュ I/O が終了してからでないと、書き込み処理が戻ってこないため、非同期アトミック書き込みデータ・ロギング I/O と比べるとスループットが低くなる場合

があります。同期 I/O を使用し、アトミック書き込みデータ・ロギングを有効にすると、I/O は同期アトミック書き込みデータ・ロギングとなります。

書き込み進行中にクラッシュが発生すると、ファイルセットが再度マウントされたときにログ・ファイルからデータが復元されます。非同期アトミック書き込みデータ・ロギング I/O の場合と同じように、ユーザのデータはログに書き込まれているのでメタデータとファイルに書き込もうとしていたデータの両方を復元できます。これにより、8K バイト倍数のサイズでデータが完全にディスクに書き込まれたか書き込まれていないかのいずれかであることが保証されます。

同期アトミック書き込みデータ・ロギングの利点は、書き込み呼び出しがアプリケーションに戻った後にクラッシュが発生した場合のデータの一貫性が保証されることです。リブートでログ・ファイルが再生され、ユーザの書き込み要求全体が適切なユーザのデータ・ファイルに書き込まれます。一方、非同期アトミック書き込みデータ・ロギングは、書き込み呼び出しがアプリケーションに戻った後の 8K バイト単位のデータの一貫性しか保証しません。

5.5.2.3 永続的なアトミック書き込みデータ・ロギング

永続的なアトミック書き込みデータ・ロギングでは、ディスク上にフラグが設定され、ファイルセットがマウント/アンマウントされても、そのファイルのロギングが継続されるようにします。ロギングを非同期に行なうか (5.5.1.1 項)、同期をとって行なうか (5.5.1.2 項) は、I/O をどのように設定したかによって決まります。

永続的なアトミック書き込みデータ・ロギング I/O のオン/オフを切り替えるには、`fcntl()` 関数を使用するか、`chfile` コマンドに `-L` オプションを指定します。

```
chfile -L on filename
```

```
chfile -L off filename
```

ファイルにフラグメント (断片) がある場合、永続的なアトミック書き込みデータ・ロギングは起動できません。フラグメントのあるファイルに対してデータ・ロギングを起動するには、次のいずれかの方法を探ります。

- 一時的なアトミック書き込みデータ・ロギングを起動する。このデータ・ロギングは、フラグメントのあるファイルにも使用できます。5.5.2.4 項を参照してください。

- `mkfset` コマンドや `chfsets` コマンドでファイルセットを作成または変更するときに、`nofrag` オプションを指定する。

フラグ・ファイルを `chfsets` コマンドに `-o nofrag` オプションを指定して無効にしても、既にフラグメントのあるファイルのフラグメントはそのままです。ファイルのフラグメントをなくすには、5.2 節を参照してください。

`-l` と `-L` オプションを `chfile` コマンドで同時に用いて、同期 I/O とアトミック書き込みデータ・ロギングを設定することはできません。しかし、`chfile` コマンドに `-L on` オプションを用いて、永続的アトミック書き込みデータ・ロギングを起動し、次にそのファイルを `open()` システム・コールに `O_SYNC` または `O_DSYNC` フラグをつけて呼び出して、そのファイルを同期 I/O としてオープンすることができます (『プログラミング・ガイド』を参照)。

永続的なアトミック書き込みデータ・ロギングを使用するファイルは、`mmap` システム・コールでメモリにマッピングすることはできません。ファイル使用の衝突については 6.4.10 項を参照してください。

5.5.2.4 一時的なアトミック書き込みデータ・ロギング

一時的なアトミック書き込みデータ・ロギングでは、ファイルセットに対して有効にするフラグをメモリ上に設定し、そのファイルセットがマウントされている間のみデータ・ロギングが行なわれるようにします。ロギングを非同期に行なうか、同期をとって行なうかは、I/O をどのように設定したかによって決まります。I/O モードの制御方法は、5.5.1 項を参照してください。

一時的なアトミック書き込みデータ・ロギングを有効にするフラグをメモリ上に設定し、対応するファイルセットがマウントされている間のみデータ・ロギングを有効にするには、`mount` コマンドに `-o adl, sync` オプションを使用します。フラグメントを持つファイルは、一時的なアトミック書き込みデータ・ロギングを使用できます。ファイルが開かれている間、永続的なアトミック書き込みデータ・ロギングの方が、一時的なアトミック書き込みデータ・ロギングよりも優先されます。

ファイルを開いているアプリケーションは、`fcntl()` 関数を呼び出すことによって一時的なアトミック書き込みデータ・ロギングを無効にできます。また、`chfile` コマンドに `-L off` オプションを使用することによって、永続的なアトミック書き込みデータ・ロギングを無効にできます。これらの処理

によるデータ・ロギングの無効化は、同じファイルを開いているすべてのアプリケーションに反映されます。

一時的なアトミック書き込みデータ・ロギングを使用しているファイルは、メモリにマッピングできます。ファイルをメモリにマッピングすると、そのファイルを使用している全スレッドでマッピングが解除されるまで、一時的なアトミック書き込みデータ・ロギングは中断されます。

5.5.2.5 アトミック書き込みデータ・ロギングのオン/オフのチューニング

ロギングを非同期に行なうか (5.5.1.1 項)、同期をとって行なうか (5.5.1.2 項) は、I/O をどのように設定したかによって決まります。

表 5-1 にアトミック書き込みデータ・ロギングの有効化をチューニングする際のパラメータを示します。

表 5-1: アトミック書き込みデータ・ロギングの有効化のチューニング

データ・ロギングのタイプ	コマンド	備考
永続的 (ファイル)	chfile コマンドに <code>-L on</code> オプションを指定、または <code>fcntl()</code> 関数を使用。	フラグメントを持つファイルについては実行できない。chfile コマンドを同期 I/O を制御するために使用する場合は、このコマンドを使用できない。 5.5.2.3 項を参照。
一時的 (ファイルセット)	非同期の場合は <code>mount</code> コマンドに <code>-o adl</code> オプションを指定、同期の場合は <code>-o adl, sync</code> オプションを指定。	フラグメントを持つファイルについて実行できる。永続的アトミック書き込みデータ・ロギングが優先される。 5.5.2.4 項を参照。

表 5-2 にアトミック書き込みデータ・ロギングの無効化をチューニングする際のパラメータを示します。

表 5-2: アトミック書き込みデータ・ロギングの無効化のチューニング

データ・ロギングのタイプ	コマンド	備考
永続的 (ファイル)	chfile コマンドに <code>-L off</code> オプションを指定, または <code>fcntl()</code> 関数を使用。	chfile コマンドを同期 I/O を制御するために使用する場合は, このコマンドは使用できない。 5.5.2.3 項を参照。
一時的 (ファイルセット)	ファイルセットがアンマウントされると, 自動的にオフになる。または <code>fcntl()</code> 関数を使用。	オープンしたファイルはすべて <code>fcntl()</code> 関数を呼び出せる。 5.5.2.4 項を参照。

5.6 ダイレクト I/O によるデータ転送速度の改善

ダイレクト I/O モードでは, キャッシュをバイパスします。データをキャッシュにコピー (AdvFS 標準の処理) することなく, ファイルに対するデータの読み取りおよび書き込みを同期的に実行します。つまり, ファイルに対するダイレクト I/O を有効にすると, 読み取りおよび書き込み要求では AdvFS キャッシュ機能がバイパスされ, 直接メモリ・アクセスによって (raw I/O と同じように), ディスク・ストレージに対して実行されます。これにより, データに 1 度しかアクセスしないようなアプリケーションの場合, I/O 処理速度を改善できます。しかし, システム・クラッシュの際はデータの一貫性は保証されません。

ダイレクト I/O は, どのようなバイト・サイズの I/O 要求でも処理しますが, 要求された転送位置がディスク・セクタ境界に合っていて, その転送サイズが基本的なセクタのサイズ (現在 512 バイト) の偶数倍である場合に, 最も高い性能を発揮します。

ダイレクト I/O は, データベースによって独占的に使用されるファイルに特に適しています。ただしキャッシングが行われなため, アプリケーションが同じデータに複数回アクセスする傾向がある場合は, 逆に, ダイレクト I/O は性能に悪影響を及ぼします。ダイレクト I/O を指定すると, すぐにそれが有効になり, すでにバッファ・キャッシュに存在するデータは自動的にディスクへフラッシュされます。

ファイルをダイレクト I/O を有効にしてオープンできるのは, アトミック書き込みデータ・ロギング用にオープン (5.5.2 項) していない場合, またはメモリにマッピング (6.4.10 項) されていない場合のみです。ダイレクト I/O を行なうようにファイルをオープンするには, `O_DIRECTIO` フラ

グを指定して `open()` 関数を呼び出します。たとえば、`file_x` に対しては、次のように呼び出します。

```
open (file_x, O_DIRECT|O_RDWR, 0644)
```

それまでのモードに関係なく、新しいモードはダイレクト I/O になり、指定したファイルがクローズされるまでこのモードが有効になります。

`fcntl()` 関数を呼び出して、ファイルがキャッシュ・モードでオープンされているか、ダイレクト I/O モードでオープンされているかを確認することができます。詳細は、`fcntl(2)` と `open(2)`、または『プログラミング・ガイド』を参照してください。

5.7 属性の変更によるシステム性能の改善

システムの性能は、さまざまな属性を変更して改善できます。『システムの構成とチューニング』に、それぞれの属性の意味と、それらを変更した場合の影響について詳しく説明しています。詳細については `sysconfig(8)` を参照してください。AdvFS の性能を改善するには、次の操作が役に立ちます。

- ダーティ・データ・キャッシングしきい値の引き上げ

ダーティ・データとは、アプリケーションによってキャッシュに書き込まれたが、まだディスクには書き込まれていないデータのことです。AdvFS でキャッシュされるダーティ・データの量は、`chvol` コマンドに `-t` オプションを使用することによってドメインのボリュームごとに変更できるほか、`AdvfsReadyQLim` 属性によって AdvFS のすべての新規ボリュームを対象に変更できます (`chvol(8)` を参照)。

`smooth sync` が無効な場合には、`chvol` コマンドによるこの変数の変更が最も効果的です。`smooth sync` が有効な場合 (省略時の動作) には、ダーティ・データをディスクにフラッシュする頻度は `smoothsync_age` 属性で調整する方が適切です。

- `smoothsync_age` 属性による連続 I/O の促進

`smoothsync_age` 属性は、バッファ・キャッシュ内の変更ページが、ディスクに書き込まれるまでの時間の長さを秒単位で指定します。この属性を使用すれば、ページへの連続した変更が終わるまでページをメモリ上に保持して I/O 回数が減るように、ページをディスクに書き込むタイミングを調整できます。

- I/O 転送サイズの変更

デバイスに最適な転送サイズで I/O を行なうために、AdvFS は、ディスク上の物理的に連続するメモリのキャッシュ・ページを結合します。最適な転送サイズはデバイス・ドライバによって決まり、ストレージの構成によって異なりますが、通常は 128 または 256 ブロックです。LSM を使用している場合は、ストライプ幅を変更することができ、さらに大きな転送サイズが設定される場合もあります。優先転送サイズの調整には、`chvol` コマンドと `-r` オプション (read) または `-w` (write) オプションを使用します (`chvol(8)` を参照)

- メモリにマッピングされた変更ページのフラッシュ

`AdvfsSyncMmapPages` 属性は、メモリにマッピングされた変更ページを、`sync` システム・コールでディスクにフラッシュするかどうかを指定します。

- アクセス構造体用に確保するメモリの拡大

AdvFS は、ページング可能メモリの一定の割合までアクセス構造体にメモリを割り当てます。この割合は、`AdvfsAccessMaxPercent` 属性によって制御されます。この値を増やせば、数多くのファイルをオープンして再使用する環境では、AdvFS の性能向上につながる可能性があります。仮想メモリ・サブシステムおよびユニファイド・バッファ・キャッシュ (UBC) で使用できるメモリの量は減りますが、この属性の値を減らすとページング可能なメモリは増えますが、多数のファイルをオープンして再使用する環境では、AdvFS の性能が低下する可能性があります。

5.8 vfast ユーティリティによるオペレーティング・システムのスループットの改善

`vfast` ユーティリティは、頻繁にオープン/クローズされるドメインのファイルに作用するバックグラウンド・プロセスです。このユーティリティは、次のようなさまざまな最適化機能を実行します。

- ファイル断片化の解消
- ボリュームの空き領域の分散化
- I/O 負荷の均等化

`vfast` ユーティリティを実行するには、`root` ユーザの特権が必要です。

`vfast` 処理は、`activate` および `deactivate` オプションでオン/オフを切

り替えることができます。suspend オプションは、vfast 処理をオフにしますが、内部統計データ収集は継続することができます。status オプションを使用すると、現在の vfast の構成、動作上の統計、ドメインの処理オプションを表示することができます。

ほとんどの vfast 処理は、デバイスに他のシステム I/O が存在しないときに実行されるので、一般的に vfast を実行しても性能は低下しません。ユーティリティが使用するシステム I/O のシェアを、`-o percent_ios_when_busy=` オプションで制限できます。省略時の設定は、周知のストレージ・デバイスの I/O 帯域幅の 1% です。

umount, rmvol, および rmfset などのユーティリティは、実行時に vfast の動作を一時的に中断させます。これらのユーティリティが終了すると、vfast はその以前の状態に戻ります。

vfast はファイルの断片化を解消し、ドメイン間のバランスを計るため、従来の AdvFS defragment および balance ユーティリティは、vfast を `-o defragment=`, `-o balance=`, または `-o topIObalance=` オプションを有効にして起動した場合使用できません。

vfast コマンドに `-o defragment=enable` オプションを指定すると、空き領域を動的に統合化し、ファイル断片化を減らし、ファイルが連続する空間を使用するようにします。AdvFS ファイル・システムでは、ファイルのデータをなるべくディスク上の連続するブロックに配置しようとはしますが、データを配置するのに十分な連続するブロックを利用できない場合は、それを連続しないブロックに分散します。断片化は、1 つのファイルにアクセスするのに多くのディスク・アドレスを参照しなければならないため、読み書きの性能を低下させます。vfast ユーティリティは、断片化されたファイルを連続するディスク・ブロックに移動することで、この性能の低下を最小限にします。ファイルは、統合化の際に再配置されることがあります。

オープン/クローズされたファイルだけが、vfast の断片化ファイルのキャッシュに置かれます。vfast でファイルを断片化解消するには、そのドメインの少なくとも 1 つのファイルセットを読み/書き用にマウントしておく必要があります。ドメインのその他のファイルセットが書き込みのためにマウントされていない場合、そのファイルセットが以前に書き込みのためにマウントされ、vfast キャッシュの中で処理を待っているファイルがない限りは、そのファイルセット中の断片化ファイルは断片化解消されません。vfast コマンドを `-l extents` オプションを指定して実行し、断片化解消を待つキュー

にあるファイルを表示します。vfast キャッシュにファイルがない場合は、断片化解消または空き領域のバランシングは行なわれません。

ドメイン全体を断片化解消するには、そのドメインのすべてのファイルをオープンしクローズする必要があります。ドメインの各ファイルセットで、次の操作を行ないます。

1. ファイルセットをマウントし、そのディレクトリに移動します。
2. 次のコマンドを実行します。

```
# find ./ -name \* >/dev/null
```

3. vfast キャッシュが一杯になる可能性があるので、一回ですべてのファイルの断片化が解消できるとは限りません。ドメインの断片化を完全に解消するには、さらに何度かファイルをオープンしクローズする必要があります。次のコマンドを実行します。

```
# vfast -L extents domain_name
```

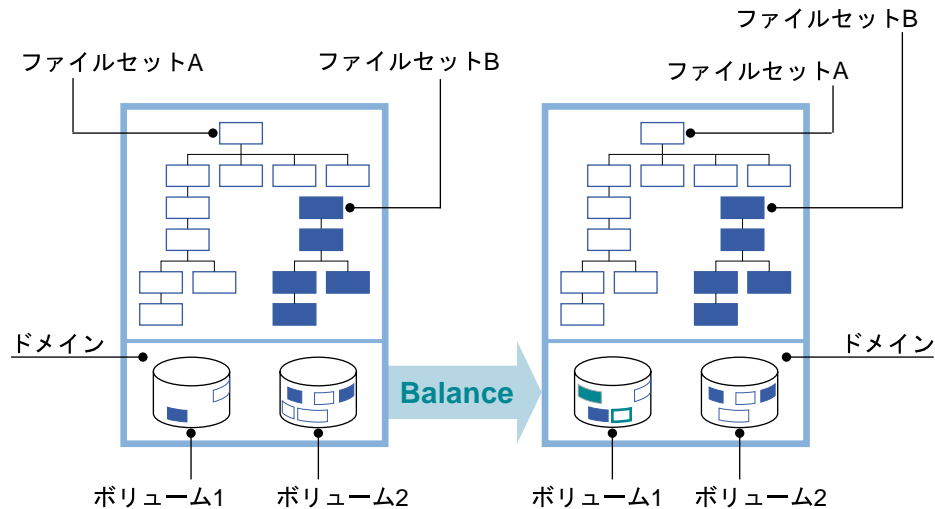
vfast でマルチボリューム・ドメインの断片化を解消する場合、ファイルの配置を制御できません。ファイルがどこに置かれているかを確認するには、showfile コマンドを実行します。特定のファイルを別のボリュームに移動したい場合は、migrate コマンドを実行します (5.12 節を参照)。

-o defragment=disable オプションを指定することで、断片化の解消は、いつでも停止できます。プロセスを途中で停止しても、ファイル・システムに影響はありません。すでに断片化解消されたファイルはその再配置された位置のままになります。

ファイルのダイレクト I/O を有効にすると (5.6 節)、vfast ユーティリティは、-o direct_io= オプションが disable に設定されるまで、そのファイルの断片化解消を行ないます。

-o defragment= オプションを有効に設定し、-o balance= も有効にした場合、vfast ユーティリティは、マルチボリューム間でファイルを分散し、I/O 負荷を均等化します。このユーティリティは、図 5-1 に示すように、ドメインの各ボリュームのスペースの使用率が可能な限り均等になるまで、ファイルをボリューム間で移動します。断片化解消のキューに入れられたファイルのみが、ドメイン間のバランスを図るのに使用されます。

図 5-1: ドメイン間のバランシング



断片化解消が必要なファイルだけが、ボリューム間で空き領域を均等化し負荷分散を図るのに使用されます。ドメイン内に断片化されたファイルがない場合は、`vfast` を実行してもバランシングは行なわれません。また、断片化のチェックはクローズ中のファイルにのみに行なわれます。そのファイルセットの最後のマウント以降アクセスしていないファイルはチェックの対象になりません。

`-o topIObalance=` オプションを有効に設定すると、`vfast` はまず I/O を監視し(`-o ss_steady_state=` オプションで設定した期間だけ)、どのボリュームの I/O 負荷が最も高いかを調べます。このユーティリティは次に、負荷のバランスを図るため高い負荷のボリュームからファイルを分散配置していきます。しかし `topIObalance` オプションでは、ダイレクト I/O が有効になっているファイルは無視されます。これは、これらのファイルへの I/O が `vfast` 統計収集の対象とならないためです。

ドメインがハードウェア RAID、LSM ストライプ化、AdvFS ファイル・ストライプ化のいずれかの方法でストライプ化されている場合、`-o topIObalance=` オプションは使用しないでください。このユーティリティは、これらの構成では効率的に I/O 負荷の分散ができません。

システム・ノードがクラスタのメンバである場合、クラスタ・ファイル・システム (CFS) がファイルをキャッシュし、キャッシュがフラッシュされ

るまで処理が発生しないため、最初にファイルがいくつか処理されない場合があります。

5.8.1 vfast ユーティリティの実行

ドメインの vfast 処理を起動するには、vfast コマンドに activate オプションを指定して実行します。処理を起動した後、さらに vfast コマンドを実行することでこのユーティリティの動作を切り替えることができます。

vfast を有効にせずにドメインの断片化の量を確認するには、-L extents オプションを使用します。エクステント数の平均値や、断片化しているファイルあたりのエクステント数が高い場合、このユーティリティの実行が役立ちます。ドメインが過度に断片化されていると思う場合は、vfast に -o defragment= オプションを指定して実行します。

ドメインのレイアウトは、showfdmn コマンドで知ることができます。このコマンドの % Used フィールドを見れば、ファイルがボリューム間を均等に分散されているかがわかります。均等に分散されていない場合は、そのドメインに対して -o balance= オプションを実行します。

vfast コマンドに -l hotfiles オプションを使用すると、ボリュームまたはドメインで最も頻繁にページングされるファイルを特定することができます。-L hotfiles オプションを指定すると、ドメインごとのボリューム内のファイルの分散状況が表示されます。

5.8.2 vfast ユーティリティの例

この節では、vfast ユーティリティの各種オプションによって、そのさまざまな機能を制御する例を示します。

下記の例では、vfast の各種機能を起動しています。

```
# vfast activate user_dmn
# vfast -o defragment=enable user_dmn
# vfast -o balance=enable user_dmn

# vfast -o topIObalance=enable user_dmn
# vfast -o percent_ios_when_busy=20 user_dmn
```

次の例では status オプションを使用してドメイン user_dmn に行なわれた断片化解消の量を調べています。

```
# vfast status user_dmn
vfast is currently running
vfast is activated on user_dmn
```

```

vfast defragment:      enabled
vfast balance:         enabled
vfast top IO balance:  enabled
Options:
  Direct IO File Processing: enabled
  Percent IOs Allocated to vfast When System Busy: 20%
  Default Hours Until Steady State: 24; Hours remaining: 0
  Total Files Defragmented: 3331
  Total Pages Moved for Defragment: 278440
  Total Extents Combined for Defragment: 21
  Total Pages Moved for Balance: 0
  Total Files Moved for Volume IO Balance: 0
  Total Pages Moved for Volume Free Space Consolidation: 50607

```

次の例では `-l extents` オプションを使用して、断片化解消と空き領域の均等化処理を待つキューに置かれた、ドメイン `user_dmn` のボリュームを表示しています。

```

# vfast -l extents user_dmn
user_dmn: Volume 1
extent
count    fileset/file
 2 user:  /u1/obj/BINARY/lp.o
 2 user:  /u1/w17/obj/kernel/test_21.o
 3 user:  /u1/obj/bs_bitfile_sets.o
 3 user:  /u1/w4/itpsa.o
 4 user:  /u1/w4/cms_utils.o

```

次の例では `-L extents` オプションを使用して、ドメイン `user_dmn` のボリュームごとの断片化の様子の要約を表示しています。

```

# vfast -L extents user_dmn
user_dmn
  Extents:                46003
  Files w/extents:        45694
  Avg exts per file w/exts: 1.01
  Free space fragments:   18858
                        <100K  <1M   <10M   >10M
  Free space:             21%    41%    31%    7%
  Fragments:              16119  2523  213    3

```

次の例では `-l hotfiles` オプションを使用して、ドメイン `user_dmn` 上にあるファイルとボリュームで、最も頻繁にページングが行なわれたものを検索しています。

```

# vfast -l hotfiles user_dmn |more
Past Week

IO Count Volume File
5487993    1   *** a reserved file, tag = -2,-10, BMT

```

```

197088 1 *** a reserved file, tag = -2, -7, SBM
147757 1 *** a reserved file, tag = -2, -9, LOG
2814 1 user: /user1/crl/BINARY/makedep
1206 1 user: /user1/crl/applications/sequoia.jar
1005 1 user: /user1/sandboxes/advfs.mod
402 1 user: /user1/alpha/arch.mod

```

5.9 ドメインの断片化の解消

vfast ユーティリティ (5.8 節) を実行していない場合、defragment ユーティリティを実行してドメインのファイル断片化の量を軽減することができます (1.3.3 項)。このユーティリティは、ファイルの連続性を高め、ファイル・エクステンツ数を減らすことにより、ドメインの断片化を減少させます。ファイルが断片化すると、ファイルをアクセスするのに多数のディスク・アドレスをチェックしなければならなくなるため、断片化を解消することで読み書き性能が改善します。さらに、ドメインの断片化解消によって、ディスク上の連続した空き領域も増えるので、それ以降に作成されるファイルも断片化が少なくなるという利点もあります。フラグ・ファイルを有効にしたり無効にしたりすることで、ドメインの断片化を解消することもできます。I/O 動作には影響しません。

vfast ユーティリティは、オペレーティング・システムに対応して最適化され、バックグラウンドで動作するため、ドメインの断片化解消には、このユーティリティを使用することをお勧めします。defragment ユーティリティの動作効率は、これを実行する前にドメイン内の不要なファイルを削除することで向上します。defragment ユーティリティは、性能が低下したと実感した場合で、ファイル・システムへのアクセスがそれほどない場合にだけ使用してください。

断片化の解消はいつでも停止できます。プロセスを中断しても、ファイル・システムに影響はありません。すでに断片化解消されたファイルはその再配置された位置のままになります。

マルチボリューム・ドメインの断片化を解消する場合、ファイルの配置を制御できません。ファイルがどこに置かれているかを確かめるには、showfile コマンドを実行します。特定のファイルを別のボリュームに移動したい場合は、migrate コマンドを実行します。ファイルの移動については、5.12 節を参照してください。

ドメイン全体を断片化解消するには、そのドメインのすべてのファイルセットをマウントしておく必要があります。ボリュームの断片化を解消するに

は、少なくとも総スペースのうちの 1% か、ボリュームにつき 5 MB (どちらか少ない方) のスペースが残っている必要があります。SysMan Menu の「AdvFS ドメインの管理」(付録 A を参照) か、グラフィカル・ユーザ・インタフェース (付録 E を参照) を使用するか、コマンド行で次のように defragment コマンドを実行します。

```
defragment domain_name
```

ドメインの断片化解消を行なうには root ユーザの特権が必要です。

defragment ユーティリティは、vfast、addvol、rmvol、balance、または rmfset コマンドを同じドメインで実行しているときには実行できません。

デフラグメント処理がシステムに及ぼす負荷を特定するのは困難です。ドメインの断片化解消に必要な時間は、次の要素に依存します。

- ボリュームのサイズ
- 利用できる空き領域の量
- システムの稼働状態
- ドメインの構成

defragment ユーティリティは、ボリュームごとにスレッドを生成します (最大 20 スレッド)。したがって、多数の小さなボリュームで構成されたドメインの断片化解消は、1 つの大きなボリュームで構成されたドメインを対象とする場合よりもより速く処理されます。ただし、複数のスレッドで断片化を解消すると、実行中の I/O のスループットが大幅に低下する可能性もあります。Version 4 のオペレーティング・システム・ソフトウェアの動作と同様に、1 つのスレッドだけでデフラグメント処理を実行するには、defragment に -N 1 オプションを指定して実行します。

ユーティリティを起動しないで、ドメインの断片化状況を調べるには、defragment コマンドに -v -n オプションを指定して実行します。エクステント数の平均や、断片化しているファイルあたりのエクステント数が高い場合、あるいは総合的な I/O 性能が低下している場合には、デフラグメント処理が役立つことがあります。多くの場合、大規模な、かなり断片化されているファイルであっても、断片化による性能の著しい低下はありません。ファイルの過剰な断片化による性能上の問題が発生していない場合には、defragment コマンドを実行する必要はありません。

かなり断片化されているファイルを見つけた場合、そのファイルの断片化だけを個別に解消することができます。個々のファイルの断片化解消については、5.10 節を参照してください。

ファイル・システムに対する操作を 1 ~ 2 か月行なっていない場合、つまり、定期的なフル・バックアップやファイル・システム全体の点検を行っていない場合は、defragment を実行する前に、/sbin/advfs/verify コマンド (6.2.4 項) を実行することをお勧めします。verify コマンドは、ファイル・システムの負荷が低いときに実行してください。

defragment を実行する前に balance ユーティリティを実行すると、デフラグメント処理が速くなる場合があります。

メール・サーバなど、大部分のファイルのサイズが AdvFS のページ・サイズである 8K バイト未満のシステムで defragment コマンドを実行するのは、ファイルセットのフラグ・ファイル /mount_point/.tags/1 の著しい断片化が示された場合だけにしてください。フラグ・ファイルの無効化については、5.2 節を参照してください。

ハードウェア・リソースと AdvFS Utilities があれば、addvol コマンドを使用してボリュームを追加し、元のボリュームを rmvol コマンドで削除することができます。古いボリュームを削除するとドメインは新しいボリュームに移動され、同時に内部のファイルの断片化も解消されます。

次の例では、accounts_domain ドメインの断片化の状態を調べ、続いて最大 15 分間にわたってそのデフラグメント処理を行います。

```
# defragment -v -n accounts_domain
defragment: Gathering data for 'accounts_domain'
Current domain data:
  Extents:                263675
  Files w/ extents:        152693
  Avg exts per file w/exts: 1.73
  Aggregate I/O perf:      70%
  Free space fragments:    85574
                        <100K  <1M   <10M  >10M
  Free space:             34%   45%   19%   2%
  Fragments:              76197  8930  440    7

# defragment -v -t 15 accounts_domain
defragment: Defragmenting domain 'accounts_domain'

Pass 1;
  Volume 2: area at block      144 ( 130800 blocks): 0% full
  Volume 1: area at block 468064 ( 539008 blocks): 49% full
```

```

Domain data as of the start of this pass:
  Extents:                7717
  Files w/extents:        6436
  Avg exts per file w/exts: 1.20
  Aggregate I/O perf:     78%
  Free space fragments:   904
                                <100K  <1M  <10M  >10M
  Free space:              4%    5%   12%   79%
  Fragments:              825    60   13    6

Pass 2;
Volume 1: area at block 924288 ( 547504 blocks): 69% full
Volume 2: area at block 144 ( 130800 blocks): 0% full
Domain data as of the start of this pass:
  Extents:                6507
  Files w/extents:        6436
  Avg exts per file w/exts: 1.01
  Aggregate I/O perf:     86%
  Free space fragments:   1752
                                <100K  <1M  <10M  >10M
  Free space:              8%    13%   11%   67%
  Fragments:             1574   157   15    6

Pass 3;
Domain data as of the start of this pass:
  Extents:                6485
  Files w/extents:        6436
  Avg exts per file w/exts: 1.01
  Aggregate I/O perf:     99%
  Free space fragments:   710
                                <100K  <1M  <10M  >10M
  Free space:              3%    11%   21%   65%
  Fragments:             546   126   32    6

Defragment: Defragmented domain 'accounts_domain'

```

各パスの前やデフラグメント処理の最後に表示される情報は、ドメインの改善度を示しています。Extents や Avg exts per file w/extents の値が減少している場合は、ファイル断片化状況が改善されていることを示しています。また、Aggregate I/O perf の値が増加している場合は、ファイル・エクステントの割り当ての全体的な効率が改善されていることを示しています。

詳細については、defragment(8) と Best Practice ドキュメントの『*Defragmenting an AdvFS Domain*』を参照してください。

5.10 ファイルの断片化解消

ドメイン全体ではなく、ファイルを対象に断片化を解消できます。 `vfast` ユーティリティが動作している間に、ファイルを個別に断片化解消できます。

そのファイルを断片化解消すべきか、つまり、そのファイルに多くのエクステンツがあるかを調べるには、`showfile` コマンドに `-x` オプションを指定して実行します。エクステンツが多くある場合、次のいずれかを行なって断片化を解消します。

