

Oracle Solaris の管理: デバイスとファイルシステム

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS:

Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

このソフトウェアもしくはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアもしくはハードウェアは、危険が伴うアプリケーション（人的傷害を発生させる可能性があるアプリケーションを含む）への用途を目的として開発されていません。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用する際、安全に使用するために、適切な安全装置、バックアップ、冗長性（redundancy）、その他の対策を講じることは使用者の責任となります。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはOracle Corporationおよびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

Intel、Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD、Opteron、AMDロゴ、AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

はじめに	19
1 リムーバブルメディアの管理(概要/タスク)	21
リムーバブルメディアの管理(概要)	21
リムーバブルメディアの機能と利点	21
手動によるマウントと自動マウントの比較	23
リムーバブルメディアへのアクセスの概要	24
リムーバブルメディアの管理(タスク)	25
リムーバブルメディアに関する考慮事項	26
フロッピーディスクのフォーマット	27
▼リムーバブルメディアを読み込む方法	27
▼フロッピーディスクをフォーマットする方法 (rmformat)	28
▼リムーバブルメディア上にファイルシステムを作成する方法	29
▼DVD-RAM 上にファイルシステムを作成する方法	30
▼リムーバブルメディア上のファイルシステムを検査する方法	30
▼リムーバブルメディア上の不良ブロックを修復する方法	31
リムーバブルメディアに読み取り/書き込み保護とパスワードによる保護を適用する	31
▼リムーバブルメディアの書き込み保護を有効または無効にする方法	32
▼リムーバブルメディアの読み取り/書き込み保護を有効または無効にする方法とパスワードを設定する方法	32
リムーバブルメディアへのアクセス	33
リムーバブルメディア名の使用	33
リムーバブルメディア上のデータにアクセスするためのガイドライン	34
▼新しいリムーバブルメディアドライブを追加する方法	34
▼リムーバブルメディアサービスを無効または有効にする方法	35
▼リムーバブルメディア上の情報にアクセスする方法	35
▼リムーバブルメディアが使用中かどうかを調べる方法	36

▼ リムーバブルメディアを取り出す方法	36
リモートシステム上のリムーバブルメディアへのアクセス	37
▼ ローカルのメディアをほかのシステムで使用可能にする方法	37
▼ リモートシステム上のリムーバブルメディアにアクセスする方法	39
2 CD および DVD への書き込み (タスク)	41
オーディオ CD、データ CD、データ DVD の取り扱い	41
CD/DVD メディアに関するよく使われる用語	42
データ CD、データ DVD、オーディオ CD への書き込み	43
RBAC を使用してリムーバブルメディアへのユーザーアクセスを制限する	45
▼ RBAC を使用してリムーバブルメディアへのユーザーアクセスを制限する方 法	45
▼ CD または DVD ライターを確認する方法	46
▼ CD または DVD メディアを検査する方法	46
データ CD またはデータ DVD を作成する	47
▼ データ CD またはデータ DVD 用の ISO 9660 ファイルシステムを作成する方法 ..	47
▼ マルチセッションのデータ CD を作成する方法	48
オーディオ CD を作成する	49
▼ オーディオ CD を作成する方法	50
▼ オーディオトラックを CD から抽出する方法	51
▼ CD をコピーする方法	52
▼ CD-RW メディアを消去する方法	52
3 デバイスの管理 (タスク)	55
デバイス管理の新機能	55
USB 3.0 デバイスのサポート	55
追加デバイス管理タスクの参照先	56
Oracle Solaris のデバイスの管理	56
x86: デバイスサポートの確認	56
デバイスドライバについて	57
デバイスの自動構成	57
デバイス構成情報の表示	59
障害のあるデバイスの解決	63
システムへ周辺デバイスを追加する	65
▼ 周辺デバイスを追加する方法	66

▼ デバイスドライバを追加する方法	67
デバイスへのアクセス	68
デバイス情報が作成される方法	68
デバイスの管理方法	68
デバイス名の命名規則	69
論理ディスクデバイス名	69
論理テープデバイス名	72
論理リムーバブルメディアデバイス名	72
4 デバイスの動的構成(タスク)	73
動的再構成とホットプラグ機能	74
接続点	75
PCI または PCIe アダプタカードの取り外し	77
PCI または PCIe アダプタカードの取り付け	77
hotplug コマンドによる PCIe ホットプラグ	77
cfgadm コマンドによる SCSI ホットプラグ(タスクマップ)	79
cfgadm コマンドによる SCSI ホットプラグ	80
▼ SCSI デバイスに関する情報を表示する方法	81
▼ SCSI コントローラの構成を解除する方法	81
▼ SCSI コントローラを構成する方法	82
▼ SCSI デバイスを構成する方法	82
▼ SCSI コントローラを切り離す方法	83
▼ SPARC: SCSI コントローラを接続する方法	84
▼ SPARC: SCSI デバイスを SCSI バスに取り付ける方法	85
▼ SPARC: SCSI コントローラ上の同一デバイスを交換する方法	86
▼ SPARC: SCSI デバイスを取り外す方法	87
SCSI 構成に関する問題のトラブルシューティング	89
▼ 失敗した SCSI 構成解除操作の解決方法	90
cfgadm コマンドによる PCI または PCIe ホットプラグ(タスクマップ)	90
cfgadm コマンドによる PCI または PCIe ホットプラグ	91
PCIe LED インジケータの動作	91
▼ PCI スロット構成情報を表示する方法	92
▼ PCI アダプタカードを取り外す方法	93
▼ PCI アダプタカードを取り付ける方法	95
PCI 構成に関する問題のトラブルシューティング	96

cfgadm コマンドによる SATA ホットプラグ	97
▼ SATA デバイスの構成を解除する方法	97
▼ SATA デバイスを構成する方法	98
Reconfiguration Coordination Manager (RCM) スクリプトの概要	98
RCM スクリプトについて	99
RCM スクリプトで実行できること	99
RCM スクリプト処理の動作方法	99
RCM スクリプトでのタスク	100
アプリケーション開発者 RCM スクリプト (タスクマップ)	100
システム管理者 RCM スクリプト (タスクマップ)	101
RCM スクリプトに名前を付ける	102
RCM スクリプトのインストールまたは削除	102
▼ RCM スクリプトのインストール方法	103
▼ RCM スクリプトの削除方法	103
▼ RCM スクリプトのテスト方法	103
テープバックアップ用の RCM スクリプトの例	104
5 USB デバイスの管理 (タスク)	109
USB デバイスの新機能	109
USB 3.0 のサポート	109
Oracle Solaris の USB サポートについて	110
サポートされる USB 機能	110
USB デバイスの機能および互換性の問題	112
バス電源供給方式のデバイス	112
USB キーボードとマウス	113
USB ホストコントローラとハブ	114
USB ハブデバイス	114
SPARC: USB 電源管理	115
USB ケーブルを接続するためのガイドライン	116
USB デバイスの概要	116
よく使用される USB 関連の略語	116
Oracle Solaris USB アーキテクチャー	117
USB バスの説明	118
USB デバイスおよびドライバクラス	120
USB 大容量ストレージデバイスの管理	120

USB フロッピーディスクデバイスの使用	122
USB 大容量ストレージデバイスのホットプラグ	123
USB 大容量ストレージデバイスを使用するための準備	126
▼ USB デバイス情報を表示する方法	127
▼ USB 大容量ストレージデバイス上にファイルシステムを作成する方法	128
▼ USB 大容量ストレージデバイス上でパーティションを変更し、PCFS ファイルシステムを作成する方法	130
▼ USB 大容量ストレージデバイス上に Solaris パーティションを作成してスライスを変更する方法	133
▼ USB 大容量ストレージデバイスをマウントまたはマウント解除する方法	135
USB 大容量ストレージデバイスの問題のトラブルシューティング	137
特定の USB ドライバを無効にする	137
▼ 特定の USB ドライバを無効にする方法	138
▼ 使用されていない USB デバイスのリンクを削除する方法	138
USB オーディオデバイスの使用	139
複数の USB オーディオデバイスのホットプラグ	140
▼ USB オーディオデバイスを追加する方法	140
▼ システムのプライマリオーディオデバイスを識別する方法	141
▼ プライマリ USB オーディオデバイスを変更する方法	142
USB オーディオデバイスに関する問題のトラブルシューティング	142
cfgadm コマンドを使った USB デバイスのホットプラグ	143
▼ USB バス情報を表示する方法 (cfgadm)	144
▼ USB デバイスの構成を解除する方法	145
▼ USB デバイスの構成方法	145
▼ 論理的に USB デバイスを接続解除する方法	146
▼ 論理的に USB デバイスを接続する方法	146
▼ 論理的に USB デバイスのサブツリーを接続解除する方法	147
▼ USB デバイスのリセット方法	147
▼ 複数の構成を持つ USB デバイスのデフォルト構成を変更する方法	147
6 InfiniBand デバイスの使用 (概要/タスク)	149
InfiniBand デバイスの概要	149
IB デバイスの動的再構成 (タスクマップ)	151
IB デバイスの動的再構成 (cfgadm)	152
▼ IB デバイス情報を表示する方法	153
▼ IOC デバイスの構成を解除する方法	154

▼ IOC デバイスを構成する方法	155
▼ IB Port、HCA_SVC、VPPA デバイスの構成を解除する方法	155
▼ IB Port、HCA_SVC、VPPA デバイスを構成する方法	156
▼ IB 擬似デバイスの構成を解除する方法	156
▼ IB 擬似デバイスを構成する方法	157
▼ HCA のカーネル IB クライアントを表示する方法	157
▼ HCA に接続されている IB デバイスの構成を解除する方法	158
IB HCA を構成する方法	159
▼ IB p_key テーブルを更新する方法	159
▼ IB 通信サービスを表示する方法	159
▼ VPPA 通信サービスを追加する方法	159
▼ 既存の IB Port、HCA_SVC、VPPA 通信サービスを削除する方法	160
▼ IOC の構成を更新する方法	161
InfiniBand デバイスでの uDAPL アプリケーションインタフェースの使用	161
▼ uDAPL を使用可能にする方法	162
DAT 静的レジストリの更新	163
7 ディスクの管理 (概要)	165
ディスク管理の新機能	165
Advanced Format ディスクのサポート	165
ディスク管理タスクについての参照先	166
ディスク管理の概要	167
ディスク関連の用語	167
ディスクラベルについて	168
EFI (GPT) ディスクラベル	168
ディスクスライスについて	171
format ユーティリティ	172
ディスクをパーティションに分割する	176
パーティションテーブル関連の用語	176
パーティションテーブル情報の表示	177
free hog スライスの使用方法	179
8 ディスク使用の管理 (タスク)	181
ディスク使用の管理 (タスクマップ)	181
ファイルとディスク容量の情報の表示	182

▼ファイルとディスク容量の情報を表示する方法	183
ファイルサイズの確認	185
▼ファイルサイズを表示する方法	186
▼サイズの大きなファイルを見つける方法	187
▼指定されたサイズ制限を超えるファイルを見つける方法	188
ディレクトリサイズの確認	189
▼ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する方法	189
▼ローカルUFS ファイルシステムのユーザーの所有状況を表示する方法	190
古いファイルまたは使用されていないファイルの検索と削除	191
▼最新ファイルのリストを表示する方法	191
▼古いファイルと使用されていないファイルを見つけて削除する方法	192
▼一時ディレクトリを一度にクリアする方法	193
▼コアファイルを見つけて削除する方法	194
▼クラッシュダンプファイルを削除する方法	194
9 ディスクの管理(タスク)	197
ディスクの管理(タスクマップ)	197
システム上のディスクの確認	198
▼システム上のディスクを確認する方法	198
ディスクのフォーマット	200
▼ディスクがフォーマット済みかを調べる方法	201
▼ディスクをフォーマットする方法	201
ディスクスライスの表示	203
▼ディスクスライス情報を表示する方法	203
ディスクラベルの作成と検査	205
▼ディスクラベルを作成する方法	205
▼ディスクラベルを検査する方法	210
破損したディスクラベルの復元	212
▼破損したディスクラベルを復元する方法	212
他社製のディスクの追加	214
10 SPARC: ディスクの設定(手順)	217
SPARC: UFS ファイルシステム用のディスクの設定(タスクマップ)	217
SPARC: UFS ファイルシステム用のディスクの設定	218
▼SPARC: UFS ルートファイルシステム用のディスクを設定する方法	219

▼ SPARC: UFS ファイルシステム用のディスクを設定する方法	219
▼ SPARC: UFS ファイルシステム用のディスクスライスを作成する方法	220
▼ SPARC: UFS ファイルシステムを作成する方法	225
▼ SPARC: UFS ルートファイルシステムのブートブロックをインストールする方 法	226
SPARC: ZFS ファイルシステム用のディスクの設定 (タスクマップ)	226
SPARC: ZFS ファイルシステム用のディスクの設定	227
▼ SPARC: ZFS ルートファイルシステム用のディスクを設定する方法	228
SPARC: ZFS ルートファイルシステム用のディスクスライスの作成	229
▼ SPARC: ZFS ルートファイルシステム用のディスクスライスを作成する方法	229
▼ SPARC: ZFS ルートファイルシステムのブートブロックをインストールする方 法	233
▼ SPARC: ZFS ファイルシステム用のディスクを設定する方法	234
11 x86: ディスクの設定 (手順)	237
x86: UFS ファイルシステム用のディスクの設定 (タスクマップ)	237
x86: UFS ファイルシステム用のディスクの設定	238
x86: ZFS ファイルシステム用のディスクの設定 (タスクマップ)	243
x86: ZFS ファイルシステム用のディスクの設定	244
Solaris fdisk パーティションの作成および変更	253
x86: fdisk パーティションの作成上のガイドライン	253
▼ x86: Solaris fdisk パーティションを作成する方法	253
fdisk パーティション識別子の変更	256
▼ Solaris fdisk 識別子を変更する方法	256
12 Oracle Solaris iSCSI ターゲットの構成 (タスク)	259
Oracle Solaris iSCSI の技術 (概要)	259
Solaris iSCSI のソフトウェア要件およびハードウェア要件の識別	260
Solaris iSCSI ターゲットデバイスの構成 (タスク)	261
Oracle Solaris の iSCSI 用語	261
動的または静的ターゲット発見の構成	262
Solaris iSCSI ターゲットデバイスの構成と管理	263
iSCSI ベースのストレージネットワークにおける認証の構成	268
▼ iSCSI イニシエータの CHAP 認証を構成する方法	269
▼ iSCSI ターゲットの CHAP 認証を構成する方法	270

他社製の RADIUS サーバーを使用して iSCSI 構成内の CHAP 管理を単純化する	271
Solaris iSCSI マルチパスデバイスの設定	272
▼ ターゲットの複数の iSCSI セッションを有効にする方法	273
iSCSI 構成の監視	276
▼ iSCSI 構成の監視	276
iSCSI イニシエータと iSCSI ターゲットのパラメータの変更	278
▼ iSCSI イニシエータと iSCSI ターゲットのパラメータを変更する方法	279
iSCSI 構成に関する問題のトラブルシューティング	281
ローカルシステムから iSCSI ターゲットに接続できない	282
ローカルシステム上で iSCSI デバイスまたは iSCSI ディスクが利用できない	283
iSNS 発見方式を使用する場合の LUN マスクの使用	283
iSCSI の一般的なエラーメッセージ	284
13 format ユーティリティー (参照情報)	289
format ユーティリティーを使用する上での推奨事項および要件	289
format のメニューとコマンドの説明	290
partition メニュー	292
x86: fdisk メニュー	293
analyze メニュー	294
defect メニュー	295
format コマンドへの入力規則	296
format コマンドへ番号を指定する	296
format のコマンド名を指定する	296
format コマンドへディスク名を指定する	297
format ユーティリティーのヘルプを利用する	297
14 ファイルシステムの管理 (概要)	299
ファイルシステム管理タスクについての参照先	299
ファイルシステムの概要	300
Oracle Solaris ファイルシステムのタイプ	300
デフォルトの Oracle Solaris ファイルシステム	306
ファイルシステムのマウントおよびマウント解除の概要	307
マウントされたファイルシステムテーブル	308
仮想ファイルシステムテーブル	308

NFS 環境	309
自動マウント (autofs)	310
ファイルシステムのタイプを調べる	311
ファイルシステムのタイプを調べる方法	311
15 ファイルシステムの作成およびマウント(タスク)	313
Oracle Solaris ファイルシステムの作成	313
ZFS ファイルシステムの作成	313
一時ファイルシステムの作成	314
LOFS ファイルシステムの作成	314
Oracle Solaris ファイルシステムのマウントおよびマウント解除	315
/etc/vfstab ファイルのフィールドの説明	316
Oracle Solaris ファイルシステムをマウント解除するための前提条件	318
Oracle Solaris ファイルシステムの作成およびマウント	319
▼ ZFS ファイルシステムの作成方法	319
▼ レガシー UFS ファイルシステムを作成およびマウントする方法	319
▼ TMPFS ファイルシステムを作成およびマウントする方法	321
▼ LOFS ファイルシステムを作成およびマウントする方法	322
▼ /etc/vfstab ファイルにエントリを追加する方法	323
▼ 1つのファイルシステムをマウントする方法 (/etc/vfstab ファイル)	324
▼ NFS ファイルシステムのマウント方法 (mount コマンド)	325
▼ x86: ハードディスクから PCFS (DOS) ファイルシステムをマウントする方法 (mount コマンド)	326
▼ ファイルシステムを使用中のすべてのプロセスを終了させる方法	327
▼ 1つのファイルシステムをマウント解除する方法	328
16 追加スワップ空間の構成(タスク)	331
スワップ空間について	331
スワップ空間と仮想メモリー	331
スワップ空間と TMPFS ファイルシステム	332
スワップ空間と動的再構成	333
SAN 環境でのスワップ空間の構成	333
スワップ空間の追加が必要かどうかを調べる方法	334
スワップ関連のエラーメッセージ	334
TMPFS 関連のエラーメッセージ	334

スワップ空間の割り当て方法	335
スワップ空間と /etc/vfstab ファイル	335
スワップ空間の計画	335
ZFS ベースのシステムでのスワップ空間の割り当て	336
スワップリソースの監視	337
Oracle Solaris ZFS ルート環境でのスワップ空間の追加または変更	338
▼ Oracle Solaris ZFS ルート環境でスワップ空間を追加する方法	339
17 UFS ファイルシステムの整合性検査 (手順)	341
UFS ファイルシステムの整合性	342
UFS ファイルシステムユーティリティー (fsck、mkfs、および newfs) の拡張機能	342
UFS ファイルシステムの状態はどのように記録されるか	349
fsck コマンドで検査して修復される内容	350
UFS ファイルシステムの不整合が発生する理由	350
整合性が検査される UFS コンポーネント	351
fsck サマリーメッセージ	356
UFS ファイルシステムを対話式で検査して修復する	357
▼ 代替ブートデバイスから UFS ルート (/) ファイルシステム、/usr ファイルシステム、または /var ファイルシステムを検査する方法	358
▼ ルート (/)、/usr、または /var 以外の UFS ファイルシステムを検査する方法	360
UFS ファイルシステムの修復	362
▼ UFS ファイルシステムを修復する方法	362
fsck コマンドで修復できない UFS ファイルシステムの修正	363
不正な UFS スーパーブロックの復元	364
▼ 不正な UFS スーパーブロックを復元する方法 (Solaris 10 6/06 リリース)	365
▼ 不正なスーパーブロックを復元する方法 (Solaris 8、9、および 10 リリース)	368
fsck コマンドの構文とオプション	369
18 UFS ファイルシステム (参照情報)	371
UFS ファイルシステムのシリンダグループの構造	371
ブートブロック	372
スーパーブロック	372
i ノード	372
データブロック	374

空きブロック	374
UFS ファイルシステムパラメータのカスタマイズ	375
論理ブロックサイズ	375
フラグメントサイズ	376
最小空き容量	376
回転待ち	377
最適化のタイプ	377
i ノード数 (ファイルの数)	377
UFS ファイルとファイルシステムの最大サイズ	378
UFS サブディレクトリの最大数	378
19 UFS ファイルシステムのバックアップと復元 (概要/タスク)	379
UFS ファイルシステムのバックアップと復元の概要	379
ファイルシステムをバックアップする理由	380
どの UFS ファイルシステムをバックアップするか計画	381
バックアップタイプの選択	383
テープデバイスの選択	384
UFS ファイルシステムのバックアップおよび復元の概要 (タスクマップ)	384
バックアップスケジュールに関するその他の注意事項	386
バックアップスケジュールを設定する際のガイドライン	387
ダンプレベルを使用して増分バックアップを作成する	389
バックアップスケジュールの例	390
UFS ファイルとファイルシステムのバックアップ (タスクマップ)	396
ファイルシステムバックアップの実行準備	397
▼ UFS ファイルシステム名を検索する方法	397
▼ 完全バックアップに必要なテープ数を決定する	398
UFS ファイルシステムのバックアップ	398
▼ UFS ファイルシステムのバックアップをテープに作成する方法	399
UFS ファイルおよびファイルシステムのバックアップを復元する (タスクマップ)	404
UFS ファイルとファイルシステムを復元するための準備	405
UFS ファイルシステム名の確認	406
必要なテープデバイスのタイプの決定	406
テープデバイス名の決定	406
UFS ファイルとファイルシステムの復元	406
▼ 使用するテープを決定する方法	407

▼ 対話式で UFS ファイルを復元する方法	408
▼ 特定の UFS ファイルを復元する方法	410
▼ UFS ファイルシステム全体を復元する方法	412
▼ UFS ルート (/) および /usr ファイルシステムを復元する方法	416
20 UFS スナップショットの使用 (手順)	419
UFS スナップショットの使用 (タスクマップ)	419
UFS スナップショットの概要	420
なぜ UFS スナップショットを使用するか	421
UFS スナップショットのパフォーマンス上の問題	422
UFS スナップショットの作成と削除	422
マルチテラバイト UFS のスナップショットの作成	423
▼ UFS スナップショットを作成する方法	424
▼ UFS スナップショットの情報を表示する方法	425
UFS スナップショットの削除	426
▼ UFS スナップショットを削除する方法	426
UFS スナップショットのバックアップ	427
▼ UFS スナップショットの完全バックアップの作成方法 (ufsdump)	427
▼ UFS スナップショットの増分バックアップの作成方法 (ufsdump)	428
▼ UFS スナップショットのバックアップ方法 (tar)	428
UFS スナップショットのバックアップからのデータの復元	429
21 ファイルとファイルシステムのコピー (タスク)	431
ファイルシステムをコピーするためのコマンド	431
cpio を使用してファイルシステム間でディレクトリをコピーする	433
▼ ファイルシステム間でディレクトリをコピーする方法 (cpio)	433
ファイルとファイルシステムをテープにコピーする	434
tar を使用してファイルをテープにコピーする	435
▼ ファイルをテープにコピーする方法 (tar)	435
▼ テープ上のファイルのリストを表示する方法 (tar)	436
▼ テープからファイルを取り出す方法 (tar)	436
pax コマンドを使用してファイルをテープにコピーする	438
▼ ファイルをテープにコピーする方法 (pax)	438
cpio コマンドを使用してファイルをテープにコピーする	439
▼ ディレクトリ内のすべてのファイルをテープにコピーする方法 (cpio)	439