- `migrate` ユーティリティを使用して、対象のファイルを同じボリューム (または別のボリューム) 上で、十分な長さの連続する空き領域が存在する部分に移動します。
- 次の手順でファイルをバックアップし復元します。
 1. `vdump` コマンドを使用してファイルをバックアップします。
 2. そのファイルを削除するか、名前を変更します。
 3. `vrestore` コマンドでデータを復元します。

5.11 マルチボリューム・ドメインのバランシング

`vfast` ユーティリティ (5.8 節) を実行していない場合、`balance` ユーティリティを使用して、マルチボリューム・ドメイン内のボリューム間で、使用スペースの割合を均等にすることができます。これにより、性能を改善し、将来のファイル割り当てでスペースが均等に割り当てられるようになります。

このユーティリティは、ドメインの各ボリュームのスペースの使用率が可能な限り均等になるまで、ファイルをボリューム間で移動します。図 5-1 を参照。このプロセスは、`vfast` ユーティリティで使用するプロセスと同じです。通常、`balance` ユーティリティではファイルを分割しないので、非常に大きなファイルのあるドメインでは、小さなファイルで構成されるドメインほど均等にはバランスがとれない場合があります。

ファイルをボリューム間で再分散するには、ドメインのすべてのファイルセットをマウントしておく必要があります。SysMan Menu の「AdvFS ドメインの管理」ユーティリティ (付録 A を参照) または AdvFS GUI (付録 E を参照) を使用するか、あるいはコマンド行で次のように `balance` コマンドを使用します。

balance *domain_name*

バランス処理を中断した場合、すでに再配置されたファイルはすべてそのまま新しい位置に残ります。残りのファイルについては、元の位置にそのまま残ります。

ドメインのバランシングを行なうには、root ユーザの特権が必要です。balance ユーティリティは、vfast、addvol、rmvol、defragment、または rmfset コマンドを同じドメインで実行している間は実行できません。

ファイルが均等に分散されているか調べるには、showfdmn コマンドでドメイン情報を表示します。% Used フィールドを見るとファイル分散の状況がわかります。

addvol コマンドでボリュームを追加した後、あるいは rmvol コマンドでボリュームを削除 (複数のボリュームが残っている場合) した後に、balance ユーティリティを使用してファイルの分散を均等化させます。

次の例のマルチボリューム・ドメイン `usr_domain` のバランスはとれていません。ボリューム 1 の使用済みスペースは 63% ですが、ボリューム 2 (小さいボリューム) の使用済みスペースは 0% (追加されたばかり) です。バランシング処理後は、両方のボリュームのスペース使用率がほぼ同じになっています。

```
# showfdmn usr_domain
      Id          Date Created      LogPgs Version Domain Name
3437d34d.000ca710 Wed Apr 3 10:50:05 2002 512          4 usr_domain

Vol  512-Blks   Free % Used   Cmode Rblks   Wblks   Vol Name
1L   1488716 549232    63%   on   128    128   /dev/disk/dsk0g
2    262144 262000     0%   on   128    128   /dev/disk/dsk4a
-----
1750860 811232    54%

# balance usr_domain
balance: Balancing domain 'usr_domain'
balance: Balanced domain 'usr_domain'
# showfdmn usr_domain
      Id          Date Created      LogPgs Version Domain Name
3437d34d.000ca710 Wed Apr 3 10:50:05 2002 512          4 usr_domain

Vol  512-Blks   Free % Used   Cmode Rblks   Wblks   Vol Name
1L   1488716 689152    54%   on   128    128   /dev/disk/dsk0g
2    262144 122064    53%   on   128    128   /dev/disk/dsk4a
-----
1750860 811216    54%
```

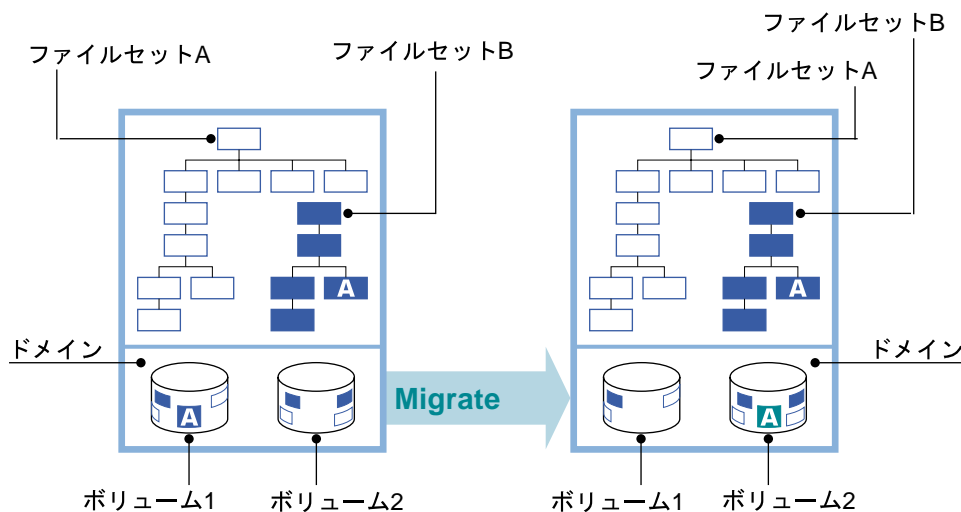
詳細については、balance(8) を参照してください。

5.12 別のボリュームへのファイルの移動

オプションの AdvFS Utilities を利用できる場合、**migrate** ユーティリティを使用して、過度にアクセスされるファイルや大きなファイルを、ドメイン内で選択した別のボリュームに移すことができます。高性能なデバイスを利用できる場合は、I/O が集中するファイルをより効率的なボリュームへ移動することもできます。

balance ユーティリティ、defragment ユーティリティ、および vfast ユーティリティに `-o topIObalance=` オプションを指定して、ファイルを移動することができますが、対象ファイルまたはページ、およびその移動先を選択できるのは、migrate ユーティリティだけです。ファイル全体または特定のページを移動できます。図 5-2 にファイル A の移動の手順を示します。このファイルを、ボリューム 1 からボリューム 2 に移動します。

図 5-2: ファイルの移動



ファイル全体を特定のボリュームに移動するには、migrate コマンドに `-d` オプションを指定して実行します。

```
migrate -d destination_vol_index filename
```

ファイル移動の際に、可能ならばその断片化が解消されます。migrate コマンドでは、指定したファイルの断片化を解消するように指定できます。migrate ユーティリティでは、どのファイルを断片化解消するように選択しても、それについて評価しません。ストライプ化したファイル・セグメン

トを別のセグメントが存在するディスクに移動することもできますが、ストライプ化の目的は果たせなくなくなります。

ファイルを移動するには、root ユーザの特権が必要です。1つのファイルに対して同時に複数の移動操作を行なうことはできません。また、一度に1つのボリュームからのみ移動できます。

次の例では showfile コマンドに -x オプションを使用して、ファイル src のエクステント・マップ (1.3.3 項) と性能を監視しています。2つのボリュームからなるドメインに属するこのファイルは、別のボリュームに移動されます。11のファイル・エクステントから1つのエクステントになり、18% から 100% に性能効率が向上しています。画面の最初の行はメタデータについてです。メタデータは新しいボリュームに移動されません。それは最初の場所に置かれたままです。画面の extentMap 部分にはファイルの移動されたページが示されています。

```
# showfile -x src
      Id Vol PgSz Pages XtntType Segs SegSz I/O Perf File
8.8002  1  16   11  simple  **    ** async 18%  src
      extentMap: 1
            pageOff    pageCnt      vol      volBlock      blockCnt
                  0           1         1       187296           16
                  1           1         1       187328           16
                  2           1         1       187264           16
                  3           1         1       187184           16
                  4           1         1       187216           16
                  5           1         1       187312           16
                  6           1         1       187280           16
                  7           1         1       187248           16
                  8           1         1       187344           16
                  9           1         1       187200           16
                 10           1         1       187232           16
      extentCnt: 11

# migrate -d 2 src
# showfile -x src
      Id Vol PgSz Pages XtntType Segs SegSz I/O Perf File
8.8002  1  16   11  simple  **    ** async 100% src
      extentMap: 1
            pageOff    pageCnt      vol      volBlock      blockCnt
                  0          11         2       45536           176
      extentCnt: 1
```

5.13 ファイルのストライプ化

ストライプ化では、ファイルが複数のボリュームに分散されます。異なるディスク・ドライブへの I/O 要求はオーバーラップさせることができるため、ストライプ化によって、シーケンシャルな読み取りおよび書き込みの性能が向上します。LSM、ハードウェア RAID、およびストレージ・エリア・ネットワーク (SAN) などの仮想ストレージ・ソリューションでは、すべてのファイルをストライプ化し、通常はシステムのセットアップ時に構成されます。AdvFS のストライプ化は単一のファイルに適用され、いつでも実行できます。

注意

AdvFS のストライプ化は、LSM、RAID、または SAN ボリュームを含まない、直接接続されているストレージにだけ使用してください。AdvFS のストライプ化とシステムのストライプ化を組み合わせると、システムの性能が低下する可能性があります。

AdvFS の `stripe` ユーティリティは、ストライプ・セグメントをドメイン内の特定の複数のボリュームに分散させます。このコマンドを実行するには、AdvFS Utilities が必要です。ストライプの幅は 64K バイト固定です。ただし、ファイルをストライプ化するボリュームの数を指定できます。

AdvFS `stripe` コマンドの構文を次に示します。

```
stripe -n volume_count filename
```

AdvFS のストライプ化ユーティリティは、`/etc/fstab` ファイルには使用できません。

次のいずれかの方法でドメインの AdvFS ストライプ化を削除できます。

- ファイルのストライプ化の解除

ストライプ化されていない新しいファイルにストライプ化したファイルをコピーし、元のファイルを削除します。

- ストライプ化ボリュームの削除

AdvFS ストライプ化セグメントを含むボリュームを削除する場合、`rmvol` ユーティリティは、同じファイルのストライプ化セグメントを含んでいない他のボリュームへセグメントを移動します。

残りのすべてのボリュームにストライプ化セグメントが含まれている場合、システムはボリュームにセグメントを移動する前に確認のためのプロンプトを表示します。

どのボリュームもストライプ化されているときにストライプ化の機能を十分に発揮させるためには、新しいファイルを既存のボリューム間でストライプ化し、そのストライプ化したファイルにオリジナルのファイルをコピーします。

詳細については、`stripe(8)`を参照してください。

5.14 新しいボリュームへのドメインとそのファイルセットの移動

新しいボリュームを追加した場合や、ファイルセットあるいはドメインがシステム・リソースを使い過ぎていると思われる場合は、ドメインを別のボリュームに移動することができます。ドメインを移動するかどうかは、そのドメインが置かれているデバイスの I/O 性能を調べます。`iostat` ユーティリティを、SysMan Menu の「モニタリング/チューニング - 入出力 (I/O) 統計の参照」ユーティリティ (付録 A を参照) から実行するか、コマンド行から実行します (`iostat(1)` を参照)。

ドメイン全体とそのファイルセットを新しいボリュームに移動するには、次の手順に従います。

1. 新しいデバイス上で新しいドメインを作成します。一時的に新しい名前が必要になります。
2. 新しいドメインに、移動元のドメイン内と同じ名前のファイルセットを作成します。
3. ファイルセットの一時的なマウント・ポイント・ディレクトリを作成します。
4. 一時的なマウント・ポイント・ディレクトリに新しいファイルセットをマウントします。
5. `vdump` コマンドを使用して、古いデバイスからファイルセットをコピーします。さらに、`vrestore` コマンドを使用して、コピーしたファイルセットを新たにマウントしたファイルセットにリストアします。
6. 古いファイルセットと新しいファイルセットをアンマウントします。

7. 新しいドメインの名前を古い名前に変更します。ドメインおよびファイルセットの名前は変更していないため、`/etc/fstab` ファイルを編集する必要はありません。古いドメインを削除します。
8. 古いファイルセットのマウント・ポイントに新しいファイルセットをマウントします。これにより、ディレクトリ構造を変更せずにファイルセットを移動できます。最後に一時的なマウント・ポイント・ディレクトリを削除します。

ドメイン内に複数のファイルセットが存在する場合には、2 ~ 8 の手順をファイルセットごとに実行します。

新しいドメインは新しいドメイン・バージョン番号 (DVN) 4 で作成されます。ドメイン・バージョン番号については、2.3.3.1 項を参照してください。古いバージョンのオペレーティング・システムを使用するためにドメイン・バージョン 3 を維持したい場合は、`mkfdmn(8)` を参照してください。ドメイン・バージョン番号が変わっても、`vdump` と `vrestore` ユーティリティは影響を受けません。

次の例では、ファイルセット `technical` のドメイン `accounts` をボリューム `dsk3c` へ移動しています。`new_accounts` は一時的なドメインで、最初に `/tmp-mnt` にマウントします。このファイルセットは `/technical` にマウントされていると想定します。また、`/etc/fstab` ファイルには、システムを、`/technical` の `accounts#technical` にマウントするためのエントリが含まれていると想定します。

```
# mkfdmn /dev/disk/dsk3c new_accounts
# mkfset new_accounts technical
# mkdir /tmp_mnt
# mount new_accounts#technical /tmp_mnt
# vdump -dx - /technical|vrestore -xf - -D /tmp_mnt
# umount /technical
# umount /tmp_mnt
# rmfdmn accounts
# rmdir /tmp_mnt
# mv /etc/fdmns/new_accounts/ /etc/fdmns/accounts/
# mount accounts#technical /technical
```

5.15 ドメイン・パニック情報の制御

`AdvfsDomainPanicLevel` 属性を使用して、ドメイン・パニックが発生した場合にクラッシュ・ダンプを作成するかどうかを選択します。ドメイン・パニックの典型的な原因は、デバイスの I/O エラーによるものです。AdvFS

はメタデータを書き込めない場合、ドメインをパニック状態にします。現在の実装では、このタイプの I/O エラーでは、システムがクラッシュしない限りクラッシュ・ダンプは作成されません。

クラッシュ・ダンプを適用するには、この属性の値を次のように設定します。

- 0 – ドメインのクラッシュ・ダンプを作成しません。
- 1 – マウント済みのファイルセットを含むドメインのクラッシュ・ダンプのみを作成します (省略時の値)。
- 2 – すべてのドメインのクラッシュ・ダンプを作成します。
- 3 – ドメイン・パニックが発生したらシステム・パニックが発生させます。この場合システムはクラッシュします。

属性の変更については `sysconfig(8)`、ドメイン・パニックからの回復については 6.3.1 項をそれぞれ参照してください。

トラブルシューティング

この章では、起こり得る問題からの回復手順と、問題の修復または回避手順について説明します。この章で取り上げるトピックは次のとおりです。

- ユーザ・ファイルの復旧 (6.1 節)
- システムの復旧 (6.2 節)
- 一般的な問題でのシステムの復旧とリストア (6.3 節)
- 予防的保守 (6.4 節)

6.1 ユーザ・ファイルの復旧

ユーザ・ファイルの復旧について、あらかじめ計画を立てておかない限り、削除されたファイルを復旧する簡便な手段はありません。次の 3 つの方法のいずれかで、削除されたファイルにアクセスできます。

- ディレクトリのバックアップ (`vdump` コマンド) とファイルのリストア (`vrestore` コマンド)。バックアップ・データが保存されているディレクトリへの書き込み許可が必要です。詳細については、第 4 章を参照してください。
- ファイルセット・クローンの設定。その日に変更のあったファイルにアクセスするためには、ファイルセット・クローンをその日のうちに作成します。クローンを作成するには、ルート特権が必要です。詳細については、2.4.10 項を参照してください。
- ディレクトリのゴミ箱の設定。この機能はユーザが自由に設定できます。ルート特権は必要ありません。詳細については、2.6 節を参照してください。

ファイルが削除されたときに、バックアップもゴミ箱もクローンもない場合、ファイルを復旧する手段はありません。しかし、ファイルが壊れている場合、`/sbin/advfs/salvage` コマンドを実行することで内容を部分的に復旧できる場合があります。手順については、6.2.6 項を参照してください。

6.2 一般的な復旧手順

次の復旧手順は、すべてのドメイン・パニックと多くのシステム・クラッシュに適用できます。クラッシュが1つのドメインによって発生しているならば、次に示す手順によって、そのドメインのファイルセットに再びアクセスできるようになります。必要なステップまでを行なってください。すべてのステップを行なう必要は必ずしもありません。ルート・ドメインを復旧する場合は、6.3.4 項を参照してください。

6.2.1 問題の報告のためのデータの保存

`sys_check` ユーティリティに `-escalate` オプションを指定して実行し、技術サポート担当者に報告するエスカレーション・ファイルを作成します。このユーティリティは、収集したデータを `/TMPDIR/escalate.tar` ファイルとして出力します。

`sys_check` コマンドは、SysMan Menuの「サポートとサービス — エスカレーション・レポートの作成」ユーティリティ (付録 Aを参照)を使用をして実行するか、コマンド行から実行します。

`sys_check` ユーティリティを実行するには、`root` ユーザまたは適切な特権が必要です。詳細については、`sys_check(8)`を参照してください。

6.2.2 システム・メタデータのコピーの保存

ドメインが、破損しているか、何らかの問題の原因になっていると思われる場合は、`/sbin/advfs/savemeta` コマンドを実行します。これにより、サポート担当者が調査するための、ドメインのメタデータのコピーが保存されます。`savemeta` コマンドに `-f` オプションを指定して実行すると、それぞれのファイルセットのフラグ・ファイルから構造情報を保存します。このユーティリティは、ログ・ファイル、ビットファイル・メタデータ・テーブル、各ボリュームのストレージ・ビットマップ、ドメインのルート・タグ・ファイル、およびファイルセット・タグ・ファイルを保存します。

たとえば、`domain_1` のメタデータを `/tmp/saved_domain_1` に保存するには、次のコマンドを入力します。

```
# /sbin/advfs/savemeta domain_1 /tmp/saved_domain_1
```

ファイルセット `fsetN` のフラグ・ファイルから、メタデータを
`/tmp/saved_fsetN` に保存するには、次のコマンドを入力します。

```
# /sbin/advfs/savemeta domain_1 -f fsetN /tmp/saved_fsetN
```

このコマンドを実行するには、`root` ユーザの特権が必要です。詳細については、`savemeta(8)`を参照してください。

6.2.3 ダメージを受けていないファイルの保存

影響のあるドメインのできるだけ多くのファイルセットをバックアップすることをお勧めします。バックアップ方式については、第4章を参照してください。ドメインを再作成するのに、このバックアップ、他の最新のバックアップ、および復旧ユーティリティ (`/sbin/advfs/verify` , `/sbin/advfs/fixfdmn` , および `/sbin/advfs/salvage`) の出力を使用することができます。

6.2.4 ファイル・システムの整合性の確認

メタデータの整合性を維持するため、`/sbin/advfs/verify` コマンドを実行して、ファイル・システムの構造を確認します。`verify` コマンドを実行すると、ビットファイル・メタデータ・テーブル (BMT)、ストレージ・ビットマップ、タグ・ディレクトリ、フラグ・ファイルなどのディスク構造がファイルセットごとにチェックされ、ディレクトリ構造が正しいかどうか、すべてのディレクトリ・エントリが有効なファイルを参照しているかどうか、およびすべてのファイルにディレクトリ・エントリがあるかどうかを確認できます。このコマンドを実行するには、`root` 権限が必要です。

`verify` コマンドは、次のような場合に実行します。

- 問題が明白な場合 (破損、ドメイン・パニック、データの喪失、I/O エラー)
- アップデート・インストレーションの前
- ファイルにアクセスしない状態が3～6か月以上続いた後、ドメイン内の全ファイルにアクセスするユーティリティ (`balance` , `defragment` , `migrate` , `quotacheck` , `repquota` , `rmfset` , `rmvol` , または `vdump` など) を実行する前

SysMan Menuの「AdvFS ドメインの管理」ユーティリティ (付録A参照) を使用するか、コマンド行で次のように `verify` コマンドを実行します。

```
/sbin/advfs/verify domain_name
```

verify コマンドは、ファイルセットを特殊なディレクトリにマウントします。ドメイン障害のために verify コマンドでファイルセットをマウントできない場合には、最後の手段として verify コマンドに -F オプションを指定して実行します。-F オプションを指定した場合、verify コマンドは mount コマンドの -d オプションを使用してファイルセットをマウントします。つまり、AdvFS は当該ドメインを復旧せずに、トランザクション・ログ・ファイルを初期化します。

注意

verify コマンドに -F オプションを指定しすると、直前の不完全な動作に対するドメインの復旧処理が行われないため、データが破損するおそれがあります。

状況によっては、verify コマンドでファイルセットがアンマウントされないことがあります。その場合には、ファイルセットを手動でアンマウントし、/etc/fdmns ファイル内に作成されているマウント・ポイントを削除する必要があります。

ファイル数が数百万にも及ぶマシンで verify コマンドを正常に動作させるためには、同コマンドに十分なスワップ領域を割り当てる必要があります。verify ユーティリティに必要なメモリ・サイズがカーネル変数の proc/max_per_proc_data_size プロセス変数の値を超過している場合には、このユーティリティは正常に終了しません。この問題に対処するには、ドメイン・サイズの 10% までスワップ領域の割り当てを増やして verify コマンドを実行します。詳細については、swapon(8)を参照してください。

verify コマンドが成功した場合、クラッシュをまねいたイベントが繰り返されたときのために、ドメインをバックアップすることをお勧めします。

setx と sety という 2 つのファイルセットを含むドメイン domainx を、verify コマンドでチェックしたときの出力例を次に示します。

```
# /sbin/advfs/verify domainx
+++Domain verification+++
Domain Id 2f03b70a.000f1db0
Checking disks ...
Checking storage allocated on disk /dev/disk/dsk10g
Checking storage allocated on disk /dev/disk/dsk10a
```

```

Checking mcell list ...
Checking mcell position field ...
Checking tag directories ...

+++ Fileset verification +++
+++ Fileset setx +++
Checking frag file headers ...
Checking frag file type lists ...
Scanning directories and files ...
    1100
Scanning tags ...
    1100
Searching for lost files ...
    1100

+++ Fileset sety +++
Checking frag file headers ...
Checking frag file type lists ...
Scanning directories and files ...
    5100
Scanning tags ...
    5100
Searching for lost files ...
    5100

```

この例では、ドメインに問題は見つかっていませんので、復旧手順を行なう必要はありません。詳細については、`verify(8)`を参照してください。

6.2.5 fixfdm ユーティリティを使用したディスク上のメタデータ損傷への対処

`/sbin/advfs/fixfdmn` ユーティリティは、壊れたメタデータを修復することによってドメインを使用可能な(マウント可能な)状態にすることを主な目的として設計されています。次の状況で、ファイルセットをアンマウントし、このユーティリティを実行します。

- 6.2.4 項で説明した `/sbin/advfs/verify` ユーティリティが動作しない場合。
- `verify` ユーティリティがディスク上の(メタデータ)損傷を検出した場合。
- ドメインに、マウントすると必ずドメイン・パニックを引き起こすファイルセットがあり、`verify` コマンドに `-F` オプションを指定して実行したくない場合。

`fixfdmn` ユーティリティはメタデータを走査し、壊れたメタデータを検索して、利用可能なメタデータが十分ある場合、これを修復しようとします。利用可能なメタデータが十分でない場合は、できるだけ壊れたメタデータを移動または削除し、さらに必要に応じてファイルを削除することによって損傷部分を取り除こうとします。

このユーティリティは、ファイルの内容のチェックと修正は行いません。ファイルからデータを復元することが急務である場合や、ドメインの大きな部分が壊れてしまった場合は、`/sbin/advfs/salvage` ユーティリティの使用法について、6.2.6 項を参照してください。

`fixfdmn` ユーティリティを実行した後は、ファイルセットを再度マウントします。ファイルセットを再マウントできたら、ファイルが削除されていないかチェックします。削除されている場合、バックアップからリストアします。ファイルセットをマウントできない場合、`fixfdmn` コマンドに `-u` オプションを指定してプロセスを取り消し、`salvage` コマンドで変更のないデータにアクセスできるようにします。

`fixfdmn` コマンドが成功した場合、クラッシュをまねくイベントが繰り返されたときのために、ドメインをバックアップすることをお勧めします。`fixfdmn` コマンドが成功すれば、さらに回復手順を実行する必要はありません。

`fixfdmn` コマンドに `-n` オプションを指定すると、ドメインをチェックだけして、修復は行なわれません。システム・メッセージを出力だけし、変更をディスクに書き込みません。

6.2.6 損傷したドメインからのファイル・データの救済

ドメインのファイルを復旧するには、`/sbin/advfs/salvage` ユーティリティを使用します。このコマンドは、壊れたドメインから救済可能なファイルを抽出し、復旧したファイルを置くためのファイルセットにそれらをコピーします。ドメインの損傷の性質によっては、すべてのデータを救済できる場合もあれば、いくつかしか救済できない場合もあります。

`/sbin/advfs/fixfdmn` コマンドを実行し、成功しなかった場合、`salvage` コマンドを実行する前に、まず `fixfdmn` コマンドに `-u` オプションを指定して実行し、このユーティリティが行なった変更を元に戻す必要があります。

salvage 機能を使用するには、SysMan Menuの「AdvFS ドメインの管理」ユーティリティ (付録 Aを参照) を使用するか、コマンド行で次のように salvage コマンドを実行します。

```
/sbin/advfs/salvage domain_name fileset_name
```

復元されたデータのコピー先には、ディスクとテープのどちらでも使用できます。復元できるデータの量は、ドメインの損傷の種類によって左右されます。詳細については、salvage(8) を参照してください。

salvage コマンドを実行しても、ドメインのすべてのデータを復旧できるという保証はありません。ファイル、ディレクトリ、ファイル名、またはファイルの一部が失われる可能性もあります。このユーティリティは、復元されたファイルの状態を記録したログ・ファイルを生成します。-l オプションを指定すると、このユーティリティが処理したすべてのファイルがログ・ファイルに記録されます。

salvage コマンドは、復元したファイルを、ファイルセット名に基づいて付けられた名前のディレクトリに格納します。salvage コマンドはファイルセットごとにlost+found ディレクトリを作成し、その中に親ディレクトリが見つからないファイルを格納します。復元されたファイルセット・ディレクトリが配置されるディレクトリのパス名は明示的に指定できます。このディレクトリを指定しない場合、salvage コマンドは、復元したファイルセットを現在の作業ディレクトリの下に格納します。

-F tar オプションを指定することによって、損傷のあるドメインから復元したデータを、tar 形式でテープに格納することも可能です。

6.2.6.1 ファイル・データの復元

最新のバックアップ・ソースがある場合、/sbin/advfs/salvage ユーティリティに -d オプションを指定して、バックアップの日付以降に変更されたデータのみを復元することができます。これによって salvage ユーティリティが処理しなければならないデータ量を軽減することができます。バックアップ・ソースがない場合は、指定したドメインのすべてのファイルに対して、salvage ユーティリティを実行します。

システムに、復旧するすべてのファイルを保持するスペースがない場合は、データをテープに復元し、その後元ディスクの場所に読み込むことがで

きます。また、salvage コマンドを `-F` オプションと `-f` オプションを指定して実行し、データを tar 形式で復元することもできます。

この手順は、次のとおりです。

1. 破損したドメインのすべてのファイルセットをアンマウントします。
2. 救済するデータをテープに保存する場合、ローカル・ドライブにテープを装着します。詳細については、『ハードウェア管理ガイド』を参照してください。
3. tar 形式でデータを救済しない場合は、復旧した情報を一時的に保持するドメインとファイルセットを作成し、ファイルセットをマウントします。
4. salvage コマンドを実行します。
5. salvage.log ファイルを調べ、すべての必要なファイルを復旧したかを確認します。失われたファイルがある場合、バックアップ・ソースから、それらを探す必要があります。
6. ドメインとマウント・ポイント・ディレクトリを再度作成します。ファイルセットを作成し、そこにマウントします。
7. バックアップからファイルをリストアします。
8. 救済したファイルを、一時的ドメインまたは tar ファイルからリストアします。

次の例では、ドメインをディスクに救済しています。壊れたドメインは PERSONNEL で、このドメインにはファイルセット personnel_a-e があり、/personnel_a-e にマウントされています。元のドメインは、ボリューム /dev/disk/dsk12c にあり、salvage コマンドの出力は、/dev/disk/dsk3c に置かれます。バックアップ・テープ /dev/tape/tape0 が利用できるので、損傷した PERSONNEL ドメインから、2001 年 12 月 5 日の午後 1 時 30 分 (最後にバックアップした日時) 以降に変更されたファイルだけを抽出します。ドメインは元のボリュームにリストアされます。

```
# umount /PERSONNEL
# mkfdmn /dev/disk/dsk3c RECOVER
# mkfset RECOVER recover_fset
# mkdir /recover
# mount RECOVER#recover_fset /recover
# /sbin/advfs/salvage -d 200112051330 -D /recover PERSONNEL
salvage: Domain to be recovered 'PERSONNEL'
```



```
salvage: Volume(s) to be used '/dev/disk/dsk12c'
salvage: Files will be restored to '/recover'
salvage: Logfile will be placed in './salvage.log'
salvage: Starting search of all filesets: 09-Dec-2001
salvage: Starting search of all volumes: 09-Dec-2001
salvage: Loading file names for all filesets: 09-Dec-2001
salvage: starting recovery of all filesets: 09-Dec-2001
```

mkfdmn コマンドを実行する前に、salvage.log ファイルを表示して、必要なすべてのファイルが復元されたことを確認します。失われたファイルがある場合、salvage コマンドの -d オプションにもっと前の日付を指定して実行することで、さらに情報を取得できる場合があります。新しいドメインを作成すると、salvage を再び実行することはできません。ただし、6.2.6.3 項に説明するように、-s オプションを指定して実行することはできます。

```
# mkfdmn /dev/disk/dsk12c PERSONNEL
# mkfset PERSONNEL personnel_a-e
# mkdir /personnel_a-e
# mount PERSONNEL#personnel_a-e /personnel_a-e
# vrestore /dev/tape/tape0 /personnel_a-e
# cp -Rp /RECOVER/personnel_a-e/* /personnel_a-e
# umount /recover
# rmfdmn RECOVER
rmfdmn: remove domain RECOVER [y/n] y
rmfdmn: domain RECOVER removed.
```

6.2.6.2 破損したルート・ドメインからのデータの復元

ルート・ドメインが破損してシステムをブートできない場合、インストール CD-ROM からシステムをブートし、/sbin/advfs/salvage コマンドを使用することができます。6.3.4 項の手順に従ってください。ルート・ドメインの損傷の種類と程度によって、ファイルの復旧が成功するかは左右されます。

6.2.6.3 ブロック単位でのデータの復元

salvage ユーティリティを実行しても復元できないファイルが大量にある場合は、salvage コマンドに -s オプションを指定して実行します。この処理では、ユーティリティがすべてのディスク・ブロックを少なくとも一度は読み取ることになるため、完了までにかかなりの時間を要します。

注意

正常なドメインに対して誤って `mkfdmn` コマンドを実行してしまった場合は、`salvage` コマンドに `-s` オプションを指定して実行する以外にファイルを復元する方法はありません。

注意

`salvage` ユーティリティはブロック型デバイスを直接オープンして読み取りを実行するため、セキュリティ上の問題につながる可能性があります。 `salvage` ユーティリティを `-s` オプション付きで実行すると、現在の AdvFS ドメインからデータの復元を試みるときに、削除済みの古い AdvFS ドメインのデータも取り込まれる可能性があります。

次の例では、ドメイン `PERSONNEL` からデータをブロック単位で復元します。

```
# /sbin/advfs/salvage -s PERSONNEL
salvage: Domain to be recovered 'PERSONNEL'
salvage: Volume(s) to be used '/dev/disk/dsk12c'
salvage: Files will be restored to '.'
salvage: Logfile will be placed in './salvage.log'
salvage: Starting sequential search of all volumes: 07-Oct-2001
salvage: Loading file names for all filesets: 07-Oct-2001
salvage: Starting recovery of all filesets: 07-Oct-2001
```

6.3 問題の修復

この節では、システムをリストアするさまざまな方法を説明します。

6.3.1 ドメイン・パニックからの回復

メタデータ書き込みエラーが発生するか、AdvFS ドメインの 1 つで破損が検出された場合、システムは、そのドメインに対してドメイン・パニック (システム・パニックではない) を発生させます。これにより、障害の発生したドメインは隔離され、システムは他のすべてのドメインで正常に機能し続けます。ドメイン・パニック後は、障害の発生したドメインのディスク・コントローラへの I/O 要求を AdvFS が発行することはありません。障害の発生したドメインへのアクセスはできませんが、そのドメイン内のファイルセットのアンマウントは可能です。

ドメイン・パニックが発生すると、EVM イベントのログが記録され (EVM(5) を参照)、次のようなメッセージがシステム・ログおよびコンソールに出力されます。

AdvFS Domain Panic; Domain name Id domain_Id

例:

```
AdvFS Domain Panic; Domain staffb_domain Id 2dad7c28.0000dfbb
An AdvFS domain panic has occurred due to either a
  metadata write error or an internal inconsistency.
This domain is being rendered inaccessible.
```

特に設定しなければ、アクティブなドメインでドメイン・パニックが発生すると、ライブ・ダンプが生成され、`/var/adm/crash` ディレクトリに置かれます。さらに、AdvFS 関連のエラーが `/var/adm/binary.errlog` に記録される場合もあります。

ドメイン・パニックから回復するには、次の手順を実行します。

1. `mount` コマンドに `-t advfs` オプションを指定して実行し、影響を受けているドメイン内のファイルセットをすべて把握します。
2. これらのファイルセットをすべてアンマウントします。
3. システムの障害レポート情報を保存します (6.2.1 項)。
4. ドメインのメタデータを保存します。詳細については、`savemeta(8)` および 6.2.2 項を参照してください。
5. 障害がハードウェアに起因するものである場合は、それを解決してから続けます。
6. ドメイン上で `/sbin/advfs/verify` ユーティリティを実行します (6.2.4 項を参照)。
 - エラーが発生しない場合は、アンマウントしたファイルセットをすべてマウントし、通常の操作を再開します。
 - `verify` コマンドは実行できても、エラーが表示される場合、`/sbin/advfs/fixfdmn` コマンド (6.2.5 項) を実行し、メタデータの破損を修復します。
7. 障害のために完全な回復が行えない場合は、新しいドメインを作成します (6.2.6 項)。

次の例では、ドメイン `staffb_dmn` 上のシステム・パニックの後で、必要な一連のコマンドを実行しています。

```
# mount -t advfs
staffb_dmn#staff3_fs on /usr/staff3 type advfs (rw)
staffb_dmn#staff4_fs on /usr/staff4 type advfs (rw)
# umount /usr/staff3
# umount /usr/staff4
# sys_check -escalate
# /sbin/advfs/savemeta staffb_dmn /tmp/saved_dmn
# /sbin/advfs/savemeta -f staff3_fs /tmp/saved_staff3
# /sbin/advfs/savemeta -f staff4_fs /tmp/saved_staff4
# /sbin/advfs/verify staffb_dmn
```

ドメイン・パニックが発生しても、リブートする必要はありません。

ドメイン・パニック前に、I/O エラーを報告するメッセージが表示された場合、修復すべきハードウェア障害に起因する可能性があります。次に、I/O エラー・メッセージの例を示します。

```
AdvFS I/O error:
Domain#Fileset: staffb_dmn#staff3_fs
Mounted on: /usr/staff3
Volume: /dev/disk/dsk6c
Tag: 0x00000006.8001
Page: 56
Block: 1200
Block count: 128
Type of operation: Write
Error: 19
EEI: 0x6200 (Advfs cannot retry this)
AdvFS initiated retries: 0
Total AdvFS retries on this volume: 0
I/O error appears to be due to a hardware problem.
Check the binary error log for details.
To obtain the name of the file on which
the error occurred, type the command:
/sbin/advfs/tag2name /usr/staff3/.tags/6
```

I/O エラーを示さないドメイン・パニックが繰り返し起こる場合、まず、`fixfdmn` ユーティリティを実行してみます (6.2.5 項)。`fixfdmn` を実行してもドメイン・パニックが起こる場合、`AdvfsDomainPanicLevel` 属性 (5.15 節) を調整して、問題解決に役立つ情報を取得することもできます。

6.3.2 読み取り専用でマウントされたファイルセットの復旧

ファイルセットのマウント時に、AdvFS はドメイン内のすべてのボリュームにアクセスできることを確認します。ドメインのメタデータに記録され

ているサイズは、各ボリュームでボリュームのサイズと一致しなければなりません。サイズが一致すると、マウント処理が続行されます。記録されているサイズよりも実際のボリュームが小さい場合、AdvFS は、ファイルセットで使用中とマークされている最後のブロックを読み込もうとします。このブロックが読み込めればマウントは成功しますが、ファイルセットは読み取り専用になります。使用中の最後のブロックが読み取れないボリュームがドメイン内にある場合、マウントは失敗します。詳細については、`mount(8)`を参照してください。

ファイルセットが読み取り専用でマウントされている場合、エラー・メッセージにフラグ付きで示されたボリュームのラベルをチェックしてください。よくあるエラーには、次の2つがあります。

- RAID アレイ上のディスクのラベルが適切でない。
- AdvFS ドメインが置かれているボリュームが、元のサイズより縮小された。

AdvFS Utilities を利用でき、かつドメインが複数のボリュームで構成されており、さらに障害の発生したボリューム (フラグのついた) を削除できるだけの空きスペースがある場合、ファイルセットを削除する必要はありません。ただし、次の作業に進む前に、バックアップを作成してください。

1. `rmvol` コマンドを使って、ドメインからボリュームを削除します。これにより、データは自動的に残りのボリュームに移されます。残りのボリュームに十分な空きスペースがない場合は、ボリュームを削除する前に `addvol` コマンドを使用してスペースを追加します。
2. `disklabel` コマンドを使って、障害のあるボリュームのディスク・ラベルを修復します。
3. `addvol` コマンドを使って、修復したボリュームをドメインに戻します。
4. `balance` コマンドを実行して、データを新しいボリュームに分散させます。

たとえば、ドメイン `data5` 内の `/dev/disk/dsk2c` (`disk_a` と呼ぶデバイスにあるとする) のラベルが正しくない場合、以下のコマンドで、そのボリューム上のファイルを移動し (`rmvol` コマンドで自動的に)、ボリュームを修復しデータをリストアした後に、それらのファイルを元に戻すことができます。最初の `addvol` のステップは、削除するボリューム上の情報のためにドメインのスペースが必要な場合のみ行ないます。

```
# addvol /dev/disk/dsk3c data5
# rmvol /dev/disk/dsk2c data5
# disklabel -z dsk2
# disklabel -rw dsk2 disk_a
# addvol /dev/disk/dsk2c data5
# balance data5
```

AdvFS Utilities を利用できない場合、次の手順を実行します。

1. ドメイン内のすべてのファイルセットのバックアップを取ります。
2. `rmfdmn` コマンドを使って、ドメインを削除します。
3. `disklabel` コマンドを使って、ボリュームのディスク・ラベルを修復します。
4. 新しいドメインを作成します。
5. バックアップからファイルセットをリストアします。

たとえば、ドメイン `data3` を含む `/dev/disk/dsk1c` (`disk_a` と呼ぶデバイスにあるとする) のラベルが正しくない場合、次のようにコマンドを実行します。

```
# vdump -0f -u /data3
# rmfdmn data3
# disklabel -z dsk1
# disklabel -rw dsk1 disk_a
# mkfdmn data3
# mkfset data3 data3fset
# mount data3#data3fset /data3
# vrestore -xf - /data3
```

6.3.3 /etc/fdmns ディレクトリのリストア

ファイルセットをマウントするためには、AdvFS に有効な `/etc/fdmns` ディレクトリがなければなりません (2.3.1 項を参照)。`/etc/fdmns` ディレクトリが間違っているか、または損傷していると、ドメインにアクセスできなくなりますが、ドメイン内のデータはそのまま残っています。`/etc/fdmns` ディレクトリはバックアップからリストアするか、あるいは再作成することができます。

このディレクトリの最新状態のバックアップ・コピーがある場合には、バックアップから `/etc/fdmns` ディレクトリをリストアすることをお勧めします。`/etc/fdmns` ディレクトリのバックアップには、標準のバックアップ機

能 (vdump, tar, または cpio) を使用できます。ディレクトリをリストアするには、バックアップ・プロセスと互換性のある回復手順を使用します。

/etc/fdmns ディレクトリをリストアできない場合、このディレクトリは手作業または /sbin/advfs/advscan コマンドによって再構成できます。手作業での再構成については 6.3.3.1 項、/sbin/advfs/advscan コマンドによる再構成については 6.3.3.2 項をそれぞれ参照してください。/etc/fdmns ディレクトリを再構成するための手順は、シングル・ボリュームとマルチボリュームのいずれのドメインの場合も類似しています。失われているドメイン、失われているリンク、またはディレクトリ全体に関して、ディレクトリの再構成を行うことができます。

6.3.3.1 手作業による /etc/fdmns ディレクトリの再構成

ディレクトリを手作業で再構築する場合は、それぞれのドメイン名とその対応するボリュームを知っておく必要があります。

注意

既存のドメインを再構築するために mkfdmn コマンドを実行しないでください。実行するとそのボリューム上のデータが破損します。ファイルが破損した場合、salvage コマンドに -S オプションを指定して実行することで、一部のファイルを復旧できる場合があります。

次の例では、/etc/fdmns ディレクトリと2つのドメインを再構築します。この例では、ドメインは存在してその名前も知っています。各ドメインは、シングル・ボリューム (特殊デバイス) からなります。リンクを作成する順序は問いません。この2つのドメインは、/dev/disk/dsk1c 上の domain1 と、/dev/disk/dsk2c 上の domain2 です。

```
# mkdir /etc/fdmns
# mkdir /etc/fdmns/domain1
# cd /etc/fdmns/domain1
# ln -s /dev/disk/dsk1c .
# mkdir /etc/fdmns/domain2
# cd /etc/fdmns/domain2
# ln -s /dev/disk/dsk2c .
```

次の例では、1つのマルチボリューム・ドメインを再構成しています。domain1 ドメインは、3つのボリューム /dev/disk/dsk1c、/dev/disk/dsk2c、および /dev/disk/dsk3c を含みます。

```
# mkdir /etc/fdmns
# mkdir /etc/fdmns/domain1
# cd /etc/fdmns/domain1
# ln -s /dev/disk/dsk1c .
# ln -s /dev/disk/dsk2c .
# ln -s /dev/disk/dsk3c .
```

6.3.3.2 advscan コマンドによる /etc/fdmns ディレクトリの再構成

/sbin/advfs/advscan コマンドを使用すると、ディスクを新しいシステムに移動した場合、デバイス番号を変更した場合、またはドメインの位置が分からなくなった場合に、どのディスク・パーティションや、どの Logical Storage Manager (LSM) ボリュームが AdvFS のドメインの一部になっているかを確認することができます。このコマンドはまた、間違えて /etc/fdmns ディレクトリを削除してしまった場合、/etc/fdmns ディレクトリからドメインを削除してしまった場合、または /etc/fdmns ディレクトリのドメインのサブディレクトリからのリンクを削除してしまった場合に、/etc/fdmns ディレクトリのすべてまたは一部を再構築する目的でも使用できます。詳細については、advscan(8)を参照してください。

AdvFS ファイル・システムを正しく動作させるためには、各ドメインで、次の3つの数値が一致していなければなりません。

1. advscan コマンドによって検索された、同一のドメイン ID を持つ物理パーティションの数。
2. ドメインのボリューム・カウント。AdvFS のディスク上のメタデータに記憶され、ドメイン内のパーティション数を示します。
3. /etc/fdmns のパーティションへのリンク数。各パーティションがリンクで表されるためです。

これらの数値が一致しない場合、advscan コマンドは、そのドメインを修正しようとします。

これらの数値が一致なくなる原因はいくつかあります。一般的にadvscan コマンドは、パーティション数や /etc/fdmns リンクの数よりもドメインのボリューム・カウントの方を信頼性が高いものとして扱っています。以

下の表では、不一致状態、考えられる原因、および advscan ユーティリティで実行できる修正処置の一覧を示します。

表 6-1 は、`/etc/fdmns/domain_name` ディレクトリ内のリンク数が、パーティション数とドメイン・ボリューム・カウン트의予想値と一致しない場合の、考えられる原因と修正処置を示しています。

表 6-1: リンク数の不一致

リンク数	考えられる原因	修正処置
少なすぎる	addvol コマンドが早期に終了してしまった。または <code>/etc/fdmns/domain_name</code> 内のリンクが手動で削除された。	advscan コマンドに <code>-f</code> オプションを指定して実行する前にドメインがアクティブ化されており、不一致の原因が addvol の中断によるものである場合、この状態は自動的に修正されます。それ以外の場合には、advscan ユーティリティが、 <code>/etc/fdmns/domain_name</code> ディレクトリにパーティションを追加します。
多すぎる	rmvol コマンドが早期に終了してしまった。または <code>/etc/fdmns/domain_name</code> 内のリンクが手動で追加された。	ドメインがアクティブ化されており、不一致の原因が rmvol の中断である場合、状況は自動的に修正されます。 <code>/etc/fdmns/domain_name</code> に手動で追加したリンクが原因であれば、 <code>/etc/fdmns/domain_name</code> ディレクトリ内のリンクを体系的に削除し、ドメインをアクティブ化します。削除するリンクの数は、 <code>/etc/fdmns/domain_name</code> ディレクトリ内のリンク数から、advscan コマンドで表示されるドメイン・ボリューム・カウンートを引いた数です。

表 6-2 は、ドメイン・ボリューム・カウン트가、パーティション数と `/etc/fdmns/domain_name` ディレクトリ内のリンク数の予想値と一致しない場合の、考えられる原因と修正処置を示しています。

表 6-2: ドメイン・ボリューム・カウンットの不一致

ドメイン・ボ リューム・カ ウント	考えられる原因	修正処置
少なすぎる	原因不明	修正できない。 /sbin/advfsfixfdmn コ マンドを実行し、ドメインを 使用可能な状態にします。
多すぎる	addvol コマンドが早期に終了 してしまった。追加しようとし ているパーティションが存在し ない、あるいは再利用された。	修正できません。 salvage ユーティリティを実行し、ド メインに残っているボリュー ムからできるだけ多くのデー タを復元します。

表 6-3 は、パーティション数が、ドメイン・ボリューム・カウンットと
/etc/fdmns/domain_name ディレクトリ内のリンク数の予想値と一致しな
い場合の、考えられる原因と修正処置を示しています。

表 6-3: パーティション数の不一致

パーティショ ン数	考えられる原因	修正処置
少なすぎる	パーティションの消失	修復できません。 fixfdmn コ マンドを使用してドメインを 使用可能な状態にします。
多すぎる	addvol コマンドが早期に 異常終了した。	addvol コマンドを再度 実行します。

次の例では、見つからなかったドメインはありません。 advscan コマンド
はデバイス dsk0 と dsk5 を走査して、AdvFS パーティションを探し、2
つのパーティション、dsk0c と dsk5c が見つかっています。ドメイン・
ボリューム・カウンットとして、2 が報告され、2つのリンクが /etc/fdmns
ディレクトリ内に登録されています。

```
# /sbin/advfs/advscan dsk0 dsk5
Scanning disks  dsk0 dsk5
Found domains:
usr_domain
          Domain Id      2e09be37.0002geb40
          Created        Thu Feb 28 09:54:15 2002
          Domain volumes      2
          /etc/fdmns links    2
          Actual partitions found:
                                dsk0c
                                dsk5c
```

次の例では、dsk6 を含むドメインを定義しているディレクトリが /etc/fdmns ディレクトリから削除されています。このため、/etc/fdmns リンクの数、パーティションの数、およびドメインのボリューム・カウントが一致していません。この例では advscan コマンドが、デバイス dsk6 を走査し、見つからないドメインを次のようにして再生しています。

1. AdvFS ドメインを含むパーティションを見つけます。ドメインのボリューム・カウントとして 1 を報告していますが、/etc/fdmns ディレクトリにはこのパーティションを含むドメイン・ディレクトリはありません。
2. 別の AdvFS ドメインを含む別のパーティションを見つけます。ドメインのボリューム・カウントは同様に 1 です。このパーティションを含むドメイン・ディレクトリはありません。
3. それ以外に AdvFS パーティションは見つかっていません。ドメイン・ボリューム・カウントと見つかったパーティションの数は、検出されたどちらのドメインでも一致しています。
4. advscan コマンドは、/etc/fdmns ディレクトリ内に、2 つのドメインのためのディレクトリを作成しています。
5. advscan コマンドは、/etc/fdmns/domain_name ドメイン・ディレクトリ内にそれらのデバイスに対するシンボリック・リンクを作成しています。

/etc/fdmns ディレクトリにそのパーティションのディレクトリのエントリがない場合、そのパーティション名にはアスタリスク (*) がつけられ、ユーティリティによってそのドメイン名が作成されます。復旧したドメインを元のドメインの名前に変更する必要があります。この例では、元のドメインの名前は revenue と expenses です。

```
# /sbin/advfs/advscan -r dsk6
Scanning disks  dsk6
Found domains:
*unknown*

Domain Id      2f2421ba.0008c1c0
Created        Sun Jan 20 13:38:02 2002

Domain volumes      1
/etc/fdmns links    0

Actual partitions found:
                    dsk6a*
```

```

*unknown*
      Domain Id      2f535f8c.000b6860
      Created        Mon Feb 25 09:38:20 2002

      Domain volumes      1
      /etc/fdmns links    0

      Actual partitions found:
                          dsk6b*

Creating /etc/fdmns/domain_dsk6a/
      linking dsk6a

Creating /etc/fdmns/domain_dsk6b/
      linking dsk6b
# mv /etc/fdmns/domain_dsk6a /etc/fdmns/revenue
# mv /etc/fdmns/domain_dsk6b /etc/fdmns/expenses

```

6.3.4 AdvFS ルート・ドメインのデータの損傷からの復旧

通常，AdvFS のルート・ドメインに回復不可能な損傷が生じた場合には，ルート・ファイル・システムを作り直して，システムをブート可能な必要があります。クラスタ構成を採用している場合は，『システム管理ガイド』および『クラスタ管理ガイド』を参照してください。また，ルート・ボリュームが LSM ボリュームの場合には，『*Logical Storage Manager*』を参照してください。ルート・ドメインを再構築するには，root ユーザの特権が必要です。

インストール CD-ROM か，RIS (Remote Installation Service) サーバが必要です。ルート・ドメインは，ブート・デバイス上に作成します。ルート・ドメインの適切なバックアップがない場合は，損傷の性質と程度によっては，ルート・ファイルを `/sbin/advfs/salvage` ユーティリティで復旧できる場合があります。詳細については，6.2.6 項を参照してください。`salvage` ユーティリティを使用して，最新のバックアップ後に修正または作成されたファイルを復旧できる場合もあります。

まず，ハードウェア・リソースを特定します。

1. インストール CD-ROM からシステムをリブートする場合，SRM (System Reference Manual) コンソールのプロンプトに次のコマンドを入力し，CD-ROM デバイスの名前を調べます。

```

>>> show device
...
DKA500          RRD47    1206    dka500.4.0.5.0

```

...

この例では、CD-ROM デバイスの名前は DKA500 です。

RIS サーバからブートする場合は、次のコマンドを入力してネットワーク・インタフェース・デバイスの名前を調べます。

```
>>> show device | more
....
ewa0.0.0.8.0      EWA0      08-00-2B-C3-E3-DC
....
```

この例では、ネットワーク・インタフェース・デバイスの名前は EWA0 です。

2. コンソール・ブート・デバイスの名前を調べます。デフォルト・ブート・デバイスの場合、次のコマンドで調べることができます。

```
>>> show bootdef_dev
bootdef_dev      dkb400.4.0.5.1
```

上記の例の dkb400.4.0.5.1 は、dkb400 がブート・デバイスであること、dk はデバイスが SCSI ディスクであること、b はデバイスが SCSI バス b に接続されていること、400 はデバイスの SCSI ターゲット ID が 4 で 論理ユニット番号 (LUN) が 00 であることを示しています。そこで、この例では、バス/ターゲット/LUN は、1/4/00 です。

デフォルト・ブート・デバイスでない場合は、show device コマンドを実行し、表示されるリストから特定します。

3. システムをインストレーション CD-ROM または RIS サーバからブートします。たとえば、特定した上記のデバイスを、CD-ROM を使ってブートする場合、次のコマンドを入力します。

```
>>> boot dka500
```

RIS サーバを使用する場合は、次のコマンドを入力します。

```
>>> boot ewa0
```

4. インストールを終了します。VGA グラフィック・コンソールを利用している場合は、インストレーションの終了を選択するか、「Installation and Configuration Welcome」ダイアログ・ボックスの [File] メニューからシェル・ウィンドウを選択します。シリアル・コンソール・ターミナルを使用している場合には、3) Exit Installation を選択します。

システム構成を確認したら、次の手順を実行します。

1. インストール・メディアからブートしたときに、ルート・ドメインがマウント可能な場合には、インストール手順では、まずインストールしたルート・ドメインの既存のデバイス・データベースを読み取ろうとします。ハードウェア・データベースの読み取りに失敗すると、エラー・メッセージが表示されます。このような状況が発生した場合は、UNIX が割り当てたそのデバイス名を、適切なハードウェア・デバイス名に変換する必要があります。

/sbin/hwmgr コマンドに -view devices オプションを指定することで、(前の手順で説明したように) 復旧するデバイスをそのバス/ターゲット/LUN 情報から調べ、バックアップに使用するディスクを特定し、テープ・デバイスがインストールされていることを確認します。

```
# /sbin/hwmgr -view devices
HWID: Device Name      Mfg Model              Location
-----
38:/dev/disk/floppy0c   3.5in floppy          fdi0-unit-0
41:/dev/disk/dsk0c     DEC RZ1DB-CA (C)      DEC bus-1-targ-4-lun-0
42:/dev/disk/dsk1c     DEC RZ1CB-CA (C)      DEC bus-1-targ-5-lun-0
43:/dev/disk/dsk2c     DEC RZ1CB-CA (C)      DEC bus-1-targ-6-lun-0
44:/dev/disk/cdrom0    DEC RRD47 (C)         DEC bus-0-targ-5-lun-0
46:/dev/ntape/tape0    DEC TLZ10 (C)         DEC
bus-1-targ-4-lun-0
```

この例のハードウェア・データベースでは、バス/ターゲット/LUN が 1/4/00 のディスクが dsk0 として認識されています。/dev/disk/dsk0a は、損傷のあるルート・ドメインを含むボリュームです。

特定したデバイスが妥当かどうかは、hwmgr コマンドに -flash オプションを指定すれば視覚的に確認できます。このコマンドを実行すると、指定したディスクのランプが 30 秒間にわたって点滅します。

```
# /sbin/hwmgr -flash light -dsf /dev/disk/dsk0a
```

2. この時点でテープ・デバイスをインストールする必要がある場合は、次のコマンドを入力します。

```
# /sbin/dn_setup -install_tape
```

3. 有効なバックアップがある場合は、そこからリストアできます。バックアップが古いものである場合は、そこからリストアした後、salvage コマンドに -d オプションを指定してその後の情報を復元します。有効な

バックアップがある場合は、ステップ 9 に進みます。バックアップに関連したファイルを救済する必要がある場合は、ステップ 8 に進みます。

4. 古いルート・ドメインをマウントします。

```
# mount old_root_domain#root /var/mnt
```

マウントに失敗した場合は、ステップ 7 に進みます。バックアップを行なうことはできません。

5. 可能ならバックアップを作成します。第 4 章を参照してください。
6. `/sbin/advfs/verify` コマンドに `-a` を指定して実行します (6.2.4 項を参照)。このコマンドが成功してエラーが表示されない場合、ステップ 12 に進みます。
7. `/sbin/advfs/fixfdmn` コマンドを実行します。詳細については、6.2.5 項を参照してください。成功した場合は、ステップ 12 に進みます。
8. `salvage` コマンドを実行してファイルを復旧し、一時ファイルセットがテープにそれを保存します。古いバックアップがある場合、`salvage` コマンドに `-d` を指定して実行します。オプションを指定しないと、ドメイン全体を救済することになります。

たとえば、ボリューム `/dev/disk/dsk0a` 上の壊れたルート・ドメインのファイルを、ローカルの未使用ディスク `/dev/disk/dsk2c` に復旧するには、次の手順を実行します。

- a. 復旧した情報を一時的に保管するドメインとファイルセットを作成し、ファイルセットをマウントします。

```
# mkfdmn /dev/disk/dsk2c RECOVER
# mkfset RECOVER recover_fset
# mkdir /var/recover
# mount RECOVER#recover_fset /var/recover
```

- b. `salvage` コマンドを実行します。コマンドの対象のボリュームを指定するために、`-v` オプションを使用する必要があります。古いバックアップとともに用いるファイルを救済している場合は、`-d` オプションを使用し、バックアップ後に変更になった情報のみを救済するようにします。

この例では、`root` ドメインのファイルが抽出され、`/recover` にマウントされたファイルセットに保存されます。

```
# /sbin/advfs/salvage -D /var/recover -V /dev/disk/dsk0a
salvage: Volume(s) to be used '/dev/disk/dsk0a'
salvage: Files will be restored to '/recover'
salvage: Logfile will be placed in './salvage.log'
salvage: Starting search of all filesets: 09-Dec-2001
salvage: Loading file names for all filesets: 09-Dec-2001
salvage: Starting recovery of all filesets: 09-Dec-2001
```

- c. salvage.log ファイルを調べ、すべての必要なファイルを復旧できたことを確認します。いったん次のステップで mkfdmn コマンドを実行すると、さらに情報を抽出するために salvage コマンドを実行することはできません。

9. 新しいルート・ドメインとルート・ファイルセットを作成します。このファイルセットを /var/mnt にマウントします。

```
# mkfdmn -r /dev/disk/dsk0a root_domain
Warning: /dev/disk/dsk0a is marked in use for AdvFS.
If you continue with the operation you can
possibly destroy existing data.
CONTINUE? [y/n] y
# mkfset root_domain root
# mkdir /var/mnt
# mount root_domain#root /var/mnt
```

10. 有効なバックアップ・ソースがある場合、そこからファイルをリストアします。

```
# vrestore -xf /dev/tape/tape0 -D /var/mnt
```

11. 救済したファイルがあれば、復旧先からルート・ドメインにコピーします。そして復旧用のドメインを削除します。

```
# cd /
# cp -Rp /recover/root/* /var/mnt
# umount /mnt /recover
# rmfdmn RECOVER
rmfdmn: remove domain RECOVER [y/n] y
rmfdmn: domain RECOVER removed.
```

ルート・ドメインがリストアされました。

12. システムを停止しリブートします。

この手順が成功しない場合は、オペレーティング・システムを再インストールし、ファイルを再作成します。

6.3.5 ハードウェア障害からの回復

ハードウェア障害のために AdvFS に問題が発生することがあります。たとえば、ファイル・システムへの書き込みが、ハードウェア障害によって失敗した場合、メタデータの損傷として現れることがあります。ハードウェアの問題は、AdvFS ファイル・システム側では解決できません。

マルチボリューム・ドメインの 1 つのドメインに原因不明のエラーが発生した場合、次の手順の 1 つ以上を行ないます。

- `/var/adm/messages` ファイルを、AdvFS の I/O エラー・メッセージがないか調べます。このファイルを見るには、`root` 特権が必要です。このファイルには、エラーが発生したドメイン、ファイルセット、ボリュームが示されます。また、I/O エラーの影響を受けたファイルを特定する方法も示されます。次のようなファイルが表示されます。

```
Dec 05 15:39:16 systemname vmunix: AdvFS I/O error:
Dec 05 15:39:16 systemname vmunix: Domain#Fileset:test1#tstfs
Dec 05 15:39:16 systemname vmunix: Mounted on: /test1
Dec 05 15:39:17 systemname vmunix: Volume: /dev/rz11c
Dec 05 15:39:17 systemname vmunix: Tag: 0x00000006.8001
Dec 05 15:39:17 systemname vmunix: Page: 76926
Dec 05 15:39:17 systemname vmunix: Block: 5164080
Dec 05 15:39:17 systemname vmunix: Block count: 256
Dec 05 15:39:17 systemname vmunix: Type of operation: Read
Dec 05 15:39:17 systemname vmunix: Error: 5
Dec 05 15:39:17 systemname vmunix: To obtain the name of
Dec 05 15:39:17 systemname vmunix: the file on which the
Dec 05 15:39:17 systemname vmunix: error occurred, type the
Dec 05 15:39:17 systemname vmunix: command
Dec 05 15:39:17 systemname vmunix: /sbin/advfs/tag2name
Dec 05 15:39:17 systemname vmunix: /test1/.tags/6
```

AdvFS の I/O エラー・メッセージが記録されていないにもかかわらず、ファイル・システムで不可解な動作が見られる場合には、できるだけ早くドメインをアンマウントし、`/sbin/advfs/verify` ユーティリティ (6.2.4 項を参照)、または `/sbin/advfs/fixfdmn` コマンドに `-n` オプションを指定して実行し (6.2.5 項)、ドメインのメタデータの整合性をチェックしてください。

- AdvFS の I/O エラー・メッセージに示されているボリュームの、デバイス・ドライバの I/O エラー・メッセージをチェックします。AdvFS の I/O エラー・メッセージに対応するこの I/O エラー・メッセージが存在しない場合は、ファイル・システムは、それが置かれているハー

ドウェアの障害による影響を受けています。できるだけ早くドメインをアンマウントし、`verify` ユーティリティ、または `fixfdmn` コマンドに `-n` オプションを指定して実行し、ドメインのメタデータの整合性をチェックしてください。

- マルチボリューム・ドメインを使用していて、十分な空き領域がある場合は、`rmvol` ユーティリティを 2.3.5 項の説明に従って使用して、問題のあるボリュームを削除してみます。これに成功すれば、ファイル・システムの問題は繰り返されなくなります。

`rmvol` に失敗した場合、ボリュームを再作成する必要があります。

1. 最新のバックアップがあれば、ドメインをリストアすることができます。バックアップがない場合、またはあっても情報が古すぎて使えない場合は、`salvage` ユーティリティを使用して (6.2.6 項)、破損したドメインの内容を抽出します。
2. `rmfdmn` コマンドを使用して問題のあるドメインを削除します。
3. `mkfdmn` コマンドを使用して、ドメインを再作成します。新しいドメインは、省略時の設定でドメイン・バージョン番号 (DVN) が 4 になります (2.3.3 項を参照)。必要に応じてボリュームを追加します。問題のあるボリュームは、新しいドメインには入れないでください。
4. 再作成したドメインの内容を、救済しバックアップした情報を使用してリストアします。
5. ドメイン内のファイルセットを再度マウントします。