▼ テープ上のファイルのリストを表示する方法 (cpio)	440
▼ テープからすべてのファイルを取り出す方法 (cpio)	441
▼ テープから特定のファイルを取り出す方法 (cpio)	442
ファイルをリモートテープデバイスにコピーする	443
▼ ファイルをリモートテープデバイスにコピーする方法 (tar と dd)	443
▼ ファイルをリモートテープデバイスから抽出する方法	444
22 テープドライブの管理 (タスク)	445
使用するメディアの選択	445
バックアップデバイス名	446
テープドライブの巻き戻しオプションを指定する	447
テープドライブに別の密度を指定する	448
テープドライブのステータスを表示する	448
▼ テープドライブのステータスを表示する方法	448
磁気テープカートリッジの取り扱い	449
磁気テープカートリッジのたるみを直す	449
磁気テープカートリッジを巻き戻す	450
ドライブの管理とメディア処理のガイドライン	450
23 UFS バックアップおよび復元コマンド (参照情報)	451
ufsdump コマンドの機能	451
デバイス特性の判断	451
メディアの終わりの検出	452
ufsdump コマンドを使用したデータのコピー	452
/etc/dumpdates ファイルの目的	452
バックアップデバイス (dump-file) 引数	453
バックアップを作成するファイルを指定する	455
テープの性質を指定する	455
ufsdump コマンドの制限	455
ufsdump コマンドオプションおよび引数の指定	456
ufsdump のデフォルトオプション	456
ufsdump とセキュリティーに関する注意事項	456
ufsrestore オプションおよび引数の指定	457

索引 459

はじめに

『システムの管理: デバイスとファイルシステム』は、Oracle Solaris システム管理に関する重要な情報が含まれるセットの一部です。このガイドには、SPARCおよびx86の両方のシステムに関する情報が含まれています。

本書は、読者が次のタスクを終了済みであることを前提としています。

- インストールされている Oracle Solaris ソフトウェア
- 使用する予定のすべてのネットワークソフトウェアを設定済み

システム管理者にとって重要な Oracle Solaris の新機能については、該当する章にある新機能に関するセクションを参照してください。

注 - この Oracle Solaris のリリースでは、SPARC および x86 系列のプロセッサアーキテクチャーをサポートしています。サポートされるシステムは、<http://www.oracle.com/webfolder/technetwork/hcl/index.html> の『Oracle Solaris Hardware Compatibility List』に記載されています。本書では、プラットフォームにより実装が異なる場合は、それを特記します。

対象読者

このドキュメントは、本リリースが稼働しているシステムの管理者を対象としています。このマニュアルを読むには、UNIX のシステム管理について1-2年の経験が必要です。UNIX システム管理のトレーニングコースに参加することも役に立ちます。

Oracle サポートへのアクセス

Oracle のお客様は、My Oracle Support を通じて電子的なサポートを利用することができます。詳細は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> を参照してください。聴覚に障害をお持ちの場合は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> を参照してください。

表記上の規則

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

表 P-1 表記上の規則

字体または記号	意味	例
<code>AaBbCc123</code>	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を使用してすべてのファイルを表示します。 - <code>machine_name% you have mail.</code>
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	<code>machine_name% su</code> <code>Password:</code>
<i>AaBbCc123</i>	コマンド行プレースホルダ: 実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、 rm <i>filename</i> と入力します。
<i>AaBbCc123</i>	参照する書名を示します。	『ユーザーズガイド』の第6章を参照してください。 これらのオプションを「 <i>class</i> 」オプションと呼びます。 参照する章、セクション、ボタンやメニュー名、強調する単語を表示します。

コマンド例のシェルプロンプト

次の表に、C シェル、Bourne シェル、および Korn シェルのデフォルトのシステムプロンプト、およびルート (管理者) のプロンプトを示します。

表 P-2 シェルプロンプト

シェル	プロンプト
C シェル	<code>machine_name%</code>
C シェルのルートプロンプト	<code>machine_name#</code>
Bourne シェルおよび Korn シェル	<code>\$</code>
Bourne シェルおよび Korn シェルのルートプロンプト	<code>#</code>

リムーバブルメディアの管理 (概要/タスク)

この章では、Oracle Solaris OS でリムーバブルメディアを管理し、使用方法について説明します。

この章の内容は次のとおりです。

- 21 ページの「リムーバブルメディアの管理 (概要)」
- 25 ページの「リムーバブルメディアの管理 (タスク)」
- 33 ページの「リムーバブルメディアへのアクセス」
- 37 ページの「リモートシステム上のリムーバブルメディアへのアクセス」

リムーバブルメディアの管理 (概要)

このセクションでは次の情報について説明します。

- 21 ページの「リムーバブルメディアの機能と利点」
- 23 ページの「手動によるマウントと自動マウントの比較」
- 24 ページの「リムーバブルメディアへのアクセスの概要」

リムーバブルメディアの機能と利点

Oracle Solaris リリースには、ユーザーとソフトウェア開発者用に、リムーバブルメディアを扱うための標準インタフェースが用意されています。リムーバブルメディアサービスには、次の利点があります。

- リムーバブルメディアを自動的にマウントします。手動によるマウントと自動マウントの比較については、次のセクションを参照してください。
- 管理者でなくても、リムーバブルメディアにアクセスできるようになります。
- ネットワーク上のほかのシステムがローカルシステム上のリムーバブルメディアに自動的にアクセスできるようになります。詳細は、37 ページの「リモートシステム上のリムーバブルメディアへのアクセス」を参照してください。

vold はサービス管理機能 (Service Management Facility、SMF) により管理される

ボリューム管理デーモン vold はサービス管理機能 (SMF) によって管理されます。つまり、該当する場合は、svcadm disable コマンドを使用して、次の新しい volfs サービスを無効にすることができます。

```
# svcadm disable volfs
```

次のコマンドを使用して、volfs サービスのステータスを識別できます。

```
$ svcs volfs
STATE          STIME      FMRI
online         Sep_29    svc:/system/filesystem/volfs:default
```

詳細は、[smf\(5\)](#) のマニュアルページを参照してください。

svccfg コマンドを使用して、追加の vold プロパティーを表示および設定できます。たとえば、問題の解決を容易にするために、vold ロギングを一時的に有効にできます。例:

```
# svccfg
svc:> select system/filesystem/volfs
svc:/system/filesystem/volfs> setprop vold/log_debuglevel=3
svc:/system/filesystem/volfs> exit
# svcadm disable volfs
# svcadm enable volfs
```

svccfg コマンドを使用して、設定可能な vold プロパティーのリストを表示することもできます。

```
# svccfg
svc:> select volfs
svc:/system/filesystem/volfs> listprop vold/*
vold/config_file          astring
vold/log_debuglevel       count      3
vold/log_file             astring
vold/log_nfs_trace        boolean    false
vold/log_verbose          boolean    false
vold/root_dir             astring
vold/never_writeback_label boolean    false
svc:/system/filesystem/volfs> exit
```

これらのプロパティーの説明については、[vold\(1M\)](#) のマニュアルページを参照してください。

ボリューム管理 (vold) の改善

メディアを含まないデバイスのデバイスリンクが、次のように適切に作成されます。

```
lrwxrwxrwx  1 root    root      28 Jun 13 13:09 /vol/dev/aliases/cdrom0 ->
/vol/dev/rdisk/c2t2d0/nomedia
```

voldの実行中に、`cdwr`および`rmformat` コマンドを使用して、メディアを含まないデバイスのリストを作成できます。

`/etc/vold.conf` ファイル内にある、次の `support nomedia` エントリを変更することで、以前の `vold` の動作に戻すことができます。

```
support media
```

次に、`vold` を再起動します。詳細は、[vold.conf\(4\)](#) のマニュアルページを参照してください。

さらに、`vold` はホットプラグを認識するようになりました。この改善により、リムーバブルメディアを挿入すると、`vold` がメディアを自動的に検出してマウントします。リムーバブルメディアデバイスのファイルシステムを認識してマウントするために、`vold` を手動で再起動する必要はありません。

レガシーまたは非 USB フロッピーディスクデバイスを使用している場合は、`vold` がメディアを認識できるように `volcheck` コマンドの発行が必要になることがあります。

メディアが検出されたが何らかの理由でマウント解除されている場合は、次のコマンドを実行する必要があります。

```
# volrmount -i rmdisk0
```

リムーバブルメディアデバイスをホットリムーブする前に、まずメディアを取り出します。例:

```
# eject rmdisk0
```

手動によるマウントと自動マウントの比較

次の表は、リムーバブルメディアの手動によるマウント (リムーバブルメディア サービスを使用しない場合) と自動マウント (リムーバブルメディア管理を使用する場合) に関する手順を比較したものです。

表 1-1 リムーバブルメディアの手動によるマウントと自動マウントの比較

手順	手動によるマウント	自動マウント
1	メディアを挿入します。	メディアを挿入します。
2	スーパーユーザーになります。	USB フロッピーディスクの場合は、 <code>volcheck</code> コマンドを使用します。

表 1-1 リムーバブルメディアの手動によるマウントと自動マウントの比較 (続き)

手順	手動によるマウント	自動マウント
3	メディアデバイスの位置を確認します。	リムーバブルメディアサービスは、リムーバブルメディアを手動でマウントして使用するのに必要な多くのタスクを自動的に実行します。
4	マウントポイントを作成します。	
5	マウントポイントが現在のディレクトリではないことを確認します。	
6	mount に適切なオプションを付けて、デバイスをマウントします。	
7	スーパーユーザーアカウントを終了します。	
8	メディア上のファイル进行操作します。	メディア上のファイル进行操作します。
9	スーパーユーザーになります。	
10	メディアデバイスのマウントを解除します。	
11	メディアを取り出します。	メディアを取り出します。
12	スーパーユーザーアカウントを終了します。	

リムーバブルメディアへのアクセスの概要

リムーバブルメディアサービスを使用すると、手動によるマウントの場合と同様にリムーバブルメディアにアクセスできますが、その操作ははるかに容易になり、管理者のアクセス権も必要ありません。

システムに複数の種類のリムーバブルメディアデバイスがある場合は、そのアクセスポイントについて、次の表を参照してください。

表 1-2 リムーバブルメディアへのアクセス先

このパスをファイルシステムへのアクセスに使用	raw データにアクセスするための場所
/floppy/floppy0	/vol/dev/aliases/floppy0
/floppy/floppy1	/vol/dev/aliases/floppy1
/cdrom/cdrom0	/vol/dev/aliases/cdrom0
/cdrom/cdrom1	/vol/dev/aliases/cdrom1

表 1-2 リムーバブルメディアへのアクセス先 (続き)

このパスをファイルシステムへのアクセスに使用	raw データにアクセスするための場所
/rmdisk/rmdisk0 または	/vol/dev/aliases/rmdisk0 または
/rmdisk/rmdisk1	/vol/dev/aliases/rmdisk1
/pcmem/pcmem0	/vol/dev/aliases/pcmem0

rmmount -l コマンドを使用して、システムにマウントされているメディアを識別できます。例:

```
# rmmount -l
/dev/dsk/c5t0d0p0      rmdisk6,/media/FD-05PUB
/dev/dsk/c4t0d3p0      rmdisk5,/media/223UHS-SD-MMC
/dev/dsk/c2t0d0s2      cdrom1,cd1,sr1,Oracle_Solaris-11_1-AI-SPARC,/media/Oracle_Solaris-11_1-AI-SPARC
/dev/dsk/c3t0d0p0      rmdisk2,/media/00JB-00CRA0
```

上の出力では、次のデバイスがマウントされています。

```
/dev/dsk/c5t0d0p0      USB フロッピーディスク
/dev/dsk/c4t0d3p0      USB カードリーダー内の CF カード
/dev/dsk/c2t0d0s2      DVD-ROM
/dev/dsk/c3t0d0p0      リムーバブル USB ディスク
```

リムーバブルメディアの管理(タスク)

このセクションでは次のガイドラインと手順について説明します。

- [26 ページの「リムーバブルメディアに関する考慮事項」](#)
- [27 ページの「リムーバブルメディアを読み込む方法」](#)
- [28 ページの「フロッピーディスクをフォーマットする方法 \(rmformat\)」](#)
- [29 ページの「リムーバブルメディア上にファイルシステムを作成する方法」](#)
- [30 ページの「DVD-RAM 上にファイルシステムを作成する方法」](#)
- [30 ページの「リムーバブルメディア上のファイルシステムを検査する方法」](#)
- [31 ページの「リムーバブルメディア上の不良ブロックを修復する方法」](#)
- [31 ページの「リムーバブルメディアに読み取り/書き込み保護とパスワードによる保護を適用する」](#)
- [32 ページの「リムーバブルメディアの書き込み保護を有効または無効にする方法」](#)
- [32 ページの「リムーバブルメディアの読み取り/書き込み保護を有効または無効にする方法とパスワードを設定する方法」](#)

リムーバブルメディアに関する考慮事項

フロッピーディスクを操作する場合の考慮事項を次に示します。

- Oracle Solaris のファイルシステム形式は、Oracle Solaris ファイルシステムをサポートするための構造に加えて、基本「ビット」形式から構成されます。DOS ファイルシステムの完全な形式は、MS-DOS または NEC-DOS ファイルシステムのいずれかをサポートする構造に加え、基本的な「ビット」形式で構成されます。メディアの準備に必要な手順は、ファイルシステムのタイプごとに異なります。このため、フロッピーディスクをフォーマットする前に、どの手順に従うかを考慮してください。詳細は、[25 ページの「リムーバブルメディアの管理\(タスク\)」](#)を参照してください。
- リムーバブルメディアの名前については、[33 ページの「リムーバブルメディア名の使用」](#)を参照してください
- 名前のない(つまり、「ラベル」のない)フロッピーディスクには、デフォルト名の `unnamed_floppy` が割り当てられます。
- 名前のない(つまり、「ラベル」のない)フロッピーディスクには、デフォルト名の `floppy` が割り当てられます。

Oracle Solaris システムは、次の種類のファイルシステムをフォーマットできません。

- ZFS または UFS
- MS-DOS または NEC-DOS (PCFS)
- UDFS

Oracle Solaris システム (SPARC または x86) では、次の密度のフロッピーディスクをフォーマットできます。

フロッピーディスクのサイズ	フロッピーディスクの密度	容量
3.5"	高密度 (HD)	1.44 MB
3.5"	倍密度 (DD)	720 KB

デフォルトでは、フロッピーディスクドライブは、その標準密度でフロッピーディスクをフォーマットします。ここでのデフォルトとは、1.44 MB のドライブは、ほかの密度が指定されない限り、フロッピーディスクのフォーマットを (それが実際に 1.44 MB のフロッピーディスクであるかどうかに関係なく) 1.44 MB で行おうとする、という意味です。言い換えると、フォーマットは、フロッピーディスクの容量またはそれ以下で、ドライブの容量またはそれ以下で行えます。

フロッピーディスクのフォーマット

`rmformat` コマンドを使用して、リムーバブルメディアのフォーマットとその他の管理タスクを実行できます。ファイルシステムは自動的にマウントされます。このため、メディアに既存のファイルシステムが含まれる場合は、メディアをフォーマットする前にマウント解除が必要になることがあります。

`rmformat` コマンドには、3つのフォーマットオプションがあります。

- `quick` - このオプションは、フロッピーディスクを認証なしで、またはメディアの特定トラックを一部認証してフォーマットします。
- `long` - このオプションは、フロッピーディスクを完全にフォーマットします。一部のデバイスでは、このオプションを使用すると、ドライブによるメディア全体の認証が含まれることがあります。
- `force` - このオプションは、ユーザーの確認なしに完全にフォーマットします。パスワード保護メカニズムを備えるフロッピーディスクでは、このオプションによりフォーマットの前にパスワードが解除されます。この機能は、パスワードを忘れてしまった場合に役立ちます。パスワード保護されていないフロッピーディスクでは、フォーマットはこのオプションにより強制的に長形式になります。

▼ リムーバブルメディアを読み込む方法

リムーバブルメディアハードウェアに関する考慮事項については、[26 ページの「リムーバブルメディアに関する考慮事項」](#)を参照してください。

- 1 メディアを挿入します。
- 2 メディアがフォーマットされていることを確認します。
確信が持てない場合は、メディアを挿入し、[手順 3](#) の記述に従ってシステムコンソールウィンドウ内のステータスメッセージを確認します。メディアのフォーマットが必要な場合は、[28 ページの「フロッピーディスクをフォーマットする方法 \(rmformat\)」](#)に進みます。
- 3 (省略可能) **USB** ではないレガシーフロッピーディスクドライブを使用している場合は、ボリューム管理を通知します。

```
$ volcheck -v
```

表示される可能性があるステータスメッセージは2つです。

`media was found` ボリューム管理でメディアが検出されたため、[33 ページの「リムーバブルメディア名の使用」](#)に記載されたディレクトリへのマウントを試みます。

メディアが適切にフォーマットされている場合は、エラーメッセージはコンソールに表示されません。

メディアがフォーマットされていない場合は、「media was found」メッセージが引き続き表示されます。ただし、次のようなエラーメッセージがシステムコンソールウィンドウに表示されます。

```
fd0: unformatted diskette or no diskette in the drive
```

```
fd0: read failed (40 1 0)
```

```
fd0: bad format
```

ボリューム管理がメディアをマウントするためには、メディアがフォーマットされている必要があります。詳細は、[28 ページの「フロッピーディスクをフォーマットする方法 \(rmformat\)」](#)を参照してください。

```
no media was found
```

ボリューム管理がメディアを検出しませんでした。メディアが正しく挿入されていることを確認して、`volcheck`を再度実行します。成功しない場合は、メディアが破損していないかどうか確認してください。メディアのマウントを手動で試みることもできます。

- 4 内容を一覧表示して、メディアがマウントされたことを確認します。
たとえばフロッピーディスクの場合、次を実行します。

```
$ ls /media/floppy
lost+found myfiles
```

▼ フロッピーディスクをフォーマットする方法 (rmformat)

`rmformat` コマンドを使用して、フロッピーディスクをフォーマットできます。デフォルトでは、このコマンドはメディア上にパーティション0とパーティション2 (メディア全体) の2つのパーティションを作成します。

- 1 リムーバブルメディアサービスが実行中であることを確認します。その場合は、より短いニックネームをデバイス名に使用できます。

```
# svcs hal dbus rmvolmgr
STATE      STIME    FMRI
online     Apr_09   svc:/system/dbus:default
online     Apr_09   svc:/system/hal:default
online     Apr_09   svc:/system/filesystem/rmvolmgr:default
```

リムーバブルメディアサービスの再起動については、[35 ページ](#)の「リムーバブルメディアサービスを無効または有効にする方法」を参照してください。メディアのデバイス名の識別については、[33 ページ](#)の「リムーバブルメディア名の使用」を参照してください。

- 2 フロッピーディスクをフォーマットします。

```
$ rmformat -F [ quick | long | force ] device-name
```

rmformat フォーマットオプションの詳細は、[27 ページ](#)の「フロッピーディスクのフォーマット」を参照してください。

rmformat の出力が不正なブロックを示している場合は、[31 ページ](#)の「リムーバブルメディア上の不良ブロックを修復する方法」を参照してください。

- 3 (オプション)フロッピーディスクに8文字のラベルを付けます。

```
$ rmformat -b label device-name
```

DOS ラベルの作成方法については、[mkfs_pcfs\(1M\)](#) のマニュアルページを参照してください。

例 1-1 フロッピーディスクのフォーマット

この例では、フロッピーディスクをフォーマットする方法を示します。

```
$ rmformat -F quick /dev/rdiskette
Formatting will erase all the data on disk.
Do you want to continue? (y/n) y
.....
```

▼ リムーバブルメディア上にファイルシステムを作成する方法

- 1 (省略可能) 必要に応じてフロッピーディスクをフォーマットします。
USB フロッピーディスクをフォーマットするには、次のような構文を使用します。

```
$ rmformat -F long /dev/rdisk/c11t0d0p0
```

- 2 (オプション)代替の Solaris パーティションテーブルを作成します。

```
$ rmformat -s slice-file device-name
```

スライスファイルの例は次のようになります。

```
slices: 0 = 0, 30MB, "wm", "home" :
        1 = 30MB, 51MB :
        2 = 0, 94MB, "wm", "backup" :
        6 = 81MB, 13MB
```

- 3 スーパーユーザーになります。
- 4 適切なファイルシステムの種類を決定し、次のいずれかの作業を選択します。
 - PCFS ファイルシステムを作成します。例:

```
# mkfs -F pcfs -o nofdisk,size=9800 /dev/rdisk/c11t0d0p0
```
 - UDFS ファイルシステムを作成します。例:

```
# mkfs -F udfs /dev/rdisk/c0t1d0p0
```

▼ DVD-RAM 上にファイルシステムを作成する方法

この手順を使用して、DVD-RAM 上にファイルシステムを作成します。

- 1 スーパーユーザーになります。
- 2 **DVD-RAM** デバイス上にファイルシステムを作成します。
たとえば、次のように UDFS ファイルシステムを作成します。

```
# mkfs -F udfs /dev/rdisk/c0t0d0s2
```
- 3 ファイルシステムをマウントします。
たとえば、次のように UDFS ファイルシステムをマウントします。

```
# mount -F udfs /dev/dsk/c0t0d0s2 /mnt
```
- 4 ファイルシステムの読み取りや書き込みができることを確認します。
- 5 作業が完了したら、**DVD-RAM** を取り出します。

▼ リムーバブルメディア上のファイルシステムを検査する方法

- 1 スーパーユーザーになります。
- 2 ファイルシステムの種類を確認し、次のいずれかの作業を選択します。
 - 次のように、**UDFS** ファイルシステムを検査します。

```
# fsck -F udfs device-name
```
 - 次のように、**PCFS** ファイルシステムを検査します。

```
# fsck -F pcfs device-name
```

例1-2 リムーバブルメディア上のPCFS ファイルシステムを検査する

次の例は、メディア上のPCFS ファイルシステムの整合性を検査する方法を示しています。

```
# fsck -F pcfs /dev/rdisk/c0t4d0s2
** /dev/rdisk/c0t4d0s2
** Scanning file system meta-data
** Correcting any meta-data discrepancies
1457664 bytes.
0 bytes in bad sectors.
0 bytes in 0 directories.
0 bytes in 0 files.
1457664 bytes free.
512 bytes per allocation unit.
2847 total allocation units.
2847 available allocation units.
```

▼ リムーバブルメディア上の不良ブロックを修復する方法

ドライブが不良ブロック管理をサポートしている場合にのみ、検証中に見つかった不良セクタを `rmformat` コマンドで検証、解析、および修復できます。ほとんどのUSBメモリースティックは不良ブロック管理をサポートしていません。

ドライブが不良ブロック管理をサポートしている場合、不良ブロックを修復するための最大の努力が行われます。それでも不良ブロックを修復できなかった場合、メディアの修復に失敗したことを示すメッセージが表示されます。

- 1 リムーバブルメディア上の不良ブロックを修復します。

```
$ rmformat -c block-numbers device-name
```

`block-numbers` には、前の `rmformat` セッションで獲得したブロック番号を10進数、8進数、または16進数形式で指定します。

- 2 リムーバブルメディアを検証します。

```
$ rmformat -V read device-name
```

リムーバブルメディアに読み取り/書き込み保護とパスワードによる保護を適用する

この機能をサポートしているリムーバブルメディアには、読み取り保護または書き込み保護を適用し、パスワードを設定できます。

▼ リムーバブルメディアの書き込み保護を有効または無効にする方法

- 1 書き込み保護を有効にするか無効にするかを決定し、次のいずれかの作業を選択します。
 - 書き込み保護を有効にします。
`$ rmformat -w enable device-name`
 - 書き込み保護を無効にします。
`$ rmformat -w disable device-name`
- 2 リムーバブルメディアの書き込み保護が有効または無効になっていることを確認します。
`$ rmformat -p device-name`

▼ リムーバブルメディアの読み取り/書き込み保護を有効または無効にする方法とパスワードを設定する方法

パスワードによる保護機能をサポートしているリムーバブルメディアには、最大 32 文字のパスワードを適用できます。

パスワード機能をサポートしていないリムーバブルメディア上でパスワードを適用しようすると、警告メッセージが表示されます。

- 1 読み取り保護または書き込み保護を有効または無効のどちらにするかを決定し、パスワードを設定します。次のいずれかの手順に従います。
 - 読み取り保護または書き込み保護を有効にします。
`$ rmformat -W enable device-name`
Please enter password (32 chars maximum): xxx
Please reenter password:

`$ rmformat -R enable device-name`
Please enter password (32 chars maximum): xxx
Please reenter password:

- 読み取り保護または書き込み保護を無効にし、パスワードを削除します。

```
$ rmformat -W disable device-name
Please enter password (32 chars maximum): xxx
```

```
$ rmformat -R disable device-name
Please enter password (32 chars maximum): xxx
```

- 2 リムーバブルメディアの読み取り保護または書き込み保護が有効または無効になっていることを確認します。

```
$ rmformat -p device-name
```

リムーバブルメディアへのアクセス

リムーバブルメディア上の情報にアクセスするには、ボリューム管理を使用する方法と使用しない方法があります。GNOMEのファイルマネージャーを使用してリムーバブルメディア上の情報にアクセスする方法については、GNOMEデスクトップのドキュメントを参照してください。

このセクションでは次の手順について説明します。

- [33 ページの「リムーバブルメディア名の使用」](#)
- [34 ページの「リムーバブルメディア上のデータにアクセスするためのガイドライン」](#)
- [34 ページの「新しいリムーバブルメディアドライブを追加する方法」](#)
- [35 ページの「リムーバブルメディアサービスを無効または有効にする方法」](#)
- [35 ページの「リムーバブルメディア上の情報にアクセスする方法」](#)
- [36 ページの「リムーバブルメディアが使用中かどうかを調べる方法」](#)
- [36 ページの「リムーバブルメディアを取り出す方法」](#)

リムーバブルメディア名の使用

リムーバブルメディアには、さまざまな名前でアクセスできます。次の表に、ボリューム管理を使用して/使用しないでアクセスできるさまざまなメディア名を示します。

表1-3 リムーバブルメディア名

メディア	ボリューム管理デバイスの名前	ボリューム管理デバイスの別名	デバイス名
最初のフロッピーディスクドライブ	/floppy	/vol/dev/aliases/floppy0	/dev/rdiskette /vol/dev/rdiskette0/ volume-name

表 1-3 リムーバブルメディア名 (続き)

メディア	ボリューム管理デバイスの名前	ボリューム管理デバイスの別名	デバイス名
最初、2 番目、3 番目の CD-ROM または DVD-ROM ドライブ	<code>/cdrom0</code> <code>/cdrom1</code> <code>/cdrom2</code>	<code>/vol/dev/aliases/cdrom0</code> <code>/vol/dev/aliases/cdrom1</code> <code>/vol/dev/aliases/cdrom2</code>	<code>/vol/dev/rdisk/cnt n[dn] /</code> <i>volume-name</i>
USB メモリースティック	<code>/rmdisk/noname</code>	<code>/vol/dev/aliases/rmdisk0</code>	<code>/vol/dev/dsk/cnt ndn/volume-name</code> <code>:c</code>

リムーバブルメディア上のデータにアクセスするためのガイドライン

ほとんどの DVD は、ISO 9660 標準でフォーマットされています。このフォーマットには可搬性があります。そのため、DVD をボリューム管理によってマウントできます。

さまざまなフォーマットに対応するために、DVD はスライスに分割されます。スライスはハードディスクのパーティションに似ています。9660 部分には可搬性があります。DVD のマウントで問題が生じた場合、特にそれがインストール DVD の場合は、そのファイルシステムが、使用しているシステムのアーキテクチャーに適しているかどうかを確認してください。たとえば、DVD 上のラベルを確認できます。

▼ 新しいリムーバブルメディアドライブを追加する方法

一般に、最近のバスタイプのほとんどで、ホットプラグ機能がサポートされています。これは、ユーザーが空のスロットにディスクを挿入すれば、システムがそのディスクを認識してくれることを意味します。

ホットプラグ対応デバイスの詳細は、[第 4 章「デバイスの動的構成 \(タスク\)」](#) を参照してください。

- 1 スーパーユーザーになります。
- 2 新しいリムーバブルメディアドライブを接続します。
詳細な手順については、該当するハードウェアのマニュアルを参照してください。
- 3 システムが新しいメディアドライブを認識することを確認します。

```
# rmformat
Looking for devices...
```

▼ リムーバブルメディアサービスを無効または有効にする方法

場合によっては、リムーバブルメディアサービスを使用しないで、メディアを管理した方がよいことがあります。このセクションでは、リムーバブルメディアサービスを無効または有効にする方法について説明します。

これらのサービスを無効にした場合は、`mount` コマンドを使ってすべてのメディアを手動でマウントする必要があります。

- 1 メディアが使用中でないことを確認します。
メディアを使用中のすべてのユーザーを確認できた保証がない場合は、[36 ページ](#)の「リムーバブルメディアが使用中かどうかを調べる方法」の手順に従って `fuser` コマンドを使用してください。
- 2 スーパーユーザーになります。
- 3 次のいずれかの手順に従います。
 - リムーバブルメディアサービスを無効にします。

```
# svcadm disable svc:/system/filesystem/volfs:default
```
 - リムーバブルメディアサービスを有効にします。

```
# svcadm enable svc:/system/filesystem/volfs:default
```

▼ リムーバブルメディア上の情報にアクセスする方法

- 1 メディアを挿入します。
数秒後にメディアがマウントされます。
- 2 メディアの内容をリスト表示します。例:

```
# ls /media
```
- 3 (省略可能) 前の手順で特定したファイルをコピーします。

例 1-3 リムーバブルメディア上の情報にアクセスする

次の例は、USB メモリースティック上の情報にアクセスする方法を示しています。

```
$ ls /rmdisk
rmdisk0/          rmdisk1/
```

次の例は、DVD 上の情報にアクセスする方法を示しています。

```
$ ls /cdrom
cdrom0          sol_10_811_sparc
```

▼ リムーバブルメディアが使用中かどうかを調べる方法

- 1 スーパーユーザーになります。
- 2 メディアにアクセスしているプロセスを特定します。

```
# fuser -u /media
```

-u は、メディアを使用しているユーザーなどを表示します。
詳細は、[fuser\(1M\)](#) のマニュアルページを参照してください。
- 3 (オプション) メディアにアクセスしているプロセスを強制終了します。

```
# fuser -u -k /media
```

-k オプションは、メディアにアクセスしているプロセスを強制終了します。



注意-メディアにアクセスしているプロセスの強制終了は、緊急の場合にのみ行います。

- 4 プロセスが終了していることを確認します。

```
# pgrep process-ID
```

▼ リムーバブルメディアを取り出す方法

- 1 メディアが使用中でないことを確認します。
シェルまたはアプリケーションがメディア上のファイルまたはディレクトリのいずれかにアクセスしている場合、メディアは「使用中」であることを忘れないでください。DVD を使用しているシェルやアプリケーションなどをすべて検出したかどうかかわからない(デスクトップツールの背後に隠れているシェルがアクセスしている可能性がある)場合は、`fuser` コマンドを使用してください。[36 ページの「リムーバブルメディアが使用中かどうかを調べる方法」](#)を参照してください。
- 2 メディアを取り出します。

```
# eject media
```

たとえば、DVD の場合は次のように入力します。

```
# eject cdrom
```

たとえば、USB メモリースティックの場合は次のように入力します。

```
# eject rmdisk0
```

ヒント - リムーバブルデバイスの名前は `eject -l` コマンドで表示できます。

リモートシステム上のリムーバブルメディアへのアクセス

リムーバブルメディアは NFS を使用してリモートシステムと共有できます。NFS の使用に関する詳細については、ネットワークサービスに関する Oracle システム管理ガイドを参照してください。

このセクションでは次の手順について説明します。

- [37 ページの「ローカルのメディアをほかのシステムで使用可能にする方法」](#)
- [39 ページの「リモートシステム上のリムーバブルメディアにアクセスする方法」](#)

▼ ローカルのメディアをほかのシステムで使用可能にする方法

システムのメディアドライブを共有するようにシステムを構成すると、そのドライブに読み込まれているメディアがほかのシステムでも使用できるようになります。ただし、音楽用 CD は例外です。メディアドライブを共有すると、そのドライブをマウントするだけで、ドライブにロードされているメディアをほかのシステムでも使用できます。手順については、[39 ページの「リモートシステム上のリムーバブルメディアにアクセスする方法」](#)を参照してください。

- 1 スーパーユーザーになります。
- 2 メディアがロードされていることを確認します。
- 3 次のエントリを `/etc/dfs/dfstab` ファイルに追加します。

例:

```
share -F nfs -o ro /cdrom/sol_10_811_sparc
```

- 4 NFS サーバーサービスが実行中かどうかを確認します。

```
# svcs *nfs*
```

NFS サーバーサービスが実行中であれば、svcs コマンドの出力は次のようになります。

```
online          14:28:43 svc:/network/nfs/server:default
```

- 5 NFS サーバーのステータスを確認し、次のいずれかの手順を選択します。

- NFS サーバーサービスが実行されている場合は、[手順8](#)に進みます。
- NFS サーバーサービスが実行されていない場合は、次の手順に進みます。

- 6 NFS サーバーサービスを起動します。

```
# svcadm enable network/nfs/server
```

- 7 NFS デモンが実行されていることを確認します。

例:

```
# svcs -p svc:/network/nfs/server:default
STATE      STIME      FMRI
online     Aug_30     svc:/network/nfs/server:default
           Aug_30         319 mountd
           Aug_30         323 nfsd
```

- 8 メディアがほかのシステムで使用できるかどうかを確認します。

メディアが使用可能な場合は、その共有構成が表示されます。

```
# share
-          /cdrom/sol_10_811_sparc  sec=sys,ro  ""
```

例 1-4 ローカルの DVD をほかのシステムで使用可能にする

次の例は、ローカル DVD をネットワーク上のほかのシステムでも使用できるようにする方法を示しています。

```
# share -F nfs -o ro /media
# svcs *nfs*
# svcadm enable network/nfs/server
# svcs -p svc:/network/nfs/server:default
# share
-          /cdrom/sol_10_811_sparc      ro  ""
```

▼ リモートシステム上のリムーバブルメディアにアクセスする方法

リモートシステム上のメディアを手動でファイルシステムにマウントすることにより、そのメディアにアクセスできるようになります。ただし、リモートシステムが [37 ページの「ローカルのメディアをほかのシステムで使用可能にする方法」](#) の手順に従って、そのメディアを共有していることが必要です。

- 1 マウントポイントとして使用する既存のディレクトリを指定します。または、マウントポイントを作成します。

```
$ mkdir /directory
```

ここで `/directory` は、リモートシステムの DVD のマウントポイントとして作成するディレクトリの名前です。

- 2 マウントするメディアの名前を確認します。

```
$ showmount -e system-name
```

- 3 管理者としてメディアをマウントします。

```
# mount -F nfs -o ro system-name:/media/media-name local-mount-point
```

`system-name:` マウントするメディアを持つシステムの名前です。

`media-name` マウントするメディアの名前です。

`local-mount-point` リモートメディアのマウント先となるローカルディレクトリです。

- 4 管理者としてログアウトします。

- 5 メディアがマウントされていることを確認します。

```
$ ls /mnt
```

例 1-5 リモートシステム上の DVD または CD にアクセスする

次の例は、リモートシステム `starbug` の `sol_10_811_sparc` という名前のリモート DVD に、自動的にアクセスする方法を示しています。

```
$ showmount -e starbug
export list for starbug:
/cdrom/sol_10_811_sparc (everyone)
$ ls /net/starbug/cdrom/
/cdrom/sol_10_811_sparc
```


CD および DVD への書き込み (タスク)

この章では、`cdwr` コマンドを使用して、データ CD、データ DVD、およびオーディオ CD を作成およびコピーする手順について説明します。

この章の内容は次のとおりです。

- 41 ページの「オーディオ CD、データ CD、データ DVD の取り扱い」
- 43 ページの「データ CD、データ DVD、オーディオ CD への書き込み」

オーディオ **CD**、データ **CD**、データ **DVD** の取り扱い

`cdwr` コマンドを使用すると、Rock Ridge 拡張または Joliet 拡張を備えた ISO 9660 フォーマットで、CD-R、CD-RW、DVD-RW、または DVD+RW のメディアデバイス上に CD および DVD のファイルシステムを書き込むことができます。

`cdwr` コマンドを使用して、次のタスクを実行できます。

- データ CD およびデータ DVD を作成します。
- オーディオ CD を作成します。
- オーディオ CD からオーディオデータを抽出します。
- CD および DVD をコピーします。
- CD-RW メディアを消去します。

`cdwr` コマンドは、次のリリースで利用可能です。

- Oracle Solaris 10 リリース (SUNWcdwr)
- Oracle Solaris 11 リリース、`media/cdwr` パッケージ

推奨される CD-R または CD-RW デバイスについては、<http://www.oracle.com/us/sun/index.html> を参照してください。

CD/DVD メディアに関するよく使われる用語

ここでは、CD/DVD メディアに関連してよく使われる用語を定義します。

用語	説明
CD-R	CD 読み取りメディア。1 度だけ書き込みができ、その後は読み取り専用となります。
CD-RW	書き換え可能な CD メディア。書き込みと消去が可能です。CD-RW デバイスだけが CD-RW メディアを読み取れます。
DVD-R	デジタルビデオディスク (記録可能)。1 度だけ書き込みができ、その後は読み取り専用となります。このデバイスは、CD-R メディアよりも大容量です。
DVD+R	デジタルビデオディスク (記録可能)。1 度だけ書き込みができ、その後は読み取り専用となります。DVD+R デバイスは、DVD-R よりも総合的なエラー管理システムを備えているため、メディアの品質に関係なくより正確な書き込みを行います。
DVD-RW	DVD-R と記憶容量の等しいデジタルビデオディスク (書き換え可能)。このメディアは、最初にディスク全体を消去したあとで、再度記録できます。
DVD+RW	DVD+R と記憶容量の等しいデジタルビデオディスク (ランダムアクセスによる書き換えが可能)。このメディアを使用すると、ディスク全体を消去せずに、個別のブロックを上書きできます。
DVD-RAM	デジタルビデオディスク (ランダムアクセスメモリー、書き換え可能)。このメディアでは、トラックおよびハードセクタがらせん状ではなく、円状になっています。
ISO 9660	ISO (Industry Standards Organization の略)。コンピュータの標準記憶フォーマットを設定する組織です。 ISO 9660 ファイルシステムは、CD や DVD の標準ファイルシステムであり、主なコンピュータプラットフォームで同じ CD や DVD を読み取れます。この標準は 1988 年に発行され、High Sierra (ネバダ州の High Sierra Hotel にちなんで名づけられた) という業界団体によって作成されました。CD ドライブや DVD ドライブを備えたほぼすべてのコンピュータが ISO 9660 ファイルシステムからファイルを読み取れます。

用語	説明
Joliet 拡張	Windows ファイルシステム情報を追加します。
Rock Ridge 拡張	UNIX ファイルシステム情報を追加します。(Rock Ridge は映画「Blazing Saddles」に出てくる町にちなんで名づけられました。) 注- これらの拡張は相互に排他的ではありません。mkisofs コマンドに -R と -j の両オプションを指定して両方のシステムとの互換性を確保できます。(詳細は mkisofs を参照してください。)
MMC 準拠のレコーダー	Multi Media Command の略。これらのレコーダが共通のコマンドセットに準拠していること意味します。ある MMC 準拠レコーダに書き込めるプログラムは、ほかのすべての MMC 準拠レコーダにも書き込むことができます。
Red Book CDDA	Compact Disc Digital Audio の略。デジタルオーディオをコンパクトディスクに格納するための業界標準方式です。「Red Book」形式とも呼ばれます。この公式の業界仕様では、1 つまたは複数のオーディオファイルが 44.1 kHz のサンプリングレートで 16 ビットのステレオサウンドにサンプリングされることが要求されます。

CD メディアに書き込む場合によく使われる用語を次の表に一覧表示しています。

用語	説明
ブランキング	CD-RW メディアからデータを消去する処理。
セッション	リードイン/リードアウト情報を持つ完全なトラック。
トラック	完全なデータまたはオーディオの単位。

データ CD、データ DVD、オーディオ CD への書き込み

このセクションでは次の手順について説明します。

- 45 ページの「RBAC を使用してリムーバブルメディアへのユーザーアクセスを制限する」
- 45 ページの「RBAC を使用してリムーバブルメディアへのユーザーアクセスを制限する方法」
- 46 ページの「CD または DVD ライターを確認する方法」
- 46 ページの「CD または DVD メディアを検査する方法」
- 47 ページの「データ CD またはデータ DVD を作成する」
- 47 ページの「データ CD またはデータ DVD 用の ISO 9660 ファイルシステムを作成する方法」

- 48 ページの「マルチセッションのデータ CD を作成する方法」
- 49 ページの「オーディオ CD を作成する」
- 50 ページの「オーディオ CD を作成する方法」
- 51 ページの「オーディオトラックを CD から抽出する方法」
- 52 ページの「CD-RW メディアを消去する方法」

CD や DVD への書き込み処理は、途中で中断されることなく、データストリームが一定に保たれている必要があります。CD または DVD に十分な速度でデータを書き込めることを確認するために、`cdwr -s` オプションを使ってメディアへの書き込みをシミュレートすることを検討してください。

次のような問題がある場合には、書き込みエラーが発生することがあります。

- メディアがドライブの速度に対応できない場合。たとえば、メディアの中には 2x または 4x の速度しか保証されていないものもあります。
- システムが書き込み処理に支障をきたすほど多数の大きなプロセスを実行している場合。
- イメージがリモートシステムにあり、ネットワークの輻輳によってイメージの読み取りに遅延が発生しています。
- 送り側ドライブの速度が受け側ドライブよりも遅い場合。

上記の問題が発生した場合は、`cdwr -p` オプションを使用して、デバイスの書き込み速度を遅くしてください。

たとえば、次のコマンドは、4x の速度での書き込みをシミュレートします。

```
$ cdwr -is -p 4 image.iso
```

注 - CD-R、CD-RW (MRW フォーマット以外)、DVD-R、および DVD-RW メディアでは、シミュレーションモード (-s) がサポートされますが、DVD-RAM、DVD+R、DVD+RW、MRW フォーマットされたメディア、およびその他一部のメディアではシミュレーションモードはサポートされません。シミュレーションモードがサポートされない場合、次のメッセージが表示されます。

```
Media does not support simulated writing
```

メディアタイプの詳細は、[42 ページの「CD/DVD メディアに関するよく使われる用語」](#)を参照してください。

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

RBAC を使用してリムーバブルメディアへのユーザーアクセスを制限する

デフォルトでは、すべてのユーザーがリムーバブルメディアにアクセスできます。ただし、役割によるアクセス制御 (RBAC) で役割を設定して、リムーバブルメディアへのユーザーアクセスを制限できます。リムーバブルメディアへのアクセスを制限するには、その役割を一部のユーザーだけに割り当てます。

役割の使用方法については、『[Solaris のシステム管理: セキュリティーサービス](#)』の「[役割に基づくアクセス制御 \(概要\)](#)」を参照してください。

▼ RBAC を使用してリムーバブルメディアへのユーザーアクセスを制限する方法

- 1 スーパーユーザーになります。
- 2 デバイス管理の権利が含まれている役割を設定します。
詳細は、『[Solaris のシステム管理: セキュリティーサービス](#)』の第 9 章「[役割に基づくアクセス制御の使用 \(タスク\)](#)」を参照してください。

```
# roleadd -m -d /export/home/muser -c "mediauser role" -A solaris.device.cdrw -P All muser
```

- 3 cdrw コマンドを使用する必要があるユーザーを、新しく作成した役割に追加します。

```
# usermod -R muser joe
```

- 4 /etc/security/policy.conf ファイルの次の行をコメントにします。

```
AUTHS_GRANTED=solaris.device.cdrw
```

この手順を実行しないと、デバイス管理役割のメンバーだけでなく、すべてのユーザーが引き続き cdrw コマンドを利用できます。

このファイルを変更したあとは、デバイス管理役割のメンバーだけが cdrw コマンドを使用できるようになります。メンバー以外のユーザーがこのコマンドを使おうとすると、アクセスが拒否され、次のメッセージが表示されます。

```
Authorization failed, Cannot access disks.
```

▼ CD または DVD ライターを確認する方法

- 1 システム上の CD または DVD ライターを確認します。

例:

```
$ cdrw -l
Looking for CD devices...
Node | Connected Device | Device type
-----+-----+-----
cdrom0 | YAMAHA CRW8824S | 1.0d | CD Reader/Writer
```

- 2 特定の CD または DVD ライターを確認します。

例:

```
$ cdrw -a filename.wav -d cdrom2
```

- 3 メディアが空であるか、または既存の目次があるかどうかを確認します。

例:

```
$ cdrw -M

Device : YAMAHA CRW8824S
Firmware : Rev. 1.00 (26/04/00)
Media is blank
%
```

▼ CD または DVD メディアを検査する方法

cdrw コマンドは、リムーバブルメディアサービスが動作している状態でも動作していない状態でも機能します。リムーバブルメディアサービスを無効または有効にする方法については、[35 ページの「リムーバブルメディアサービスを無効または有効にする方法」](#)を参照してください。

- 1 CD または DVD をドライブに挿入します。

そのドライブで読み取ることができる CD または DVD であれば、どのような種類でもかまいません。

- 2 デバイスのリストを表示して、そのドライブが正しく接続されていることを確認します。

```
$ cdrw -l
Looking for CD devices...
Node | Connected Device | Device type
-----+-----+-----
cdrom1 | YAMAHA CRW8824S | 1.0d | CD Reader/Writer
```

- 3 (オプション) そのドライブがリスト内がない場合、システムにそのデバイスを認識させるために、次のいずれかの作業を選択します。

- システムをリブートせずにドライブを追加します。

```
# devfsadm
```

次に、リムーバブルメディアサービスを再起動します。

データ CD またはデータ DVD を作成する

まず `mkisofs` コマンドを使用してファイルとファイル情報を CD や DVD で使用される High Sierra 形式に変換し、データを準備します。

▼ データ CD またはデータ DVD 用の ISO 9660 ファイルシステムを作成する方法

- 1 空の CD または DVD をドライブに挿入します。
- 2 その新しい CD または DVD 上に ISO 9660 ファイルシステムを作成します。

```
$ mkisofs -r /pathname > cd-file-system
```

 - r Rock Ridge 情報を作成し、ファイル所有権を 0 にリセットします。
 - /pathname ISO 9660 ファイルシステムの作成に使われるパス名を指定します。
 - > cd-file-system CD または DVD に書き込むファイルシステムの名前を指定します。
- 3 ファイルシステムを CD または DVD にコピーします。

```
$ cdrw -i cd-file-system
```

 - i cd-file-system は、CD または DVD を作成するためのイメージファイルを指定します。

例 2-1 データ CD またはデータ DVD 用の ISO 9660 ファイルシステムを作成する

次の例は、データ CD またはデータ DVD 用の ISO 9660 ファイルシステムを作成する方法を示しています。

```
$ mkisofs -r /home/dubs/dir > dubs_cd
Total extents actually written = 56
Total translation table size: 0
Total rockridge attributes bytes: 329
Total directory bytes: 0
Path table size(bytes): 10
Max brk space used 8000
56 extents written (0 Mb)
```

次に、ファイルシステムを CD にコピーします。

```
$ cdrw -i dubs_cd
Initializing device...done.
Writing track 1...done.
Finalizing (Can take several minutes)...done.
```

▼ マルチセッションのデータ CD を作成する方法

この手順では、複数のセッションを CD に書き込む方法について説明します。また、infoA と infoB の各ディレクトリを CD にコピーする例も示します。

- 1 最初の CD セッション用のファイルシステムを作成します。

```
$ mkisofs -o infoA -r -V my_infoA /data/infoA
Total translation table size: 0
Total rockridge attributes bytes: 24507
Total directory bytes: 34816
Path table size(bytes): 98
Max brk space used 2e000
8929 extents written (17 Mb)
```