6.3.6 AdvFS ディスクの別のマシンへの移動

マシンに障害が発生した場合、AdvFS ドメインを含むディスクを AdvFS ソフトウェアが動作している他のコンピュータに移すことができます。ディスクを新しいマシンに接続し、`/etc/fdmns` ディレクトリを変更して、移動したボリュームを新しいシステムが認識するようにします。新しいシステムとの競合を回避するために、ディスクの SCSI ID を割り当て直す必要がある場合があります。詳細については、ディスクの製造元が提供する指示書を参照してください。この処理を実行するには、`root` ユーザでなければなりません。

DVN4 ドメインを、オペレーティング・システム・ソフトウェアの Version 4 が動作しているシステムに移動することはできません。これを行なうとエ

ラー・メッセージが生成されます (互換性については、6.4.6 項を参照)。
Version 4 が動作しているマシンの DVN3 ドメインを Version 5 が動作しているマシンに移動することはできます。新しいオペレーティング・システムは、以前の OS で作成されたドメインを認識します。

注意

ボリュームを新しいマシンに移動するのに `addvol` コマンドや `mkfdmn` コマンドは使用しないでください。このコマンドを実行すると、移動しようとしているディスク上のすべてのデータが削除されます。すでにこのコマンドを実行してしまった場合は、6.2.6 項を参照してください。

ドメインがどのパーティション上にあるか分からない場合は、新しいマシンにディスクを追加し、`/sbin/advfs/advscan` ユーティリティを実行すると、この情報を再作成できることがあります。また、ディスクのラベルをチェックして、過去にどのパーティションが AdvFS パーティションとして作成されていたかを確認することもできます。パーティションがどのドメインに含まれていたかをディスク・ラベルで確認することはできません。

次の例では、古いマシン上の 2 つのディスク、`dsk3` と `dsk4` にドメイン `testing_domain` が存在するものとします。このドメインには、2 つのファイルセット、`sample1` と `sample2` が含まれています。これらのファイルセットは、`/sample1` と `/sample2` にマウントされています。移動するドメインにパーティション `dsk3c`、`dsk4a`、`dsk4b`、および `dsk4g` があることが既に分かっているものとします。この場合、ディスクは次の手順で移動します。

1. ディスクの移動先となる稼働中のマシンをシャットダウンします。
2. 異常のあるマシンから取り出したディスクを正常なマシンに取り付けます。
3. システムをリブートします。シングルユーザ・モードでリブートする必要はありません。
4. この新しいディスク用に作成されたデバイス・ノードを確認します。

```
# /sbin/hwmgr -show scsi -full
```

Device Name 欄をチェックして、今追加したディスクの名前を確認します。ディスク番号が最も大きいものが、それに相当します。

5. 新しいディスク ID はこの例では、disk 7 と disk 8 です。
/etc/fdmns ディレクトリを、移動したドメインからのシンボリック・リンクを含むように変更します。

```
# mkdir -p /etc/fdmns/testing_domain
# cd /etc/fdmns/testing_domain
# ln -s /dev/disk/dsk7c .
# ln -s /dev/disk/dsk8a .
# ln -s /dev/disk/dsk8b .
# ln -s /dev/disk/dsk8g .
# mkdir /sample1
# mkdir /sample2
```

6. /etc/fstab ファイルに、移動したファイルセットのマウント・ポイント情報を追加します。

```
testing_domain#sample1 /sample1 advfs rw 1 0
testing_domain#sample2 /sample2 advfs rw 1 0
```

7. ボリュームをマウントします。

```
# mount /sample1
# mount /sample2
```

6.3.7 マルチボリューム usr ドメインのリストア

マルチボリューム /usr ファイル・システムをリストアするには、その前に usr_domain ドメインとそのすべてのボリュームを再構成する必要があります。ただし、マルチボリューム・ドメインのリストアには License Management Facility (LMF) が必要です。LMF によって、マルチボリューム・ドメインの作成に必要な addvol コマンドを含む AdvFS Utilities が制御されます。

最初に 1 つのボリュームを含む usr ドメインを作成して addvol コマンドをリストアします。続いて LMF をリストアし、それを使用して addvol コマンドを有効化します。以上の処理が完了すれば、usr ドメインにボリュームを追加して、マルチボリューム・ドメイン全体を復元できます。

LMF は 2 つのコンポーネント、すなわち /usr/sbin/lmf に格納されているユーティリティと /var/adm/lmf 内のデータベースから構成されています。システムによっては /var が /usr にリンクされており、これらのディレクトリがともに usr ファイルセットに位置している場合もあります。その

ような構成のシステムでは、`addvol` コマンドと LMF の 2 つのコンポーネントを `usr` ファイルセットにリストアします。`/usr` ディレクトリと `/var` ディレクトリが `usr_domain` 内の異なるファイルセットに配置されているシステムでは、`addvol` コマンドと LMF ユーティリティを `usr` ファイルセット、LMF データベースを `var` ファイルセットにそれぞれリストアします。

次の例は、`dsk1g`、`dsk2c`、`dsk3c` の 3 つのボリュームで構成されるマルチボリューム・ドメイン `usr_domain` のリストア方法を示しています。`usr_domain` ドメイン内のファイルセット `usr` 内には、`/var` ディレクトリと `/usr` ディレクトリがともに配置されています。`/usr` と `/var` のバックアップ・テープが存在します。(バックアップ・テープを作成する必要がある場合は第 4 章を参照してください。) この手順では、ルート・ファイル・システムは既にリストアされているものとします。ルート・ファイル・システムのリストア方法については、6.3.4 項を参照してください。

1. ルート・ファイルセットを読み書き両用でマウントします。

```
# mount -u /
```

2. 古い `usr_domain` へのリンクを削除して、1 つ目のボリュームを使用して新しい `usr_domain` を作成します。

```
# rm -rf /etc/fdmns/usr_domain
# mkfdmn /dev/disk/dsk1g usr_domain
```

3. `/usr` および `/var` ファイルセットを作成しマウントします。

```
# mkfset usr_domain usr
# mount -t advfs usr_domain#usr /usr
```

4. `/usr` にソフト・リンクを作成します。`lmf` コマンドは、このディレクトリでデータベースを探すようになります。

```
# ln -s /var /usr/var
```

5. `/usr` のバックアップ・テープをセットし、そこからリストアします。

```
# cd /usr
# vrestore -vi
(/) add sbin/addvol
(/) add sbin/lmf
(/) add var/adm/lmf
(/) extract
(/) quit
```

6. ライセンス・データベースをリセットします。

```
# /usr/sbin/lmf reset
```

- usr_domainにボリュームを追加します。

```
# /usr/sbin/addvol /dev/disk/dsk2c usr_domain
# /usr/sbin/addvol /dev/disk/dsk3c usr_domain
```

- /usr バックアップの完全なリストアを行います。

```
# cd /usr
# vrestore -xv
```

次の例も、dsk1g、dsk2c、dsk3cの3つのボリュームで構成されるマルチボリューム・ドメインusr_domainのリストア方法を示しています。この例では、/varディレクトリと/usrディレクトリはusr_domainドメイン内の異なるファイルセットに配置されています。/usrと/varのバックアップ・テープが存在します。(バックアップ・テープを作成する必要がある場合は第4章を参照してください。)この場合は、/varと/usrのバックアップ・テープをともにマウントしなければなりません。この手順では、ルート・ファイル・システムは既にリストアされているものとします。ルート・ファイル・システムのリストア方法については、6.3.4項を参照してください。

- ルート・ファイルセットを読み書き両用でマウントします。

```
# mount -u /
```

- 古いusr_domainへのリンクを削除し、1つ目のボリュームを使用して新しいusr_domainを作成します。

```
# rm -rf /etc/fdmns/usr_domain
# mkfdmn /dev/disk/dsk1g usr_domain
```

- /usrおよび/varファイルセットを作成しマウントします。

```
# mkfset usr_domain usr
# mkfset usr_domain var
# mount -t advfs usr_domain#usr /usr
# mount -t advfs usr_domain#var /var
```

- /varバックアップ・テープをセットし、そこからリストアします。

```
# cd /var
# vrestore -vi
(/) add adm/lmf
(/) extract
(/) quit
```

- /usrバックアップ・テープをセットし、そこからリストアします。

```
# cd /usr
# vrestore -vi
(/) add sbin/addvol
```

```
(/) add sbin/lmf  
(/) extract  
(/) quit
```

6. ライセンス・データベースをリセットします。

```
# /usr/sbin/lmf reset
```

7. ボリュームを `usr_domain` に追加します。

```
# /usr/sbin/addvol /dev/disk/dsk2c usr_domain  
# /usr/sbin/addvol /dev/disk/dsk3c usr_domain
```

8. `/usr` バックアップの完全なリストアを行います。

```
# cd /usr  
# vrestore -xv
```

9. `/var` バックアップの完全なリストアを行います。

```
# cd /var  
# vrestore -xv
```

6.3.8 mkfdmn コマンドまたは addvol コマンドの誤った使用からの復旧

`mkfdmn` および `addvol` コマンドは、AdvFS で使用するようにボリュームを初期化します。そのボリュームがすでに他の AdvFS ドメインの一部である場合、このコマンドを使用するとそのボリュームに存在するドメインのメタデータを壊します。すでに初期化したボリュームに対して、これらのコマンドを誤って使用した場合、`salvage` ユーティリティに `-s` オプションを使用することで、そのファイルの一部を回復することができる場合があります。詳細については、6.2.6.3 項を参照してください。

`mkfdmn` または `addvol` コマンドを実行した後に実行したコマンドが多いほど、元のボリュームの内容にはその分だけ変更が加えられ、`salvage` コマンドで救済できる情報は少なくなります。`salvage` コマンドを使用しても十分な情報が得られない場合、唯一の選択肢は、バックアップからデータをリストアすることです。

6.4 問題の予防

クラッシュ発生後に各ドメインがマウントされると、システムによって回復コードが自動的に実行されます。このコードはトランザクション・ログ・ファイルを見て、システム・クラッシュ時に実行していたファイル・システム操作が完了したか、バックアップされたかをチェックします。これにより、AdvFS メタデータは、クラッシュ後も一貫した状態を保つことができま

す。クラッシュ発生時とは異なるオペレーティング・システムでファイル・システムを回復することはお勧めできません。

システム・クラッシュによってデータがダメージを受けた場合、5.5 節を参照して、そのような問題が将来起きないようにするために、データをディスクに書き込む方法を改善してください。

6.4.1 未使用スペースとディスク使用状況のチェック

ディスク・スペースの割り当て状況は、ファイル、ファイルセット、またはドメイン別に確認できます。表 6-4 は、ディスク・スペースの使用状況を調べるために使用できるコマンドを示しています。

表 6-4: ディスク・スペースの使用状況に関するコマンド

コマンド	機能
df ^{ab}	ディスク・スペースの使用状況をファイルセット別に表示します。ファイルセットが使用できるスペースは、ファイルセット・クォータが設定されている場合、それによって制限されます。ソフトとハードの両方のクォータが設定されている場合は、そのうちの小さい方のクォータ限界値を使って、利用できるディスク・スペースの値が計算されます。ドメインに、指定したファイルセット・クォータで利用可能な値より少ないスペースしかない場合は、このコマンドは、そのドメインで実際に使用できるスペースの値を表示します。複数のファイルセットを含み、ファイルセット・クォータを設定していないドメインの場合、すべてのファイルセットの容量の合計が 100% を超える場合があります。各ファイルセットで使用可能なスペースはドメイン全体であるため、この計算で使われる利用可能なスペースの数値が、ドメインの全スペースになるためです。
du	ファイルのブロック割り当てに関する情報を表示します。-a オプションを指定すると、個々のファイルの情報が表示されます
ls	ファイルごとにディスク・スペース使用量を表示します。-l オプションを指定すると、スパーズ・ファイルの未割り当て部分も含めて、ディスク・スペースの大きさが表示されます。-s オプションを指定すると、実際に割り当てられているブロック数が表示されます。このオプションは、特にスパーズ・ファイルの占有領域を確認する手段として利用できます。
showfdmn ^b	アクティブなドメイン内のそれぞれのボリュームの属性とブロック使用状況を表示します。マルチボリューム・ドメインでは、その他のボリューム情報も表示されます。

表 6-4: ディスク・スペースの使用状況に関するコマンド (続き)

コマンド	機能
showfile	特定のファイルの、またはディレクトリ内のファイルの、ブロック使用状況とボリューム情報を表示します。
showfsets ^b	ドメイン内のファイルセットに関する情報を表示します。 -q オプションを指定すると、ファイルセット・クォータの限界値が表示されます。
vdf ^{ab}	showfdmn, showfsets, およびdf コマンドの出力を再フォーマットします。この3つのコマンドの機能は、vdf コマンドにもあてはまります。

^a結果をより正確にするためには、quotacheck コマンドを最初に実行します。休止しているシステムの場合、このことは特に重要です。ユーザとグループのクォータを設定していない場合は、一時的に設定します。

^b表示する単位を選択します。

df および showfsets コマンドでは、メタデータとクォータ・ファイルが使用するディスク・スペースを含みません。そこで Used フィールドには、実際にドメインが使用している容量より小さい値が示されている可能性があります。vdf コマンドでは、メタデータとクォータ・ファイルの情報を含むディスク使用量を表示します。詳細については、3.4.2.4 項 とこのコマンドのリファレンス・ページを参照してください。

状況によっては、AdvFS のディスク使用状況の情報が破損することもあります。この情報を修復するには、quotacheck コマンドに -v オプションを指定して実行します。

クォータのディスク使用量を調べるには、3.4 節を参照してください。

6.4.2 ボリュームの再利用

既存ドメイン (/etc/fdmns ディレクトリにエントリが存在する) のストレージを他のドメインに追加するには、rmvol コマンドを使ってそのドメインからボリュームを削除して、それを他のドメインに追加します。AdvFS は自動的にドメインのディスク使用量を更新します。

たとえば、ドメイン old_domain 内のボリューム /dev/disk/dsk5c を、ドメイン new_domain に追加する場合には、old_domain 内のファイルセットをすべてマウントした後、次のコマンドを実行します。

```
# rmvol /dev/disk/dsk5c old_domain
# addvol /dev/disk/dsk5c new_domain
```

追加するディスクまたはディスク・パーティションが既存のドメインの一部でなくても、ラベルが設定されているために警告メッセージが表示される場合は、ディスク・ラベルをリセットします。これを `disklabel` コマンドを使用せずに行なうことができます。 `addvol` または `mkfdmn` コマンドの実行時に表示されるプロンプトに `yes` と答えると、ディスク・ラベルがリセットされます。この操作を行うと、追加するディスクまたはディスク・パーティション上のすべての情報が失われます。ディスクを他のシステムで使用するために移動したい場合は、6.3.6 項を参照してください。

6.4.3 ディスク障害

データを定期的かつ頻繁にバックアップし、ディスク障害の兆候を見逃さないようにします。障害が顕在化する前に、問題のあるディスクからファイルを削除することにより、多くのトラブルを未然に防ぐことができます。AdvFS Utilities を利用している場合、`addvol` コマンドを使用して新しいディスクを追加し、`rmvol` コマンドを使用して問題のあるディスクを削除することができます。ディスクの動作状況の確認方法についての詳細は、『システム管理ガイド』のイベント管理に関する情報を参照してください。

6.4.4 ディスク使用量の制御

リソースの使用についての制限が一切ない状態でシステムが稼働している場合は、システムにクォータを追加して、ユーザがアクセスできるディスク・スペースの容量を制限することができます。ユーザ・クォータおよびグループ・クォータは、UFS のクォータと同様に、ユーザまたはグループが 1 つのファイルセットに対して割り当てることができるスペースの量を制限します。ファイルセット・クォータは、AdvFS で導入されたもので、ドメインで利用できるすべてのスペースを 1 つのファイルセットで使い切ってしまうのを防ぎます。詳細については、第 3 章を参照してください。

6.4.5 クォータ限界値と猶予期間の上限の設定

クォータ限界値に達しつつある場合、または限界値に達するとは思えない場合でも、クォータ・システムの限界値を超過する可能性があります。クォータに関する詳細は第 3 章を参照してください。

6.4.5.1 クォータ限界値の超過

エディタの使用中に、編集データを保存するとクォータの限界値を超えることが分かった場合、エディタを強制終了したりファイルに書き出すとデータが失われる可能性があります。その代わりに、次のいずれかを行ないます。

- 編集中のファイルを保存する前に、他のファイルを削除して、編集中のファイルのために十分な空きスペースを作ります。
- 編集中のファイルを `tmp` などの別のファイルセットに保存し、他のファイルをいくつか削除するか、そのファイルセットのクォータを増やし、その後でファイルを元のファイルセットに戻します。

6.4.5.2 クォータ限界値までの保存

ハードおよびソフト限界値は、その値自体を含みません。その値未満の場合だけ、ファイルを作成することができます。たとえば、ハード限界値が1000ファイルの場合、作成できるのは 999 ファイルまでです。

ブロック数のハードおよびソフト限界値も同様に、その値自体を含みません。AdvFS ではまれに、ユーザ、グループ、またはファイルセット・クォータ値に達するまでに残り 8K バイトとなった時点で、残りのスペースの一部または全部を使用しようとする、クォータに制約される場合があります。これは、AdvFS がストレージを 8K バイト単位で割り当てているためです。ファイルに 8K バイトのデータを追加するとクォータの限界値に達するか超えてしまう場合、そのファイルは拡張されません。

6.4.5.3 クォータ限界値の変更

マウントしていないか、または `/etc/fstab` ファイル (2.4.1 項) に `userquota` と `groupquota` オプションが設定されていないファイルセットに対して、クォータ限界値または猶予期間を設定しても、設定した値は保持されません。 `edquota` コマンドを実行する前に、`/etc/fstab` のエントリをアップデートし、ファイルセットをマウントする必要があります。

6.4.5.4 猶予期間の無効化

猶予期間を無効にするには、`edquota` コマンドで値を 1 秒に設定します。値を 0 日に設定すると、省略値の猶予期間の7日が適用されます。

6.4.6 ディスク構造の互換性の問題の回避

Version 5.0 以降のオペレーティング・システムで作成されるドメインは、以前のバージョンとは互換性のない新しいオンディスク・フォーマットになります (2.3.3 項を参照)。新しいオペレーティング・システムは以前のディスク構造を認識しますが、以前のオペレーティング・システムは新しいディスク構造を認識しません。Version 5.0 以降のオペレーティング・システムのフル・インストールを行う代わりに、Version 4 からのアップデート・インストールを行う場合、`/root`、`/usr`、および `/var` のファイルは、DVN が 3 のままになります。一方、Version 5.0 のフル・インストールを行う場合には、これらの各ディレクトリに含まれるファイルの DVN は 4 になります。

旧バージョンのオペレーティング・システムから DVN4 のファイルセットにアクセスするには、Version 5.0 以降のオペレーティング・システム・ソフトウェアが稼働しているサーバからファイルセットを NFS マウントします。Version 5.0 より前のバージョンのオペレーティング・システムが稼働している環境で、DVN4 のドメインに属するファイルセットをマウントしようとすると、エラー・メッセージが表示されます。

DVN3 のドメインを DVN4 に自動的にアップグレードするツールはありません。ドメインを DVN4 にアップグレードするには、2.3.3.4 項の手順に従ってください。

6.4.7 ユーティリティの互換性の問題の回避

Version 5.0 以降のオペレーティング・システムで作成されるドメインは、新しいオンディスク・ファイル・フォーマットになっているため、以前のリリースの AdvFS Utilities の中には、使用すると DVN4 ドメインを壊す可能性があるものがあります。旧バージョンのオペレーティング・システムの、静的にリンクされた AdvFS 固有のユーティリティは、Version 5.0 以降では動作しません。このようなユーティリティは、通常 Version 4.0 より前のオペレーティング・システムのユーティリティです。また、旧バージョンの Tru64 UNIX に含まれる、以下に示す動的にリンクされた AdvFS Utilities は、Version 5.0 以降では動作しません。

- `advfsstat`
- `balance`
- `chvol`

- defragment
- rmvol
- showfdmn
- verify

6.4.8 ログ・ファイルの不整合の回避

システムがクラッシュすると、AdvFS はリブート時に復旧しようとします。クラッシュ時にマウントされていたファイルセットのあるドメインは、そのドメインのファイルセットを再マウントすると回復します。この回復処理では、AdvFS トランザクション・ログ・ファイルを使用して、AdvFS メタデータの一貫性を保ちます。

オペレーティング・システムのバージョンによってトランザクション・ログ・ファイルの構造は異なります。そのため、AdvFS の回復操作は、クラッシュ時に使用していたオペレーティング・システムと同じバージョンで実行することが重要です。そうしないと、ドメイン・メタデータの破壊やドメイン・パニック (あるいはその両方) が発生する危険性があります。

AdvfsDomainPanicLevel 属性 (5.7 節を参照) で、ドメイン・パニックをシステム・パニックに発展させるように設定したためにシステム・クラッシュが発生した場合、パニックが発生したドメインで `/sbin/advfs/verify` コマンドを実行して、データが損傷していないことを確認します。クラッシュ時にファイルセットがアンマウントされていた場合、または再マウントを実行し、`verify` コマンドを実行した場合、そのファイルセットを、適切であれば、異なるバージョンのオペレーティング・システムにマウントすることができます。

6.4.9 AdvFS ルート・ドメインのサイズ拡張

AdvFS のルート・ドメインを置くことができるボリューム (パーティション) の数は、クラスタ構成の場合を除いて 1 つだけです。したがって、ルート・ドメインのサイズを拡張するには、ルート・ドメインを含むボリュームのサイズを増やすか、より大きなボリューム上にルート・ドメインを作り直す必要があります。

6.4.9.1 ルート・ボリ्यूムのサイズ拡張

ルート・ドメインを含むボリ्यूムのサイズを拡張した場合、2.3.4.3 項の手順に従って、mount コマンドに `-o extend` オプションを指定して、システムにそれを知らせる必要があります。ルート・ボリ्यूムが、LSM ボリ्यूムの場合は、『*Logical Storage Manager*』を参照してください。

6.4.9.2 ルート・ボリ्यूムの大きなデバイスへの移動

ルート・ドメインを別のデバイスに再作成するには、インストールした新しいデバイスと、インストレーション CD-ROM または RIS サーバ、およびドメインをバックアップするためのバックアップ・テープまたは未使用のディスク・パーティションが必要です。

ルート・ドメインを移動するには、まず新しいディスクをインストールし、それを認識させる必要があります。ルート・ボリ्यूムに使用するディスクがすでにインストールされている場合は、ステップ 5 を省略し、次に進みます。

1. システムをシャットダウンします。
2. 新しいディスク・デバイスをインストールします。この方法については、ハードウェア・マニュアルを参照してください。
3. 追加したディスクがコンソールで認識されることを確認します。次の例では、DKB300 が追加されています。

```
>>> show device
polling ncr0(NCR 53C810) slot 1, bus 0 PCI,
      hose 1 A SCSI Bus ID 7
dka500.5.0.1.1      DKA500                      RRD45      1645
polling isp0(QLogic ISP1020) slot 4, bus 0 PCI,
      hose 1 SCSI Bus ID 7
dkb0.0.0.4.1      DKB0                          RZ1DB-CA   LYJ0
dkb100.1.0.4.1    DKB100                       RZ1CB-CA   LYJ0
dkb200.2.0.4.1    DKB200                       RZ1CB-CA   LYJ0
dkb300.3.0.4.1    DKB300                       RZ1BB-CS   0656
mkb400.4.0.4.1    MKB400                       TLZ10      02ab
...
```

4. システムをリブートします。ブート中にオペレーティング・システムが新しいデバイスを認識し、それをデバイス情報データベースに適宜反映します。追加したディスクには、オペレーティング・システムによって `dsk6` というデバイス名が付けられているとします。

特定したデバイスが妥当かどうかは、`/sbin/hwmgrr` コマンドに `-flash` オプションを指定すれば視覚的に確認できます。このコマンドを実行すると、指定したディスクのランプが 30 秒間にわたって点滅します。

```
# /sbin/hwmgrr -flash light -dsf /dev/disk/dsk6a
```

5. `root` としてログインし、`/etc/fdmns` ディレクトリでルート・ドメインを含むボリュームを調べます。

```
# ls -l /etc/fdmns/root_domain
total 0
lrwxrwxrwx  1 root      system 15 Feb 1 2002 dsk3a
-> /dev/disk/dsk3a
```

この例では `/dev/disk/dsk3a` がルート・ドメインを含むボリュームです。

6. 必要に応じ `disklabel` コマンドに `-t advfs` オプションを指定して、または `diskconfig` ユーティリティを使用して (Boot Block: リストから AdvFS を選択)、新しいディスクを AdvFS のブート可能ディスクとして構成します。この作業には `root` 権限が必要です。

```
# disklabel -r dsk6 > /tmp/backupdisklabel
# disklabel -R -r -t advfs dsk6 /tmp/backupdisklabel
# rm /tmp/backupdisklabel
```

7. 新しいデバイスにルート・ドメインを作成します。

- a. 古いルート・ドメインからブートしたシステムをシャットダウンします。
- b. オペレーティング・システム CD-ROM または RIS サーバからブートします。たとえば CD-ROM を使用する場合、次のコマンドを入力します。

```
>>> boot dka500
```

たとえば RIS サーバを使用する場合、次のコマンドを入力します。

```
>>> boot ewa0
```

- c. インストールを終了します。VGA グラフィック・コンソールを使用している場合は、インストールの終了を選択するか、「Installation and Configuration Welcome」ダイアログ・ボックスの [File] メニューからシェル・ウィンドウを選択します。シリアル・コンソール・ターミナルを使用している場合には、3) Exit Installation を選択します。

- d. 新しいルート・デバイスに新しいルート・ドメインとルート・ファイルセットを作成します。元のデバイスにリンクするために `/etc/fdmns` ディレクトリに新しいエントリを作成します。

```
# mkfdmn -r /dev/disk/dsk6a root_domain
# mkfset root_domain root
# mkdir /var/new_root
# mount root_domain#root /var/new_root
# mkdir /etc/fdmns/orig_root
# ln -s /dev/disk/dsk3a /etc/fdmns/orig_root
# mount orig_root#root /var/orig_root
```

- e. 古いルート・ドメインからファイルをダンプし、新しいルート・ドメインにリストアします。

```
# vdump -of - /var/orig_root | vrestore -xf \
-D /var/new_root
```

8. システム情報を新しいルート・ボリュームを指すようにアップデートします。この例では、ルート・ドメインとスワップ・パーティションは `dsk3` から `dsk6` に移されています。他に変更はありません。

`/etc/fdmns` ディレクトリを、新しいルート・ドメインを認識するようにアップデートします。ここでは、`dsk6a` が新しいルート・ドメインを含むボリュームであり、`dsk3a` が元のルート・ドメインを含むボリュームです。

```
# cd /var/new_root/etc/fdmns/root_domain
# ln -s /dev/disk/dsk6a dsk6a
# rm dsk3a
```

9. システムを停止し、省略時のブート・デバイスを変更します。

```
# halt
. . .
>>> set bootdef_dev dkb300
```

10. 新しいルート・ドメインをブートします。

```
>>> boot
```

6.4.10 メモリ・マッピング，ダイレクト I/O，およびデータ・ロギングの互換性の問題

`mount` コマンドに `-o adl` オプションを指定して実行し—時的アトミック書き込みデータ・ロギングを有効にした場合を除き，メモリ・マッピング，アトミック書き込みデータ・ロギング，およびダイレクト I/O は相互に排他的です。ファイルがこれらのモードのいずれかでオープンされている場合，同

じファイルを競合するモードでオープンしようとするとう失敗します。詳細については、5.5 節、5.6 節、および `mmap(2)` を参照してください。

メモリ・マッピング・モードのファイルは、`smoothsync` でフラッシュされますが、`smoothsync_age` によりフラッシュしない場合もあります。これは、ファイルがマッピング解除された後で一時的にメモリ・マッピング・モードにとどまることがあることを意味します。詳細については、『システムの構成とチューニング』を参照してください。

6.4.11 古いデータへのアクセスの禁止

ディスク上の古くなったデータは削除しない限り、クラッシュの後、または `salvage` コマンドを実行した後に復旧することができます。これらのデータがアクセスされないようにするには、`chfsets` コマンドを `-o objectsafety` オプションを指定して実行します。オブジェクト・セーフティを選択すると、アプリケーションは古いデータにアクセスできなくなりますが、データが完全に破壊されるわけではありません。

オブジェクト・セーフティ・オプションを有効にすると、そのファイルセットに属するディスク上のページがゼロで充填され、ファイルによって使用される前にディスクに強制的に書き込まれます。これにより、そのファイルの書き込み中にシステムがクラッシュしても、古いデータが見えてしまうことはなくなります。オブジェクト・セーフティを有効にする直前にストレージを割り当てられたファイルでは、その割り当てに関連するバッファがディスクに書き込みされると、オブジェクト・セーフティが起動します。

たとえば、`xyz_domain` のファイルセット `xyz3` のオブジェクト・セーフティを有効にするには、次のコマンドを入力します。

```
# chfsets -o objectsafety xyz_domain xyz3
```

オブジェクト・セーフティを使用すると、ページを2度ディスクに書き込まなければならないため性能は低下します。

6.4.12 無効または破損したセーブセット・フォーマット

ディスクに書き込まれたセーブセットのリストア時に、フォーマットが無効であるか、または壊れていることを示すメッセージが表示された場合には、そのセーブセットがパーティション `a` またはパーティション `c` にバックアップされていないかを確認してください。これらのパーティションにはブロック `0` が含まれています。ブロック `0` はディスク・ラベル・ブロックであり、

不慮の書き込みから保護されています。そこで、これらのパーティションにセーブセットをバックアップしようとしても、適切に保存されません。

ディスクのブロック 0 から始まるパーティションにセーブセットを正しくダンプするには、あらかじめディスク・ラベルをクリアする必要があります。ディスク・ラベルをクリアしないで `vdump` コマンドを実行した場合でも、コマンド出力では有効なセーブセットが含まれているように見えます。ところが、`vrestore` コマンドを実行すると、セーブセットに含まれるディスク・ラベルをセーブセットとして解釈したときに、エラーが返されます。ディスク・パーティションへのダンプについては、4.3.7 項を参照してください。

6.4.13 性能低下の対策

ディスクの性能は、それへの I/O 要求に依存します。1 つのボリュームに高負荷のアクセスが集中するようにドメインが構築されている場合、システムの性能が低下します。マルチボリューム・ドメインがある場合、さまざまな方法で負荷のバランスを改善し、スループットを向上させることができます。詳細については、『システムの構成とチューニング』、および第 5 章を参照してください。

性能を低下させている原因を見つけるには、まずシステムの動作をチェックします。性能を改善する方法は、数多くあります。

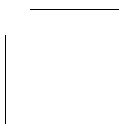
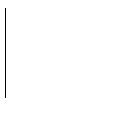
- ドメインのアップグレード (2.3.3.4 項を参照)
- ディスク・アクセスの互換性の問題の回避 (5.6 節を参照)
- ドメインの断片化解消 (5.9 節を参照)
- 他のボリュームへのファイルセットの移動 (5.14 節を参照)