-o infoA ISO ファイルシステムの名前を指定します。

-r Rock Ridge 情報を作成し、ファイル所有権を 0 にリセットします。

-V my_infoA リムーバブルメディアサービスがマウントポイントとして使用するボリュームラベルを指定します。

/data/infoA 作成する ISO イメージディレクトリを指定します。

- 2 最初のセッションの ISO ファイルシステムを CD にコピーします。

```
$ cdrw -i0 infoA
Initializing device...done.
Writing track 1...done.
done.
Finalizing (Can take several minutes)...done.
```

-i infoA CD に書き込むイメージファイルの名前を指定します。

-0 書き込むために CD を開いたままにしておきます。

- 3 CD が排出されたあとで、再度挿入します。

- 4 次の書き込みセッションに含める CD メディアのパス名を確認します。

```
$ eject -n
.
.
.
cdrom0 -> /vol/dev/rdisk/c2t4d0/my_infoA
/vol/dev/... パス名を書き留めてください。
```


- 5 次のセッションを書き込む CD 上の次書き込み可能なアドレスを確認します。

```
% cdrw -M /cdrom
Device : YAMAHA CRW8424S
Firmware : Rev. 1.0d (06/10/99)

Track No. |Type      |Start address
-----+-----+-----
1          |Audio     |0
2          |Audio     |33057
3          |Data      |60887
4          |Data      |68087
5          |Data      |75287
Leadout    |Data      |84218
```

```
Last session start address: 75287
Next writable address: 91118
```

Next writable address: 出力に記述されているアドレスを書き留めて、次のセッションの書き込み時にこのアドレスを使用できるようにします。

- 6 次の CD セッション用の ISO ファイルシステムを作成し、CD に書き込みます。

```
$ mkisofs -o infoB -r -C 0,91118 -M /vol/dev/rdisk/c2t4d0/my_infoA
/data/infoB
Total translation table size: 0
Total rockridge attributes bytes: 16602
Total directory bytes: 22528
Path table size(bytes): 86
Max brk space used 20000
97196 extents written (189 Mb)
```

-o infoB

ISO ファイルシステムの名前を指定します。

-r

Rock Ridge 情報を作成し、ファイル所有権を 0 にリセットします。

-C 0,91118

最初のセッションの開始アドレスと、次に書き込み可能なアドレスを指定します。

-M /vol/dev/rdisk/c2t4d0/my_infoA

マージする既存の ISO イメージのパスを示します。

/data/infoB

作成する ISO イメージディレクトリを指定します。

オーディオ CD を作成する

cdrw コマンドを使用すると、個々のオーディオトラックまたは .au と .wav ファイルからオーディオ CD を作成できます。

次の表に、サポートされているオーディオ形式を示します。

表記形式	説明
sun	Red Book CDDA 形式のデータが入る Oracle .au ファイル
wav	Red Book CDDA 形式のデータが入る RIFF (.wav) ファイル
cda	raw CD オーディオデータ (「リトルエンディアン」バイト順序により、44.1 kHz のサンプリングレートでサンプリングされた 16 ビットの PCM ステレオ) が入る .cda ファイル
aur	「ビッグエンディアン」バイト順序による raw CD データが入る .aur ファイル

オーディオ形式を指定しなかった場合、`cdrw` コマンドはファイル拡張子に基づいてオーディオファイルの形式を判断しようとします。ファイル拡張子の大文字と小文字は区別されません。

▼ オーディオ CD を作成する方法

この手順では、オーディオファイルを CD にコピーする方法について説明します。

- 1 空の CD を CD-RW ドライブに挿入します。
- 2 オーディオファイルが入っているディレクトリに移動します。
\$ `cd /myaudiodir`
- 3 オーディオファイルを CD にコピーします。
\$ `cdrw -a track1.wav track2.wav track3.wav`
-a オプションによってオーディオ CD が作成されます。

例 2-2 オーディオ CD を作成する

次の例は、オーディオ CD を作成する方法を示しています。

```
$ cdrw -a bark.wav chirp.au meow.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
```

次の例は、マルチセッションのオーディオ CD を作成する方法を示しています。最初のセッションの書き込みが終わると、CD が排出されます。次の書き込みセッションの前に CD を再度挿入する必要があります。

```
$ cdrw -a0 groucho.wav chico.au harpo.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
<Re-insert CD>
$ cdrw -a zeppo.au
Initializing device...done.
Writing track 1...done.
done.
Finalizing (Can take several minutes)...done.
```

▼ オーディオトラックを CD から抽出する方法

オーディオトラックを CD から抽出して、そのオーディオトラックを新しい CD にコピーする場合は、次の手順に従います。

`cdrw -T` オプションを使ってオーディオファイル形式を指定しなかった場合、`cdrw` コマンドはファイル名拡張子を使ってオーディオファイル形式を判断します。たとえば、`cdrw` コマンドは、このファイルが `.wav` ファイルであることを検知します。

```
$ cdrw -x 1 testme.wav
```

- 1 オーディオ CD を CD-RW ドライブに挿入します。
- 2 オーディオトラックを抽出します。

```
$ cdrw -x -T audio-type 1 audio-file
```

<code>-x</code>	オーディオ CD からオーディオデータを抽出します。
<code>T audio-type</code>	抽出されるオーディオファイルの形式を指定します。サポートされているオーディオ形式は、 <code>sun</code> 、 <code>wav</code> 、 <code>cda</code> 、 <code>aur</code> です。
<code>audio-file</code>	抽出されるオーディオトラックを指定します。

- 3 抽出したトラックを新しい CD にコピーします。

```
$ cdrw -a audio-file
```

例 2-3 オーディオトラックを CD から抽出してオーディオ CD を作成する方法

次の例は、オーディオ CD から最初のトラックを抽出し、そのファイルに `song1.wav` という名前を付ける方法を示しています。

```
$ cdrw -x -T wav 1 song1.wav
Extracting audio from track 1...done.
```

次の例は、オーディオ CD にトラックをコピーする方法を示しています。

```
$ cdrw -a song1.wav
Initializing device...done.
Writing track 1...done.
Finalizing (Can take several minutes)...done.
```

▼ CD をコピーする方法

この手順では、まずすべてのトラックをオーディオ CD から抽出してディレクトリに入れ、次にそれらのトラックをすべて空の CD にコピーする方法について説明します。

注-デフォルトでは、cdrw コマンドは CD を /tmp ディレクトリにコピーします。コピーを行うには、最大 700M バイトの空き領域が必要です。CD をコピーするのに必要な空き領域が /tmp ディレクトリに不足している場合は、-m オプションを使って代替ディレクトリを指定します。

- 1 オーディオ CD を CD-RW ドライブに挿入します。
- 2 オーディオファイル用のディレクトリを作成します。

```
$ mkdir /music_dir
```
- 3 オーディオ CD からトラックを抽出します。

```
$ cdrw -c -m music_dir
```

トラックごとに Extracting audio ... メッセージが表示されます。

すべてのトラックが抽出されると、CD が排出されます。
- 4 空の CD を挿入して、Return キーを押します。
トラックの抽出が終わると、オーディオ CD が排出されます。空の CD を挿入するよう指示するプロンプトが表示されます。

例 2-4 CD をコピーする

次の例は、CD 間でコピーする方法を示しています。このタスクを行うには、CD-RW デバイスが 2 台必要です。

```
$ cdrw -c -s cdrom0 -d cdrom1
```

▼ CD-RW メディアを消去する方法

CD を書き換える前に、既存の CD-RW データを消去する必要があります。

- 次のいずれかの手順を選択して、メディア全体を消去するか、CD 上の最後のセッションだけを消去します。

- 最後のセッションだけを消去します。

```
$ cdrw -d cdrom0 -b session
```

-b session オプションを使って最後のセッションだけを消去する場合は、-b all オプションを使ってメディア全体を消去する場合に比べて短い時間で済みます。cdrw コマンドを使用して、1 セッションだけでデータ CD またはオーディオ CD を作成した場合でも、-b session オプションを使用できます。

- メディア全体を消去します。

```
$ cdrw -d cdrom0 -b all
```


デバイスの管理 (タスク)

Oracle Solaris リリースのデバイス管理には、通常、システムでの周辺デバイスの追加と削除、デバイスをサポートするための Sun 製品以外のデバイスドライバの追加、システム構成情報の表示が含まれます。この章では、Oracle Solaris リリースのディスク、DVD ドライブ、テープデバイスなどの周辺デバイスを管理する方法について説明します。

この章の内容は次のとおりです。

- 55 ページの「デバイス管理の新機能」
- 56 ページの「追加デバイス管理タスクの参照先」
- 56 ページの「Oracle Solaris のデバイスの管理」
- 65 ページの「システムへ周辺デバイスを追加する」
- 68 ページの「デバイスへのアクセス」

デバイス管理の新機能

ここでは、Oracle Solaris リリースにおけるデバイス管理の新機能について説明します。Solaris の新機能の一覧および Solaris リリースについての説明は、『[Oracle Solaris 10 1/13 の新機能](#)』を参照してください。

USB 3.0 デバイスのサポート

この Oracle Solaris リリースでは USB 3.0 をサポートしています。新しい USB ホストコントローラドライバ xhci が導入されています。詳細は、[第 5 章「USB デバイスの管理 \(タスク\)」](#)を参照してください。

追加デバイス管理タスクの参照先

次の表に、デバイスのホットプラグの実行手順、およびシリアルデバイス(プリンタやモデムなど)の追加手順の参照先を示します。

表 3-1 デバイスを追加する場合の参照先

デバイス管理タスク	参照先
ホットプラグ不可ディスクを追加します。	第 10 章「SPARC: ディスクの設定 (手順)」または 243 ページの「x86: ZFS ファイルシステム用のディスクの設定 (タスクマップ)」
SCSI または PCI デバイスのホットプラグを実行します。	80 ページの「cfgadm コマンドによる SCSI ホットプラグ」または 91 ページの「cfgadm コマンドによる PCI または PCIe ホットプラグ」
USB デバイスのホットプラグを実行します。	120 ページの「USB 大容量ストレージデバイスの管理」
モデムを追加します。	『System Administration Guide: Advanced Administration』の第 3 章「Managing Serial Ports With the Service Access Facility (Tasks)」
プリンタを追加します。	印刷に関するシステム管理ガイド
デバイスをセキュリティ保護します。	『Solaris のシステム管理: セキュリティサービス』の第 4 章「デバイスアクセスの制御 (タスク)」

Oracle Solaris のデバイスの管理

以降のセクションでは、Oracle Solaris でデバイスを管理する機能の概要を説明します。

- 57 ページの「デバイスドライバについて」
- 57 ページの「デバイスの自動構成」
- 59 ページの「デバイス構成情報の表示」
- 63 ページの「障害のあるデバイスの解決」

デバイスへのアクセスに関する情報については、68 ページの「デバイスへのアクセス」を参照してください。

x86: デバイスサポートの確認

デバイス検出ツールを使用すると、使用している x86 ハードウェアがこの Oracle Solaris リリースでサポートされているかどうかを確認できます。詳細は、http://www.oracle.com/webfolder/technetwork/hcl/hcts/device_detect.jspを参照してください。

デバイスドライバについて

コンピュータは通常、広範囲の周辺デバイスと大容量ストレージデバイスを使用します。たとえば、各システムには、ディスクドライブ、キーボードとマウス、何らかのDVDライターなどがあります。

これ以外に一般に使用されるデバイスには、次のようなものがあります。

- DVDドライブ
- プリンタとプロッタ
- ライトペン
- タッチセンサー式画面
- デジタイザー
- タブレットとスタイラスのペア

Oracle Solaris ソフトウェアは、これらのすべてのデバイスと直接には通信を行いません。各タイプのデバイスに異なるデータ形式、プロトコル、および転送速度が必要になります。

デバイスドライバは、オペレーティングシステムが特定のハードウェアと通信できるようにする低レベルのプログラムです。このドライバは、そのハードウェアに対するオペレーティングシステムの「インタプリタ」として機能します。

デバイスの自動構成

カーネルは、プラットフォーム固有のコンポーネントを備えた汎用コアと、一連のモジュールから成ります。カーネルは Oracle Solaris リリースで自動的に構成されます。

カーネルモジュールとは、システムで固有のタスクを実行するために使用されるソフトウェアのコンポーネントのことです。「ロード可能」なカーネルモジュールの例としては、デバイスのアクセス時にロードされるデバイスドライバがあげられます。

カーネルモジュールについては、次の表で説明します。

表 3-2 Solaris カーネルモジュール

場所	内容
/platform/`uname -m`/kernel	プラットフォーム固有のカーネルコンポーネント
/kernel	システムのブートに必要なすべてのプラットフォームに共通のカーネルコンポーネント
/usr/kernel	特定の命令セット内にあるすべてのプラットフォームに共通のカーネルコンポーネント

システムは、ブート時にどのデバイスが接続されるかを判断します。さらに、カーネルは、それ自体を動的に構成して、必要なモジュールだけをメモリーにロードします。ディスクデバイスやテープデバイスなどのデバイスがはじめてアクセスされると、対応するデバイスドライバがロードされます。このプロセスは、「自動構成」と呼ばれます。これは、すべてのカーネルモジュールが、必要に応じて自動的にロードされるためです。

`/etc/system` ファイルを修正することによって、カーネルモジュールがロードされる方法をカスタマイズできます。このファイルを修正する方法については、[system\(4\)](#) のマニュアルページを参照してください。

自動構成の機能と利点

自動構成の利点は次のとおりです。

- モジュールが必要に応じてロードされるため、主メモリーをより効率的に使用できます。
- 新しいデバイスがシステムに追加されるときに、カーネルを再構成する必要がありません。
- カーネルを再構成しないでドライバをロードし、テストできます。

自動構成プロセスは、新しいデバイス（およびドライバ）をシステムに追加するときに使用されます。以前のリリースでは、シャットダウンしたシステムにデバイスを追加した場合、再ブート（boot-r）が必要でした。デバイス構成の拡張により、シャットダウンしたシステムにデバイスを追加したときに再ブート（boot-r）が不要になります。

Oracle Solaris OS でシステムコンポーネントがホットプラグ機能をサポートする場合、システムが動作しているときにも、デバイスを追加、削除、または交換できます。ホットプラグ対応デバイスについては、[第4章「デバイスの動的構成（タスク）」](#)を参照してください

標準サポートされていないデバイスを使用する場合

Oracle Solaris リリースには、各種の標準デバイスをサポートするために必要なデバイスドライバが組み込まれています。これらのドライバは、`/kernel/drv` および `/platform/`uname -m`/kernel/drv` ディレクトリにあります。

ただし Solaris で標準にサポートされていないデバイスを購入した場合は、その製造元から、デバイスを正しくインストール、保守、管理するために必要なソフトウェアを提供してもらう必要があります。

そのようなデバイス用ソフトウェアには、少なくともデバイスドライバとその関連構成（.conf）ファイルが含まれます。.conf ファイルは、drv ディレクトリにもあります。また、サポートされていないデバイスは、Oracle Solaris で提供されるユーティリティと互換性を持たないので、保守および管理用のユーティリティが必要になる場合があります。

サポートされていないデバイスに必要な対策については、デバイスのご購入先にお問い合わせください。

デバイス構成情報の表示

次の表に、システムおよびデバイス構成情報を表示するために使用する 3 つのコマンドを説明します。

コマンド	説明	マニュアルページ
<code>prtconf</code>	メモリーの総量、システムのデバイス階層によって記述されたデバイス構成を含む、システム構成情報を表示します。このコマンドによる出力は、システムのタイプによって異なります。	prtconf(1M)
<code>sysdef</code>	システムハードウェア、疑似デバイス、ロード可能なモジュール、および指定のカーネルパラメータを含む、デバイス構成情報を表示します。	sysdef(1M)
<code>dmesg</code>	最後のリブート以降にシステムに接続されたデバイスのリストと、システム診断情報を表示します。	dmesg(1M)

システムのデバイスの識別に使用されるデバイス名については、[69 ページの「デバイス名の命名規則」](#)を参照してください。

driver not attached メッセージ

次のドライバ関連メッセージが、`prtconf` コマンドと `sysdef` コマンドによって表示されることがあります。

device, instance #number (driver not attached)

このメッセージは、このデバイスのドライバが使用できないことをいつも示すわけではありません。このメッセージは、ノードにデバイスがないか、あるいはデバイスが使用中ではないために、デバイスインスタンスに「現在」接続されているドライバがないことを示します。ドライバは、デバイスがアクセスされると自動的にロードされます。それらのドライバは、デバイスが使用されなくなるとアンロードされます。

使用中のデバイスのエラーチェック

次のユーティリティーは、指定されたデバイスが使用されているときを検出します。

- `dumppadm`
- `format`

- `mkfs` と `newfs`
- `swap`

これらの拡張により、これらのユーティリティーが、次のような使用状況のシナリオを検出する可能性があります。

- デバイスが ZFS ストレージプールに含まれている
- デバイスがダンプデバイスまたはスワップデバイスである
- マウントされたファイルシステムまたはデバイスのエントリが `/etc/vfstab` ファイルに存在する

たとえば、`format` ユーティリティーを使用してアクティブデバイスにアクセスしようとすると、次のようなメッセージが表示されます。

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
    0. clt0d0 <FUJITSU-MAY2073RCSUN72G-0401 cyl 8921 alt 2 hd 255 sec 63>
       /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@0,0
       /dev/chassis/SYS/HD0/disk
    1. clt1d0 <FUJITSU-MAY2073RCSUN72G-0401-68.37GB>
       /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@1,0
       /dev/chassis/SYS/HD1/disk
Specify disk (enter its number): 0
selecting clt0d0
[disk formatted]
/dev/dsk/clt0d0s0 is part of active ZFS pool rpool. Please see zpool(1M).
```

FORMAT MENU:

```
.
.
.
```

▼ システム構成情報を表示する方法

`prtconf` および `sysdef` コマンドの出力から、システムに接続されているディスク、テープ、および DVD デバイスを識別できます。デバイスインスタンスの出力の横に `driver not attached` メッセージが表示されます。これらのデバイスは、何らかのシステムプロセスによって常に監視されているため、「`driver not attached`」メッセージは通常、そのデバイスインスタンスにデバイスがないことを示す良い標識になります。

疑似デバイス、ロード可能なモジュール、および指定のカーネルパラメータを含むシステム構成情報を表示するには、`sysdef` コマンドを使用してください。

- システムおよびデバイスの構成情報を表示します。

- システムに接続されているすべてのデバイスを表示します。

たとえば、SPARC システムに対する次の `prtconf -v` の出力は、システムに接続されているディスクデバイスを識別しています。ディスクの詳細情報は、Device Minor Nodes セクションに記述されます。

```
$ /usr/sbin/prtconf -v | more
Memory size: 32640 Megabytes
System Peripherals (Software Nodes):

SUNW,SPARC-Enterprise-T5220
.
.
.
location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__2/disk
Device Minor Nodes:
    dev=(27,40)
        dev_path=/pci@0/pci@0/pci@8/pci@0/pci@a/LSILogic,sas@0/sd@2,0:a
        spectype=blk type=minor
        dev_link=/dev/dsk/c4t2d0s0
        dev_path=/pci@0/pci@0/pci@8/pci@0/pci@a/LSILogic,sas@0/sd@2,0:a,raw
        spectype=chr type=minor
        dev_link=/dev/rdisk/c4t2d0s0
        Device Minor Layered Under:
        mod=zfs accesstype=blk
        dev_path=/pseudo/zfs@0
        Minor properties:
            name='Nblocks' type=int64 items=1 dev=(27,40)
                value=0000000074702c8f
            name='Size' type=int64 items=1 dev=(27,40)
                value=0000000e8e0591e00
        .
        .
        .
```

- システムに接続されている特定のデバイスに関する情報を表示します。

たとえば、SPARC システムに対する次の `prtconf` の出力は、`/dev/dsk/c4t2d0s0` の `sd` インスタンス番号を示しています。

```
# prtconf -v /dev/dsk/c4t2d0s0 | grep instance
sd, instance #5
```

- システムに接続されているデバイスだけを表示します。

```
# prtconf | grep -v not
```

- デバイスの使用状況を表示します。

たとえば、次の `fuser` コマンドは、どのプロセスが `/dev/console` デバイスにアクセスしているかを示しています。

```
# fuser -d /dev/console
/dev/console:    5742o    2269o    20322o    858o
```

例 3-1 システム構成情報の表示

x86 ベースのシステムでは、次の prtconf 出力が表示されます。

```
# prtconf
System Configuration: Oracle Corporation i86pc
Memory size: 8192 Megabytes
System Peripherals (Software Nodes):

i86pc
  scsi_vhci, instance #0
  pci, instance #0
    pci108e,4843, instance #0
    pci8086,25e2, instance #0
      pci8086,3500, instance #7
        pci8086,3510, instance #9
        pci8086,3518, instance #10
          pci108e,4843, instance #0
          pci108e,4843, instance #1
      pci8086,350c, instance #8
    pci8086,25e3 (driver not attached)
    pci8086,25f8, instance #2
      pci108e,286, instance #0
        disk, instance #0
        disk, instance #2
        disk, instance #3
        disk, instance #1
    pci8086,25e5 (driver not attached)
    pci8086,25f9 (driver not attached)
    pci8086,25e7 (driver not attached)
    pci108e,4843, instance #0 (driver not attached)
    pci108e,4843, instance #1
    pci108e,4843, instance #2 (driver not attached)
    pci108e,4843 (driver not attached)
    pci108e,4843 (driver not attached)
    pci108e,4843 (driver not attached)
    pci108e,4843 (driver not attached)
    pci8086,2690, instance #6
      pci108e,125e, instance #2
      pci108e,125e, instance #3
    pci108e,4843, instance #0
    pci108e,4843, instance #1
      device, instance #0
        keyboard, instance #0
        mouse, instance #1
    pci108e,4843, instance #2
    pci108e,4843, instance #3
    pci108e,4843, instance #0
      storage, instance #0
        disk, instance #4
  .
  .
  .
```

x86 ベースのシステムからは、次の sysdef 出力が表示されます。

```
# sysdef
* Hostid
```

```

*
* 29f10b4d
*
* i86pc Configuration
*
*
* Devices
*
+boot (driver not attached)
memory (driver not attached)
aliases (driver not attached)
chosen (driver not attached)
i86pc-memory (driver not attached)
i86pc-mmio (driver not attached)
openprom (driver not attached)
options, instance #0
packages (driver not attached)
delayed-writes (driver not attached)
itu-props (driver not attached)
isa, instance #0
    motherboard (driver not attached)
    pnpADP,1542, instance #0
    asy, instance #0
    asy, instance #1
    lp, instance #0 (driver not attached)
    fdc, instance #0
        fd, instance #0
        fd, instance #1 (driver not attached)
    kd (driver not attached)
    kdmouse (driver not attached)
.
.
.

```

障害のあるデバイスの解決

デバイスのリタイアメントメカニズムでは、障害管理フレームワーク (FMA) により障害のあるデバイスを特定します。この機能では、障害のあるデバイスを安全な方法で自動的に非アクティブにして、データ消失、データ破壊、パニック、システムのダウンタイムなどを回避できます。リタイアメント処理は、デバイスのリタイア後のシステムの安定性を考慮して、安全な方法で実行されます。

重要なデバイスがリタイアされることはありません。リタイア済みのデバイスを手動で交換する必要がある場合は、手動交換の手順に加え、デバイスの交換後に `fmadm repair` コマンドを使用して、デバイスが交換されたことをシステムに認識させます。

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

デバイスのリタイアメントに関してユーザーに通知するための一般的なメッセージが、コンソールに表示され、`/var/adm/messages` ファイルにも書き込まれます。例:

```
Aug 9 18:14 starbug genunix: [ID 751201 kern.notice] NOTICE: One or more I/O devices have been retired
```

prtconf コマンドを使用して、特定のリタイア済みデバイスを確認することもできます。例:

```
# prtconf
.
.
.
pci, instance #2
    scsi, instance #0
        disk (driver not attached)
        tape (driver not attached)
        sd, instance #3
        sd, instance #0 (retired)
    scsi, instance #1 (retired)
        disk (retired)
        tape (retired)
pci, instance #3
    network, instance #2 (driver not attached)
    network, instance #3 (driver not attached)
os-io (driver not attached)
iscsi, instance #0
pseudo, instance #0
.
.
.
```

▼ 障害のあるデバイスを解決する方法

障害のあるデバイスまたはリタイア済みのデバイスを解決するには、次の手順を使用します。

zpool status または fmadm コマンドを使用して、ZFS デバイスの問題や障害情報を確認することもできます。ZFS デバイスの問題や障害情報については、『[Oracle Solaris ZFS 管理ガイド](#)』の第 10 章「[Oracle Solaris ZFS のトラブルシューティングとプールの回復](#)」を参照してください。

- 1 fmadm faulty コマンドで、障害のあるデバイスを特定します。例:

```
# fmadm faulty
-----
TIME                EVENT-ID                MSG-ID SEVERITY
-----
Jun 20 16:30:52 55c82fff-b709-62f5-b66e-b4e1bbe9dcb1 ZFS-8000-LR Major

Problem Status      : solved
Diag Engine         : zfs-diagnosis / 1.0
System
  Manufacturer      : unknown
  Name               : ORCL,SPARC-T3-4
  Part_Number        : unknown
  Serial_Number      : 1120BDRCCD
  Host_ID             : 84a02d28
-----
Suspect 1 of 1 :
```



```

Fault class : fault.fs.zfs.open_failed
Certainty   : 100%
Affects     : zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/
pool_name=pond/vdev_name=id1,sd@n5000c500335dc60f/a
Status      : faulted and taken out of service

FRU
  Name       : "zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/
pool_name=pond/vdev_name=id1,sd@n5000c500335dc60f/a"
  Status     : faulty

Description : ZFS device 'id1,sd@n5000c500335dc60f/a' in pool 'pond' failed to
              open.