AdvFS Utilities を使用できる場合には、以下の作業を行うこともできます。

- リソース割り当ての変更。ドメインのサイズを増やすか、遅いボリュームを速いボリュームに交換します。
- `vfast` ユーティリティの実行 (5.8 節を参照)
- マルチボリューム・ドメインのバランシング (5.11 節を参照)
- ファイル単位でのストライプ化 (5.13 節を参照)
- ファイル単位での移動 (5.12 節を参照)

6.4.14 **rmvol** または **migrate** コマンドを起動できない

rmvol コマンド (2.3.5 項) を使用してボリュームの削除を実行し、そのコマンドが異常終了すると、まれにボリュームがアクセスできない状態になり、書き込みができなくなることがあります。これらのボリュームには、**showfdmn** コマンドの出力結果に、“data unavailable” というマークがつきます。この場合、**chvol** コマンドに **-A** オプションを指定して実行し、ボリュームを再度アクティブ化します。



SysMan Menu によるファイル・システム情報へのアクセス

SysMan Menu のすべての機能を使用するためには root 特権が必要です。
SysMan Menu は次の方法でオープンできます。

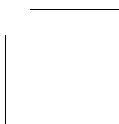
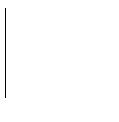
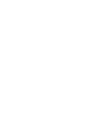
- コマンド行で次のコマンドを実行します。

```
# /usr/sbin/sysman
```
- システムで CDE が起動されている場合は、CDE のフロント・パネルから起動できます。
 1. フロント・パネルから「アプリケーション・マネージャ」を選択します。
 2. 「システム管理」アイコンを選択します。
 3. 「SysMan Menu」アイコンを選択します。

システム管理の作業リストから AdvFS のタスクにアクセスするには、次に示すアイコンをダブルクリックします。

1. ストレージ
2. ファイルシステム管理ユーティリティ
3. Advanced File System (AdvFS) ユーティリティ
4. 実行したいタスク

詳細については、`sysman(8)` を参照してください。



B

AdvFS コマンド

ここでは、AdvFS コマンドについて要約します。個々のコマンドの詳細については、リファレンス・ページを参照してください。

B.1 AdvFS ベース・システム・コマンド

次の表では、AdvFS の基本的なコマンドを示します。これらのコマンドは Tru64 UINX のライセンスに含まれており、レイヤード・ソフトウェアのライセンスは必要ありません。Tru64 UINX システムにリファレンス・ページのサブセットをインストールしてある場合は、man コマンドを使用して、各コマンドのリファレンス・ページにアクセスすることができます。

表 B-1: AdvFS 構成コマンド

コマンド名	説明
chfile	ファイルの属性を変更します。
chvol	ボリュームの属性を変更します。
defragment	ドメインのファイルを、より連続的に作成します。
mkfdmn	ドメインを作成します。
mkfset	ドメイン内にファイルセットを作成します。
renamefset	既存のファイルセットの名前を書き換えます。
rmfdmn	ドメインを削除します。
rmfset	ドメインからファイルセットを削除します。
/sbin/ad- vfs/switchlog	AdvFS ログ・ファイルをドメイン内の別の ボリュームに移動します。

表 B-2: AdvFS 情報表示コマンド

コマンド名	説明
advfsstat	ファイル・システムの統計値を表示します。
ncheck	ファイル・システム内の各ファイルのタグと絶対パス名を表示します。
showfdmn	ドメインの属性を表示します。
showfile	ファイルの属性を表示します。
showfsets	ドメイン内のファイルセット属性を表示します。
/sbin/advfs/vdf	ファイルセットまたはドメインの使用済みディスク容量と空きディスク容量を表示します。

表 B-3: AdvFS バックアップ・コマンド

コマンド名	説明
rvdump	全体および追加型ファイルセット・バックアップをリモートで実行します。
rvrestore	バックアップ・メディアからファイルをリモートでリストアします。
vdump	全体および追加型ファイルセット・バックアップを実行します。
vrestore	バックアップ・メディアからファイルをリストアします。

表 B-4: AdvFS チェックおよび修復コマンド

コマンド名	説明
/sbin/advfs/advscan	ディスク上の AdvFS パーティションを検索します。
/sbin/advfs/fixfdmn	ディスク上のメタデータの損傷を修復します。
/sbin/advfs/mountlist	マウントされている AdvFS ファイルセットをチェックします。
/sbin/advfs/salvage	損傷のある AdvFS ドメインからファイル・データを取り込みます。
/sbin/advfs/tag2name	タグ番号に対応するファイルのパス名を出力します。
/sbin/advfs/verify	ファイル・システムの一貫性のチェックおよび修復を行います。

表 B-5: AdvFS クォータ・コマンド

コマンド名	説明
chfsets	ファイルおよびブロックのクォータの変更と、オブジェクト・セーフティ、フラグ処理、およびデータ管理 API のオン/オフの切り替えを行います。
edquota	ユーザおよびグループ・クォータを編集します。
quot	ファイルセットの所有権を要約します。
quota	ユーザまたはグループ別にディスク使用状況と限界値を表示します。
quotacheck	ファイル・システム・クォータの矛盾をチェックします。
quotaoff	クォータの適用を無効にします。
quotaon	クォータの適用を有効にします。
repquota	ファイル・システムのクォータを要約します。

表 B-6: AdvFS ディスク構造コマンド

コマンド名	説明
/sbin/advfs/nvbmtpg	フォーマットされたビットファイル・メタデータ・テーブル (BMT) のページを表示します。
/sbin/advfs/nvfragpg	フラグ・ファイル情報を表示します。
/sbin/advfs/nvlogpg	フォーマットされたログのページを表示します。
/sbin/advfs/nvtagpg	フォーマットされたタグ・ディレクトリのページを表示します。
/sbin/advfs/savemeta	ドメインのメタデータを取得します。
/sbin/advfs/shblk	フォーマットされていないディスク・ブロックを表示します。
/sbin/advfs/shfragbf	フラグ・ファイル情報を表示します。
/sbin/advfs/vfilepg	マウントされていないドメインのファイルの内容を表示します。
/sbin/advfs/vsbmpg	ストレージ・ビットマップのページを表示します。

B.2 AdvFS Utilities のコマンド

以下の表では、AdvFS Utilities のコマンドについて説明します。これらのコマンドは、オプションの AdvFS Utilities のライセンスを必要とします。AdvFS Utilities のリファレンス・ページ・サブセットをインストールしてある場合は、`man` コマンドを使用して、これらの各コマンドのリファレンス・ページにアクセスできます。

表 B-7: AdvFS Utilities コマンド

コマンド名	説明
<code>addvol</code>	既存のドメインにボリュームを追加します。
<code>advfsd</code>	AdvFS GUI エージェント (デーモン) を起動します。
<code>balance</code>	ボリューム間で使用スペース率を平均化します。
<code>clonefset</code>	ファイルセットの読み取り専用コピーを作成します。
<code>dmapi</code>	データ管理アプリケーション・プログラミング・インタフェース (DMAPI) にアクセスします。
<code>dtadvfs</code>	AdvFS GUI を起動します。
<code>lsmsa</code>	LSMSA (Logical Storage Manager Storage Administrator) を起動します。
<code>migrate</code>	ファイルをドメイン内の別のボリュームに移動します。
<code>mktrashcan</code>	ディレクトリを、削除されたファイルを格納するゴミ箱ディレクトリに接続します。
<code>rmtrashcan</code>	ゴミ箱ディレクトリから指定ディレクトリを切り離します。
<code>rmvol</code>	既存のドメインからボリュームを削除します。
<code>shtrashcan</code>	特定のディレクトリに接続されるゴミ箱ディレクトリがあれば、それを表示します。
<code>stripe</code>	ファイルのストレージ割り当てを、ドメイン内の2つ以上のボリューム間で分散します。
<code>vfast</code>	ファイルの断片化を減らし、ボリューム間で空き領域と I/O 負荷のバランスをとります。

B.3 AdvFS コマンドと UFS コマンドの比較

次の表に、AdvFS コマンドと、同等または類似の UFS ファイル・システム・コマンドをリストします。すべての AdvFS コマンドとすべての UFS コマンドが 1 対 1 に対応しているわけではありません。

表 B-8: AdvFS コマンドと UFS コマンドの比較

AdvFS コマンド	UFS コマンド
addvol	なし。UFS はボリュームを使用しません。
advfsd	なし。UFS には GUI はありません。
advfsstat	iostat か、sys_check や Collect for Tru64 UNIX などのシステム監視ツールを使用します。ファイル・システム・イベントを監視するには、イベント監視ツールを使用します。
advscan	なし。UFS はボリュームを使用しません。
balance	なし。UFS はボリュームを使用しません。
chfile	なし。変更できる属性は、所有権と保護だけです。
chfsets	なし。変更できる属性は、所有権と保護だけです。
chvol	なし。UFS はボリュームを使用しません。
clonefset	なし。
defragment	なし。
dtadvfs, lsmsa	なし。UFS には GUI はありません。
migrate	cp および mv を使用します。
mkfdmn, mkfset	newfs を使用します。
mktrashcan, rmtrashcan, shtrashcan	延期削除機能のある CDE ウィンドウ環境を使用します。
mountlist	mount を使用します。
ncheck	AdvFS と同じです。
fixfdmn, nvbmtpg, nvfragpg, nvlogpg, nvtagpg, savemeta, shblk, shfragbf, vfilepg, vsbmpg	なし。類似の UFS コマンドはありません。

表 B-8: AdvFS コマンドと UFS コマンドの比較 (続き)

AdvFS コマンド	UFS コマンド
quota, edquota, quot, quotacheck, quotaoff, quotaon, repquota	AdvFS と同じです。
renamefset	なし。
rmfdmn, rmfset	umount を使用してファイル・システムをアンマウントし, /etc/fstab ファイルからエントリを削除します。
rmvol	なし。UFS はボリュームを使用しません。
vdump, rvdump, vrestore, rvrestore	dump, rdump, restore, および rrestore を使用します。AdvFS コマンドを使用できます。
salvage	なし。
showfdmn	なし。
showfile	file および ls -l を使用します。
stripe	なし。
switchlog	なし。
tag2name	なし。
vdf	df および du を使用します。
verify	fsck および bcheckrc を使用します。

ファイル・システムの変換

この付録には、`/usr` ファイル・システム、ルート・ファイル・システム、およびデータ・ファイル・システムを AdvFS に変換するために使用できる手順を記載しています。また、システム全体を AdvFS から UFS に変換する方法についても扱っています。

この付録で説明している変換方法は、変換プロセスを説明するためのガイドラインです。具体的なファイル名、テープ・ドライブ、およびディスク・パーティションはシステムごとに異なります。

`vdump` および `vrestore` のファイル・フォーマットは、オペレーティング・システム・ソフトウェアの Version 4.0 と Version 5 の間で互換性があります。システムを Version 5 にアップグレードし、ドメインを再作成し、バックアップからデータをリストアすると (第 4 章を参照)、AdvFS ファイル・セットおよびドメインは新しい構造を持つことになります (2.3.3 項を参照)。

C.1 `/usr` ファイル・システムの AdvFS への変換

AdvFS の初期インストール時に、`/usr` ファイル・システムに AdvFS を使用できます。UFS のファイル・システム `/usr` を AdvFS に変換すると、障害時のシステム停止時間を短縮することができます。`/usr` を AdvFS にインストールしていない場合は、バックアップ・テープ、中間ファイル、別のディスクのいずれかを使用して実行できます。

C.1.1 バックアップ・テープの使用

既存のファイル・システムをテープにバックアップした後 AdvFS 環境にリストアすることにより、`/usr` (UFS) ファイル・システムを同等な AdvFS ファイル・システムに変換できます。

この処理には次のものが必要になります。

- root ユーザ特権
- バックアップ・デバイスおよびメディア

- 変換されたファイル・システムの 5% 増しのディスク・スペース
- システムにインストールされた AdvFS

次のようなシステム構成を想定します。

- 既存の UFS 構成
ファイル・システム /usr
ディスク・パーティション /dev/disk/dsk3g
- 新しい AdvFS 構成
ファイル・システム /usr
ディスク・パーティション /dev/disk/dsk3g
ドメイン usr_domain
ファイルセット usr

次のような手順で、ファイル・システムを変換します。

1. /usrファイル・システムを含むシステムに root としてログインします。
2. AdvFS vdump コマンドを使用して、/usr ファイル・システムを省略時のテープ・ドライブである /dev/tape/tape0 にバックアップします。

```
# mt rewind
# cd /usr
# vdump -0 .
```
3. 次の手順で /etc/fstab ファイルを編集します。
 - a. /usr を UFS ファイル・システムとしてマウントする次のようなエントリを検索します。

```
/dev/disk/dsk3g /usr ufs rw 1 2
```
 - b. 上記の行を、/usr を AdvFS ファイル・システムとしてマウントする次のようなエントリに置換します。

```
usr_domain#usr /usr advfs rw 1 0
```
4. 次のコマンドを入力することにより、システムをシャットダウンします。

```
# shutdown -h now
```
5. システムをシングルユーザ・モードにリブートします。シングルユーザ・モードへのブート方法については、『システム管理ガイド』を参照してください。

6. I シングルユーザ・モードで、ルート・ファイル・システムを `rw` としてマウントし、`usr_domain` ドメインを作成し、さらに `usr` ファイルセットを作成します。mkfdmn コマンドに `-F` オプションを指定してパーティション・ラベルを UFS から AdvFS に強制的に変更し、警告メッセージを回避します。

```
# mount -u /  
# mkfdmn -F /dev/disk/dsk3g usr_domain  
# mkfset usr_domain usr
```

7. 次のコマンドを入力することにより、`usr` ファイルセットを `/usr` ディレクトリにマウントします。

```
# mount -t advfs usr_domain#usr /usr
```

8. `/usr` ファイル・システムをテープから `usr` ファイルセットにリストアします。

```
# vrestore -x -D /usr
```

9. システムをマルチユーザ・モードにブートします。システム・プロンプトが戻った時点で、変換された `/usr` ファイル・システムが使用できる状態になります。

C.1.2 中間ファイルの使用

既存のファイル・システムをファイルにバックアップした後 AdvFS 環境にリストアすることにより、`/usr` (UFS) ファイル・システムを同等な AdvFS ファイル・システムに変換できます。

この処理には次のものが必要になります。

- root ユーザ特権
- 中間ファイルのためのディスク・スペース (中間ファイルを含むファイル・システムは、同一ディスク、異なるディスクのどちらでも構いません。)
- 変換されたファイル・システムの 5% 増しのディスク・スペース
- システムにインストールされた AdvFS

次のようなシステム構成を想定します。

- 既存の UFS 構成
ファイル・システム `/usr`

ディスク・パーティション /dev/disk/dsk3g

中間ファイル /tmp/usr_bck

- 新しい AdvFS 構成

ファイル・システム /usr

ディスク・パーティション /dev/disk/dsk3g

ドメイン usr_domain

ファイルセット usr

次のような手順で、/usr ファイル・システムを変換します。

1. /usr ファイル・システムを含むシステムに root としてログインします。
2. AdvFS vdump コマンドを使用して、/usr ファイル・システムを中間ファイルである /tmp/usr_bck にバックアップします。

```
# cd /usr
# vdump -0f /tmp/usr_bck /usr
```

3. 次の手順で /etc/fstab ファイルを編集します。

- a. /usr を UFS ファイル・システムとしてマウントする次のようなエントリを検索します。

```
/dev/disk/dsk3g /usr ufs rw 1 2
```

- b. 上記の行を、/usr を AdvFS ファイル・システムとしてマウントする次のようなエントリに置換します。

```
usr_domain#usr /usr advfs rw 1 0
```

4. 次のコマンドを入力することにより、システムをシャットダウンします。

```
# shutdown -h now
```

5. システムをシングルユーザ・モードにリブートします。シングルユーザ・モードへのリブートについては、『システム管理ガイド』を参照してください。

6. シングルユーザ・モードで、ルート・ファイル・システムを rw としてマウントし、usr_domain ドメインを作成し、さらに usr ファイルセットを作成します。mkfdomn コマンドに -F オプションを指定してパーティション・ラベルを UFS から AdvFS に強制的に変更し、警告メッセージを回避します。


```
# mount -u /  
# mkfdmn -F /dev/disk/dsk3g usr_domain  
# mkfset usr_domain usr
```

7. 次のコマンドを入力することにより, `usr` ファイルセットを `/usr` ディレクトリにマウントします。

```
# mount -t advfs usr_domain#usr /usr
```

8. `/usr` ファイル・システムを中間ファイルから `usr` ファイルセットにリストアします。

```
# vrestore -xf /tmp/usr_bck -D /usr
```

9. システムをマルチユーザ・モードにブートします。システム・プロンプトが戻った時点で, 変換された `/usr` ファイル・システムが利用できるようになります。

C.1.3 別のディスクの使用

あるディスク上の `/usr` (UFS) ファイル・システム変換して, 同等な `/usr` (AdvFS) ファイル・システムとして異なるターゲット・ディスク上に配置することができます。

この処理には次のものが必要になります。

- `root` ユーザ特権
- 変換されたファイル・システムの 5% 増しのディスク・スペースを持つ別のディスク
- システムにインストールされた AdvFS

次のようなシステム構成を想定します。

- 既存の UFS 構成
ファイル・システム `/usr`
ディスク・パーティション `/dev/disk/dsk3g`
- 新しい AdvFS 構成
ファイル・システム `/usr`
ディスク・パーティション `/dev/disk/dsk2c`
マウント・ディレクトリ `/usr.advfs`
ドメイン `usr_domain`

ファイルセット `usr`

次のような手順で、`/usr` ファイル・システムを変換します。

1. `/usr` ファイル・システムを含むシステムに `root` としてログインします。
2. ドメインとファイルセットを作成します。

```
# mkfdmn /dev/disk/dsk2c usr_domain
# mkfset usr_domain usr
```
3. マウント・ポイント・ディレクトリを作成し、ディレクトリに新しいファイルセットをマウントします。

```
# mkdir /usr.advfs
# mount -t advfs usr_domain#usr /usr.advfs
```
4. `/usr` ディレクトリに移動します。

```
# cd /usr
```
5. システム上で他のユーザが作業していないことを確認して (システムをシングルユーザ・モードにすると確実です)、UFS ファイル・システムの内容を AdvFS ファイル・システムにコピーします。

```
# vdump -0f - -D . | vrestore -xf - -D /usr.advfs
```
6. 次の手順で `/etc/fstab` ファイルを編集します。
 - a. `/usr` を UFS ファイル・システムとしてマウントするために、次のようなエントリを検索します。

```
/dev/disk/dsk3g /usr ufs rw 1 2
```
 - b. 上記の行を、`/usr` を AdvFS ファイル・システムとしてマウントするための次のようなエントリと置換します。

```
usr_domain#usr /usr advfs rw 1 0
```
7. 次のコマンドを実行してテンポラリ・ディレクトリを削除します。

```
# umount /usr.advfs
# rmdir /usr.advfs
```
8. システムをシャットダウンおよびリブートします。システム・プロンプトが戻った時点で、変換された `/usr` ファイル・システムが使用できるようになります。

C.2 ルート・ファイル・システムの AdvFS への変換

ルート・ファイル・システムを AdvFS に変換することにより、システムを AdvFS ドメインからブートしてルート (/) ファイル・システムとして AdvFS を使用することができるようになります。AdvFS ルート・ドメインは、単一ディスクに置かれていなければなりません。オペレーティング・システムの初期インストール時に、AdvFS ファイル・システム上にルートをインストールすることができます。それ以外の方法としては、次のような方法があります。

注意

対象となる AdvFS のルート・ドメインは、ボリュームを 1 つだけ含むことができ、また変換されたルート・ファイル・システムを格納するのに十分なサイズでなければなりません。

この処理には、root ユーザ特権、別のブート可能ディスク (パーティション a または c を使用)、システムにインストールされた AdvFS ファイル・システムが必要です。

次のようなシステム構成を想定します。

- 既存の UFS 構成
ファイル・システム root
マウント・ディレクトリ /newroot
ディスク・パーティション /dev/disk/dsk1a
- 新しい AdvFS 構成
ファイル・システム root
マウント・ディレクトリ /newroot
ディスク・パーティション /dev/disk/dsk2a
ドメイン root_domain
ファイルセット root

次のような手順で、ルート・ファイル・システムを変換します。この例では `disk_name` というディスクを対象とします。

1. ルート・ファイル・システムを含むシステムに `root` としてログインします。
2. ドメインとファイルセットを作成します。

```
# mkfdmn -r /dev/disk/dsk2a root_domain
# mkfset root_domain root
```
3. 次のコマンドを入力することにより、マウント・ポイント・ディレクトリを作成し、ディレクトリに新しいファイルセットをマウントします。

```
# mkdir /newroot
# mount -t advfs root_domain#root /newroot
```
4. システム上で他に処理が行われていないことを確認し (システムをシングルユーザ・モードにすると確実です)、UFS ルート・ファイル・システムをルート・ファイルセットにリストアします。

```
# vdump 0f - / | (cd /newroot; vrestore -xf -)
```
5. 次のコマンドを使用して、ルート・ドメインを伴うディスクをブート可能なディスクにします。

```
# disklabel -r /dev/rdisk/dsk2a > /tmp/dsk2label
# disklabel -t advfs -r -R /dev/rdisk/dsk2a \
/tmp/dsk2label disk_name
```
6. AdvFS ルート・ファイルセット内の `/etc/fstab` ファイルを編集して、新しいルート・エントリを示すようにします。
 - a. `/newroot/etc/fstab` ファイルで、UFS ファイル・システムとしてルートをマウントしている次のようなエントリを検索します。

```
/dev/disk/dsk1a / ufs rw 1 1
```
 - b. 上記の行を、AdvFS ファイル・システムとしてルートをマウントする次のようなエントリと置換します。

```
root_domain#root / advfs rw 1 0
```
7. 編集が完了したら、次のコマンドを入力することにより、システムをシャットダウンします。

```
# shutdown -h now
```
8. 省略時のブート・デバイスである `BOOTDEF_DEV` を再設定して、新しいルート・ドメインを伴うディスクに変更します。この手順は、使用

しているハードウェアによって異なります。ハードウェア・マニュアルを参照してください。

9. システムをリブートして、AdvFS ルート・ファイル・システムを使用できるようにします。

T変換されたルート・ファイル・システムを使用するための準備がこれで整いました。

AdvFS ルート・ドメインは 1 つのディスクに制限されていますので、`addvol` コマンドを使用してルート・ドメインを拡張することはできません。

C.3 ファイル・システムの AdvFS への変換

ファイル・システムを AdvFS に変換することにより、リブート時間を短縮することができます。また、システム要求条件の変更に合わせてファイル・システム構成を簡単に修正することができます。

ファイル・システムを UFS から AdvFS に変換するには、バックアップ・テープ、中間ファイル、別のディスク、別のシステムのいずれかが必要です。

C.3.1 バックアップ・テープの使用

既存のファイル・システムを `vdump` コマンドでテープにバックアップし、`vrestore` コマンドで AdvFS 環境にリストアすることにより、UFS ファイル・システムを同等な AdvFS ファイル・システムに変換することができます。

この処理には次のものが必要になります。

- root ユーザ特権
- バックアップ・デバイスおよびメディア
- 変換されたファイル・システムの 5% 増しのディスク・スペース
- システムにインストールされた AdvFS

次のようなシステム構成を想定します。

- 既存の UFS 構成
ファイル・システム `/staff2`
マウント・ディレクトリ `/staff2`
ディスク・パーティション `/dev/disk/dsk2c`

- 新しい AdvFS 構成

ファイル・システム /staff2

ディスク・パーティション /dev/disk/dsk2c

ドメイン staff_domain

ファイルセット staff2

次のような手順で、/staff2 ファイル・システムを変換します。

1. /staff2 ファイル・システムを含むシステムに root としてログインします。

2. AdvFS vdump コマンドを使用して、/staff2 ファイル・システムを省略時のテープ・ドライブである /dev/tape/tape0 にバックアップします。

```
# mt rewind
# mount /staff2
# vdump -0f /dev/tape/tape0 /staff2
# umount /staff2
```

3. staff_domain ドメインと staff2 ファイルセットを作成します。mkfdmn コマンドに -F を使用してパーティション・ラベルを UFS から AdvFS に強制的に変更し、警告メッセージを回避します。

```
# mkfdmn -F /dev/disk/dsk2c staff_domain
# mkfset staff_domain staff2
```

4. 新しいファイルセットをディレクトリにマウントします。

```
# mount -t advfs staff_domain#staff2 /staff2
```

5. 次のコマンドを入力することにより、/staff2 ファイル・システムをテープから staff2 ファイルセットにリストアします。

```
# vrestore -xvf /dev/tape/tape0 -D /staff2
```

6. 次の手順で /etc/fstab ファイルを編集します。

- a. /staff2 を UFS ファイル・システムとしてマウントするためのエントリを検索します。

```
/dev/disk/dsk2c /staff2 ufs rw 1 2
```

- b. 上記の行を、/staff2 を AdvFS ファイル・システムとしてマウントするための次のようなエントリと置換します。

```
staff_domain#staff2 /staff2 advfs rw 1 0
```

変換された `/staff2` ファイル・システムを使用するための準備がこれで整いました。

C.3.2 中間ファイルの使用

既存のファイル・システムをファイルにバックアップした後 AdvFS 環境にリストアすることにより、UFS ファイル・システムを同等な AdvFS ファイル・システムに変換できます。

この処理には次のものが必要になります。

- root ユーザ特権
- 中間ファイルのためのディスク・スペース (中間ファイルを含むファイル・システムは、同一ディスク、異なるディスクのどちらでも構いません。)
- 変換されたファイル・システムの 5% 増しのディスク・スペース
- システムにインストールされた AdvFS

次のようなシステム構成を想定します。

- 既存の UFS 構成
ファイル・システム `/staff2`
ディスク・パーティション `/dev/disk/dsk3g`
中間ファイル `/tmp/staff_bck`
- 新しい AdvFS 構成
ファイル・システム `/staff2`
ディスク・パーティション `/dev/disk/dsk3g`
ドメイン `staff_domain`
ファイルセット `staff2`

次のような手順で `/staff2` ファイル・システムを変換します。

1. `/usr` ファイル・システムを含むシステムに root としてログインします。
2. AdvFS `vdump` コマンドを使用して、`/staff2` ファイル・システムを中間ファイルである `/tmp/staff_bck` にバックアップします。

```
# vdump -0f /tmp/staff_bck /staff2
```

3. `staff_domain` ドメインと `staff2` ファイルセットを作成します。
`mkfdomn` コマンドに `-F` オプションを指定してパーティション・ラベルを UFS から AdvFS に強制的に変更し、警告メッセージを回避します。

```
# mkfdomn -F /dev/disk/dsk3g staff_domain
# mkfset staff2
```

4. 次の手順で `/etc/fstab` ファイルを編集します。
 - a. `/staff2` を UFS ファイル・システムとしてマウントするための次のようなエントリを検索します。

```
/dev/disk/dsk3g /staff2 ufs rw 1 2
```

- b. 上記の行を、`/staff2` を AdvFS ファイル・システムとしてマウントするための次のようなエントリに置換します。

```
/staff_domain#staff2 /staff2 advfs rw 1 0
```

5. 次のようなコマンドを入力して、`staff2` ファイルセットを `/staff2` ディレクトリにマウントします。

```
# mount -t advfs staff_domain#staff2 /staff2
```

6. 中間ファイルから `staff2` ファイルセットへ `/staff2` ファイル・システムをリストアします。

```
# vrestore -xf /tmp/staff_bck -D /staff2
```

これで、変換された `/staff2` ファイル・システムが使用できる状態になります。

C.3.3 別のディスクの使用

あるディスク上の UFS ファイル・システムを変換して、同等な AdvFS ファイル・システムとして異なるターゲット・ディスク上に配置することができます。

この処理には次のものが必要になります。

- root ユーザ特権
- 変換されたファイル・システムの 5% 増しのディスク・スペースを持つ別のディスク system
- システムにインストールされた AdvFS

次のようなシステム構成を想定します。

- 既存の UFS 構成
 ファイル・システム /staff2
 ディスク・パーティション /dev/disk/dsk3g
- 新しい AdvFS 構成
 ファイル・システム /staff2
 ディスク・パーティション /dev/disk/dsk2c
 マウント・ディレクトリ /staff2
 ドメイン staff_domain
 ファイルセット staff2

次のような手順で、/staff2 ファイル・システムを変換します。

1. /staff2 ファイル・システムを含むシステムに root としてログインします。
2. ドメイン staff_domain およびファイルセット staff2 を作成します。

```
# mkfdmn /dev/disk/dsk2c staff_domain
# mkfset staff_domain staff2
```
3. マウント・ポイント・ディレクトリを作成し、そのディレクトリに新しいファイルセットをマウントします。

```
# mkdir /new_staff2
# mount -t advfs staff_domain#staff2 /new_staff2
```
4. システム上で他の作業が行われていないことを確認し、UFS ファイル・システムの内容を AdvFS ファイル・システムにコピーします。

```
# vdump -0f - -D /staff2 | vrestore -xf - -D /new_staff2
```
5. 次の手順で /etc/fstab ファイルを編集します。
 - a. /staff2 を UFS ファイル・システムとしてマウントするための次のようなエントリを検索します。

```
/dev/disk/dsk3g /staff2 ufs rw 1 2
```
 - b. 上記の行を、/staff2 を AdvFS ファイル・システムとしてマウントするための次のようなエントリと置換します。

```
staff_domain#staff2 /staff2 advfs rw 1 0
```
6. /new_staff2 をアンマウントします。

- ```
umount /new_staff2
```
7. /staff2 をアンマウントします。

```
umount /staff2
```
  8. 古いディレクトリを削除します。

```
rmdir /staff2
```
  9. 新しいファイルセットをマウントします。

```
mount /staff2
```

### C.3.4 別のシステムの使用

既存のファイル・システムを新しいファイル・システムに移動し、さらにそのファイル・システムを AdvFS に変換することができます。

この処理には次のものが必要になります。

- 2つのシステムと tar ユーティリティなどのファイル転送のための機能 (tar(1) を参照)
- ターゲット・システムの root ユーザ特権
- 変換されたファイル・システムの 5% 増しのディスク・スペース
- ターゲット・システムにインストールされた AdvFS

次のようなシステム構成を想定します。

- 既存の UFS 構成  
ファイル・システム /staff4
- 新しい AdvFS 構成  
ファイル・システム /staff4  
ディスク・パーティション /dev/disk/dsk2c  
マウント・ディレクトリ /staff4  
ドメイン staff\_domain  
ファイルセット staff4

次のような手順で、staff4 ファイル・システムを変換します。

1. /staff4 ファイル・システムを含むシステムにログインし、次のコマンドを使用してファイル・システムのバックアップを取ります。

```
tar c /staff4
```

2. ターゲット・システムの root としてログインします。

3. `staff_domain` ドメインと `staff4` ファイルセットを作成します。

```
mkfdmn /dev/disk/dsk2c staff_domain
mkfset staff_domain staff4
```