Response    : An attempt will be made to activate a hot spare if available.

Impact      : Fault tolerance of the pool may be compromised.

Action      : Use 'fmadm faulty' to provide a more detailed view of this event.
              Run 'zpool status -lx' for more information. Please refer to the
              associated reference document at
              http://support.oracle.com/msg/ZFS-8000-LR for the latest service
              procedures and policies regarding this diagnosis.

```

- 2 障害のある、またはリタイア済みのデバイスを交換するか、デバイスエラーをクリアします。例:

```
# zpool clear pond c0t5000C500335DC60Fd0
```

デバイスエラーが断続的に発生するが、デバイスを交換しなかった場合は、前のエラーのクリアを試みることができます。

- 3 FMA 障害をクリアします。例:

```
# fmadm repaired zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/ \
pool_name=pond/vdev_name=id1,sd@n5000c500335dc60f/a
fmadm: recorded repair to of zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/
pool_name=pond/vdev_name=id1,sd@n5000c500335dc60f/a
```

- 4 障害がクリアされたことを確認します。

```
# fmadm faulty
```

エラーがクリアされると、fmadm faulty コマンドは何も返しません。

システムへ周辺デバイスを追加する

ホットプラグ不可の新しい周辺デバイスを追加する場合、通常、次の作業が必要になります。

- システムのシャットダウン
- システムへのデバイスの接続
- システムのリブート

システムにホットプラグインできない次のデバイスを追加する場合は、[66 ページ](#)の「[周辺デバイスを追加する方法](#)」の手順に従ってください。

- DVD ドライブ
- セカンダリディスクドライブ
- テープドライブ

場合によっては、新しいデバイスをサポートするために、他社製のデバイスドライバを追加しなければなりません。

ホットプラグ対応デバイスについては、[第4章「デバイスの動的構成\(タスク\)」](#)を参照してください。

▼ 周辺デバイスを追加する方法

- 1 スーパーユーザーになります。
- 2 (省略可能) デバイスをサポートするためにデバイスドライバを追加する必要がある場合は、[67 ページ](#)の「[デバイスドライバを追加する方法](#)」の手順を実行します。
- 3 システムをシャットダウンします。

```
# shutdown -i0 -g30 -y
```

-i0 システムを init0 状態に戻します。システムの電源を落としてデバイスの追加、削除を行うのに適した状態になります。

-g30 システムを 30 秒以内にシャットダウンします。デフォルト値は 60 秒です。

-y ユーザーの介入なしに、システムのシャットダウンを続けます。このオプションを指定しないと、シャットダウンプロセスを続けるかどうか、プロンプトでたずねられます。

- 4 システムがシャットダウンしたら、次のいずれかを選択して電源を落とします。
 - SPARC プラットフォームでは、ok プロンプトが表示されたら電源を落としても安全です。
 - x86 プラットフォームでは、type any key to continue プロンプトが表示されたら電源を落としても安全です。
- 5 すべての周辺デバイスの電源を落とします。

周辺デバイスの電源スイッチの位置については、各自の周辺デバイスに添付のハードウェアガイドを参照してください。

- 6 周辺デバイスをインストールし、ハードウェア構成がメーカーの仕様に従っていることを確認します。
デバイスの設置と接続については、周辺デバイスに添付のハードウェアガイドを参照してください。
- 7 システムの電源を入れます。
システムがブートされてマルチユーザーモードになり、ログインプロンプトが表示されます。
- 8 周辺デバイスにアクセスし、そのデバイスが追加されたことを確認してください。
デバイスにアクセスする方法については、[68 ページの「デバイスへのアクセス」](#)を参照してください。

▼ デバイスドライバを追加する方法

この手順では、デバイスがすでにシステムに追加されていることを前提としています。そうでない場合は、[66 ページの「周辺デバイスを追加する方法」](#)を参照してください。

- 1 スーパーユーザーになります。
- 2 メディアをドライブに挿入します。
- 3 ドライバをインストールします。
`# pkgadd [-d] device package-name`
- 4 パッケージが正常に追加されたことを確認します。
`# pkgchk package-name`
`#`
パッケージが正しくインストールされている場合は、何も表示されません。

例 3-2 デバイスドライバを追加する

次の例では、XYZdrv というパッケージをインストールして確認します。

```
# pkgadd XYZdrv
(licensing messages displayed)
.
.
.
Installing XYZ Company driver as <XYZdrv>
.
.
```

```
.  
Installation of <XYZdrv> was successful.  
# pkgchk XYZdrv  
#
```

デバイスへのアクセス

コマンドを使用してディスク、ファイルシステムなどのデバイスを管理する場合、デバイス名を指定する方法を知っている必要があります。通常、論理デバイス名を使用して、システムに接続されたデバイスを表すことができます。論理デバイス名と物理デバイス名は、システム上でそれぞれ論理デバイスファイルと物理デバイスファイルによって表現されます。

デバイス情報が作成される方法

システムがブートされると、デバイス階層が作成されて、システムに接続されたすべてのデバイスが表示されます。カーネルは、このデバイス階層情報を使用して、ドライバを該当するデバイスに対応づけます。また、カーネルは、特定の操作を実行するドライバへの一連のポインタを提供します。

デバイスの管理方法

devfs ファイルシステムは、システム上のすべてのデバイスの名前空間である **/devices** ディレクトリを管理します。このディレクトリは、実際のバスとデバイスのアドレスから成る「物理」デバイスを表します。

dev ファイルシステムは、「論理」デバイス名の名前空間である **/dev** ディレクトリを管理します。

デフォルトでは、**devfsadm** コマンドはすべてのドライバをシステムに読み込もうとし、可能な限りすべてのデバイスインスタンスに接続しようとします。次に、**devfsadm** コマンドはデバイスファイルを **/devices** ディレクトリに作成し、論理リンクを **/dev** ディレクトリに作成します。**devfsadm** コマンドはまた、**path_to_inst** インスタンスデータベースの保守も行います。

動的再構成イベントまたはファイルシステムアクセスに対応する **/dev** および **/devices** ディレクトリの更新は、**devfsadmd** (**devfsadm** コマンドのデーモンバージョン) によって処理されます。このデーモンは、システムのブート時にサービス管理機能によって起動されます。

devfsadmd デーモンは再構成イベントによるデバイス構成の変化を自動的に検出するため、このコマンドを対話的に実行する必要はありません。

詳細は、次のマニュアルページを参照してください。

- [devfsadm\(1M\)](#)
- [devfs\(7FS\)](#)
- [path_to_inst\(4\)](#)

デバイス名の命名規則

Oracle Solaris でデバイスは3つの方法のいずれかで参照されます。

- 物理デバイス名 – デバイス情報階層の完全デバイスパス名を表します。物理デバイス名は、そのデバイスが最初にシステムに追加されるときまでに作成されます。物理デバイスファイルは、`/devices` ディレクトリにあります。
- インスタンス名 – システム上のデバイスすべてのカーネル短縮名を表します。たとえば、`sd0` と `sd1` は、2つのディスクデバイスのインスタンス名を表します。インスタンス名は、`/etc/path_to_inst` ファイル内でマップされます。
- 論理デバイス名 – 論理デバイス名は、そのデバイスが最初にシステムに追加されるときまでに作成されます。論理デバイス名は、デバイスを参照する際に、ほとんどのファイルシステムコマンドで使用されます。論理デバイス名を使用するファイルコマンドの一覧は、[表 3-3](#) を参照してください。`/dev` ディレクトリの論理デバイスファイルは、`/devices` ディレクトリの物理デバイスファイルにシンボリックリンクされています。

上記のデバイス名の情報は、次のコマンドによって表示できます。

- `dmesg`
- `format`
- `sysdef`
- `prtconf`

論理ディスクデバイス名

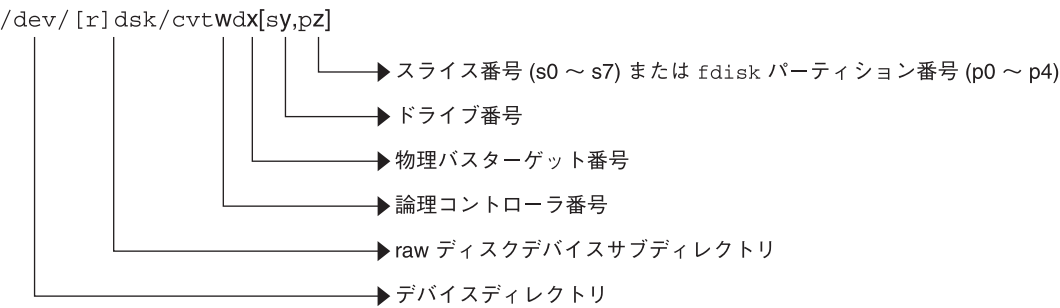
論理デバイス名は、次のタスクを行う場合に、ディスクデバイスにアクセスするために使用されます。

- システムに新しいディスクを追加します。
- システム間でディスクを移動します。
- ローカルディスク上にあるファイルシステムにアクセスまたはそれをマウントします。
- ローカルファイルシステムのバックアップをとります。

管理コマンドの多くは、ディスクスライスまたはファイルシステムを参照する引数を使用します。

シンボリックリンクされるサブディレクトリ(/dev/dsk または /dev/rdisk のどちらか)に続けて、特定のコントローラ、ディスク、およびスライスを識別する文字列を指定することによって、ディスクデバイスを参照してください。

図 3-1 論理デバイス名の説明



ディスクサブディレクトリの指定

一部のディスクとファイルの管理コマンドには、*raw* (またはキャラクタ型) デバイスインタフェースか、ブロックデバイスインタフェースを使用する必要があります。デバイスからどのような方法でデータを読み取るかによって使い分けます。

raw デバイスインタフェースは、一度に少量のデータだけを転送します。ブロックデバイスインタフェースには、大量のデータブロックが一度に読み取られるバッファが含まれます。

コマンドによって、必要なインタフェースは異なります。

- コマンドが *raw* デバイスインタフェースを必要とする場合は、/dev/rdisk サブディレクトリを指定してください。(rdisk の「r」は、「raw」を表します。)
- コマンドがブロックデバイスインタフェースを必要とする場合は、/dev/dsk サブディレクトリを指定してください。
- コマンドが /dev/dsk または /dev/rdisk のどちらを必要とするかがわからない場合は、そのコマンドのマニュアルページの説明を参照してください。

次の表に、一般的に使用されるディスクコマンドとファイルシステムコマンドの一部に必要なインタフェースを示します。

表 3-3 使用頻度の高いコマンドに必要なデバイスインタフェースのタイプ

コマンド	インタフェースのタイプ	使用例
dumpadm(1M)	ブロック	<code>dumpadm -d /dev/zvol/dsk/rpool/dump</code>

表 3-3 使用頻度の高いコマンドに必要なデバイスインタフェースのタイプ (続き)

コマンド	インタフェースのタイプ	使用例
<code>prtvtoc(1M)</code>	Raw	<code>prtvtoc /dev/rdisk/c0t0d0s0</code>
<code>swap(1M)</code>	ブロック	<code>swap -a /dev/zvol/dsk/rpool/swap</code>

直接コントローラとバス指向コントローラ

ディスクのパーティションやスライスにアクセスする方法は、そのディスクデバイスが直接コントローラとバス指向コントローラのどちらに接続されているかによって異なる場合があります。一般的に、直接コントローラの論理デバイス名には「ターゲット」識別子が含まれません。

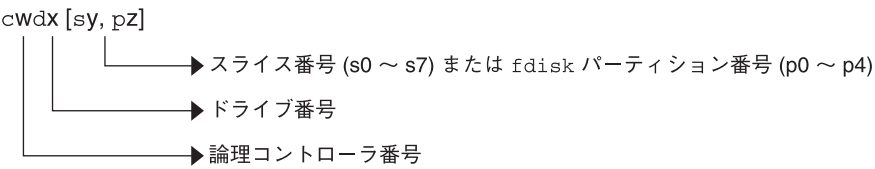
両方のタイプのコントローラについて、次のセクションで説明します。

注- コントローラ番号は、システム初期設定時に自動的に割り当てられます。この番号は、厳密に論理的なものであり、物理コントローラに直接対応するものではありません。

直接コントローラでアクセスされるディスク

IDE コントローラでアクセスされるディスクにスライスを指定するには、下の図に示す命名規則に従ってください。

図 3-2 直接コントローラでアクセスされるディスク

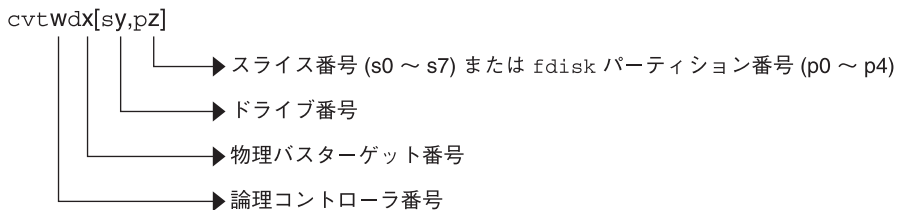


fdisk パーティション全体を示すには、スライス 2 (s2) を指定してください。

バス指向コントローラでアクセスされるディスク

バス指向コントローラ (SCSI など) でアクセスされるディスクにスライスを指定するには、下の図に示す命名規則に従ってください。

図 3-3 パス指向コントローラでアクセスされるディスク

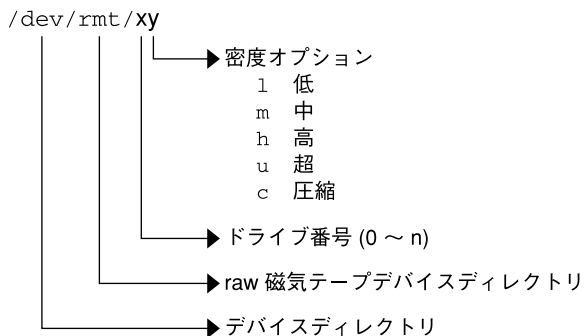


ディスク全体を示すには、スライス 2 (s2) を指定してください。

論理テープデバイス名

論理テープデバイスファイルは、`/devices` ディレクトリからのシンボリックリンクとして `/dev/rmt/*` ディレクトリ内にあります。

図 3-4 論理テープデバイス名



システムに接続された最初のテープデバイスは 0 (`/dev/rmt/0`) です。テープ密度の値 (l、m、h、c、および u) の詳細は、[第 22 章「テープドライブの管理 \(タスク\)」](#) を参照してください。

論理リムーバブルメディアデバイス名

リムーバブルメディアは、リムーバブルメディア管理サービスによって管理されるため、論理デバイス名は、手動でメディアをマウントしないかぎり、通常使用されません。

システムのリムーバブルメディアデバイスを表す論理デバイス名については、[33 ページの「リムーバブルメディアへのアクセス」](#) を参照してください。

デバイスの動的構成 (タスク)

この章では、Oracle Solaris OS でデバイスを動的に構成する手順について説明します。Oracle Solaris OS でシステムコンポーネントがホットプラグ機能をサポートする場合、システムが動作しているときにも、デバイスを追加、削除、または交換できます。ホットプラグをサポートしないシステムコンポーネントの場合は、新しいコンポーネントがシステムにインストールされたあとで、ブート時に新しいデバイスが構成されます。

動的に構成するデバイスに関する手順については、次のセクションを参照してください。

- 79 ページの「[cfgadm コマンドによる SCSI ホットプラグ \(タスクマップ\)](#)」
- 90 ページの「[cfgadm コマンドによる PCI または PCIe ホットプラグ \(タスクマップ\)](#)」
- 97 ページの「[cfgadm コマンドによる SATA ホットプラグ](#)」
- 100 ページの「[アプリケーション開発者 RCM スクリプト \(タスクマップ\)](#)」
- 101 ページの「[システム管理者 RCM スクリプト \(タスクマップ\)](#)」

cfgadm コマンドを使用して USB デバイスのホットプラグを実行する方法については、[143 ページの「cfgadm コマンドを使った USB デバイスのホットプラグ」](#)を参照してください。

cfgadm コマンドを使用して InfiniBand デバイスのホットプラグを実行する方法については、[第 6 章「InfiniBand デバイスの使用 \(概要/タスク\)」](#)を参照してください。

デバイスへのアクセスに関する情報については、[68 ページの「デバイスへのアクセス」](#)を参照してください。

動的再構成とホットプラグ機能

このセクションでは次の情報について説明します。

- 75 ページの「接続点」
- 77 ページの「PCI または PCIe アダプタカードの取り外し」
- 77 ページの「PCI または PCIe アダプタカードの取り付け」
- 77 ページの「hotplug コマンドによる PCIe ホットプラグ」

「ホットプラグ機能」とは、システムの動作中に、システムコンポーネントを物理的に取り付け、取り外し、または交換できる機能のことです。「動的再構成」とは、システムコンポーネントのホットプラグを実行できる機能のことです。また動的再構成は、システムリソースをシステムから物理的に取り外さなくても (なんらかの方法で) システムリソース (ハードウェアとソフトウェアの両方) をシステム内で移動したり、無効にできる機能のことです。

一般に、次のバスタイプはホットプラグ対応です。

- USB
- ファイバチャネル
- 1394
- ATA
- SCSI
- Infiniband

さらに、SPARC プラットフォームと x86 プラットフォームの両方で、`cfgadm` コマンドを使用して、次のデバイスをホットプラグできます:

- USB デバイス
- SCSI または SAS デバイス
- PCI デバイス
- PCIe デバイス
- SATA デバイス
- InfiniBand デバイス

`cfgadm` コマンドには、次のような機能があります。

- システムコンポーネントのステータスの表示
- システムコンポーネントの検査
- システムコンポーネントの構成の変更
- 構成ヘルプメッセージの表示

`cfgadm` コマンドでシステムコンポーネントを再構成する利点は、システムが動作しているときでも、システムコンポーネントを取り付け、取り外し、または交換できることです。さらに、`cfgadm` コマンドでは、システムコンポーネントを取り付け、取り外し、または交換するために必要な手順が示されます。

コンポーネントのホットプラグを実行する手順については、次を参照してください。

- 80 ページの「[cfgadm コマンドによる SCSI ホットプラグ](#)」
- 91 ページの「[cfgadm コマンドによる PCI または PCIe ホットプラグ](#)」
- 97 ページの「[cfgadm コマンドによる SATA ホットプラグ](#)」
- [cfgadm\(1m\)](#)

注 - すべての SCSI と PCI のコントローラが [cfgadm](#) コマンドによるホットプラグ機能をサポートしているわけではありません。

Oracle が提供する高可用性の一部として、動的再構成はほかの階層化製品 (代替パス指定やフェイルオーバーソフトウェアなど) とともに使用することをお勧めします。両方の製品ともデバイス障害に対する耐性を提供します。

高可用性ソフトウェアがなくても、障害が発生したデバイスを交換できます。この場合、適切なアプリケーションを手動で停止し、重要でないファイルシステムのマウントを手動で解除し、デバイスを取り付けまたは取り外します。

注 - システムによっては、ホットプラグ対応のスロットとホットプラグに対応しないスロットを持つ場合があります。企業レベルのシステムなど、特定のハードウェア構成に対するデバイスのホットプラグについては、お使いのハードウェア構成のドキュメントを参照してください。

接続点

[cfgadm](#) コマンドは接続点についての情報を表示します。「接続点」とは、動的再構成を行うことができるシステム内の特定の場所のことです。

接続点は、次の要素から構成されています。

- 占有装置 (*occupant*): システムに構成できるハードウェアコンポーネントのことです。
- 受容体 (*receptacle*): 占有装置を受け入れる場所のことです。

接続点は、論理と物理の両方の接続点 ID ([Ap_Id](#)) で表現されます。物理 [Ap_Id](#) は接続点の物理的なパス名です。論理 [Ap_Id](#) は物理 [Ap_Id](#) に代わるユーザーに理解しやすい ID です。[Ap_Id](#) の詳細は、[cfgadm\(1m\)](#) のマニュアルページを参照してください。

通常、SCSI HBA (Host Bus Adapter)、つまり、SCSI コントローラの論理 [Ap_Id](#) はコントローラ番号 ([c0](#) など) で表現されます。

コントローラ番号が SCSI HBA に割り当てられていない場合、内部的に生成された固有の識別子が提供されます。たとえば、SCSI コントローラの固有な識別子は次のようになります。

fas1:scsi

通常、SCSI デバイスの論理 Ap_Id の形式は次のようになります。

HBA-logical-apid::device-identifier

次の例において、c0 は SCSI HBA の論理 Ap_Id です。

c0::dsk/c0t3d0

通常、デバイス識別子は /dev ディレクトリ内にある論理デバイス名から導き出されます。たとえば、論理デバイス名が /dev/rmt/1 のテープデバイスの論理 Ap_Id は次のようになります。

c0::rmt/1

SCSI デバイスの論理 Ap_Id を /dev ディレクトリ内にある論理デバイス名から導き出すことができない場合、内部的に生成された固有の識別子が提供されます。たとえば、/dev/rmt/1 テープデバイスの識別子は、次のようになります。

c0::st4

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

cfgadm コマンドはすべてのリソースと動的再構成の操作を、一般的な状態 (configured、unconfigured など) や操作 (connect、configure、unconfigure など) を示す用語で表現します。一般的な状態や操作の詳細は、[cfgadm\(1m\)](#) のマニュアルページを参照してください。

次の表に、SCSI HBA 接続点の受容体と占有装置の状態を示します。

受容体の状態	説明	占有装置の状態	説明
empty	SCSI HBA には利用できません	configured	1つまたは複数のデバイスがバス上で構成されています
disconnected	バスは休止しています	unconfigured	デバイスは構成されていません
connected	バスはアクティブです		

次の表に、SCSI デバイス接続点の受容体と占有装置の状態を示します。

受容体の状態	説明	占有装置の状態	説明
empty	SCSI デバイスには利用できません	configured	デバイスが構成されています

受容体の状態	説明	占有装置の状態	説明
disconnected	バスは休止しています	unconfigured	デバイスは構成されていません
connected	バスはアクティブです		

SCSI 接続点の状態は特別なハードウェアによって示されない限り未知です。SCSI コンポーネントの構成情報を表示する手順については、[81 ページの「SCSI デバイスに関する情報を表示する方法」](#)を参照してください。

PCI または PCIe アダプタカードの取り外し

デバイスドライバがホットプラグ機能をサポートしている場合、必須でないシステムリソースとして機能している PCI アダプタカードは取り外すことができます。重要なシステムリソースとして機能している PCI アダプタカードは取り外すことができません。

PCI アダプタカードが取り外し可能であるためには、次の条件が必要です。

- デバイスドライバはホットプラグ機能をサポートしていなければなりません。
- 重要なリソースには代替バスでアクセスできなければなりません。

たとえば、システムにインストールされている Ethernet カードが 1 つしかない場合、ネットワーク接続を切断せずにこの Ethernet カードを取り外すことは不可能です。このような環境でネットワーク接続をアクティブに保ったまま Ethernet カードを取り外すには、別の階層化ソフトウェアサポートが必要です。

PCI または PCIe アダプタカードの取り付け

PCI アダプタカードをシステムに取り付けるには、次の条件が必要です。

- スロットが利用できなければなりません。
- デバイスドライバが当該アダプタカードのホットプラグ機能をサポートしていなければなりません。

PCI アダプタカードの取り付けまたは取り外しの手順については、[91 ページの「cfgadm コマンドによる PCI または PCIe ホットプラグ」](#)を参照してください。

hotplug コマンドによる PCIe ホットプラグ

hotplug コマンドを使ってホットプラグ処理可能な接続を管理できます。ここで言う接続とは、PCI Express (PCIe) および PCI SHPC デバイス上のコネクタまたはポートの

みを指します。ホットプラグコネクタとは、コンポーネントの挿入や取り外しが行われるシステムの物理的機構のことです。ホットプラグポートとは、システムへのデバイスの接続が管理される、システムデバイスツリー内の論理表現です。

hotplug 機能を使用することで、オンボードデバイスを含むデバイスをシステムに物理的に追加したり取り外したりせずに、オンラインやオフラインにできます。

hotplug コマンドを使ってデバイスを管理するには、hotplug サービスを有効にする必要があります。