4. マウント・ポイント・ディレクトリを作成し、そのディレクトリに新しいファイルセットをマウントします。

```
mkdir /staff4
mount -t advfs staff_domain#staff4 /staff4
```

5. 省略時テープ・ドライブ `/dev/rmt0h` から `/staff4` ファイル・システムをリストアします。

```
mt rewind
tar x /staff4
```

6. `/etc/fstab` ファイルを編集します。 `/staff4` を UFS ファイル・システムとしてマウントするための次のような行を追加します。

```
staff_domain#staff4 /staff4 advfs rw 1 0
```

これで、`staff_domain` ドメインに、使用可能状態の `staff4` ファイルセットが含まれることになります。

## C.4 AdvFS から UFS への変換

ファイル・システム全体を AdvFS から UFS への変換する場合は、2 段階のステップを踏むことになります。最初に、AdvFS ルート・ファイル・システムを UFS に変換します。次に、各 AdvFS ファイルセットを UFS ファイル・システムに変換します。

### C.4.1 ルート・ファイル・システムの UFS への変換

ルート・ファイル・システムの変換を行う場合、AdvFS ルート・ファイルセットがマウントされている状態で UFS ディスクをマウントしなければなりません。

この処理には次のものが必要になります。

- root ユーザ特権
- 別のブート可能ディスク (パーティション a を使わなければならない)

次のようなシステム構成を想定します。

- 既存の AdvFS 構成  
ファイル・システム root  
ディスク・パーティション /dev/disk/dsk1a  
ドメイン root\_domain  
ファイルセット root
- 新しい UFS 構成  
ファイル・システム root  
ディスク・パーティション /dev/disk/dsk2a

ファイル・システムの変換は、次のような手順に従って行います。

1. root としてログインします。
2. UFS ファイル・システムを作成します (ここで disk\_name と呼ぶデバイスに作成)。  

```
newfs /dev/disk/dsk2a disk_name
```
3. マウント・ポイント・ディレクトリを作成し、UFS ファイル・システムをマウントします。  

```
mkdir /newroot
mount -t ufs /dev/disk/dsk2a /newroot
```
4. AdvFS ルート・ファイル・システムを /dev/disk/dsk2a UFS ファイル・システムにリストアします。  

```
vdump -0f - / | (cd /newroot; vrestore -xf -)
```
5. UFS ファイル・システムを含むディスクをブート可能なディスクにします。  

```
disklabel -r /dev/rdisk/dsk2a > /tmp/dsk2label
disklabel -t ufs -r -R /dev/rdisk/dsk2a \
/tmp/dsk2label <disk>
```
6. UFS ファイル・システムの /etc/fstab ファイルを編集して、新しいルート・エントリを参照するようにします。
  - a. /newroot/etc/fstab を検索して、AdvFS ファイル・システムのルートとしてマウントするための次のようなエントリを探します。  

```
root_domain#root / advfs rw 1 0
```

- b. 上記のエントリを、ルートを UFS ファイル・システムとしてマウントするための次のようなエントリと置換します。

```
/dev/disk/dsk2a / ufs rw 1 1
```

7. 次のコマンドを入力して、システムをシャットダウンします。

```
shutdown -h now
```

8. ブート省略時デバイス、BOOTDEF\_DEV を新しいルート・ディスクに再設定します (特定の情報については、ハードウェア・マニュアルを参照してください)。
9. システムをリブートして、UFS ルート・ファイル・システムを使用可能にします。

## C.4.2 ファイルセットの UFS への変換

ルート・ファイル・システムを UFS に変換したら、ファイルセットを UFS に変換することができます。

この処理には次のものが必要になります。

- root ユーザ特権
- バックアップのためのテープあるいはディスク

次のようなシステム構成を想定します。

- 既存の AdvFS 構成

マウント・ディレクトリ /staff2

ドメイン staff\_domain

ファイルセット staff2

- 新しい UFS 構成

マウント・ディレクトリ /staff2

ディスク・パーティション /dev/disk/dsk2c

次の例では、AdvFS ドメインに 1 つのボリューム、1 つのファイルセットが含まれるものとします。AdvFS ドメインに複数のファイルセットが含まれる場合には、ファイルセットごとに別々の UFS ファイル・システムを作成しなければなりません。

---

## 注意

---

変換を開始する前に、必ずすべての AdvFS ファイルセットのフル・バックアップを取ってください。

---

1. AdvFS ファイルセットのバックアップを取ります。

```
vdump -0f /dev/tape/tape0 /staff2
```

2. ファイルセットをアンマウントします。

```
umount /staff2
```

3. ファイルセットを削除します。

```
rmfset staff_domain staff2
```

4. ドメインを削除します。

```
rmfdmn staff_domain
```

5. 指定されたディスク・タイプの UFS ファイル・システムを作成します (ここで <disk> と呼ぶデバイスに作成)。作成しているパーティションの現在のラベルが AdvFS の場合、警告メッセージが表示されます。  
**yes** と応答して AdvFS の指定を無視します。

```
newfs /dev/disk/dsk2c <disk>
```

6. 次の手順で /etc/fstab ファイルを編集します。

- a. /staff2 を AdvFS ファイルセットとしてマウントするための次のようなエントリを検索します。

```
staff_domain#staff2 /staff2 advfs rw 1 0
```

- b. 上記の行を、/staff2 を UFS ファイル・システムとしてマウントするための次のようなエントリと置換します。

```
/dev/disk/dsk2c /staff2 ufs rw 1 2
```

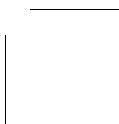
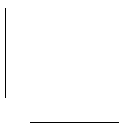
7. UFS ファイル・システムをマウントします。

```
mount -t ufs /dev/disk/dsk2c /staff2
```

8. vrestore コマンドを使用して、ファイルをバックアップから UFS ファイル・システムにロードします。

```
vrestore -xvf /dev/tape/tape0 -D /staff2
```

ドメインに複数のボリュームが含まれる場合には、ファイルセットに割り当てられたディスク・スペースが、UFS ファイル・システム・ディスク・パーティションの限界値を超えていないかをチェックしなければなりません。複数の UFS ファイル・システムを作成して、ドメイン内にファイルセットを保持する必要があるかもしれません。





# D

## データ管理アプリケーション・プログラミング・インタフェース (DMAPI)

DMAPI は、CAE (Common Applications Environment) 規格の 1 つであり、データ管理アプリケーションと基になるオペレーティング・システムのインタフェースとなります。データ管理アプリケーションの 1 つには、DMAPI 機能を使ってデータを 3 次ストレージとの間で移動したり、ファイル情報を追跡する、階層型ストレージ・マネージャ (HSM) があります。

DMAPI 機能を使用するには、AdvFS Utilities のライセンスが必要です。サポートされるデータ管理操作については、`dmapi(3)` を参照してください。

あるファイルが DMAPI 対応かどうかは、`showfsets` コマンドを使用して調べます。ファイルセットを DMAPI 対応にするには、`chfsets` コマンドを実行します。これには、`root` ユーザの特権が必要です。

DMAPI 対応のファイルセットは次のような点で制約されます。

- DMAPI 対応のファイルセットは、そのドメインの唯一のファイルセットでなければなりません。DMAPI 対応のファイルセットのあるドメインでは、そのファイルセットのクローンを作成したり、別のファイルセットを追加することはできません。
- `root`、`usr`、および `var` として指定したファイルセットは、DMAPI を使用できません。また、DMAPI 対応のファイルセットは `/`、`/usr`、および `/var` にマウントできません。
- DMAPI は、DVN4 ドメインだけをサポートします。
- DMAPI 対応のファイルセットを管理できるのは、1 つのアプリケーションだけです。複数のアプリケーションによる 1 つのファイルセットへの操作は、サポートされていません。
- DMAPI 機能を有効にする前に、ファイルセットをアンマウントしておく必要があります。

- DMAPI 対応のファイルセットは、DMAPI をサポートしないバージョンの Tru64 UNIX にはマウントできません。これを行なうと、データが破壊される恐れがあります。
- Tru64 UNIX では、既存のファイルセットあるいはインポートしたファイルセットが DMAPI 対応の場合、他のシステムからのファイルセットのインポートをサポートしていません。
- DMAPI 対応のファイルセットは、`vdump` コマンドを使用してダンプできません。データ管理アプリケーションのベンダが推奨するバックアップ手法を使用してください。
- DMAPI 対応のファイルセットでは、クォータはサポートされません。
- DMAPI 対応のファイルセット上の失われたデータは、`salvage` コマンドではリストアできません。`salvage` ユーティリティは 1 次ストレージにしか機能しません。DMAPI 対応のファイルセットに対してこのコマンドを実行すると、DMAPI アプリケーションによって非常駐となった、いくつかのデータが失われる可能性があります。DMAPI アプリケーションのベンダが用意している手順に従ってください。
- `defragment`, `balance`, `migrate`, および `/sbin/advfs/verify` などのストレージ管理ユーティリティは、DMAPI 対応のファイルセット上でも機能しますが、これらのユーティリティを使用した結果は、DMAPI を使用しない場合よりも、ストレージ管理の効率が低下する場合があります。
- 現在の DMAPI の実装では、非同期イベント・ポスティングはいくつかのまれなケースを除いて信頼のおけるものです。詳細については、CAE DMAPI 標準の Event Types を参照してください。
- DMAPI 対応のファイルセット上ではメモリ・マッピングはサポートされません。
- `vfilepg` コマンドは、ページがスパースであった場合、異常終了します。DMAPI によって 3 次のストレージに移されたページは、1 次ストレージ側ではスパースとして見えます。そこで `vfilepg` コマンドは DMAPI 対応のファイルセット上では正しく動作しない可能性があります。

新しくドメインとファイルセットを作成し、そのファイルセットで DMAPI 機能を有効にするには、次の手順を行ないます。

1. 新しいドメインとファイルセットを作成します。
2. `chfsets` に `-o dmapi` オプションを指定して、DMAPI 機能を有効にします。
3. `showfsets` コマンドを使用して、ファイルセットのステータスをチェックします。

たとえば、ファイルセット `sales_fset` をドメイン `receipt_dmn` に作成するには次のコマンドを入力します。

```
mkfdmn /dev/disk/dsk1c receipt_dmn
mkfset receipt_dmn sales_fset
chfsets -o dmapi receipt_dmn sales_fset
showfsets receipt_dmn
```

既存のファイルセットで DMAPI 機能を有効にするには、次の手順を実行します。

1. ドメインに 1 つのファイルセットのみが存在するようにします。複数存在する場合は次の操作を行ないます。
  - a. `vdump` コマンドを使用して、そのファイルセットをダンプします。
  - b. 新しいドメインとファイルセットを作成します。新しいドメインなので、そのファイルセットには元のファイルセットと同じ名前をつけることができます。
  - c. `vrestore` コマンドを使用して、ファイルセットを新しいドメインにリストアします。
2. 必要ならばファイルセットをアンマウントします。
3. `chfsets` コマンドを使用して、DMAPI を有効にします。
4. `showfsets` コマンドを使用して、ファイルセットのステータスをチェックします。

ファイルセットの DMAPI を無効にするには次の操作を行ないます。

1. ファイルセットをアンマウントします。
2. `chfsets` コマンドを `-o nodmapi` オプションを指定して呼び出します。

3. `showfsets` コマンドを使用して、ファイルセットのステータスをチェックします。

DMAPI を無効にできるのは、データ管理アプリケーションがファイルセットの修正を始めていない場合だけなので注意してください。データが破壊される可能性をなくすために、修正が開始されたら DMAPI を無効にできません。

そのファイルで、DMAPI 管理領域が有効になっているかチェックするには、次のように `showfile` コマンドを入力します。

```
showfile /mnt/file1
 Id Vol PgSZ Pages XtntType Segs SegSZ I/O Perf File
6.8006 1 16 4 simple ** ** async 100% file1
DMAPI regions present on this file.
```

# E

## グラフィカル・ユーザ・インタフェース

AdvFS ファイル・システムは、2 つのインタフェースで視覚化することができます。1 つは LSMSA (Logical Storage Manager Storage Administrator)、もう 1 つは AdvFS GUI (AdvFS Graphical User Interface) です。LSMSA では LSM オブジェクトを管理できるほか、LSM および AdvFS ファイル・システムの管理機能を実行できます。AdvFS GUI は AdvFS ファイル・システムを視覚的に表現します。LSMSA と AdvFS GUI は、どちらもライセンス登録によって使用可能になります。ライセンス登録については、最寄りの弊社代理店または各支店/営業所にお問い合わせください。

LSMSA は Java で実装されたアプリケーションであり、ローカル環境とリモート (クライアント) 環境のどちらでも使用できます。LSMSA の操作方法は、『*Logical Storage Manager*』で詳しく説明しています。

この付録では、AdvFS GUI の使用方法について説明します。AdvFS GUI は共通デスクトップ環境 (CDE) 上で動作するよう設計されています。AdvFS GUI はローカル環境のほか、リモート・システム・マネージャとしても使用できます。

クラスタ構成で実行している場合、AdvFS GUI にはローカル・ノードからアクセスします。クラスタ全体を管理するのにグラフィカル・インタフェースを使用することができますが、AdvFS GUI はクラスタ環境のために最適化されてはいません。

### E.1 AdvFS GUI のインストール

AdvFS GUI をロードするには、Tru64 UNIX オペレーティング・システムのアップデート時またはインストール時に、AdvFS Utilities サブセットを選択します。

表 E-1 では、AdvFS GUI を構成するサブセットのサブセット・タイトル、サブセット名 (xxx はサブセットの識別番号)、および内容について説明しています。表 E-2 では、AdvFS Utilities のソフトウェア・サブセットをロードおよび実行するために必要なディスク・スペースを示します。

表 E-1: AdvFS GUI サブセット

| サブセット                                    | 名前                  | 説明                                                 |
|------------------------------------------|---------------------|----------------------------------------------------|
| AdvFS Utilities                          | (OSFADVFSxxx)       | AdvFS 管理のための拡張ユーティリティのセット。使用に際しては、OS とは別のライセンスが必要。 |
| AdvFS Graphical User Interface (dtadvfs) | (OSFXADVFSxxx)      | AdvFS GUI とオンライン・ヘルプ・ファイル                          |
| AdvFS Agent (advfsd)                     | (OSFADVFSDAEMONxxx) | 従来型の UNIX デーモンで動作する AdvFS エージェント                   |
| AdvFS Japanese Graphical User Interface  | (OISJPXADVFSxxx)    | 日本語版 GUI の動作に必要なローカライゼーション・ファイル                    |

表 E-2: AdvFS GUI に必要なディスク・スペース

| ユーティリティ            | / (ルート) | /var | /usr |
|--------------------|---------|------|------|
| Utilities サブセット    | 0       | 0    | 300  |
| GUI (dtadvfs)      | 30      | 20   | 7000 |
| Agent (advfsd)     | 0       | 50   | 800  |
| 日本語ローカライゼーション・ファイル | 0       | 1    | 661  |

## E.2 GUI の構成要素

GUI は、advfsd (エージェント) と dtadvfs (実際のグラフィカル・インタフェース) の 2 つの部分からなります。GUI が動作するためには、エージェントが動作している必要があります。

### E.2.1 GUI エージェント (advfsd)

AdvFS GUI エージェント (**advfsd**) は、GUI に代わってコマンドを実行し、システム情報を取得します。このエージェントが動作していなければ、GUI は機能しません。このエージェントは、OSFADVFSDAEMON サブセットがインストールされている場合、ブート時に自動的に起動されます。エージェントはバックグラウンドで動作します。

通常は、advfsd を手動で実行する必要はありません。エージェントの起動および停止が必要な場合には、いつでも、コマンド行から実行できます

## E-2 グラフィカル・ユーザ・インタフェース

(advfsd(8) を参照)。エージェントを停止して、GUI を使用するときには再起動することができます。これは、エージェントのためにシステム動作が遅くなっていると感じるときに便利です。

エージェントを停止するには、次のコマンドを実行します。

```
/sbin/init.d/advfsd stop
```

エージェントを再起動するには、次のコマンドを実行します。

```
/sbin/init.d/advfsd start
```

システム上で同時に実行できるエージェントの数は 1 つです。advfsd を重複して実行しようとしても失敗します。

このエージェントにより、NetView などの SNMP (Simple Network Management Protocol) クライアントが AdvFS 情報を取得することができます。情報の流れは双方向ではなく、SNMP クライアントは advfsd に対してシステム構成コマンドを実行することはできません。

エージェントはシステムからの情報を定期的 to 取得し、ファイル・システムとストレージ・デバイスの情報を更新します。この情報は GUI 表示に渡されます。エージェントは、空きスペース警告状況の評価も行います。エージェントがシステム・ディスクのスキャンを開始する時間間隔は、エージェントの監視間隔で指定されます。エージェントの状態の監視間隔を変更する方法については、E.3.1 項を参照してください。この間隔の値は、エージェントを再起動しても保存されます。

## E.2.2 AdvFS GUI (dtadvfs)

AdvFS GUI (**dtadvfs**) を起動できるのは root ユーザだけです。dtadvfs は、端末ウィンドウと CDE アイコンのどちらからでも起動できます。GUI で実行可能なファイル・システム操作については、E.4 節で説明します。

コマンド行から GUI を起動するには、次のように入力します。

```
/usr/bin/X11/dtadvfs &
```

CDE アイコンから GUI を起動するには、次の手順に従います。

1. CDE ツールバーで「アプリケーション・マネージャ」を選択します。
2. 「アプリケーション・マネージャ」ウィンドウで「システム管理」アイコンをダブルクリックします。

3. 「システム管理」ウィンドウで「ストレージ管理」アイコンをダブルクリックします。
4. 「ストレージ管理」ウィンドウで「Advanced File System」アイコンをダブルクリックします。

GUI の動作中はウィンドウ表示が定期的に更新され、ファイル・システムの属性の変化が反映されます。GUI 更新間隔 (更新の時間間隔) は、デフォルトでは 15 分です。GUI の更新間隔の変更方法については、E.3.1 項を参照してください。この間隔の値は、エージェントが再起動されるまで有効です。

### E.2.2.1 GUI セキュリティ

GUI で管理される各システムには、セキュリティ上の必要に応じて 2 つのファイルに関連付けることができます。1 つはパスワード・ファイル、もう 1 つは許可ホストのファイルです。パスワード・ファイルでは、特定のシステムを管理する GUI の使用を、パスワードを知っている管理者だけに制限できます。許可ホストのファイルでは、そのファイルに記述されているシステムで動作している GUI のみ、システムのリモート管理を許可するよう設定できます。これらのファイルは、任意のテキスト・エディタを使って編集します。GUI からこれらのファイルに直接アクセスすることはできません。

root ユーザは、パスワードを使用するシステム上でパスワード・ファイル `/var/advfs/daemon/socket/gui.passwd` を作成します。システムにつき 1 つのパスワードを記述します (root パスワードの使用は避けてください)。このファイルにはプレーン・テキストでパスワードを記述します。正しいパスワードを入力しなければ、そのシステムに接続された GUI を実行することはできません。GUI を使用してリモートからシステムを管理しようとするユーザは、このパスワードを使ってシステムに接続する必要があります。

許可ホストのファイル `/var/advfs/daemon/socket/hosts.allow` も、root ユーザが作成します。このファイルには、GUI で管理できるシステム (ホスト) のリストをプレーン・テキストで記述します。hosts.allow ファイルには、ローカル・システム上で動作中の GUI が自動的に含まれます。つまり、そのシステム自体が含まれることになります。他のシステムからのリモート管理を許可するには、hosts.allow ファイルにそれらのシステムを記述して、当該システムのファイル・システム情報をそれらのシステムに送信する許可をエージェントに与えます。システムの gui.passwd ファイル



にパスワードを設定してシステムを保護している場合には、リモート・ユーザはそのパスワードも知っている必要があります。

詳細については、`advfsd(8)` と `dtadvfs(8)` を参照してください。

#### E.2.2.2 ディスクの無視

`/var/advfs/daemon/disks.ignore` ファイルを作成してディスクのリストを指定すると、エージェントはそれらのディスクのチェックを行いません。`disks.ignore` ファイルには、プレーン・テキストで各行に 1 つずつディスク・ドライブを指定します。オフラインの HSZ デバイスやスピンドウンしたディスク・ドライブがあると性能が低下する場合がありますため、このファイルが役立ちます。

エージェントは、ディスクをチェックするたびに `disks.ignore` ファイルを処理するため、故障したディスクをファイルのリストに追加することにより、利用可能になった時点でディスクをリストから削除できます。このときエージェントを停止する必要はありません。詳細については、`advfsd(8)` を参照してください。

`disks.ignore` ファイルに LSM ボリューム名を記述しても、LSM ボリュームは無視されません。LSM ボリュームを無視するには、そのボリュームの元のディスクを指定する必要があります。1 つのディスク・グループ全体を無視するには、そのグループ内のすべてのディスクを指定します。リストに指定されたディスクのすべてのパーティションが無視されるので、1 つのディスクのパーティションが 2 つ以上のディスク・グループに属する場合は、予期しない結果になることがあります。

#### E.2.2.3 ログ・ファイル

プロセスの進行に従い、エージェントのログ・ファイル `/var/advfs/daemon/logs/advfsd` と GUI のログ・ファイル `/var/advfs/gui/advfs_gui.log` にエントリが生成されます。これらのファイルは定期的にチェックしてください。エージェントと GUI のどちらのログ・ファイルも、ディスク・スペースが無制限に消費されないよう、定期的に改名されます。

#### E.2.2.4 ヘルプ

ヘルプ・システムには、AdvFS ファイル・システムに関するバックグラウンド情報と、GUI に関する情報が収められています。また、個々の操作につい

て解説されており、ファイル・システムの処理を行うために使用されるダイアログ・ボックスについても詳細に説明されています。

#### E.2.2.5 リモート・ファイル・システムの監視

自分が管理するシステムから、`hosts.allow` ファイルに指定されているファイル・システムをリモートで監視することができます。監視対象のシステムには、GUI を通じて次の手順で接続できます。

1. メイン・ウィンドウの [AdvFS] メニューから、[ホスト...] を選択します。
2. 「ホストの選択」ダイアログ・ボックスが表示されます。この中で、ホスト・システムを選択できます。または、ホスト・リストを選択して修正し、ホスト・システムの追加または削除を行うことができます。このリストはユーザ自身が作成し、定期的に接続するホストを識別するためにシステムに保存します。
3. ホストを選択し、[接続] をクリックします。

ホストの接続に失敗すると、どのホストにも接続できなくなります。この場合は、もう一度別のシステムに対して接続処理を試みてください。

ホスト・リストは、単に便宜上の目的で存在します。このリストによってセキュリティが強化されることはありません。ホスト名はいつでも、「ホストの選択」ダイアログ・ボックスの「選択」フィールドに入力できます。

#### E.2.2.6 LSM ボリュームの管理

LSM ボリュームを使用している場合は、GUI が LSM ボリュームをどのように処理するかについて、次のような知識が必要です。

- LSM ボリュームに、`unused` のラベルが付いている場合は、リストには `available` と示されます。
- AdvFS ドメインで使用している LSM ボリュームのサイズを変更した場合、AdvFS GUI を使用してこの変更をシステムに通知することはできません。mount コマンドに `-o extend` を指定してコマンド行から実行します。(2.3.4.3 項 および `mount(8)` を参照)
- コマンド行からドメインに LSM ボリューム `/dev/vol/volname` を追加した場合は、AdvFS GUI はそれが `/dev/vol/rootdg/volname` と同一のものであるとは認識しません。そのボリュームにファイルセットがマウントされている場合は、「デバイスとボリューム」ウィンド

うには `unknown` と表示され、マウントされているファイルセットがない場合には `unavailable` と表示されます。そのボリュームをドメインに追加しようとする、使用中であることを示すエラー・メッセージが表示されます。

- AdvFS が使用しているボリュームを GUI の動作中に LSM に取り込んだ場合、GUI ウィンドウではボリューム名は更新されません。正しい名前を表示するには、GUI を終了してから再起動する必要があります。
- `disks.ignore` ファイルに LSM ボリューム名を含めても、LSM ボリュームを無視することはできません。LSM ボリュームを構成しているディスクをリストしなければなりません。

### E.3 ファイル・システム情報の表示

ファイル・システムの構成要素であるオブジェクト (ボリューム、ドメイン、ファイルセット、ファイルセット・クローン) を、GUI ウィンドウのオブジェクト・ツリーに表示できます。オブジェクト・ツリーには、オブジェクト相互の階層構造による関係が示されます。オブジェクトに他のオブジェクトが包含されている場合は、左側にフォルダ・アイコンが表示されます。フォルダ・アイコンをクリックすることにより、オブジェクト・ツリーの表示を展開したり縮小したりすることができます。つまり、ツリー内のそのオブジェクトより下のオブジェクトの表示と非表示が切り替わります。たとえば、あるドメインのフォルダ・アイコンをクリックすると、そのドメインに属するファイルセットの表示と非表示が切り替わります。

ウィンドウ・ビューを変更して、オブジェクトについての重要な情報をさまざまな方法で表示できます。次の情報を表示できます。

- メイン・ウィンドウの情報

メイン・ウィンドウが最初に表示されたときには、このウィンドウ内にはファイル・システムの全体図が表示されています。このウィンドウは、すべての GUI 管理タスクの出発点であり、ファイル・システムの状態を監視するための主要な画面です。[表示] メニューのメニュー項目を選択することにより、ドメインとファイルセット、ドメインとボリューム、ドメイン、ファイルセット、およびボリュームの状態が表示できます。また、[表示] メニューを使うと、それらの表示からファイルセット・クォータの表示に切り替えたり、元の表示に戻したりすることができます。

- デバイスとボリューム

[構成] メニューで [デバイスとボリューム] を選択すると、詳細画面が表示されます。「デバイスとボリューム」ウィンドウには、システム上のすべての物理ディスク、ディスク・グループ、論理ボリュームが表示されるため、ボリューム・タイプやパーティションが利用可能かどうか判断することができます。

- オブジェクトの特性

[構成] メニューで [表示...] を選択すると、メイン・ウィンドウで選択し、強調表示されたオブジェクトの詳細な情報が表示されます。

通常は、[表示...] を選択することにより、後で説明するファイル・システム管理タスクの代わりとすることができます。たとえば、ドメインを強調表示にしてから [表示...] を選択すると、オブジェクト・ダイアログ・ボックスが表示され、そのドメインの作成日時とログのページ数が示されます。

ドメイン、ファイルセット、クローン、またはボリュームをオブジェクト・ツリー内でダブルクリックすることは、そのオブジェクトを強調表示にしてから [表示...] を選択することと同じです。ダブルクリック操作は、「デバイスとボリューム」ウィンドウ内のホスト、デバイス、パーティション、およびボリューム・アイコンに対しては機能しません。オブジェクト・ツリー内のオブジェクトをシングル・クリックすると、そのオブジェクトが強調表示になりますが、新しい情報は表示されません。

オブジェクト上でマウスの右ボタンをクリックすると、そのオブジェクトに対して有効なコマンドのリストが表示されます。

選択したメニュー項目が利用できない (淡色表示される) ときは、適切なオブジェクトが反転表示になっているか確認してください。たとえば、ファイルセットのクローンを作成するには、先にファイルセットを選択しておく必要があります。

### E.3.1 情報の表示単位の選択

[AdvFS] メニューから [オプション] を選択すると、ファイル・システム情報の表示をカスタマイズできます。エージェントの状態の監視間隔を除いて、すべてのオプションは GUI が再起動されるたびにリセットされます。

- 単位

ディスク・スペースの利用状況を表示する単位を選択します。512K バイト・ブロック，K バイト，M バイト，G バイト，T バイトが選択できます。

- 更新の間隔

GUI の更新間隔を指定します。

エージェントの状態の監視間隔を指定します。

- 空き容量警告

使用量 > と 空容量 < のどちらかを選択します。

利用可能なスペースの表示方法として，割合による表示と，任意の単位による表示のどちらかを選択します。

- ファイルセット・クォータ

クォータの計算方法として，スペースの割合による方法と使用量による方法のどちらかを選択します。

### E.3.2 メイン・ウィンドウ

最初にメイン・ウィンドウが表示されたときには，ホストとドメインについて，ディスク・スペースの利用状況が表示されます。[表示] メニューのオプションを選択することにより，ドメイン，ファイルセット，ボリュームをさまざまな組み合わせで表示できます。

メイン・ウィンドウの [表示] メニューで [ファイルセット・クォータ] を選択することにより，ファイルセット・クォータの情報を表示することもできます。ドメイン，ファイルセット，ボリューム情報の表示に戻すには，同じ [表示] メニューから表示したい項目を選択します。

メイン・ウィンドウには，ドメイン，ファイルセット，クローン，ボリュームに対してシステム管理タスクを実行するためのメニューがあります。メニュー・コマンドを実行すると，タスクの実行に役立つダイアログ・ボックスが呼び出されます。

### E.3.3 「デバイスとボリューム」ウィンドウ

「デバイスとボリューム」ウィンドウには，ディスク・グループと論理ボリュームが表示されます。このウィンドウでは，ディスクのサイズとタイプについて，また個々のパーティションが何に使用されているかについて識別します。この画面は，ドメインを拡大しようとするとき，および使用

するボリュームを探したいときに特に役立ちます。「デバイスとボリューム」ウィンドウの [表示] メニューでは、オブジェクトをソートでき、利用可能なボリュームのみを表示するよう選択できます。また、重複しているボリュームを表示するよう選択することもできます。

## E.4 ファイル・システム操作の実行

ドメイン、ファイルセット、AdvFS ファイルセット・クローン、およびボリュームに対して、ファイル・システム管理タスクを GUI から直接実行することができます。通常は、最初に適切なオブジェクトを選択してから作業を進めます。以下の項では、「ファイル・ドメイン」と「ドメイン」は同じ意味で使用しています。

### E.4.1 ドメインに対する操作

ドメインに対するファイル・システム管理タスクを開始するには、メイン・ウィンドウの [構成] メニューで [表示] または [ファイル・ドメイン] を選択します。また、「デバイスとボリューム」ウィンドウでも、新しいドメインの作成、および既存のドメインへのボリュームの追加を行うことができます。

#### E.4.1.1 ドメインの作成

ドメインの定義は、少なくとも 1 つのファイルセットを作成するまでは不完全です。ドメインは、マウントされたファイルセットがない限り、アクティブではありません。メイン・ウィンドウを使ってドメインを作成する手順は、次のとおりです。

1. [構成] メニューで [ファイル・ドメイン] を選択します。
2. [ファイル・ドメイン] メニューで [新規] を選択します。
3. 「新しいファイル・ドメイン」ダイアログ・ボックスでドメイン名を入力し、オブジェクト・ツリーで利用可能なボリュームを強調表示します。デバイス・リストの表示を展開するには、デバイス名の左のフォルダ・アイコンをクリックします。空き容量警告を設定することもできます。

新しいドメインにどのストレージ・デバイスを使用するか分からない場合は、次の手順に従って「デバイスとボリューム」ウィンドウ ([構成] メニュー) から作業を開始し、システム上のストレージの全体図を表示します。

1. メイン・ウィンドウの [構成] メニューで、[デバイスとボリューム] を選択します。
2. 「デバイスとボリューム」ウィンドウで、[表示] メニューから [使用可能なボリュームだけを表示] を選択します。
3. available のラベルが付いたボリュームを選択します。
4. [構成] メニューで、[新しいファイル・ドメイン] を選択します。
5. 「新しいファイル・ドメイン」ダイアログ・ボックスで、ドメインを作成し、空き容量警告を設定します。監視対象のファイル・システムが Tru64 UNIX Version 4.0E より前のバージョンの場合は、[Advanced] ボタンの使い方について、メイン・ウィンドウの説明を参照してください。