```
# svcadm enable svc:/system/hotplug:default
```

次の例は、hotplug コマンドの使用方法を示しています。

システムのすべての PCI/PCIe ホットプラグ可能コネクタ/ポート (仮想と物理) を次のように表示します。

```
# hotplug list -lv
```

PCIe スロットの Ethernet カードを構成します。例:

```
# hotplug enable /pci0,0 pcie0
```

PCIe スロットの Ethernet カードの構成を解除します。例:

```
# hotplug disable /pci0,0 pcie0
```

PCI デバイスノードをオフラインにして、ノードのデバイスドライバを切り離します。例:

```
# hotplug offline /pci0,0/pci1 pci.0,2
```

PCI デバイスノードをオンラインにして、ノードのデバイスドライバを接続します。例:

```
# hotplug online /pci0,0/pci1 pci.0,2
```

IOV 物理機能の依存するポートをインストールします。例:

```
# hotplug install /pci@400/pci@1/pci@0/pci@4 pci.0,1
```

次に、結果となる IOV 仮想機能で、探索されたものを表示します。例:

```
# hotplug list -v /pci@400/pci@1/pci@0/pci@4 pci.0,1
<pci.0,1> (ONLINE)
{ IOV physical function }
{ IOV virtual function 'pci.0,81' }
{ IOV virtual function 'pci.0,83' }
{ IOV virtual function 'pci.0,85' }
{ IOV virtual function 'pci.0,87' }
```

```
<pci.0,81> (OFFLINE)
ethernet@0,81
<pci.0,83> (OFFLINE)
ethernet@0,83
<pci.0,85> (OFFLINE)
ethernet@0,85
<pci.0,87> (OFFLINE)
ethernet@0,87
```

IOV 物理機能の依存するポートをアンインストールします。例:

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
```

依存する IOV 仮想機能がビジー状態の場合、この操作は失敗します。例:

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
ERROR: devices or resources are busy.
ethernet@0,81:
{ Network interface igbvf1 }
{ igbvf1: hosts IP addresses: 10.0.0.1 }
{ Plumbed IP Address }
```

PCI ホットプラグ処理 (hotplug) のトラブルシューティング

hotplug コマンドを使用するには、次のサービスが動作している必要があります。

```
svc:/system/hotplug:default
```

動作していない場合は、次のメッセージが表示されます。

```
ERROR: hotplug service is not available.
```

サポートされる入出力バスがシステムに存在しない場合、次のエラーメッセージが表示されます。

```
ERROR: there are no connections to display.
(See hotplug(1m) for more information.)
```

上記のエラーメッセージは、ホットプラグ対応のほかの入出力デバイスがシステムに存在するが、それらのデバイスの管理には hotplug コマンドではなく cfgadm コマンドを使用する必要があることを意味する場合があります。

cfgadm コマンドによる SCSI ホットプラグ (タスクマップ)

タスク	説明	参照先
SCSI デバイスについての情報を表示します。	SCSI コントローラおよびデバイスについての情報を表示します。	81 ページの「SCSI デバイスに関する情報を表示する方法」

タスク	説明	参照先
SCSI コントローラの構成を解除します。	SCSI コントローラの構成を解除します。	81 ページの「SCSI コントローラの構成を解除する方法」
SCSI コントローラを構成します。	構成を解除された SCSI コントローラを構成します。	82 ページの「SCSI コントローラを構成する方法」
SCSI デバイスを構成します。	特定の SCSI デバイスを構成します。	82 ページの「SCSI デバイスを構成する方法」
SCSI コントローラを切り離します。	特定の SCSI コントローラを切り離します。	83 ページの「SCSI コントローラを切り離す方法」
SCSI コントローラを接続します。	切り離された特定の SCSI コントローラを接続します。	84 ページの「SPARC: SCSI コントローラを接続する方法」
SCSI デバイスを SCSI バスに取り付けます。	特定の SCSI デバイスを SCSI バスに取り付けます。	85 ページの「SPARC: SCSI デバイスを SCSI バスに取り付ける方法」
SCSI コントローラ上の同一デバイスと交換します。	SCSI バス上のデバイスを、同じタイプの別のデバイスに交換します。	86 ページの「SPARC: SCSI コントローラ上の同一デバイスを交換する方法」
SCSI デバイスを取り外します。	SCSI デバイスをシステムから取り外します。	87 ページの「SPARC: SCSI デバイスを取り外す方法」
SCSI 構成に関する問題のトラブルシューティングを行います。	失敗した SCSI 構成解除操作を解決します。	90 ページの「失敗した SCSI 構成解除操作の解決方法」

cfgadm コマンドによる SCSI ホットプラグ

このセクションでは、cfgadm コマンドを使用してさまざまな SCSI ホットプラグ処理を実行する方法について説明します。

注- 通常、SCSI フレームワークは SCSI デバイスのホットプラグ機能をサポートしています。ただし、使用している SCSI デバイスでホットプラグ機能がサポートされているかどうか、ハードウェアのドキュメントを参照して確認してください。

この節で説明する手順では、特定のデバイスを使用して、cfgadm コマンドで SCSI コンポーネントのホットプラグを実行する例を示します。cfgadm コマンドで提供されるデバイス情報や表示されるデバイス情報は、システム構成によって異なります。

▼ SCSI デバイスに関する情報を表示する方法

次の手順では、SCSI コントローラ c2 と c3、およびコントローラに接続されたデバイスを例として使用して、cfgadm コマンドを使用して表示可能なデバイス構成情報を示します。

注 - SCSI デバイスが cfgadm コマンドでサポートされていない場合、その SCSI デバイスは cfgadm コマンドの出力には表示されません。

- 1 スーパーユーザーになります。
- 2 システム上の接続点に関する情報を表示します。

```
# cfgadm -l
```

Ap_Id	Type	Receptacle	Occupant	Condition
c2	scsi-bus	connected	configured	unknown
c3	scsi-sas	connected	configured	unknown

この例では、c2 と c3 は2つの SCSI コントローラを表しています。

- 3 システムの SCSI コントローラとこれらに接続されているデバイスについての情報を表示します。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c2	scsi-bus	connected	configured	unknown
c2::dsk/c2t0d0	CD-ROM	connected	configured	unknown
c3	scsi-sas	connected	configured	unknown
c3::dsk/c3t0d0	disk	connected	configured	unknown
c3::dsk/c3t1d0	disk	connected	configured	unknown
c3::dsk/c3t2d0	disk	connected	configured	unknown
c3::dsk/c3t3d0	disk	connected	configured	unknown

注 - cfgadm -l コマンドは、SCSI デバイスではなく、SCSI HBA についての情報を表示します。ディスクやテープなどの SCSI デバイスについての情報を表示するには、cfgadm -al コマンドを使用してください。

▼ SCSI コントローラの構成を解除する方法

次の手順では、SCSI コントローラ c2 を使用して、SCSI コントローラの構成を解除する例を示します。

- 1 スーパーユーザーになります。
- 2 SCSI コントローラの構成を解除します。

```
# cfgadm -c unconfigure c2
```

3 SCSI コントローラの構成が解除されていることを確認します。

```
# cfgadm -al
Ap_Id                                Type      Receptacle  Occupant    Condition
c2_                                   scsi-bus  connected   unconfigured unknown
c2::dsk/c2t0d0                       unavailable connected   unconfigured unknown
c3_                                   scsi-sas  connected   configured   unknown
c3::dsk/c3t0d0                       disk      connected   configured   unknown
c3::dsk/c3t1d0                       disk      connected   configured   unknown
c3::dsk/c3t2d0                       disk      connected   configured   unknown
c3::dsk/c3t3d0                       disk      connected   configured   unknown
```

c2 の Occupant の列に unconfigured と表示されていることに注目してください。これは、SCSI バスに占有装置が構成されていないことを示します。

構成解除処理が失敗した場合は、[90 ページの「失敗した SCSI 構成解除操作の解決方法」](#)を参照してください。

▼ SCSI コントローラを構成する方法

次の手順では、SCSI コントローラ c2 を使用して、SCSI コントローラを構成する例を示します。

- 1 スーパーユーザーになります。
- 2 SCSI コントローラを構成します。

```
# cfgadm -c configure c2
```

- 3 SCSI コントローラが構成されていることを確認します。

```
# cfgadm -al
Ap_Id                                Type      Receptacle  Occupant    Condition
c2_                                   scsi-bus  connected   configured   unknown
c2::dsk/c2t0d0                       unavailable connected   unconfigured unknown
c3_                                   scsi-sas  connected   configured   unknown
c3::dsk/c3t0d0                       disk      connected   configured   unknown
c3::dsk/c3t1d0                       disk      connected   configured   unknown
c3::dsk/c3t2d0                       disk      connected   configured   unknown
c3::dsk/c3t3d0                       disk      connected   configured   unknown
```

前述の構成を解除する例では、SCSI バス上のすべてのデバイスを削除しました。この例では、すべてのデバイスをシステムに構成し直します。

▼ SCSI デバイスを構成する方法

次の手順では、SCSI ディスク c2t0d0 を使用して、SCSI デバイスを構成する例を示します。

- 1 スーパーユーザーになります。

2 構成するデバイスを確認します。

```
# cfgadm -al
Ap_Id                                Type          Receptacle  Occupant    Condition
c2                                   scsi-bus      connected   configured  unknown
c2::dsk/c2t0d0                       unavailable   connected   unconfigured unknown
c3                                   scsi-sas      connected   configured  unknown
c3::dsk/c3t0d0                       disk         connected   configured  unknown
c3::dsk/c3t1d0                       disk         connected   configured  unknown
c3::dsk/c3t2d0                       disk         connected   configured  unknown
c3::dsk/c3t3d0                       disk         connected   configured  unknown
```

3 SCSI デバイスを構成します。

```
# cfgadm -c configure c2::dsk/c2t0d0
```

4 SCSI デバイスが構成されていることを確認します。

```
# cfgadm -al
Ap_Id                                Type          Receptacle  Occupant    Condition
c2                                   scsi-bus      connected   configured  unknown
c2::dsk/c2t0d0                       CD-ROM       connected   configured  unknown
c3                                   scsi-sas      connected   configured  unknown
c3::dsk/c3t0d0                       disk         connected   configured  unknown
c3::dsk/c3t1d0                       disk         connected   configured  unknown
c3::dsk/c3t2d0                       disk         connected   configured  unknown
c3::dsk/c3t3d0                       disk         connected   configured  unknown
```

▼ SCSI コントローラを切り離す方法



注意-SCSI デバイスを切り離すときには注意が必要です。特に、ルートファイルシステムの重要なコンポーネントが入っているディスクのコントローラを扱っているときは十分に注意してください。動的再構成ソフトウェアは、システムがハングする原因をすべて発見できるわけではありません。この手順は、十分注意して実行してください。

次の手順では、SCSI コントローラ c2 を使用して、SCSI デバイスを切り離す例を示します。

1 スーパーユーザーになります。

2 デバイスを切り離す前に、デバイスが接続されていることを確認します。

```
# cfgadm -al
Ap_Id                                Type          Receptacle  Occupant    Condition
c2                                   scsi-bus      connected   configured  unknown
c2::dsk/c2t0d0                       CD-ROM       connected   configured  unknown
c3                                   scsi-sas      connected   configured  unknown
c3::dsk/c3t0d0                       disk         connected   configured  unknown
c3::dsk/c3t1d0                       disk         connected   configured  unknown
c3::dsk/c3t2d0                       disk         connected   configured  unknown
c3::dsk/c3t3d0                       disk         connected   configured  unknown
```

3 SCSI コントローラを切り離します。

```
# cfgadm -c disconnect c2
WARNING: Disconnecting critical partitions may cause system hang.
Continue (yes/no)? y
```



注意 - このコマンドは、cfgadm -c connect コマンドを使用するまで、SCSI バス上のすべての入出力動作を中断します。cfgamd コマンドは基本的な検査を行い、重要なパーティションが切り離されるのを防ぎます。しかし、すべての場合を発見できるわけではありません。このコマンドの使い方が不適切な場合、システムがハングし、システムをリブートしなければならない可能性があります。

4 SCSI バスが切り離されていることを確認します。

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c2             unavailable  disconnected  configured  unknown
c2::dsk/c2t0d0 unavailable  disconnected  configured  unknown
c3             scsi-sas     connected   configured  unknown
c3::dsk/c3t0d0 disk         connected   configured  unknown
c3::dsk/c3t1d0 disk         connected   configured  unknown
c3::dsk/c3t2d0 disk         connected   configured  unknown
c3::dsk/c3t3d0 disk         connected   configured  unknown
```

コントローラとそれに接続されていたすべてのデバイスがシステムから切り離されました。

▼ SPARC: SCSI コントローラを接続する方法

次の手順は、SCSI コントローラ c2 を使用して、SCSI コントローラを接続する例を示します。

- 1 スーパーユーザーになります。
- 2 接続する前に、デバイスが切り離されていることを確認します。

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c2             unavailable  disconnected  configured  unknown
c2::dsk/c2t0d0 unavailable  disconnected  configured  unknown
c3             scsi-sas     connected   configured  unknown
c3::dsk/c3t0d0 disk         connected   configured  unknown
c3::dsk/c3t1d0 disk         connected   configured  unknown
c3::dsk/c3t2d0 disk         connected   configured  unknown
c3::dsk/c3t3d0 disk         connected   configured  unknown
```

3 SCSI コントローラを接続します。

```
# cfgadm -c connect c2
```

4 SCSI コントローラが接続されていることを確認します。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c2	scsi-bus	connected	configured	unknown
c2::dsk/c2t0d0	CD-ROM	connected	configured	unknown
c3	scsi-sas	connected	configured	unknown
c3::dsk/c3t0d0	disk	connected	configured	unknown
c3::dsk/c3t1d0	disk	connected	configured	unknown
c3::dsk/c3t2d0	disk	connected	configured	unknown
c3::dsk/c3t3d0	disk	connected	configured	unknown

▼ SPARC: SCSI デバイスを SCSI バスに取り付ける方法

SCSI コントローラ c3 を使用して、SCSI デバイスを SCSI バスに取り付ける方法を説明します。

注- デバイスを取り付けるときは、デバイス自身の Ap_Id ではなく、デバイスを取り付ける SCSI HBA (コントローラ) の Ap_Id を指定します。

1 スーパーユーザーになります。

2 現在の SCSI 構成を確認します。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c2	scsi-bus	connected	configured	unknown
c2::dsk/c2t0d0	CD-ROM	connected	configured	unknown
c3	scsi-sas	connected	configured	unknown
c3::dsk/c3t0d0	disk	connected	configured	unknown
c3::dsk/c3t1d0	disk	connected	configured	unknown
c3::dsk/c3t2d0	disk	connected	configured	unknown

3 SCSI デバイスを SCSI バスに取り付けます。

a. 次の cfgadm コマンドを入力します。

例:

```
# cfgadm -x insert_device c3
Adding device to SCSI HBA: /devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2
This operation will suspend activity on SCSI bus: c3
```

b. Continue (yes/no)? というプロンプトに y と入力して、次に進みます。

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

ホットプラグ処理の実行中、SCSI バス上の入出力動作は中断されます。

c. デバイスを接続して、電源を入れます。

d. Enter y if operation is complete or n to abort (yes/no)? というプロンプトに y と入力します。

Enter y if operation is complete or n to abort (yes/no)? y

4 デバイスが取り付けられていることを確認します。

```
# cfgadm -al
Ap_Id                                Type      Receptacle  Occupant    Condition
c2                                   scsi-bus  connected   configured  unknown
c2::dsk/c2t0d0                       CD-ROM    connected   configured  unknown
c3                                   scsi-sas  connected   configured  unknown
c3::dsk/c3t0d0                       disk      connected   configured  unknown
c3::dsk/c3t1d0                       disk      connected   configured  unknown
c3::dsk/c3t2d0                       disk      connected   configured  unknown
c3::dsk/c3t3d0                       disk      connected   configured  unknown
```

コントローラ c3 に新しいディスクが取り付けられました。

▼ SPARC: SCSI コントローラ上の同一デバイスを交換する方法

次の手順では、SCSI ディスク c3t3d0 を使用して、SCSI コントローラ上の同一デバイスを交換する例を示します。

SCSI コントローラ上で同一デバイスの交換を試みる場合、次の条件を確認します。

- ZFS ルートプールの一部であるミラー化されている、またはミラー化されていないブートデバイスの交換を試みる場合は、次を参照してください。

<http://www.oracle.com/technetwork/articles/servers-storage-admin/011-091-sol-dis-recovery-489183.html>

- デバイスが従来のボリューム管理ソフトウェアにより制御されている場合は、ボリューム管理ドキュメントでデバイスの具体的な交換手順を確認してください。有効な Oracle サポートプランをお持ちの場合は、次を参照してください。
 - Solaris ボリュームマネージャー (SVM) を使用している場合は、MOS ドキュメント 1010753.1 を参照してください。
 - Veritas Volume Manager (VxVM) を使用している場合は、MOS ドキュメント 1003122.1、1011782.1、および 1002285.1 を参照してください。

1 スーパーユーザーになります。

2 現在の SCSI 構成を確認します。

```
# cfgadm -al
c2                                   scsi-bus  connected   configured  unknown
c2::dsk/c2t0d0                       CD-ROM    connected   configured  unknown
c3                                   scsi-sas  connected   configured  unknown
```

c3::dsk/c3t0d0	disk	connected	configured	unknown
c3::dsk/c3t1d0	disk	connected	configured	unknown
c3::dsk/c3t2d0	disk	connected	configured	unknown
c3::dsk/c3t3d0	disk	connected	configured	unknown

3 SCSIバス上のデバイスを、同じタイプの別のデバイスに交換します。

a. 次の **cfgadm** コマンドを入力します。

例:

```
# cfgadm -x replace_device c3::dsk/c3t3d0
Replacing SCSI device: /devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@3,0
This operation will suspend activity on SCSI bus: c3
```

b. **Continue (yes/no)?** というプロンプトに **y** と入力して、次に進みます。

ホットプラグ処理の実行中、SCSIバス上の入出力動作は中断されます。

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

c. デバイスの電源を切ってから、そのデバイスを取り外します。

d. 交換用のデバイスを取り付けます。そして、取り付けたデバイスの電源を入れます。

交換用のデバイスは、取り外すデバイスと同じタイプで、同じアドレス(ターゲットと LUN)にするようにします。

e. **Enter y if operation is complete or n to abort (yes/no)?** というプロンプトに **y** と入力します。

```
Enter y if operation is complete or n to abort (yes/no)? y
```

4 デバイスが交換されていることを確認します。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c2	scsi-bus	connected	configured	unknown
c2::dsk/c2t0d0	CD-ROM	connected	configured	unknown
c3	scsi-sas	connected	configured	unknown
c3::dsk/c3t0d0	disk	connected	configured	unknown
c3::dsk/c3t1d0	disk	connected	configured	unknown
c3::dsk/c3t2d0	disk	connected	configured	unknown
c3::dsk/c3t3d0	disk	connected	configured	unknown

▼ SPARC: SCSI デバイスを取り外す方法

次に、SCSI ディスク c3t3d0 を使用して、SCSI コントローラ上のデバイスを取り外す例を示します。

1 スーパーユーザーになります。

2 現在の SCSI 構成を確認します。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c2	scsi-bus	connected	configured	unknown
c2::dsk/c2t0d0	CD-ROM	connected	configured	unknown
c3	scsi-sas	connected	configured	unknown
c3::dsk/c3t0d0	disk	connected	configured	unknown
c3::dsk/c3t1d0	disk	connected	configured	unknown
c3::dsk/c3t2d0	disk	connected	configured	unknown
c3::dsk/c3t3d0	disk	connected	configured	unknown

3 SCSI デバイスをシステムから取り外します。

a. 次の cfgadm コマンドを入力します。

例:

```
# cfgadm -x remove_device c3::dsk/c3t3d0
Removing SCSI device: /devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@3,0
This operation will suspend activity on SCSI bus: c3
```

b. Continue (yes/no)? というプロンプトに y と入力して、次に進みます。

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

ホットプラグ処理の実行中、SCSI バス上の入出力動作は中断されます。

c. デバイスの電源を切ってから、そのデバイスを取り外します。

d. Enter y if operation is complete or n to abort (yes/no)? というプロンプトに y と入力します。

```
Enter y if operation is complete or n to abort (yes/no)? y
```

注 - この手順は、SCSI RAID デバイスを SCSI RAID アレイから取り外す場合に実行する必要があります。

4 デバイスがシステムから取り外されていることを確認します。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c2	scsi-bus	connected	configured	unknown
c2::dsk/c2t0d0	CD-ROM	connected	configured	unknown
c3	scsi-sas	connected	configured	unknown
c3::dsk/c3t0d0	disk	connected	configured	unknown
c3::dsk/c3t1d0	disk	connected	configured	unknown
c3::dsk/c3t2d0	disk	connected	configured	unknown

SCSI 構成に関する問題のトラブルシューティング

このセクションでは、SCSI 構成に関する問題のトラブルシューティングを行うために、エラーメッセージとその解決策について説明します。SCSI 構成に関する問題のトラブルシューティングの詳細は、[cfgadm\(1m\)](#) のマニュアルページを参照してください。

エラーメッセージ

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
  Resource          Information
  -----
  /dev/dsk/clt0d0s0  mounted filesystem "/file-system"
```