ドメインを作成すると、`/etc/fdmns` ディレクトリに自動的にエントリが生成されます。

#### E.4.1.2 ドメインの空き容量警告の設定

空き容量警告を設定しておくと、ドメイン内の空きスペースが限界値に達するか超過するとユーザに通知されます。警告の限界値を超過すると自動的にスクリプトを実行するかどうか、警告の設定時に選択できます。`/usr/advfs/daemon/scripts` に、エージェントがコマンドの実行に使用するスクリプトのサンプルがあります。

メイン・ウィンドウで空き容量警告を設定する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のドメインを強調表示にします。
2. [構成] メニューで [ファイル・ドメイン] を選択します。
3. [ファイル・ドメイン] メニューで [空き容量警告を設定] を選択します。
4. 「ファイル・ドメイン」ダイアログ・ボックスで、空き容量警告を設定します。

ドメインのサイズを変更したときは、空き容量警告を必ず再設定してください。空き容量警告の値は、ディスク・スペースの割合ではなく実際のサイズ (K バイトなど) を使って指定されます。

#### E.4.1.3 ドメインの拡大

ボリュームを追加することで、単一ボリュームのドメインをマルチボリューム・ドメインに変更したり (クラスタ構成でない限り、単一のボリュームしか持てないルート・ドメインを除く)、マルチボリュームのドメインを拡大したりできます (2.3.4 項を参照)。

ドメインにボリュームを追加する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のドメインを強調表示にします。
2. [構成] メニューで [ファイル・ドメイン] を選択します。
3. [ファイル・ドメイン] メニューで [ボリュームを追加] を選択します。
4. 「ボリュームの追加」ダイアログ・ボックスで、追加する利用可能ボリュームを強調表示にします。ボリュームの表示を展開するには、デバイス名の左のフォルダをクリックします。

どのボリュームをドメインに追加するべきか分からない場合は、次の手順に従って「デバイスとボリューム」ウィンドウから作業を開始し、システム上のストレージの全体図を表示します。

1. メイン・ウィンドウの [構成] メニューで、[デバイスとボリューム] を選択します。
2. 「デバイスとボリューム」ウィンドウで、[表示] メニューから [使用可能なボリュームだけを表示] を選択します。
3. available のラベルが付いたボリュームを選択します。
4. [構成] メニューで [ファイル・ドメインに追加] を選択します。
5. 「ボリュームの追加」ダイアログ・ボックスで、新しいボリュームを追加するドメインを選択します。

ボリュームを追加したら、ドメインのバランシングを実行するとよいでしょう。こうすると、既存のファイルが新しいボリュームに分散されます。

#### E.4.1.4 ドメインの削除

ドメインは、そのドメイン内のすべてのファイルセットとファイルセット・クローンをアンマウントした後でのみ削除できます。ファイルセットがマウントされているドメインを削除しようとすると、エラー・メッセージが表示されます。すべてのファイルセットを削除するためにドメインを削除する



と、`/etc/fdmns` ディレクトリ内のドメイン・エントリが削除されます。しかし、この方法で削除した場合、セキュリティ・ホールが存在する可能性があります。これは、ポインタのみが変更されボリューム上のデータは削除されないため、コマンド行から `salvage` ユーティリティ (6.2.6 項を参照) を使ってそのファイルセット・データにアクセスできる場合があるからです。セキュリティを確保するには、ファイルセットを個別に削除してください。

ドメインを削除すると、そのファイルセットにはアクセスできなくなります。ダイアログ・ボックスで「`/etc/fstab` を変更」オプションを選択してファイルを変更するか、あるいはコマンド行からファイルを変更するまで、削除されたドメインの名前は `/etc/fstab` ファイルから削除されません。このファイルを更新しないと、システムのリブート時、またはコマンド行から `mount` に `-a` オプションを指定して入力したときに、エラー・メッセージが表示されます。`/etc/fstab` ファイルからファイル名を削除しなかった場合、後でコマンド行から削除することができます。

ドメインを削除する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のドメインを強調表示にします。
2. [構成] メニューで [ファイル・ドメイン] を選択します。
3. [ファイル・ドメイン] メニューで [削除] を選択します。
4. 「ファイル・ドメインの削除」ダイアログ・ボックスでドメインを削除し、`/etc/fstab` ファイルをバックアップして修正します。

#### E.4.1.5 ドメイン名の変更

既存のドメインに新しい名前を付けることができます (2.3.7 項を参照)。ダイアログ・ボックスで「`/etc/fstab` を変更」オプションを選択してファイルを変更するか、あるいはコマンド行からファイルを変更するまで、元の名前は `/etc/fstab` ファイルから削除されません。`/etc/fstab` ファイルを更新しないと、システムのリブート時、またはコマンド行から `mount` コマンドを `-a` オプションを指定して入力したときに、このドメイン内のファイルセットはマウントされません。後で `/etc/fstab` を変更する場合は、コマンド行から実行してください。

ドメイン名を変更するには、その前にすべてのファイルセットをアンマウントする必要があります。ファイルセットがマウントされているドメインの名

前を変更しようとする、エラーが発生します。ドメインを既存のドメインの名前に変更することはできません。

ドメイン名の変更手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のドメインを強調表示にします。
2. [構成] メニューで [ファイル・ドメイン] を選択します。
3. [ファイル・ドメイン] メニューで [名前の変更] を選択します。
4. 「ファイル・ドメイン名の変更」ダイアログ・ボックスで新しい名前を入力し、`/etc/fstab` ファイルをバックアップして修正します。

## E.4.2 ファイルセットに対する操作

ファイルセットに対するファイル・システム管理タスクを開始するには、操作対象のファイルセットを強調表示にしてから、メイン・ウィンドウの[構成] メニューで [表示...] または [ファイルセット] を選択します。ファイルセットの操作を行うときは、必ず `/etc/fstab` の変更を選択してください。このファイルを変更しないと、それ以後の操作に対して `/etc/fstab` ファイルの内容が適正でなくなる可能性があります。`/etc/fstab` ファイルを後で更新する場合は、コマンド行から実行してください。

### E.4.2.1 ファイルセットの作成

ファイルセットを作成するには、そのファイルセットを関連付けるドメインが存在しなければなりません (E.4.1.1 項を参照)。

ファイルセットを作成する手順は、次のとおりです。

1. オブジェクト・ツリーで、ファイルセットを作成するドメインを強調表示します。
2. [構成] メニューで [ファイルセット] を選択します。
3. [ファイルセット] メニューで [新規] を選択します。
4. 「新しいファイルセット」ダイアログ・ボックスでファイルセットを作成し、マウントし、`/etc/fstab` のバックアップを取って変更を加えます。さらに、空き容量警告を設定し、必要ならファイルセット・クォータを設定します。

ファイルセットを作成する場合は、空き容量警告は利用可能なスペースの割合としてのみ設定できます。

#### E.4.2.2 ファイルセットのマウント

ファイルセットをマウントすると、そのファイルセットのファイルが利用可能になります。メイン・ウィンドウには、どのファイルセットがマウントされているかが表示されます。ドメインは、少なくとも1つのファイルセットがマウントされているときにアクティブになります。

ファイルセットをマウントする手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のファイルセットを強調表示にします。
2. [構成] メニューで [ファイルセット] を選択します。
3. [ファイルセット] メニューで [マウント] を選択します。
4. 「ファイルセットのマウント」ダイアログ・ボックスで、マウント・ポイントがない場合はそれを作成した後、`/etc/fstab` ファイルのバックアップを取って、ファイルに変更を加えます。

#### E.4.2.3 ファイルセットのアンマウント

ファイルセットをアンマウントすると、ファイルは利用できなくなります。ファイルセットを削除する場合、またはそのファイルセットが属しているドメインを削除する場合には、先にそのファイルセットをアンマウントする必要があります。メイン・ウィンドウには、ファイルセットがマウントされているかどうかが表示されます。

ファイルセットをアンマウントする手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のファイルセットを強調表示にします。
2. [構成] メニューで [ファイルセット] を選択します。
3. [ファイルセット] メニューで [アンマウント] を選択します。
4. 「ファイルセットのアンマウント」ダイアログ・ボックスでファイルセットをアンマウントし、`/etc/fstab` ファイルのバックアップを取って、ファイルに変更を加えます。

#### E.4.2.4 ファイルセットの空き容量警告の設定

空き容量警告を設定すると、ファイルセット内の空きスペースが限界値に達するか、または限界値を超過したときに警告が発行されます。警告を設定するには、先にファイルセットをマウントしておく必要があります。警告の設定を行うと、警告の限界値を超えた場合に自動的にスクリプトを実行するかどうかを選択できます。

空き容量警告を設定する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のファイルセットを強調表示にします。
2. [構成] メニューで [ファイルセット] を選択します。
3. [ファイルセット] メニューで [空き容量警告を設定] を選択します。
4. 「ファイルセット」ダイアログ・ボックスで、空き容量警告を設定します。

ファイルセットのサイズを変更した場合は、空き容量警告を必ず再設定してください。空き容量警告の値は、ディスク・スペースの割合ではなく実際のサイズ (K バイトなど) を使って評価されます。

#### E.4.2.5 ファイルセット・クォータの設定

ファイルセット・クォータを設定すると、ファイルセットが消費できるディスク・スペースの容量を制限できます。クォータを設定しない場合、すべてのファイルセットがドメイン内のすべての利用可能スペースを使用できます。クォータは、マウントされているファイルセットにのみ設定できます。

ファイルセット・クォータを設定する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のファイルセットを強調表示にします。
2. [構成] メニューで [ファイルセット] を選択します。
3. [ファイルセット] メニューで [クォータ設定] を選択します。
4. 「ファイルセット」ダイアログ・ボックスで、ファイルセット・クォータを設定します。

[表示] メニューで [ファイルセット・クォータ] を選択すると、メイン・ウィンドウでファイルセット・クォータの情報を表示できます。ディスクの使用状況の表示に戻すには、[表示] メニューで表示したい項目を選択します。

#### E.4.2.6 ファイルセットの削除

ファイルセットを削除するには、その前にアンマウントを実行する必要があります。アンマウントは、削除プロセスの一環として行うことができます。クローンを持つファイルセットは削除できません。この場合、先にクローンを削除してください。ファイルセットを削除すると、そのファイルセット内のすべてのファイルが削除されます。

すべてのファイルセットがアンマウントされている場合、すべてのファイルセットを削除する最も速い方法は、それらのファイルセットが属するドメインを削除することです。ただし、この方法で削除した場合は、salvage ユーティリティ (6.2.6 項を参照) を使ってデータにアクセスできる可能性があるため、セキュリティ・ホールが存在することになります。セキュリティが問題になる場合は、ファイルセットを個々に削除してください。

ファイルセットを削除する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のファイルセットを強調表示にします。
2. [構成] メニューで [ファイルセット] を選択します。
3. [ファイルセット] メニューで [削除] を選択します。
4. 「ファイルセットの削除」ダイアログ・ボックスで、ファイルセットをアンマウントしてから削除し、`/etc/fstab` ファイルのバックアップを取って、ファイルに変更を加えます。

#### E.4.2.7 ファイルセット名の変更

ファイルセット名を変更するには、その前にアンマウントを実行する必要があります。アンマウントは、名前変更プロセスの一部として行うことができます。新しいファイルセット名は、ドメイン内で一意でなければなりません。

ファイルセット名の変更手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のファイルセットを強調表示にします。
2. [構成] メニューで [ファイルセット] を選択します。
3. [ファイルセット] メニューで [名前の変更] を選択します。
4. 「ファイルセット」ダイアログ・ボックスでファイルセットをアンマウントし、新しい名前を入力した後、`/etc/fstab` ファイルのバックアップを取って、ファイルに変更を加えます。

---

## 注意

---

一度もマウントされたことのない (/etc/fstab ファイルにエントリがない) ファイルセット名を変更する場合は、「/etc/fstab を変更」ボックスをチェックしても、そのファイルセット名の新しいエントリは追加されません。手動で /etc/fstab ファイルを編集し、新しいエントリを追加する必要があります。

---

ファイルセット名を変更すると、そのファイルセットのクローンはファイルセットを追跡できなくなります。クローン名は変更できないので、この場合、古いクローンを削除してから新規にクローンを作成してください。

### E.4.3 AdvFS ファイルセット・クローンに対する操作

クローンに対する操作を行う場合は、操作の実行時に /etc/fstab ファイルのバックアップを取り、/etc/fstab ファイルに変更を加えることが重要です。このファイルを変更しないと、それ以後の操作で /etc/fstab ファイルの内容が適正なものでなくなる可能性があります。/etc/fstab ファイルを後で変更する場合は、コマンド行から実行してください。

#### E.4.3.1 AdvFS ファイルセット・クローンの作成

ファイルセット・クローンを作成すると、ファイル・システムがオンラインの間に、ファイルのバックアップを行うことができます (4.1.2.1 項参照)。1つのファイルセットに対して作成できるクローンの数は1つです。

クローンを作成する手順は、次のとおりです。

1. オブジェクト・ツリーで、クローンを作成するファイルセットを強調表示にします。
2. [構成] メニューで [クローン] を選択します。
3. [クローン] メニューで [新規] を選択します。
4. 「新しいクローン」ダイアログ・ボックスでクローン名を入力し、マウント・ポイントを作成した後、/etc/fstab ファイルのバックアップを取って、ファイルに変更を加えます。

#### E.4.3.2 クローンのマウント

クローンにアクセスするには、クローンをマウントする必要があります。アンマウントされたクローンはファイルセットに加えられた変更の追跡は行いますが、マウントされていないクローンを読み取ることはできません。クローンは、読み取り専用としてマウントされます。

クローンをマウントする手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のクローンを強調表示にします。
2. [構成] メニューで [クローン] を選択します。
3. [クローン] メニューで [マウント] を選択します。
4. 「クローンのマウント」ダイアログ・ボックスで、マウント・ポイントがない場合はそれを作成した後、`/etc/fstab` ファイルのバックアップを取って、ファイルに変更を加えます。

#### E.4.3.3 クローンのアンマウント

マウントされていないクローンにアクセスすることはできませんが、クローンによるファイルセットの変化の追跡は行われます。

クローンをアンマウントする手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のクローンを強調表示にします。
2. [構成] メニューで [クローン] を選択します。
3. [クローン] メニューで [アンマウント] を選択します。
4. 「クローンのアンマウント」ダイアログ・ボックスで、クローンをアンマウントした後、`/etc/fstab` ファイルのバックアップを取って、ファイルに変更を加えます。

#### E.4.3.4 クローンの削除

クローンを削除するには、その前にアンマウントを実行する必要があります。アンマウントは、削除プロセスの一環として行うことができます。

クローンを削除する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のクローンを強調表示にします。
2. [構成] メニューで [クローン] を選択します。

3. [クローン] メニューで [削除] を選択します。
4. 「クローンの削除」ダイアログ・ボックスで、クローンをアンマウントしてから削除し、`/etc/fstab` ファイルのバックアップを取って、ファイルに変更を加えます。

## E.4.4 ボリュームに対する操作

ボリュームに対するファイル・システム管理タスクを開始するには、操作の対象となるボリュームを強調表示にしてから、メイン・ウィンドウの[構成]メニューで[表示] または [ボリューム] を選択します。ボリュームを表示するには、[表示] メニューを [ドメインとボリューム] または [ドメイン、ファイルセット、ボリューム] に設定する必要があります。

### E.4.4.1 ボリュームの空き容量警告の設定

空き容量警告を設定すると、ボリューム内の空きスペースが限界値に達するか、または限界値を超過したときに通知を行います。

空き容量警告を設定する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のボリュームを強調表示にします。
2. [構成] メニューで [ボリューム] を選択します。
3. [ボリューム] メニューで [空き容量警告を設定] を選択します。
4. 「ボリューム」ダイアログ・ボックスで、空き容量警告を設定します。

限界値を超過したときに自動的にスクリプトを実行するかどうかを、警告の設定時に選択できます。

### E.4.4.2 ボリュームの追加

ボリュームの追加は、ドメインの拡大と同じです (E.4.1.3 項を参照)。ボリュームの詳細については、2.2 節を参照してください。

### E.4.4.3 ボリュームの削除

ボリュームの削除は、ドメイン内のファイルセットの論理構造を壊すことなく、いつでもドメインから実行できます (2.3.5 項を参照)。そのボリュームに保存されたデータは、ドメイン内の他のボリュームに移動されます。操作が正常に完了するためには、残りのボリューム上に空きがなければなりません。



ドメインからボリュームを削除するには、その前にドメイン内のすべてのファイルセットをアンマウントする必要があります。ボリュームに関連付けられたドメインのバランシングおよび断片化解消処理の実行中は、そのボリュームを削除できません。

ボリュームを削除する手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のボリュームを強調表示にします。
2. [構成] メニューから [ボリューム] を選択します。
3. [ボリューム] メニューから [削除] を選択します。
4. 「ボリュームの削除」ダイアログ・ボックスで、削除の確認を行います。

## E.4.5 AdvFS の保守

ファイル・システムの保守タスクを開始するには、メイン・ウィンドウで [メンテナンス] メニューを選択します。保守ツールを使ってディスク上のファイルのマッピング状態を変更することにより、読み取りおよび書き込みの性能を向上させることができます。保守ツールはシステムがオンラインのときに実行でき、その動作はシステム・ユーザとアプリケーションに対して透過的です。保守タスクは、ドメインに悪影響を与えずに中断することができます。この場合、実行済みのタスクの結果は中断後もそのまま残ります。

### E.4.5.1 マルチボリューム・ドメインのバランシング

ボリューム間でファイルを平均して分散させるには、GUI を使ってバランシング処理を開始します (5.11 節を参照)。ドメインのバランシングは、同ドメイン上で `vfast` を `-o balance=` を指定して実行している間、あるいは断片化解消、ボリュームの追加と削除、またはファイルセットの削除を実行している間は実行できません。バランシングの対象となるドメインでは、すべてのファイルセットがマウントされていなければなりません。マウントされていないファイルセットが存在するドメインでバランシングを実行しようとすると、エラー・メッセージが表示されます。

この操作は、いつでも打ち切ることができます。断片化解消中のファイルは断片化されたままになります。

ドメインのバランシングを行う手順は、次のとおりです。

1. オブジェクト・ツリーで、目的のドメインを強調表示にします。

- 2. [メンテナンス]メニューで[バランス]を選択します。
- 3. 「バランス」ダイアログ・ボックスで、バランス処理を開始します。

E.4.5.2 ドメインの断片化解消

ドメインの断片化解消を実行することにより、ドメイン内のファイルの連続性を高めることができます。また、断片化解消を実行すると空きスペースが整理統合されるため、後で作成されるファイルの断片化が少なくなります(5.9節を参照)。同一のドメイン上で `vfast` を `-o defragment=` を指定して実行している間、あるいはバランシング、ボリュームの追加と削除、またはファイルセットの削除を実行している間は、断片化解消を実行できません。断片化解消操作は、いつでも打ち切ることができます。断片化解消の対象となるドメインでは、すべてのファイルセットがマウントされていなければなりません。マウントされていないファイルセットがあるドメインで断片化解消を実行しようとすると、エラー・メッセージが表示されます。

ドメインの断片化解消を行う手順は、次のとおりです。

- 1. オブジェクト・ツリーで、目的のドメインを強調表示にします。
- 2. [メンテナンス]メニューで[断片化解消]を選択します。
- 3. 「断片化解消」ダイアログ・ボックスで、断片化解消処理を開始し、実行時間を指定します。

E.5 GUI のトラブルシューティング

表 E-3 では、GUI に関連した問題とその対処方法を示します。これ以外の情報については、`dtadvfs(8)` および `advfsd(8)` を参照してください。

表 E-3: GUI のトラブルシューティング

| 問題                                                               | 原因と処置                                 |
|------------------------------------------------------------------|---------------------------------------|
| GUI の起動が遅い。                                                      | コマンド行から、少なくとも 1 つのファイルセットをマウントしてください。 |
| 「Advanced File System」アイコンが、「アプリケーション・マネージャ・ストレージ管理」ウィンドウに存在しない。 | インストールが不完全です。                         |

表 E-3: GUI のトラブルシューティング (続き)

| 問題                                                                      | 原因と処置                                                                                                                                                                                                                                                                                             |
|-------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| コマンド行から dtadvfs を実行すると、Permission denied メッセージが表示される。                   | root でないユーザが dtadvfs を実行しました。                                                                                                                                                                                                                                                                     |
| CDE アイコンから dtadvfs を実行すると、「Password Error」ダイアログが表示される。                  | 入力した root パスワードが間違っています。再入力を行わないでください。ダイアログをキャンセルしてから、やり直してください。                                                                                                                                                                                                                                  |
| Object key is invalid または Can't get the object data というエラー・メッセージが表示される。 | GUI では、複数の動作を同時に追跡できない場合があります。GUI を再起動してください。                                                                                                                                                                                                                                                     |
| 動作が遅い、または解釈できないエラー・メッセージが表示される。                                         | <p>ネットワークの過負荷 (一定回数再試行しても GUI が構成データを取得できない)、またはシステムの過負荷 (エージェントが CUP サイクルをほとんど、または全く取得できない)。</p> <p>複数のオフライン HSZ デバイス、アンマウントされたファイルセット、スピンドウンしたディスク・デバイスがある場合は、それを disks.ignore ファイルに追加してください。これらのいずれも問題がないと思われる場合は、エージェントがハングしていると考えられます。いったん GUI を終了してエージェントを停止した後、エージェントと GUI を再起動してください。</p> |
| エラー・メッセージが正しくない。                                                        | 複数の構成タスクが同時に実行されており、複数のタスクでエラーが発生しています。                                                                                                                                                                                                                                                           |
| 空き容量警告の値が正しくない。                                                         | 再設定してください。警告の値は、容量の割合ではなく実際のサイズ (K バイトなど) を使って指定されます。                                                                                                                                                                                                                                             |
| 削除したドメインに関連付けられたボリュームが使用中と表示される。                                        | デバイス上のディスク・ラベルを手動で変更してください。コマンド行で disklabel コマンドを実行するか、CDE Disk Configuration ユーティリティを使用します。                                                                                                                                                                                                      |
| 削除したボリュームが表示される。                                                        | デバイス上のディスク・ラベルを手動で変更してください。コマンド行で disklabel コマンドを実行するか、CDE Disk Configuration ユーティリティを使用します。                                                                                                                                                                                                      |
| advfsd が CPU 時間と I/O リソースを大量に消費する。                                      | エージェントの状態の監視間隔を変更して、ポーリングの頻度を減らしてください。(E.2.1 項を参照)                                                                                                                                                                                                                                                |

表 E-3: GUI のトラブルシューティング (続き)

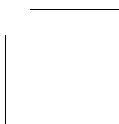
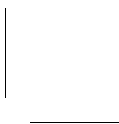
| 問題                                    | 原因と処置                                                                                                                    |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
|                                       | LSM ディスク・グループとボリュームの数を減らしてください。                                                                                          |
|                                       | マウントされていないファイルセットの数を減らしてください。アクティブでないドメインは、特に低速です。disks.ignore ファイルを使用します。(E.2.2.2 項を参照)                                 |
|                                       | これらのいずれによっても問題が解決しないと思われる場合は、エージェントがハングしていると考えられます。いったん GUI を終了してエージェントを停止した後、エージェントと GUI を再起動してください。                    |
| ブート時にシステムでパニックが発生する。                  | 不良ドメインがないか確認してください。パニックは、/etc/fstab ファイルからそのドメインのエントリが削除されていても発生します。シングルユーザ・モードで、次のいずれかの操作を実行してから、マルチユーザ・モードでリブートしてください。 |
|                                       | - ドメインを修復する。                                                                                                             |
|                                       | - 不良ドメインを、/etc/fdmns ディレクトリから別のディレクトリに移動する (修復は困難になります)。                                                                 |
| ソート結果が正しくない。                          | ソート対象のデータ・フィールドに、unavailable とマークされたものがないか確認してください。必要なら、ファイルセットをマウントしてください。                                              |
| ボリュームの追加時に Volume in use メッセージが表示される。 | 認識されていない LSM ボリュームではないか、確認してください (E.2.2.6 項を参照)。                                                                         |
| X または Motif の問題                       | これらの問題は無視しても構いません。Motif スタイル・ガイドに定義されているフル・キーボードのサポートは実装されていません。                                                         |

# F

## アプリケーション・プログラミング・インタフェース (API)

API を使用してカスタムのバックアップおよびリストア機能を構築できます。API は、次のような基本システム機能から構成されています。詳細は、それぞれのリファレンス・ページを参照してください。

- `advfs_clonefset(3)`  
アクティブなファイルセットの読み取り専用コピーを作成します。
- `advfs_get_fdmn_list(3)`  
ドメイン・リストを取得します。
- `advfs_get_file_attributes(3)`  
AdvFS ファイルの属性を取得します。
- `advfs_get_fset_list(3)`  
ドメインのすべてのファイルセットのリストを取得します。
- `advfs_get_fset_quotas(3)`  
ファイルセットのファイルセット・クォータを取得します。
- `advfs_rmfsset(3)`  
ファイルセットまたはファイルセット・クローンを削除します。
- `advfs_set_file_attributes(3)`  
AdvFS ファイル属性を設定します。
- `advfs_set_fset_quotas(3)`  
ファイルセットのファイルセット・クォータを設定します。



---

## 用語集

ここでは、AdvFS 関連のドキュメントに出てくる用語および略語について説明します。

### A

#### **AdvFS Utilities**

ライセンスによって利用できるレイヤード・プロダクト。このユーティリティを使用すると、マルチボリューム・ドメインの作成と操作ができます。

#### **advfsd**

コマンドを実行して AdvFS GUI に表示するシステム情報を入手するためのエージェント。

### B

#### **BMT (Bitfile Metadata Table)**

各ページにヘッダと各ボリューム上の mcell の配列が格納されたページの配列。BMT には、当該ボリューム上に存在する各ファイルのメタデータ (ファイル属性、ファイル・エクステント・マップ、ファイルセット属性、および POSIX ファイル統計情報を含む) が格納されます。

### C

#### **Collect**

リアルタイムの、性能監視アプリケーション。

### D

#### **dtadvfs**

AdvFS グラフィカル・ユーザ・インタフェース (GUI)。

#### **DVN (Domain Version Number)**

ドメイン・バージョン番号。ファイル構造を指定するディスク・メタデータの番号。Version 5.0 以降の Tru64 UNIX ではドメインの DVN は 4 にな

ります。それより古いバージョンの Tru64 UNIX で作成されたドメインの DVN は 3 です。

## E

**/etc/fdmns** ディレクトリ

ドメイン定義を含むディレクトリ。

**/etc/fstab** ファイル

システム・リブート時にマウントされるファイル・システムを識別するための ファイル。

## G

**GUI 更新間隔**

GUI ウィンドウ上の情報がアップデートされる時間的な間隔。

## I

**i ノード**

数値を用いたファイルの識別子。

## L

**LSM (Logical Storage Manager)**

データに冗長性を持たせ、ボリューム・レベルのストライプ化を行うストレージ管理システム。

**Logical Storage Manager**

*LSM (Logical Storage Manager)* を参照

## M

**mcell**

ファイルの特徴を表すレコードを含む、メタデータ・セル。

## N

**NetWorker**

NetWorker for Tru64 UNIX は、スケジューリングされたオンライン自動バックアップを行います。



## P

### PAK (Product Authorization Key)

弊社のソフトウェアを使用するためのライセンス。

### Product Authorization Key

*PAK (Product Authorization Key)* を参照

## U

### UBC

ユニファイド・バッファ・キャッシュ。ファイル・データと AdvFS メタデータを保持する動的に割り当てられたシステム・バッファ・キャッシュ。

## あ

### アトミック書き込みデータ・ロギング

書き込みシステム・コールで、すべてのデータ (最大 8K バイト) がディスクに書き込まれるか、あるいは全く書き込まれないかのどちらかになるようにするための機能。

## い

### 移行 (migrate)

ファイルをドメイン内のあるボリュームから別のボリュームに移動すること。

## え

### エージェント

エージェントは一般的な UNIX デーモンとして実行されます。

*advfsd* を参照

### エージェント状態監視間隔

エージェントがシステム・ディスクを走査する時間的な間隔。

### エクステント

ファイルに割り当てられたディスク・スペースの連続領域。ファイルには、0 個以上のエクステントが割り当てられています。

#### エクステント・マップ

ファイルに属するエクステントのサイズおよび位置のテーブル。単純なファイルの場合、エクステント・マップは1個です。ストライプ化されたファイルの場合は、ストライプ・セグメントごとにエクステント・マップが1個あります。

## お

#### オブジェクト

AdvFS GUI で管理されるボリューム、ドメイン、ファイルセット、ファイルセット・クローンのこと。

#### オブジェクト・ツリー

AdvFS GUI におけるオブジェクトの階層構造表示。

## か

#### 書き込み時コピー

オリジナル・ファイルのデータが変更されたときに AdvFS ファイルセット・クローンにオリジナルの情報を保存する処理。

## く

#### クォータ・ファイル

ユーザ ID またはグループ ID ごとに、クォータ制限を保存し、ファイル数、ディスク・ブロック使用状況、および猶予期間の管理を行うファイル。ファイルセット限界値情報はファイルセットのメタデータ内に記憶されます。

#### クローン

AdvFS ファイルセット・クローンは、クローンが作成されたときのファイルセット・データの読み取り専用スナップショットです。ファイルセットが更新されると、これも更新されます。AdvFS ファイルセットのクローン作成コマンドは、HSG80 コントローラの"ADD SNAPSHOT\_UNITS" コマンドに相当します。

HSG80 コントローラのクローンは、システムの完全な物理コピーであり、変更を追跡する必要はありません。

システム・クローンは、オペレーティング・システムの完全なコピーであり、これを使って別のさまざまな環境でシステムを構築することができます。

ファイルセット・クローンを参照

## こ

### ゴミ箱

ゴミ箱が設定されているディレクトリから最近削除されたファイルが置かれているディレクトリ。このディレクトリは、ユーザ・ファイル用にユーザごとに設定できます。

## さ

### 先書きログ

ファイル構造情報に対する変更が、実際の変更がディスクに書き込まれる前に、完全にトランザクション・ログ・ファイルに書き込まれるプロセス。

## す

### ストライプ

複数のストレージ領域に、データを分散させること。AdvFS は、個別のファイルをストライプ化します。LSM やハードウェア RAID では、すべてのファイルがストライプ化されます。

### ストライプ・セグメント

AdvFS では、ボリューム上に配置されたストライプ化ファイルのことです。4 つのボリュームに渡ってストライプ化されたファイルは、4 つのストライプ・セグメントを持ちます。各セグメントは別のボリュームへ移行することができます。

### ストレージ・ビットマップ

ボリューム上の割り当て済ディスク・スペースを維持管理します。

### ストレージ領域

ボリュームを参照

### スパーズ・ファイル

ページのすべてにディスク・スペースが割り当てられていないファイル。

## せ

### セーブセット

AdvFS バックアップ情報を保存するために、vdump ユーティリティが作成したひとまとまりのブロック。

### セグメント

ストライプ・セグメント を参照

## そ

### その他のメタデータ・ビットファイル

ディスク・ラベルやブート・ブロックのような AdvFS メタデータを表さない ボリュームの領域をマップするファイル。

### ソフト限界値

ディスク・ブロック使用状況、またはファイル数が 猶予期間中のみそれを超えることを認められる限界値。

## た

### ダーティ・データ

データがアプリケーションによって書き込まれているが、ファイル・システムがそれをメモリにキャッシュしていてまだディスクに書き込まれていないデータ。

### ダイレクト I/O

データをキャッシュにコピーせず、ファイルに対して同期的にデータを読み書きする I/O 方式。

### タグ

ファイルセット内にある AdvFS ファイルのユニークな識別子。

### 断片化解消

ドメイン内の各ファイルや未使用スペースの連続性を高めること。

## ち

### チェックサム

テープ・バックアップ中に作成されるエラー回復のためのブロック。

## と

### ドメイン

1 つ以上のボリュームを含む，ストレージの名前付きプール。ファイル・ドメインと呼ばれることもある。

ネットワーキングの用語では，ドメインはイントラネットまたはインターネットのサブネットワーク上の関連するシステムの集まりのことです。

Windows NT では，ドメインは，共通のディレクトリ・データベースを共有する，Windows NT Server ネットワークの管理者が定義したコンピュータの集まりです。

### ドメイン ID

システムのドメインを識別する数値の集まり。

### ドメイン・バージョン番号

*DVN (Domain Version Number)* を参照

### ドメイン・パニック

ドメインの異常が検出されたときに，ドメインにそれ以上アクセスできなくなる状態。AdvFS では，ドメイン・パニック後もドメインのファイルセットをアンマウントすることができる。

### トランザクション・ログ・ファイル

メタデータに対する変更を，実際の変更がディスクに書き込まれる前に，記録するログ・ファイル。一定の間隔で，トランザクション・ログの変更点がディスクに書き込まれます。

## は

### ハード限界値

超えることができないディスク・ブロック使用量，またはファイル数の限界値。

### バッファ・キャッシュ

ディスクから読み込まれたデータ・ブロックや，ディスクに書き込まれるのを待っているデータ・ブロックが置かれるメモリ領域。

### バランス

ドメイン内の各ボリューム間でファイルを一様に分散させること。

## ひ

### ビットファイル

AdvFS が 1 つのエンティティと見なすページのセット。予約ファイルとユーザ・ファイルはビットファイルです。

### ビットファイル・メタデータ・テーブル

*BMT (Bitfile Metadata Table)* を参照

### 非同期 I/O

ブロックしない入出力方式。データはキャッシュに書き込まれます。データのディスクへの書き込みが完了する前に制御が戻る可能性があります。

## ふ

### ファイル・エクステンツ

エクステンツ を参照

### ファイルセット

ディレクトリとファイルの階層。AdvFS ファイル・システムのディレクトリ階層のマウント可能部分を表します。

### ファイルセット ID

ファイルセットとそのドメインに関連付ける一意の識別子。

### ファイルセット・クォータ

ファイルセットが使用できるディスク・スペース量と保有できるファイル数の上限値。

### ファイルセット・クローン

クローンの作成時点における既存ファイルセットのデータと構造を保存した、ファイルセットの読み取り専用コピー。作成直後のクローンはわずかなディスク・スペースだけを使用しています。オリジナルのファイルセットに割り当てられているブロック内のデータが変更されると、オリジナル・ブロックがクローン内に保存されます。変更されるディスク・ブロックが増えるにつれて、クローンによるディスク・スペースの使用量も増加します。AdvFS ファイルセット・クローンの内容は、オリジナルのファイルセットをユーザが利用できる状態に保ったまま、バックアップすることができます。

#### ファイル・フラグメント

ファイルに割り当てられたストレージの最後のページが一部しか使用されない場合や、ファイルの合計サイズが 8K バイト以下の場合に発生します。

#### フラグ・ファイル

8K バイト (1 ページ) 以下のサイズのファイルにストレージを割り当てるためのファイル。フラグメントを使用すると、ディスク・スペースの浪費を削減できます。

#### ブロック

512 バイト単位のディスク・ストレージ。

### へ

#### ページ

8K バイトの連続するディスク・スペースの割り当て (16 ブロック)。

### ほ

#### ボリューム

AdvFS では、UNIX のブロック・デバイスと同様に動作するもののことです。ディスク、ディスク・パーティション、または論理ボリューム。

LSM ボリュームは、ディスク・グループ内のミラー、サブディスク、および LSM ディスクの階層を表すオブジェクトです。アプリケーションとファイル・システムは、LSM ボリュームに対して読み書き要求をします。

### み

#### ミラー

別のストレージ領域に同一のデータ・コピーを保存すること、あるいはそのようなコピーの 1 つ。

### め

#### メタデータ

ファイル属性、拡張領域マップ、ファイルセット属性などのファイル構成情報。

## ゆ

猶予期間

ハード限界値を超えていない限り，ソフト限界値を超えていられる期間。

ユニファイド・バッファ・キャッシュ

*UBC* を参照

## る

ルート・タグ・ファイル

ドメイン内のすべてのファイルセットの格納場所を定義するビットファイル。各ドメインに 1 つあります。

## れ

連続する領域

ディスク・ボリューム上で物理的に隣接しているストレージ。



# 索引

## A

- addvol** コマンド ..... 2-10
  - 誤った使用..... 6-31
- AdvFS GUI** エージェント ..... E-2
- AdvFS Utilities**
  - コマンド..... B-4
  - ライセンスのリストア ..... 6-28
- AdvFS Utilities** のライセンス
  - AdvFS Utilities のライセンスのリストア ..... 6-28
- AdvfsAccessMaxPercent** 属性..... 5-16
- AdvfsDomainPanicLevel** 属性..... 5-33
- advfsd** コマンド ..... E-2
- AdvfsReadyQLim** 属性 ..... 5-16
- advfsstat** コマンド ..... 5-2
- AdvfsSyncMmapPages** 属性.. 5-16
- AdvFS** の **API**..... F-1
- advscan** コマンド ..... 6-16
- API**
  - データ管理..... D-1

## B

- balance** コマンド ..... 5-27
- BMT** ..... 1-9

## C

- chfile** コマンド ..... 4-2, 5-10
- chfsets** コマンド..... 2-26, 3-18, 3-20, 6-41
- chvol** コマンド..... 2-2, 5-16, 6-43
- clonefsset** コマンド ..... 2-29
- Collect** ..... 5-2

## D

- defragment** コマンド..... 5-23
- df** コマンド ..... 3-25, 6-32
- disks.ignore** ファイル..... E-5
- DMAPI** ..... D-1
- dtadvfs** コマンド ..... E-3
- du** コマンド..... 6-32
- DVN**
  - (ドメイン・バージョン番号を参照)

## E

- edquota** コマンド ..... 3-7, 3-13
- /etc/fdmns** ディレクトリ
  - 説明 ..... 2-3
  - リストア..... 6-14
- /etc/fstab** ファイル
  - クォータ..... 3-6, 3-15

ファイルセットの指定 ..... 2-19

## F

**fdmns** ディレクトリ ..... 2-3  
**fixfdmn** コマンド ..... 6-5  
**freezefs** コマンド ..... 4-3  
**fstab** ファイル ..... 2-19

## G

### GUI

LSM ボリューム ..... E-6  
インストール ..... E-1  
エージェント ..... E-2  
許可ホスト ..... E-4  
セキュリティ ..... E-4  
ソフトウェア・サブセット .... E-1  
ディスクの無視 ..... E-5  
「デバイスとボリューム」ウィンドウ ..... E-9  
ドメイン  
    空き容量警告 ..... E-11  
    拡大 ..... E-12  
    削除 ..... E-12  
    作成 ..... E-10  
    断片化解消 ..... E-22  
    名前の変更 ..... E-13  
    バランシング ..... E-21  
    ボリュームの追加 ..... E-12  
    トラブルシューティング .... E-22  
    パスワード・ファイル ..... E-4  
    表示単位 ..... E-8  
    ファイルセット  
        空き容量警告 ..... E-16  
        アンマウント ..... E-15

クォータの設定 ..... E-16

削除 ..... E-17

作成 ..... E-14

名前の変更 ..... E-17

マウント ..... E-15

ファイルセット・クローン

アンマウント ..... E-19

削除 ..... E-19

作成 ..... E-18

マウント ..... E-19

ボリューム

空き容量警告の設定 ..... E-20

削除 ..... E-20

追加 ..... E-20

メイン・ウィンドウ ..... E-9

リモート・ファイル・システムの監視 ..... E-6

ログ・ファイル ..... E-5

**gui.passwd** ファイル ..... E-4

## H

**hosts.allow** ファイル ..... E-4

## I

### I/O

アトミック書き込み... 5-10, 6-40

ダイレクト ..... 5-15, 6-40

同期 ..... 5-9

非同期 ..... 5-9

**iostat** コマンド ..... 5-2

## L

**LSM Storage Administrator** .. E-1

**LSMSA** ..... E-1  
**ls** コマンド ..... 3-6, 6-32

## M

---

**migrate** コマンド ..... 5-29  
**mkfdmn** コマンド ..... 2-5  
誤った使用 ..... 6-31  
**mkfset** コマンド ..... 2-22  
**mktrashcan** コマンド ..... 2-34  
**mountlist** コマンド ..... B-2  
**mount** コマンド ..... 2-23

## N

---

**ncheck** コマンド ..... 3-21  
**nvbmtpg** コマンド ..... B-3  
**nvfragpg** コマンド ..... B-3  
**nvlogpg** コマンド ..... B-3  
**nvtagpg** コマンド ..... B-3

## Q

---

**quotacheck** コマンド ..... 3-23  
**quotaoff** コマンド ..... 3-16, 3-17  
**quotaon** コマンド .. 3-6, 3-7, 3-16  
**quota** コマンド ..... 3-22  
**quot** コマンド ..... 3-21

## R

---

**renamefset** コマンド ..... 2-25  
**repquota** コマンド ..... 3-24  
**rmfdmn** コマンド ..... 2-16

**rmfset** コマンド ..... 2-24  
**rmtrashcan** コマンド ..... 2-34  
**rmvol** コマンド ..... 2-14  
**rvdump** コマンド ..... 4-23  
**rvrestore** コマンド ..... 4-24

## S

---

**salvage** コマンド ..... 6-6  
**savemeta** コマンド ..... 6-2  
**shblk** コマンド ..... B-3  
**shfragbf** コマンド ..... B-3  
**showfdmn** コマンド .... 3-27, 6-32  
**showfile** コマンド . 1-7, 5-29, 6-32  
**showfsets** コマンド ..... 2-20,  
3-27, 6-32  
**shtrashcan** コマンド ..... 2-34  
**smoothsync\_age** 属性 ..... 5-16  
**stripe** コマンド ..... 5-31  
**switchlog** コマンド ..... 5-4  
**SysMan Menu** ..... A-1

## T

---

**tag2name** コマンド ..... B-2  
**thawfs** コマンド ..... 4-3

## U

---

**UBC** ..... 5-7  
**UFS**  
AdvFS からの変換 ..... C-15  
AdvFS コマンドとの比較 ..... B-5  
ファイルセットの変換先 ..... C-17

**umount** コマンド ..... 2-24  
**/usr** ファイル・システム  
    UFS から AdvFS への変換 .... C-1

## V

**vdf** コマンド ..... 3-27, 6-32  
**vdump** コマンド ..... 4-15  
**verify** コマンド ..... 6-3  
**vfast** コマンド ..... 5-17  
**vfilepg** コマンド ..... B-3  
**vrestore** コマンド ..... 4-24  
**vsbmpg** コマンド ..... B-3

## あ

空き容量警告 ..... E-9  
    ドメイン ..... E-11  
    ファイルセット ..... E-16  
    ボリューム ..... E-20  
アップグレード  
    インストレーション ..... 6-36  
    ファイルセット ..... 2-23  
アトミック書き込みデータ・ロギン  
    グ ..... 6-40  
    一時的 ..... 5-13  
    永続的 ..... 5-12  
    オン/オフのチューニング .... 5-14  
    説明 ..... 5-10  
    同期 ..... 5-11  
    非同期 ..... 5-11  
    メモリ・マッピング, ダイレクト  
        I/O ..... 6-40  
アンマウント  
    ファイルセット ..... 2-24, E-15

ファイルセット・クローン .. 2-29,  
    E-19

## い

### 移動

    AdvFS と UFS との間 ..... C-1  
    ディスク ..... 6-26  
    ファイル ..... 5-32  
    ファイルセット ..... 5-32  
インストレーション  
    アップグレード ..... 6-36  
インストール  
    AdvFS GUI ..... E-1

## え

エクステンツ ..... 1-5

## お

オブジェクト・セーフティ .... 6-41

## か

### 回復

    ゴミ箱のファイルから ..... 2-34  
    書き込み時コピー ..... 2-26

### 拡大

    ドメイン ..... 2-10, E-12  
拡張  
    ルート ..... 6-37

## き

キャッシュのチューニング ..... 5-7

## <

- クォータ ..... 3-1
  - /etc/fstab ファイル ..... 2-19, 3-6, 3-15
  - GUI による表示 ..... E-9
  - GUI の表示の変更 ..... E-9
  - グループ ..... 3-5
  - 限界値 ..... 3-2
  - 限界値の超過 ..... 6-34
  - 設定
    - ファイルセット ..... 3-18, E-16
    - 複数のファイルセット ..... 3-20
    - 複数のユーザおよびグループ ..... 3-13
    - ユーザおよびグループ ..... 3-7
  - ソフト限界値 ..... 3-2
  - タイプの比較 ..... 3-5
  - ディスク使用量の確認 ..... 3-20
  - ディスク使用量の表示 ..... 3-22
  - ハード限界値 ..... 3-2
  - ファイルセットごとの要約... 3-24
  - ファイル ..... 3-17
  - ファイル・サイズ ..... 3-6
  - ファイルセット . 3-17, E-9, E-16
  - ファイルセット所有者の確認 3-21
  - ファイルセットの開始 ..... 3-20
  - ファイルセットの解除 ..... 3-20
  - ファイルセットの削除 ..... 3-20
  - ファイルセットの適用 ..... 3-20
  - ファイルセットの適用解除... 3-20
  - ファイルセットの無効化 ..... 3-20
  - ファイルセットの有効化 ..... 3-20
  - ファイルの再配置 ..... 2-19
  - 複数のファイルセット ..... 3-20
  - 複数のユーザおよびグループ 3-13
  - 猶予期間 ..... 3-4
  - ユーザ ..... 3-5
  - ユーザおよびグループ ..... 3-7
  - ユーザおよびグループの開始
    - 手動 ..... 3-16
    - スタートアップ時 ..... 3-15
  - ユーザおよびグループの起動
    - 手動 ..... 3-16
    - スタートアップ時 ..... 3-15
  - ユーザおよびグループの削除 3-17
  - ユーザおよびグループの停止 3-17
  - ユーザおよびグループの適用解除 ..... 3-17
  - ユーザおよびグループの無効化 ..... 3-17
  - ユーザおよびグループの有効化
    - 手動 ..... 3-16
    - スタートアップ時 ..... 3-15
  - リストア ..... 4-25
- クラッシュからの復旧 ..... 6-2
- グラフィカル・ユーザ・インタフェース
  - ( GUI を参照 )
- グループ・クォータ ..... 3-5
- クローン, ファイルセット
  - アンマウント ..... 2-29, E-19
  - 削除 ..... 2-29, E-19
  - 作成 ..... 2-29, E-18
  - 名前の変更 ..... 2-30
  - バックアップ ..... 4-6
  - マウント ..... 2-29, E-19

## け

---

### 限界値

- クォータ ..... 3-2
- クォータの超過 ..... 6-34

## こ

---

### 更新間隔 ..... E-4

### 構成

- AdvFS ルート ..... 2-30
- ファイル・システム ..... 1-7

### コマンド

- AdvFS Utilities ..... B-4
- AdvFS ベース ..... B-1
- UFS との比較 ..... B-5
- ゴミ箱 ..... 2-34

## さ

---

### サイズ

- クォータ・ファイル ..... 3-6
- スペース使用状況 ..... 6-32
- ルートの拡張 ..... 6-37

### 削除

- ドメイン ..... 2-16, E-12
- ファイルセット ..... 2-24, E-17
- ファイルセット・クォータ... 3-20
- ファイルセット・クローン.. 2-29, E-19
- 不良ディスク ..... 6-34
- ボリューム ..... 2-14, E-20
- ユーザおよびグループのクォータ ..... 3-17

### 作成

- ドメイン ..... 2-5, E-10

ファイルセット ..... 2-22, E-14

ファイルセット・クローン.. 2-29, E-18

## し

---

### 修復

- ドメイン ..... 6-16

### 障害

- ハードウェア ..... 6-25

### 使用状況

- ドメイン ..... 6-32
- ファイル ..... 6-32
- ファイルセット ..... 6-32

### 使用量

- ドメイン ..... 3-20

## す

---

### ストレージ

- 選択 ..... 1-10
- ビットマップ ..... 1-9
- ストレージの割り当て ..... 1-10
- スパーズ・ファイル ..... 1-5
- スペース

壊れたディスクの使用状況の出

- 力 ..... 6-32
- ディスク使用状況 ..... 6-32

## せ

---

### 性能

- vfast による ..... 5-17
- 改善 ..... 6-42
- 監視 ..... 5-2
- 属性の変更による改善 ..... 5-16

トランザクション・ログ・ファイル  
の改善 ..... 5-4  
フラグ・ファイルの無効化による性  
能改善 ..... 5-3  
セキュリティ  
データ ..... 6-41  
ドメイン ..... 2-16, 6-41, E-12  
ファイルセット ..... 2-24, E-17  
ゼロ充填ページ ..... 6-41  
セーブセット  
圧縮 ..... 4-20  
不良 ..... 6-41  
リストア ..... 4-26  
セーブセットの圧縮 ..... 4-20

## そ

属性  
変更 ..... 5-16  
ソフト限界値 ..... 3-2  
損傷  
ルート・ドメインの修復 ..... 6-20

## た

ダイレクト I/O ..... 5-15, 6-40  
タグ ..... 3-21  
ダンプ  
エラー保護機能 ..... 4-21  
サブディレクトリ ..... 4-19  
増分 ..... 4-17  
ディスク・パーティションへ  
の ..... 4-19  
テープへ ..... 4-18

標準出力へ ..... 4-18  
ファイルへの ..... 4-19  
リモート ..... 4-23  
断片化解消 ..... 5-23, E-22  
vfast による ..... 5-17  
ファイル ..... 5-27

## ち

チューニング  
属性 ..... 5-16  
データ・キャッシュ ..... 5-7

## て

ディスク ..... 2-2  
( ボリューム も参照 )  
移動 ..... 6-26  
削除 ..... 6-34  
障害からの回復 ..... 6-25  
使用スペース .. 3-20, 6-32, 6-34  
損傷を受けた ..... 6-26  
バージョンの互換性の問題... 6-36  
パーティションへのダンプ... 4-19  
間違った使用方法の情報 ..... 6-32  
デバイス  
( ボリューム を参照 )  
「デバイスとボリューム」ウィンド  
ウ ..... E-9  
データ  
vrestore でのリストア ..... 4-24  
管理 API ..... D-1  
再使用 ..... 6-41  
セキュリティ ..... 6-41

リモートからのリストア ..... 4-23  
ロギング ..... 5-8, 6-40  
データ管理 **API** ..... D-1  
データ・キャッシュのチューニン  
グ ..... 5-7

## と

同期 **I/O** ..... 5-9  
ドメイン  
DVN3 の作成 ..... 2-8  
DVN4 の作成 ..... 2-7  
DVN4 へのアップグレード .... 2-9  
Version 5 ..... 2-6  
空き容量警告の設定 ..... E-11  
拡大 ..... 2-10, E-12  
休止 ..... 4-3  
互換性 ..... 6-36  
サイズの縮小 ..... 2-14  
削除 ..... 2-16, E-12  
作成 ..... 2-5, E-10  
修復 ..... 6-16  
ストライプ化 ..... 5-31  
スペース使用状況 ..... 6-32  
セキュリティ ..... 2-16, E-12  
説明 ..... 1-3  
損傷 ..... 6-5  
断片化解消 ..... 5-23, E-22  
ディスクの ..... 6-16  
名前の変更 ..... 2-17, E-13  
パニック ..... 6-10  
バランシング ..... 5-27, E-21  
ファイル・データの復旧 ..... 6-6  
復旧 ..... 6-2  
ボリュームの削除 ..... 2-14

ボリュームの追加 ..... 2-10, E-12  
マルチボリュームの復旧 ..... 6-28  
命名 ..... 2-3  
ルート ..... 2-30  
ルートの損傷 ..... 6-20  
ルートの名前の変更 ..... 2-31  
ドメインの拡大 ..... E-12  
ドメインの休止 ..... 4-3  
ドメインのバランシング 5-27, E-21  
ドメイン・パニック ..... 6-10  
ドメイン・バージョン番号 ..... 2-6  
トランザクション・ログ・ファイル  
互換性 ..... 6-37  
性能改善 ..... 5-4  
説明 ..... 1-4  
データ・ロギング ..... 5-8  
分離 ..... 5-4

## な

名前の変更  
ドメイン ..... 2-17, E-13  
ファイルセット ..... 2-25, E-17  
ファイルセット・クローン... 2-30

## は

破壊  
メタデータの修復 ..... 6-5  
破損  
ユーティリティが古いため... 6-36  
バックアップ  
LSM スプリット・ミラーの使  
用 ..... 4-8  
圧縮 ..... 4-20  
エラー保護機能 ..... 4-21



ゴミ箱に ..... 2-34  
 コントローラ・ベースのクローンの  
   使用 ..... 4-10  
 コントローラ・ベースのスナップ  
   ショットの使用 ..... 4-12  
 サブディレクトリ ..... 4-19  
 増分 ..... 4-17  
 ソースの作成 ..... 4-4  
 データの整合性 ..... 4-1  
 テープへ ..... 4-18  
 標準出力へ ..... 4-18  
 ファイルセット ..... 4-1  
 ファイルセット・クローンの使  
   用 ..... 4-6  
 ファイルへの ..... 4-19  
 方式 ..... 4-15  
 マウントしたファイルセットか  
   ら ..... 4-5  
 リモート ..... 4-23  
 レベル ..... 4-17  
 パニック  
   ドメイン ..... 6-10  
 バージョン番号, ドメイン ..... 2-6  
 パーティション  
   AdvFS の検索 ..... 6-16  
   再利用 ..... 6-33  
 ハード限界値 ..... 3-2

## ひ

ビットファイル・メタデータ・テー  
   ブル  
   ( BMT を参照 )  
 ビットマップ

ストレージ ..... 1-9  
 非同期 I/O ..... 5-9  
 表示  
   使用スペース ..... 3-20, 6-32  
   ドメイン ..... 2-4  
   ドメイン情報 ..... 2-4  
   ファイル ..... 5-29  
   ファイル情報 ..... 5-29

## ふ

ファイル  
   disks.ignore ..... E-5  
   gui.passwd ..... E-4  
   GUI ログ ..... E-5  
   hosts.allow ..... E-4  
   vfast による断片化解消 ..... 5-17  
   vrestore によるリストア ..... 4-25  
   移動 ..... 5-32  
   エクステンツ ..... 1-5  
   クォータ ..... 3-6, 3-17  
   ゴミ箱に移動 ..... 2-34  
   ストライプ化 ..... 5-31  
   ストレージの割り当て ..... 1-5  
   スパーズ ..... 1-5  
   断片化解消 ..... 5-27  
   復旧 ..... 6-1  
   フラグ ..... 1-5, 5-3  
   ブリアロケーション (事前割り当  
     て) ..... 1-5  
   リモートからのリストア ..... 4-23  
 ファイル・クォータ  
   ( クォータ を参照 )  
 ファイル・システム

|                                |                  |                                        |            |
|--------------------------------|------------------|----------------------------------------|------------|
| AdvFS から UFS への変換 ..           | C-15             | ファイルセット・クローンを使用し<br>たバックアップ .....      | 4-6        |
| AdvFS ルート .....                | 2-30             | マウント .....                             | 2-23, E-15 |
| GUI による表示 .....                | E-7              | 命名規則 .....                             | 2-18       |
| GUI によるリモートの監視 ...             | E-6              | 猶予期間 .....                             | 3-18       |
| UFS から AdvFS への変換              |                  | 読み取り専用でマウントされ<br>た .....               | 6-12       |
| /usr .....                     | C-1              | 読み取り専用の復旧 .....                        | 6-12       |
| ファイル・システム .....                | C-9              | ルートの名前の変更 .....                        | 2-32       |
| ルート .....                      | C-7              | ファイルセット・クローン<br>(クローン, ファイルセットを<br>参照) |            |
| 確認 .....                       | 6-3              | ファイルの移動 .....                          | 5-29       |
| 構成 .....                       | 1-7              | ファイルのストライプ化 .....                      | 5-31       |
| 構造 .....                       | 1-2              | 復旧                                     |            |
| 復旧 .....                       | 6-2              | mkfdmn または addvol から ..                | 6-31       |
| ファイルセット                        |                  | システム・クラッシュ .....                       | 6-2        |
| AdvFS から UFS への変換 ..           | C-17             | ドメイン .....                             | 6-2        |
| 空き容量警告の設定 .....                | E-16             | ファイル・データ .....                         | 6-6        |
| アップグレード .....                  | 2-23             | マルチボリューム・ドメイン                          | 6-28       |
| アンマウント .....                   | 2-24, E-15       | 読み取り専用でマウントされたファ<br>イルセット .....        | 6-12       |
| 移動 .....                       | 5-32             | ルート・ドメイン .....                         | 6-20       |
| オブジェクト・セーフティ ...               | 6-41             | フラグ・ファイル .....                         | 5-3        |
| クォータ .....                     | 3-17, E-16       | フラグメント .....                           | 1-5, 5-3   |
| クォータの適用解除 .....                | 3-20             |                                        |            |
| クローン<br>(クローン, ファイルセット<br>を参照) |                  |                                        |            |
| 削除 .....                       | 2-24, E-17       |                                        |            |
| 作成 .....                       | 2-22, E-14       |                                        |            |
| 情報の表示 .....                    | 2-20             |                                        |            |
| スペース使用状況 .....                 | 6-32             |                                        |            |
| セキュリティ ..                      | 2-24, 6-41, E-17 |                                        |            |
| 説明 .....                       | 1-3              |                                        |            |
| 属性の変更 .....                    | 2-26             |                                        |            |
| ディスク使用状況 .....                 | 6-32             |                                        |            |
| 名前の変更 .....                    | 2-25, E-17       |                                        |            |
| バックアップ .....                   | 4-1              |                                        |            |

## へ

### 変換

|                     |      |
|---------------------|------|
| AdvFS から UFS へ      |      |
| ファイルセット .....       | C-17 |
| ルート .....           | C-15 |
| UFS から AdvFS へ      |      |
| /usr ファイル・システム .... | C-1  |
| データ .....           | C-9  |

ルート・ファイル・システム ..... C-7  
ページ  
ゼロによる充填 ..... 2-26

## ほ

---

ボリューム ..... 2-2  
空き容量警告の設定 ..... E-20  
アクティブ化 ..... 6-43  
交換 ..... 2-13  
サイズの拡大 ..... 2-14  
再利用 ..... 6-33  
削除 ..... 2-14, E-20  
説明 ..... 1-3  
属性 ..... 2-2  
追加 ..... 2-10, E-20  
変更 ..... 5-32  
ボリュームの追加 ..... 2-10, E-20

## ま

---

マウント  
ファイルセット ..... 2-23, E-15  
ファイルセット・クローン.. 2-29,  
E-19

## め

---

命名  
ドメイン ..... 2-3  
ファイルセット ..... 2-18  
メタデータ  
整合性 ..... 4-3

説明 ..... 1-9  
損傷への対処 ..... 6-5  
バックアップ ..... 4-1  
保存 ..... 6-2  
メモリ・マッピング ..... 6-40

## ゆ

---

猶予期間  
設定  
ファイルセット ..... 3-19  
ユーザおよびグループ ..... 3-7  
定義 ..... 3-4  
停止 ..... 3-4  
ユーザ・クォータ ..... 3-5  
ユーティリティの互換性 ..... 6-36

## よ

---

読み取り専用のファイルセット. 6-12

## ら

---

ライセンス  
リストア ..... 6-28

## り

---

リストア  
AdvFS Utilities のライセンス 6-28  
/etc/fdmns ディレクトリ ..... 6-14  
vrestore によるファイルの ... 4-25  
クォータ ..... 4-25  
セーブセット ..... 4-26  
データ ..... 4-24

リモート ..... 4-23

## る

---

### ルート

AdvFS から UFS への変換 .. C-15

サイズの拡張 ..... 6-37

シングルユーザ・モードでのマウン  
ト ..... 2-31

損傷の修復..... 6-20

データの救済 ..... 6-9

ドメイン ..... 2-30

名前の変更..... 2-32

ファイル・システム..... 2-30

復旧 ..... 6-20

## ろ

---

ロギング ..... 5-10

### ログ

トランザクション

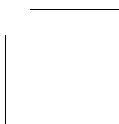
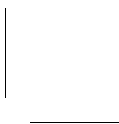
( トランザクション・ログ・  
ファイル を参照 )

## Tru64 UNIX ドキュメントの購入方法

Tru64 UNIX ドキュメントのご購入については、弊社担当営業または日本ヒューレット・パッカートの各営業所/代理店にお問い合わせください。

各ドキュメント・キットの注文番号は以下のとおりです。ドキュメント・キットに含まれるマニュアルの内容については『ドキュメント概要』を参照してください。

| キット名                                              | 注文番号        |
|---------------------------------------------------|-------------|
| Tru64 UNIX Documentation CD-ROM                   | QA-6ADAA-G8 |
| Tru64 UNIX Documentation Kit                      | QA-6ADAA-GZ |
| End User Documentation Kit                        | QA-6ADAB-GZ |
| - Startup Documentation Kit                       | QA-6ADAC-GZ |
| - General User Documentation Kit                  | QA-6ADAD-GZ |
| - System and Network Management Documentation Kit | QA-6ADAE-GZ |
| Developer's Documentation Kit                     | QA-6ADAF-GZ |
| Reference Pages Documentation Kit                 | QA-6ADAG-GZ |
| TruCluster Server Documentation Kit               | QA-6BRAA-GZ |
| Tru64 UNIX 日本語ドキュメント・キット                          | QA-6ADJB-GZ |
| スタートアップ・ドキュメント・キット                                | QA-6ADJC-GZ |
| 一般ユーザ・ドキュメント・キット                                  | QA-6ADJD-GZ |
| システム/ネットワーク管理ドキュメント・キット                           | QA-6ADJE-GZ |
| プログラミング・ドキュメント・キット                                | QA-6ADJF-GZ |
| CDE 翻訳ドキュメント・キット                                  | QA-6ADJG-GZ |
| TruCluster Server 日本語ドキュメント・キット                   | QA-05SJA-GZ |
| Advanced Server for UNIX 日本語ドキュメント・キット            | QA-5U2JA-GZ |



# マニュアルに対するご意見

Tru64 UNIX  
AdvFS 管理ガイド  
AA-RK3TD-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