エラーの発生原因

ファイルシステムがマウントされているデバイスを取り外しまたは交換しようとした。

解決方法

エラーメッセージのリストにあるファイルシステムのマウントを解除してから、もう一度 `cfgadm` コマンドを実行します。

エラーメッセージ

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
  Resource          Information
  -----
  /dev/dsk/device-name  swap area
```

エラーの発生原因

`cfgadm` コマンドを使用して、スワップデバイス、専用のダンプデバイスなどのシステムリソースを取り外すと、システムリソースがアクティブな場合、このようなエラーメッセージが表示されます。

解決方法

指定されたデバイス上のスワップ領域の構成を解除してから、再度 `cfgadm` を実行します。

エラーメッセージ

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
  Resource          Information
  -----
  /dev/dsk/device-name  dump device (swap)
```

エラーの発生原因

スワップ領域上に構成されているダンプデバイスを取り外そうとしたか、置き換えようとした。

解決方法
スワップ領域に構成されているダンプデバイスの構成を解除してから、再度
cfgadm を実行します。

エラーメッセージ

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
Resource                                Information
-----
/dev/dsk/device-name    dump device (dedicated)
```

エラーの発生原因
専用ダンプデバイスを取り外そうとしたか、置き換えようとした。

解決方法
専用ダンプデバイスの構成を解除し、cfgadm 処理を再実行します。

▼ 失敗した **SCSI** 構成解除操作の解決方法

1つ以上のターゲットデバイスが使用中である場合、および SCSI 構成解除操作が失敗した場合、次の手順を使用します。この手順を使用しないと、将来、このコントローラおよびターゲットデバイスの動的再構成操作が失敗し、dr in progress メッセージが表示されます。

- 1 スーパーユーザーになります。
- 2 コントローラを再構成します。

```
# cfgadm -c configure device-name
```

cfgadm コマンドによる **PCI** または **PCIe** ホットプラグ (タスクマップ)

次のタスクマップでは、システム上の PCI または PCIe デバイスの管理タスクについて説明します。

タスク	説明	参照先
PCI スロット構成情報を表示します。	システムの PCI ホットプラグ対応デバイスとスロットのステータスを表示します。	92 ページの「PCI スロット構成情報を表示する方法」
PCI アダプタカードを取り外します。	カードを構成解除してからスロットから電源を外し、システムからカードを取り外します。	93 ページの「PCI アダプタカードを取り外す方法」

タスク	説明	参照先
PCI アダプタカードを取り付けます。	アダプタカードをホットプラグ対応のスロットに挿入します。スロットを電源に接続し、カードを構成します。	95 ページの「PCI アダプタカードを取り付ける方法」
PCI 構成に関する問題のトラブルシューティングを行います。	PCI 構成障害に対処するために、エラーメッセージと解決策を確認します。	96 ページの「PCI 構成に関する問題のトラブルシューティング」

cfgadm コマンドによる PCI または PCIe ホットプラグ

このセクションでは、SPARC および x86 ベースのシステム上で PCI または PCIe アダプタカードのホットプラグを実行する手順について説明します。

ホットプラグ操作では、`cfgadm` コマンドに加えて、`prtconf` コマンドが便利です。`prtconf` コマンドはハードウェアに関連する追加の構成情報を表示します。

ハードウェアの追加後に `prtconf` コマンドを使用して、ハードウェアが正しく構成されているかどうかを確認します。たとえば、構成作業後に `prtconf -D` コマンドを使用して、新しくインストールしたハードウェアデバイスにドライバが接続されているかどうかを確認します。ハードウェアの構成の前にデバイスドライバがシステムに追加されていない場合、`add_drv` コマンドを使用して手動で追加された可能性があります。

詳細については、[prtconf\(1M\)](#) と [add_drv\(1M\)](#) のマニュアルページを参照してください。

次の例では、簡潔にするため、PCI 接続点だけを表示しています。画面に表示される接続点はシステムによって異なります。

PCIe LED インジケータの動作

システムの LED インジケータを監視して、スロットのホットプラグ動作のステータスを視覚的に把握できます。PCI Express の場合の LED の動作は、PCI Express の仕様で定義された動作と一致するか、そうでない場合は動作がプラットフォームに依存することがあります。

詳細は、ハードウェアガイドを参照してください。PCI Express の場合は、Attention ボタンが押されると電源インジケータが点滅し、状態遷移が始まったことを示します。状態遷移が終了すると、点滅が終了します。

▼ PCI スロット構成情報を表示する方法

この手順は、PCIe 構成情報を含めるように更新されました。

cfgadm コマンドは、システム上の PCI ホットプラグ対応デバイスとスロットのステータスを表示します。詳細については、[cfgadm\(1m\)](#) のマニュアルページを参照してください。

- 1 スーパーユーザーになります。
- 2 PCI 構成情報を表示します。
 - PCI スロット構成情報を表示します。

例:

```
# cfgadm
Ap_Id                Type      Receptacle  Occupant    Condition
pci1:hpc0_slot0      unknown   empty       unconfigured unknown
pci1:hpc0_slot1      unknown   empty       unconfigured unknown
pci1:hpc0_slot2      unknown   empty       unconfigured unknown
pci1:hpc0_slot3      ethernet/hp connected   configured  ok
pci1:hpc0_slot4      unknown   empty       unconfigured unknown
```

- 具体的な PCI デバイス情報を表示します。

例:

```
# cfgadm -s "cols=ap_id:type:info" pci
Ap_Id                Type      Information
pci1:hpc0_slot0      unknown   Slot 7
pci1:hpc0_slot1      unknown   Slot 8
pci1:hpc0_slot2      unknown   Slot 9
pci1:hpc0_slot3      ethernet/hp Slot 10
pci1:hpc0_slot4      unknown   Slot 11
```

論理 Ap_Id の pci1:hpc0_slot0 は、ホットプラグ対応のスロット Slot 7 の論理 Ap_Id です。コンポーネント hpc0 はこのスロットのホットプラグ対応のアダプタカードを示し、pci1 は PCI バスのインスタンスを示します。Type フィールドは、スロット中の PCI アダプタカードのタイプを示します。

- PCIe スロット構成情報を表示します。

例:

```
# cfgadm pci
Ap_Id                Type      Receptacle  Occupant    Condition
pcie1                unknown   empty       unconfigured unknown
pcie2                unknown   empty       unconfigured unknown
pcie3                unknown   empty       unconfigured unknown
pcie4                ethernet/hp connected   configured  ok
pcie5                pci-pci/hp connected   configured  ok
pcie6                unknown   disconnected unconfigured unknown
```

- 具体的な PCIe デバイス情報を表示します。

例:

```
# cfgadm -s "cols=ap_id:busy:o_state" pci
Ap_Id          Busy    Occupant
pcie1          n      unconfigured
pcie2          n      unconfigured
pcie3          n      unconfigured
pcie4          n      configured
pcie5          n      configured
pcie6          n      configured
```

注-ほとんどの場合、論理 Ap_Id は、システムのシャーシに貼られているスロットラベルと一致します。ハードウェアガイドでホットプラグ対応スロットの cfgadm の出力について参照してください。ホットプラグ操作を試す前に、Busy フィールドを表示して、Ap_Id が別の状態に遷移していないことを確認できます。

▼ PCI アダプタカードを取り外す方法

PCIe アダプタカードを取り外す場合のために、次の手順が更新されました。ただし、アダプタカードを取り外す手順は、PCI と PCIe のどちらを使用するかに関係なく同じです。

- 1 スーパーユーザーになります。
- 2 PCI アダプタカードが入っているスロットを確認します。

例:

```
# cfgadm pci
Ap_Id          Type      Receptacle  Occupant    Condition
pcie1          unknown  empty       unconfigured unknown
pcie2          unknown  empty       unconfigured unknown
pcie3          unknown  empty       unconfigured unknown
pcie4          etherne/hp connected  configured  ok
pcie5          pci-pci/hp connected  configured  ok
pcie6          unknown  disconnected unconfigured unknown
```

- 3 デバイスを開いているアプリケーションを停止します。
たとえば、デバイスが Ethernet カードの場合、ifconfig コマンドでインタフェースを無効にしてからカードを引き抜きます。例:

```
# ifconfig unplumb bge1
```

- 4 下に示すように cfgadm コマンドを使用してデバイスを手動で構成解除します。または、PCIe アダプタカードを使用している場合は、自動構成方式を使用します。たとえば、ハードウェアガイドに従ってスロットの **Attention** ボタンを押します。

```
# cfgadm -c unconfigure pcie4
```

- 5 デバイスの構成が解除されていることを確認します。

例:

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	empty	unconfigured	unknown
pcie4	unknown	connected	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

注 - デバイスが構成解除されると、Type と Condition も unknown になります。

- 6 スロットへの電源を手動で切り離します。自動構成方式を使用する場合は、この手順は不要です。

```
# cfgadm -c disconnect pcie4
```

詳細は、ハードウェアガイドを参照してください。

- 7 デバイスが切り離されていることを確認します。

例:

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	empty	unconfigured	unknown
pcie4	unknown	disconnected	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

- 8 プラットフォームのガイドの適切な手順に従って **PCI** アダプタカードを取り外します。カードを取り外すと、「**Receptable**」の状態が「**empty**」になります。

例:

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	empty	unconfigured	unknown
pcie4	unknown	empty	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

注 - プラットフォームの実装に応じて、ブート時に自動構成方式を有効または無効にすることができます。環境に適した自動構成方式を設定してください。

▼ PCI アダプタカードを取り付ける方法

PCIe アダプタカードを取り付ける場合のために、次の手順が更新されました。ただし、アダプタカードを追加する手順は、PCI と PCIe のどちらを使用するかに関係なく同じです。

- 1 スーパーユーザーになります。
- 2 ホットプラグ対応のスロットを確認して、ラッチを開きます。
たとえば、pcie3 の場合。

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	empty	unconfigured	unknown
pcie4	unknown	empty	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

- 3 ハードウェアガイドの適切な手順に従って、PCI アダプタカードをスロットに挿入します。
- 4 PCI アダプタカードを挿入した後に、どのスロットに PCI アダプタカードが入っているかを確認します。

例:

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	disconnected	unconfigured	unknown
pcie4	unknown	empty	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

- 5 **cfgadm** コマンドを使用して、手動で電源をスロットに接続します。または、**PCIe** アダプタカードを使用している場合は、自動構成方式を使用します。たとえば、ハードウェアガイドに従ってスロットの **Attention** ボタンを押します。

例:

```
# cfgadm -c connect pcie3
```

- 6 接続点が接続されていることを確認します。

例:

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	connected	unconfigured	unknown
pcie4	unknown	empty	unconfigured	unknown

pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

- 7 下に示すように **cfgadm** コマンドを使用して **PCI** アダプタカードを手動で構成します。自動構成方式を使用する場合は、この手順は不要です。詳細は、ハードウェアガイドを参照してください。

例:

```
# cfgadm -c configure pcie3
```

- 8 スロット中の **PCI** アダプタカードの構成を確認します。

例:

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	etherne/hp	connected	configured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

- 9 新しいデバイスの場合、サポートソフトウェアを構成します。
たとえば、デバイスが Ethernet カードの場合、**ifconfig** コマンドでインタフェースを設定します。例:

```
# ifconfig plumb bge1
```

注- プラットフォームの実装に応じて、ブート時に自動構成方式を有効または無効にすることができます。環境に適した自動構成方式を設定してください。

PCI 構成に関する問題のトラブルシューティング

エラーメッセージ

```
cfgadm: Configuration operation invalid: invalid transition
```

エラーの発生原因

無効な移行を行いました。

解決方法

cfgadm -c コマンドが適切に発行されているかどうかを確認します。**cfgadm** コマンドで現在の受容体と占有装置の状態を確認し、**Ap_id** が正しいことを確認します。

エラーメッセージ

```
cfgadm: Attachment point not found
```

エラーの発生原因

指定した接続点は見つかりません。

解決方法

接続点が正しいかどうかを確認します。cfgadm コマンドを使用して、利用可能な接続点のリストを表示します。物理パスを調べて、まだ接続点があるかどうかを確認してください。

cfgadm コマンドによる SATA ホットプラグ

SATA コントローラおよびポートマルチプライヤデバイスのポートは、デバイスツリー内の接続点で表されます。システム上で接続および構成されている SATA デバイスは、接続点の名前の拡張として表されます。接続点と SATA ポートという語は、相互に置き換えて使用できます。

SATA デバイスで使用される cfgadm の構文は、SCSI または PCI デバイス用の cfgadm の構文と若干異なります。

SATA デバイス情報は、次のようにして表示できます。

```
% cfgadm -al
Ap_Id                                Type      Receptacle  Occupant    Condition
sata0/0::dsk/c7t0d0                 disk      connected   configured  ok
sata0/1::dsk/c7t1d0                 disk      connected   configured  ok
sata0/2::dsk/c7t2d0                 disk      connected   configured  ok
sata0/3::dsk/c7t3d0                 disk      connected   configured  ok
.
.
.
```

▼ SATA デバイスの構成を解除する方法

一般に、SATA デバイスの取り外しおよび交換を行う前に、デバイスの構成を解除する必要があります。アクティブな ZFS ストレージプールの一部になっているデバイスの構成解除を試みると、次のようなエラーメッセージが表示されます。

```
# cfgadm -c unconfigure sata5/7
Unconfigure the device at: /devices/pci@2,0/pci1022...
This operation will suspend activity on the SATA device
Continue (yes/no)? y
cfgadm: Hardware specific failure: Failed to unconfig device at ap_id: /devices/pci@2,0/pci10...
```

- 1 スーパーユーザーになります。
- 2 構成を解除するデバイスを確認します。

```
# cfgadm -al | grep c7t0d0
sata0/0::dsk/c7t0d0                 disk      connected   configured  ok
```

- 3 デバイスの構成を解除します。

```
# cfgadm -c unconfigure sata0/0
```

デバイスを個別に指定して構成解除を試みた場合は、次のようなメッセージが表示されます。

```
# cfgadm -c unconfigure sata0/0::disk/c7t0d0
do_control_ioctl: open failed: errno:2
cfgadm: Library error: Cannot open ap_id: /devices/pci@0,0/pci10...
```

- 4 デバイスの構成が解除されていることを確認します。

```
# cfgadm | grep sata0/0
sata0/0                                disk          connected    unconfigured ok
```

▼ SATA デバイスを構成する方法

ディスクは物理的に取り外すか交換したあとで構成できます。

- 1 スーパーユーザーになります。

- 2 デバイスを構成します。

```
# cfgadm -c configure sata0/0
```

- 3 デバイスが構成されていることを確認します。

```
# cfgadm | grep sata0/0
sata0/0::disk/c7t0d0                disk          connected    configured    ok
```

Reconfiguration Coordination Manager (RCM) スクリプトの概要

Reconfiguration Coordination Manager (RCM) は、システムコンポーネントの動的な除去を管理するフレームワークです。RCM を使用すると、システムリソースを順番に登録および解放できます。

新しい RCM スクリプト機能を使用すると、アプリケーションを停止したり、動的な再構成の間にアプリケーションからデバイスを手際良く解放したりする独自のスクリプトを記述できます。スクリプトによって登録されたリソースに要求が影響を与える場合、RCM フレームワークは再構成要求に応じてスクリプトを自動的に起動します。

リソースを動的に除去する場合は、アプリケーションからリソースを手動で解放しておく必要があります。あるいは、`-f` オプションを指定して `cfgadm` コマンドを使用することで、再構成オペレーションを強制することも可能です。ただし、このオプションはアプリケーションを認識不能な状態のままにする可能性があります。また、アプリケーションからリソースを手動で解放すると、一般にエラーが発生します。

RCM スクリプト機能を使うと、動的再構成処理を簡単かつ効果的に実行できます。RCM スクリプトを作成すると、次の操作を実行できます。

- 動的にデバイスを取り外したときに、デバイスを自動的に解放します。デバイスがアプリケーションによって起動した場合は、この処理によって、デバイスも終了します。
- システムからデバイスを動的に取り外すときに、サイト固有のタスクを実行します。

RCM スクリプトについて

- 実行可能シェルスクリプト (Perl, sh, csh, または ksh) または RCM デーモンが実行するバイナリプログラム。推奨言語は、Perl です。
- スクリプトファイル所有者のユーザー ID を使用することにより、自分のアドレス領域で実行されるスクリプト。
- `cfgadm` コマンドを使ってシステムリソースを動的に再構成するときに、RCM デーモンによって実行されるスクリプト。

RCM スクリプトで実行できること

RCM スクリプトを使用した場合、デバイスを動的に取り外すと、デバイスがアプリケーションから解放されます。デバイスが開いている場合には、RCM スクリプトによって閉じられます。

たとえば、テープバックアップアプリケーションで RCM スクリプトを使用して、テープドライブを終了させたり、テープバックアップアプリケーションをシャットダウンしたりできます。

RCM スクリプト処理の動作方法

次のようにして RCM スクリプトを起動します。

```
$ script-name command [args ...]
```

RCM スクリプトにより、次の基本的な手順が実行されます。

1. コマンド行引数から RCM コマンドを取得します。
2. コマンドを実行する
3. 結果を名前と値のペアで `stdout` に記述します。
4. 適切な終了ステータスで終了します。

RCM デーモンは、スクリプトのインスタンスを同時に 1 つ実行します。たとえば、RCM デーモンは、スクリプトの実行中には、そのスクリプトが終了するまで同じスクリプトを実行しません。

RCM スクリプトコマンド

次の RCM コマンドを RCM スクリプトに含める必要があります。

- `scriptinfo` - スクリプト情報を収集します
- `register` - リソースに処理対象を登録します
- `resourceinfo` - リソース情報を収集します

次の RCM コマンドの一部またはすべてを RCM スクリプトに含めることができます。

- `queryremove` - リソースが解放されたかどうかを問い合わせます
- `preremove` - リソースを解放します
- `postremove` - リソースの削除後に通知します
- `undoremove` - `preremove` で実行された動作を元に戻します

これらの RCM コマンドの詳細は、[rcmscript\(4\)](#) のマニュアルページを参照してください。

RCM スクリプト処理環境

デバイスを動的に取り外すと、RCM デーモンにより次のコマンドが実行されます。

- スクリプトで識別されたリソースリスト (デバイス名) を収集するための、スクリプトの `register` コマンドが実行されます。
- スクリプトの登録されたリソースが動的な取り外し操作によって影響を受ける場合、リソースを取り外す前にスクリプトの `queryremove` および `preremove` コマンドが実行されます。
- 取り外し操作が成功した場合に、スクリプトの `postremove` コマンドが実行されます。ただし、取り外し操作に失敗した場合、RCM デーモンによりスクリプトの `undoremove` コマンドが実行されます。

RCM スクリプトでのタスク

次のセクションでは、アプリケーション開発者およびシステム管理者のために RCM スクリプトタスクについて説明します。

アプリケーション開発者 RCM スクリプト (タスクマップ)

次のタスクマップでは、RCM スクリプトを作成するアプリケーション開発者のタスクについて説明します。

タスク	説明	参照先
1. アプリケーションが使用するリソースを特定します。	アプリケーションが使用するリソース (デバイス名) を特定します。このデバイスは動的に取り外される可能性があります。	cfgadm(1m)
2. リソースを解放するコマンドを特定します。	アプリケーションからリソースを完全に解放するようにアプリケーションに通知するコマンドを特定します。	アプリケーションのドキュメント
3. リソースを取り外した後に使用するコマンドを特定します。	リソースを取り外したことをアプリケーションに通知するコマンドを含めます。	rcmscript(4)
4. リソースの取り外しに失敗した場合のコマンドを特定します。	使用可能なリソースについてアプリケーションに通知するコマンドを含めます。	rcmscript(4)
5. RCM スクリプトを記述します。	タスク 1-4 で特定した情報に基づいて RCM スクリプトを記述します。	104 ページの「テープバックアップ用の RCM スクリプトの例」
6. RCM スクリプトをインストールします。	適切なスクリプトディレクトリにスクリプトを追加します。	103 ページの「RCM スクリプトのインストール方法」
7. RCM スクリプトをテストします。	手動でスクリプトコマンドを実行し、動的再構成操作を実行してスクリプトをテストします。	103 ページの「RCM スクリプトのテスト方法」

システム管理者 RCM スクリプト (タスクマップ)

ここでは、サイトをカスタマイズするために RCM スクリプトを作成するシステム管理者のタスクについて説明します。

タスク	説明	参照先
1. 動的に削除するリソースを特定します。	<code>cfgadm -l</code> コマンドを使って削除する可能性があるリソース (デバイス名) を特定します。	cfgadm(1m)
2. 停止するアプリケーションを特定します。	アプリケーションを完全に停止させるコマンドを特定します。	アプリケーションのドキュメント
3. リソースの取り外し前および取り外し後のコマンドを特定します。	リソースを取り外す前後の動作を特定します。	rcmscript(4)

タスク	説明	参照先
4. RCM スクリプトを記述します。	タスク 1-3 で特定した情報に基づいて RCM スクリプトを記述します。	104 ページの「テープバックアップ用の RCM スクリプトの例」
5. RCM スクリプトをインストールします。	適切なスクリプトディレクトリにスクリプトを追加します。	103 ページの「RCM スクリプトのインストール方法」
6. RCM スクリプトをテストします。	手動でスクリプトコマンドを実行し、動的再構成操作を実行してスクリプトをテストします。	103 ページの「RCM スクリプトのテスト方法」

RCM スクリプトに名前を付ける

次の規則に従って、スクリプトに *vendor*、*service* という名前を付ける必要があります。

vendor スクリプトを提供するベンダーのストックシンボル、またはベンダーを識別する固有名です。

service スクリプトが表すサービス名です。

RCM スクリプトのインストールまたは削除

RCM スクリプトのインストールまたは削除を行うには、管理者になる必要があります。この表を使用して、RCM スクリプトをインストールするディレクトリを判断してください。

表 4-1 RCM スクリプトディレクトリ

ディレクトリの位置	スクリプトタイプ
/etc/rcm/scripts	特定のシステム用のスクリプト
/usr/platform/`uname -i`/lib/rcm/scripts	特定のハードウェア実装用のスクリプト
/usr/platform/`uname -m`/lib/rcm/scripts	特定のハードウェアクラス用のスクリプト
/usr/lib/rcm/scripts	任意のハードウェア用のスクリプト

▼ RCM スクリプトのインストール方法

- 1 管理者になります。
- 2 適切なディレクトリにスクリプトをコピーします。
表 4-1 を参照してください。
例:

```
# cp ABC,sample.pl /usr/lib/rcm/scripts
```
- 3 スクリプトのユーザー ID およびグループ ID を希望の値に変更します。

```
# chown user:group /usr/lib/rcm/scripts/ABC,sample.pl
```
- 4 **SIGHUP** を RCM デーモンに送信します。

```
# pkill -HUP -x -u root rcm_daemon
```

▼ RCM スクリプトの削除方法

- 1 管理者になります。
- 2 RCM スクリプトディレクトリからスクリプトを削除します。
例:

```
# rm /usr/lib/rcm/scripts/ABC,sample.pl
```
- 3 **SIGHUP** を RCM デーモンに送信します。

```
# pkill -HUP -x -u root rcm_daemon
```

▼ RCM スクリプトのテスト方法

- 1 スクリプトを実行する前にコマンド行シェルに **RCM_ENV_FORCE** などの環境変数を設定します。
たとえば、Korn シェルで次のように設定します。

```
$ export RCM_ENV_FORCE=TRUE
```
- 2 コマンド行から手動でスクリプトコマンドを実行してスクリプトをテストします。
例:

```
$ script-name scriptinfo
$ script-name register
$ script-name preremove resource-name
$ script-name postremove resource-name
```

- 3 スクリプトの各 RCM スクリプトコマンドにより、適切な出力結果が **stdout** に印刷されるかどうかを確認します。
- 4 適切なスクリプトディレクトリにスクリプトをインストールします。
詳細は、[103 ページの「RCM スクリプトのインストール方法」](#)を参照してください。
- 5 動的な削除操作を実行してスクリプトをテストします。
たとえば、スクリプトによってデバイス `/dev/dsk/c1t0d0s0` が登録されたとします。次のコマンドを実行してください。

```
$ cfgadm -c unconfigure c1::dsk/c1t0d0
$ cfgadm -f -c unconfigure c1::dsk/c1t0d0
$ cfgadm -c configure c1::dsk/c1t0d0
```



注意- 上記のコマンドは、システムの状態を変化させたり、システム障害を招くおそれもあるため、これらのコマンドを十分理解しておくことは大切です。

テープバックアップ用の RCM スクリプトの例

ここでは、RCM スクリプトを使用したテープバックアップの例を示します。

テープバックアップ用の RCM スクリプトの役割

テープバックアップ用の RCM スクリプトは、次の手順を実行します。

1. RCM コマンドのディスパッチテーブルを設定します。
2. 指定した RCM コマンドに対応するディスパッチルーチンを呼び出し、未実装の RCM コマンドのステータス 2 で終了させます。
3. `scriptinfo` セクションを設定します。

```
rcm_script_func_info=Tape backup appl script for DR
```

4. すべてのテープドライブのデバイス名を **stdout** に印刷して、すべてのテープドライバをシステムに登録します。

```
rcm_resource_name=/dev/rmt/$f
```

エラーが発生した場合、スクリプトによりエラー情報が **stdout** に出力されます。

```
rcm_failure_reason=$errmsg
```

5. テープデバイスのリソース情報を設定します。

```
rcm_resource_usage_info=Backup Tape Unit Number $unit
```

6. バックアップアプリケーションがそのデバイスを使用しているかどうか確認して、`preremove` 情報を設定します。バックアップアプリケーションがそのデバイスを使用していない場合、動的再構成操作が続行されます。バックアップアプリ

ケーションがそのデバイスを使用している場合、スクリプトにより RCM_ENV_FORCE が検査されます。RCM_ENV_FORCE が FALSE に設定されている場合、スクリプトにより動的再構成操作が拒否され、次のメッセージが印刷されます。

```
rcm_failure_reason=tape backup in progress pid=...
```

RCM_ENV_FORCE が TRUE に設定されている場合、バックアップアプリケーションが停止し、再構成操作が続行されます。

テープバックアップ再構成シナリオの結果

RCM スクリプトを使わずに `cfgadm` コマンドを使ってテープドライブを取り外した場合、次のような結果になります。

- バックアップアプリケーションがテープドライブを使用していない場合に `cfgadm` コマンドを使用すると、操作は正常に実行されます。
- バックアップアプリケーションがテープドライブを使用中に `cfgadm` コマンドを使用すると、操作が失敗します。

RCM スクリプトと `cfgadm` コマンドを使ってテープドライブを取り外した場合、次のような結果になります。

- バックアップアプリケーションがテープドライブを使用していない場合に `cfgadm` コマンドを使用すると、操作は正常に実行されます。
- バックアップアプリケーションがテープドライブを使用しているときに、`-f` オプションを指定せずに `cfgadm` コマンドを使用すると、次のようなエラーメッセージが表示され、操作が失敗します。

```
tape backup in progress pid=...
```

- バックアップアプリケーションがテープドライブを使用しているときに、`f` オプションを指定して `-cfgadm` コマンドを使用すると、スクリプトによってバックアップアプリケーションが停止され、`cfgadm` 操作が正常に実行されます。

例—テープバックアップ用の RCM スクリプト

```
#!/usr/bin/perl -w
#
# A sample site customization RCM script.
#
# When RCM_ENV_FORCE is FALSE this script indicates to RCM that it cannot
# release the tape drive when the tape drive is being used for backup.
#
# When RCM_ENV_FORCE is TRUE this script allows DR removing a tape drive
# when the tape drive is being used for backup by killing the tape
# backup application.
#

use strict;
```

```

    my ($cmd, %dispatch);
    $cmd = shift(@ARGV);
# dispatch table for RCM commands
    %dispatch = (
        "scriptinfo"    =>    \&do_scriptinfo,
        "register"       =>    \&do_register,
        "resourceinfo"  =>    \&do_resourceinfo,
        "queryremove"   =>    \&do_preremove,
        "preremove"     =>    \&do_preremove
    );

    if (defined($dispatch{$cmd})) {
        &{$dispatch{$cmd}};
    } else {
        exit (2);
    }

    sub do_scriptinfo
    {
        print "rcm_script_version=1\n";
        print "rcm_script_func_info=Tape backup appl script for DR\n";
        exit (0);
    }

    sub do_register
    {
        my ($dir, $f, $errmsg);

        $dir = opendir(RMT, "/dev/rmt");
        if (!$dir) {
            $errmsg = "Unable to open /dev/rmt directory: $!";
            print "rcm_failure_reason=$errmsg\n";
            exit (1);
        }

        while ($f = readdir(RMT)) {
            # ignore hidden files and multiple names for the same device
            if (($f !~ /\^\.\/) && ($f =~ /\^[0-9]+\$/)) {
                print "rcm_resource_name=/dev/rmt/$f\n";
            }
        }

        closedir(RMT);
        exit (0);
    }

    sub do_resourceinfo
    {
        my ($rsrc, $unit);

        $rsrc = shift(@ARGV);
        if ($rsrc =~ /\^\/dev\/rmt\/([0-9]+\$/)) {
            $unit = $1;
            print "rcm_resource_usage_info=Backup Tape Unit Number $unit\n";
            exit (0);
        } else {
            print "rcm_failure_reason=Unknown tape device!\n";
            exit (1);
        }
    }

```

```

    }
}

sub do_preremove
{
    my ($src);

    $src = shift(@ARGV);

    # check if backup application is using this resource
    #if (the backup application is not running on $src) {
        # allow the DR to continue
        #    exit (0);
    #}
    #
    # If RCM_ENV_FORCE is FALSE deny the operation.
    # If RCM_ENV_FORCE is TRUE kill the backup application in order
    # to allow the DR operation to proceed
    #
    if ($ENV{RCM_ENV_FORCE} eq 'TRUE') {
        if ($cmd eq 'preremove') {
            # kill the tape backup application
        }
        exit (0);
    } else {
        #
        # indicate that the tape drive can not be released
        # since the device is being used for backup by the
        # tape backup application
        #
        print "rcm_failure_reason=tape backup in progress pid=...\n"
;
        exit (3);
    }
}

```


USB デバイスの管理 (タスク)

この章では、Oracle Solaris OS で USB デバイスを使用する概要情報と手順について説明します。この章では特に、HAL サービスを使って USB デバイスを使用する方法について説明します。

この章の内容は次のとおりです。

- 109 ページの「USB デバイスの新機能」
- 110 ページの「Oracle Solaris の USB サポートについて」
- 116 ページの「USB デバイスの概要」
- 120 ページの「USB 大容量ストレージデバイスの管理」
- 139 ページの「USB オーディオデバイスの使用」
- 143 ページの「cfgadm コマンドを使った USB デバイスのホットプラグ」

USB デバイスの一般的な情報については、<http://www.oracle.com/technetwork/java/index.html> を参照してください。

動的再構成およびホットプラグについての一般的な情報については、第4章「デバイスの動的構成 (タスク)」を参照してください。

USB プリンタの構成については、印刷に関するシステム管理ガイドを参照してください。

USB デバイスの新機能

このリリースでは、次の機能が新しいか変更されています。

USB 3.0 のサポート

USB 3.0 のサポートは、新しい USB ホストコントローラドライバ `xhci`、ハブドライバの機能拡張、および Oracle Solaris USB Architecture (USBA) フレームワークの導入によって提供されています。USB 3.0 のハブと大容量ストレージデバイスは、USB 3.0

ポートに挿入すると、USB 3.0 モードで動作できるようになりました。さらに、レガシー USB デバイスも、USB オーディオデバイスを除き、USB 3.0 ポートに接続した場合に引き続き動作します。[110 ページの「サポートされる USB 機能」](#)を参照してください。

Oracle Solaris の USB サポートについて

このセクションでは次の情報について説明します。

- [110 ページの「サポートされる USB 機能」](#)
- [112 ページの「USB デバイスの機能および互換性の問題」](#)
- [112 ページの「バス電源供給方式のデバイス」](#)
- [113 ページの「USB キーボードとマウス」](#)
- [114 ページの「USB ホストコントローラとハブ」](#)
- [114 ページの「USB ハブデバイス」](#)
- [115 ページの「SPARC: USB 電源管理」](#)
- [116 ページの「USB ケーブルを接続するためのガイドライン」](#)

サポートされる USB 機能

この Oracle Solaris リリースには、xhci ホストコントローラドライバを使用する USB 3.0 デバイス (「SuperSpeed」デバイス) の拡張サポートが含まれます。xhci ホストコントローラドライバは、オーディオデバイスを除くすべての USB デバイスをサポートします。xhci ホストコントローラドライバは、スーパースピードおよびスーパースピードでないバスインタフェースを含む USB ホストの割り込み、制御、および一括転送をサポートします。USB 2.0、1.1、および 1.0 デバイスおよびドライバとの互換性がサポートされているため、同じケーブル、コネクタ、およびソフトウェアインタフェースを使用できます。

特に明記されていない場合、次の USB デバイスが SPARC ベースと x86 ベースの両方のシステムでサポートされます。

- USB 3.0 デバイスは、USB 3.0 ポートに挿入すると、USB 3.0 モードで動作します。レガシー USB デバイス (2.0、1.1、1.0) も、USB オーディオデバイスを除き、USB 3.0 ポートに接続した場合に引き続き動作します。
- オーディオデバイスは USB 2.0、1.1、および 1.0 オーディオデバイスのみサポートします。

USB ドライバによってサポートされていないデバイスには、gphoto2、gtkam、および pilotlink などの libusb アプリケーションが含まれていることがあります。詳細については、`/usr/share/doc/libusb/libusb.txt` 内のユーザー空間 USB ライブラリドキュメントを参照してください。

- 汎用 USB ドライバのサポート – [ugen\(7D\)](#) を参照してください。

- ヒューマンインターフェースデバイス (HID) のサポート (キーボードおよびマウスデバイス) – [hid\(7D\)](#) を参照してください。
- ハブのサポート – [hubd\(7D\)](#) を参照してください。
- プリンタのサポート
- USB CDC-ECM (USB Communication Device Class- Ethernet Control Model) デバイスのサポート
- 次のようなシリアルデバイスのサポート:
 - Edgeport – [usbser_edge\(7D\)](#) を参照してください。
 - Prolific – [usbsprl\(7D\)](#) を参照してください。
 - Keyspan – [usbsksp\(7D\)](#) を参照してください。
- ストレージデバイスのサポート – [scsa2usb\(7D\)](#) を参照してください。
- ユーザー空間 USB デバイス管理ライブラリのサポート。 [libusb\(3LIB\)](#) を参照してください。
- USB 3.0 バス速度が 480 Mビット/秒から 5 Gビット/秒に向上しました。

次のいずれかの USB 3.0 または USB 2.0 ポートを使用できます。

- USB 3.0 または USB 2.0 PCI カード上のポート
- USB 3.0 または USB 2.0 ポートに接続された USB 3.0 または USB 2.0 ハブ上のポート
- SPARC または x86 コンピュータのマザーボード上のポート

注 – 以前の SPARC プラットフォームでは、USB 2.0 PCI カードが必要な場合があります。

- 同じシステム上に USB 3.0、USB 2.0、および USB 1.0 デバイスが共存する場合でも、USB 2.0 および USB 1.1 デバイスは以前と同様に機能します。USB 2.0 および USB 1.0 ホストコントローラは、制御、一括、割り込み、アイソクロナスの 4 つすべての転送タイプをサポートします。USB 3.0 ホストコントローラは、制御、一括、割り込みの 3 つの転送タイプをサポートします。USB 3.0 ホストコントローラでは、アイソクロナス転送タイプはサポートされていません。
- USB 3.0 および USB 2.0 デバイスは USB 1.1 または 1.0 ポートでも動作しますが、USB 2.0 または USB 3.0 ポートに接続するとパフォーマンスが大幅に向上します。
- XHCI ホストコントローラには付属のコントローラがないため、単一のコントローラであらゆる速度の USB デバイスをサポートします。USB 2.0 ホストコントローラには、1 つの高速 EHCI および 1 つ以上の OHCI または UHCI 組み込みコントローラがあります。USB 2.0 ポートに接続されているデバイスは、USB 2.0 をサポートしているかどうかに応じて、EHCI または OHCI コントローラに動的に割り当てられます。

USB 3.0 デバイスのサポートの詳細については、[xhci\(7D\)](#) を参照してください。

USB 2.0 デバイスのサポートの詳細は、[ehci\(7D\)](#) および [usba\(7D\)](#) のマニュアルページを参照してください。

USB 1.1 デバイスのサポートの詳細については、[ohci\(7D\)](#) を参照してください。

USB デバイスおよび USB 用語については、[116 ページ](#)の「[USB デバイスの概要](#)」を参照してください。

USB デバイスの機能および互換性の問題

USB 3.0 デバイスは、USB 3.0 仕様に準拠した高速 (「SuperSpeed」) デバイスです。USB 2.0 デバイスは、USB 2.0 仕様に準拠した高速 (「Hi-Speed」) デバイスです。USB 3.0 および USB 2.0 仕様については、<http://www.usb.org/home> を参照してください。

USB デバイスの速度を識別するには、`/var/adm/messages` ファイルで次のようなメッセージを確認します。

```
Dec 13 17:05:57 mysystem usba: [ID 912658 kern.info] USB 2.0 device
(usb50d,249) operating at hi speed (USB 2.x) on USB 2.0 external hub:
storage@4, scsa2usb0 at bus address 4
```

この Oracle Solaris リリースでは、たとえば、次の USB デバイスがサポートされません。

- 大容量ストレージデバイス。たとえば CD-RW、ハードディスク、DVD、デジタルカメラ、フロッピーディスク、テープドライブ、メモリースティック、およびマルチフォーマットのカードリーダー
- キーボードおよびマウス
- オーディオデバイス。たとえばスピーカおよびマイク

注 - USB 3.0 では、オーディオデバイスをサポートできません。

それ以外のストレージデバイスでも、`scsa2usb.conf` ファイルを変更すれば使用することがあります。詳細は、[scsa2usb\(7D\)](#) のマニュアルページを参照してください。

バス電源供給方式のデバイス

バス電源供給方式のハブは、接続先の USB バスの電源を利用して、接続されているデバイスに電源を供給します。これらのハブの負荷が大きくなりすぎないように、十分に注意してください。これらのハブから接続先のデバイスに供給できる電源は限られているためです。

USB デバイス用の配電計画が実装されています。この機能には、次の制限があります。

- 2 台のバス電源供給方式のハブのカスケード接続は推奨されません。
- バス電源供給方式のハブの各ポートの最大消費電力は100mAです。
- バス電源供給方式のハブに接続できるのは、自己電源のデバイスまたはバス電源供給方式の低電力デバイスだけです。バス電源供給方式の高電力デバイスは接続を拒否されます。接続は予測できない場合があるため、誤った電源をレポートするハブやデバイスもあります。

USB キーボードとマウス

USB キーボードおよびマウスデバイスを使用するときは、次のことに注意してください。

- SPARC システムにおいて、リブート中または `ok` プロンプトの出ている間は、キーボードおよびマウスを移動しないでください。システムのリブート後、いつでもキーボードおよびマウスを別のハブに移動できます。キーボードおよびマウスは、差し込んだ後は再び完全に機能します。
- Sun 社製以外の USB キーボードでは、キーパッドの左側にある機能は使用できない場合があります。
- **SPARC** – SPARC システムにおいて USB キーボードおよびマウスデバイスを使用するときは、次のことに注意してください。
 - USB キーボードの電源キーとタイプ 5 キーボードの電源キーの動作は異なります。USB キーボードでは、「`SUSPEND/SHUTDOWN`」キーを使用してシステムを中断またはシャットダウンできます。ただし、そのキーを使用してシステムの電源を入れることはできません。
 - ブートプロセスが完了するまでは、OpenBoot PROM (OBP) の制限により、キーボードおよびマウスデバイスはマザーボードのルートハブのポートにしか接続できません。
 - レガシー SPARC システムでは、USB キーボードおよびマウスデバイスをタイプ 3、4、または 5 キーボードと同時に使用できません。
- 複数のキーボードおよびマウスデバイスのサポートについては、[virtualkm\(7D\)](#)のマニュアルページを参照してください。

USB ホイール付きマウスのサポート

次のホイール付きマウス機能がサポートされます。

- USB マウスまたは PS/2 マウスデバイスでは、3 ボタン以上の使用がサポートされています。
- USB マウスまたは PS/2 マウスデバイスでは、ホイール付きマウスによるスクロールが使用可能です。USB マウスまたは PS/2 マウスでホイールを回転させると、マウスが置かれたアプリケーションまたはウィンドウがスクロールします。StarSuite、Firefox、および GNOME アプリケーションはホイール付きマウスによるスクロール機能をサポートします。ただし、ほかのアプリケーションではこの機能がサポートされない場合があります。

USB ホストコントローラとハブ

USB ハブは次のタスクを実行します。

- ポートにおけるデバイスの取り付けと取り外しの監視
- ポートにおける個々のデバイスの電源管理
- ポートへの電源の制御

USB ホストコントローラにはルートハブという埋め込みハブがあります。システムのバックパネルに見えるポートは root ハブのポートです。

USB ホストコントローラは次のタスクを実行します。

- USB バスの管理。個々のデバイスはバスの調整はできません。
- デバイスによって決定されるポーリング間隔による、デバイスのポーリング。ポーリング間隔(時間)を考慮してデバイスに十分なバッファがあることを前提とします。
- USB ホストコントローラとそれに接続されているデバイス間でのデータの送信。ピアツーピア通信はサポートされません。

USB ハブデバイス

USB ハブデバイスを使用するときには、次の点に注意してください。

- SPARC システムと x86 システムのどちらにおいても、ハブは 4 段を超えて多段接続しないでください。SPARC システムでは、OpenBootPROM (OBP) は 4 段を超えるデバイスを正確に認識できません。
- バス電源供給方式のハブ同士をカスケード接続しないでください。バス電源供給方式のハブは独自の電源を持っていません。

- 大量の電源を必要とするデバイスをバス電源供給方式のハブに接続しないでください。これらのデバイスがバス電源供給方式のハブへの接続を拒否されたり、ほかのデバイス用の電源がなくなったりする可能性があります。このようなデバイスとして、USB フロッピーディスクデバイスなどがあります。

SPARC: USB 電源管理

SPARC システムでは、USB デバイスの保存停止および復元再開機能が完全にサポートされます。ただし、稼働中のデバイスを保存停止したり、システムの保存停止で電源がオフになっているときにデバイスを取り外すことは決してしないでください。

SPARC システムで電源管理を有効にしている場合、USB のフレームワークはすべてのデバイスの電源管理を最大限に試みます。USB デバイスの電源管理により、ハブドライバはデバイスが接続されているポートの中断も行います。「リモートウェイクアップ」をサポートするデバイスは、そのデバイスが利用可能な状態になるように、そのデバイスのバス上にあるすべてのデバイス呼び起こすようシステムに通知できます。アプリケーションがデバイスに入出力を送信した場合も、ホストシステムはデバイス呼び起こすことができます。

リモートウェイクアップ機能がサポートされている場合、すべての HID デバイス (キーボード、マウス、ハブ、およびストレージデバイス)、ハブデバイス、およびストレージデバイスは、デフォルトで電源管理されます。USB プリンタが電源管理されるのは、2つの印刷ジョブ間だけです。汎用の USB ドライバ (UGEN) で管理されているデバイスの電源は、デバイスが閉じているときにのみ管理されます。

電源消費を減らすために電源管理を行なっている場合は、まず USB 末端デバイスの電源が切断されます。また、ハブのポートに接続されているすべてのデバイスの電源が切断されると、しばらくしてからハブの電源が切断されます。もっとも効率的に電源管理をするためには、あまり多くのハブをカスケード接続しないでください。

SPARC システムの SUSPEND/SHUTDOWN キーの使用についての詳細は、[113 ページ](#)の「[USB キーボードとマウス](#)」を参照してください。

USB ケーブルを接続するためのガイドライン

USB ケーブルを接続する際には、次のガイドラインに従ってください。

- USB 3.0 仕様では、ケーブルの最大長を指定していません。ただし、USB 3.0 ケーブルは、USB 2.0 ケーブルとはっきり異なります。ケーブルが短いほどパフォーマンスが向上すると推定され、数ギガビットの転送速度を達成するには、3 m のケーブルが推奨されます。
- USB 2.0 デバイスの場合、これらのデバイスの接続には、常に準拠のフルレート (480 Mビット/秒) 20/28 AWG ケーブルを使用します。
- サポートされている最長ケーブル長は 5 m です。
- 延長ケーブルを使用しないでください。ケーブルを延長するには、最良の結果が得られるよう、自己電源供給方式のハブを使ってください。

詳細については、<http://www.usb.org/about/faq> を参照してください。

USB デバイスの概要

このセクションでは次の概要情報について説明します。

- 116 ページの「よく使用される USB 関連の略語」
- 117 ページの「Oracle Solaris USB アーキテクチャー」
- 118 ページの「USB バスの説明」
- 120 ページの「USB デバイスおよびドライバクラス」

Universal Serial Bus (USB) は PC 業界で開発された、周辺デバイス (キーボード、マウス、プリンタなど) をシステムに接続するための低コストのソリューションです。

USB コネクタは 1 方向 1 種類のケーブルだけに適合するように設計されています。USB が設計された主な目的は、デバイスごとに異なる何種類ものコネクタを減らすことです。USB の設計により、システムのバックパネルの混雑を軽減できます。

デバイスは、外部 USB ハブ上の USB ポートか、コンピュータ本体に設置されたルートハブ上の USB ポートのいずれかに接続されます。ハブには複数のポートがあるため、1 つのハブからデバイスツリーの複数の枝が伸びることがあります。

詳細については、[usb\(7D\)](usb(7D)) または <http://www.usb.org/home> を参照してください。

よく使用される USB 関連の略語

次の表に、Oracle Solaris OS で使用される USB の略語を示します。USB のコンポーネントと略語についての詳細は、<http://www.usb.org/home> を参照してください。

略語	定義	参照先
UGEN	USB 汎用ドライバ	ugen(7D)
USB	Universal Serial Bus (ユニバーサルシリアルバス)	usb(7D)
USBDA	USB アーキテクチャー (Solaris)	usbda(7D)
USBDAI	USBDA クライアントドライバインタフェース (Solaris)	なし
HCD	USB ホストコントロールドライバ	なし
EHCI	拡張ホストコントローラインタフェース	ehci(7D)
OHCI	オープンホストコントローラインタフェース	ohci(7D)
UHCI	ユニバーサルホストコントローラインタフェース	uhci(7D)
XHCI	拡張ホストコントローラインタフェース	xhci(7D)

Oracle Solaris USB アーキテクチャー

USB デバイスは、2つのレベルのデバイスツリーノードとして表現できます。デバイスノードは、USB デバイス全体を表します。1つまたは複数の子インタフェースノードはデバイス上にある個々のUSB インタフェースを表します。

ドライバのバインドは互換性のある名前属性の使用によって実現されます。詳細については、『IEEE 1275 USB binding (英語版)』の3.2.2.1 項と『[デバイスドライバの記述](#)』を参照してください。ドライバは、デバイス全体にバインドしてすべてのインタフェースを制御することも、1つのインタフェースだけにバインドすることも可能です。デバイス全体にバインドするドライバがベンダーにもクラスにも存在しない場合、汎用 USB マルチインタフェースドライバがデバイスレベルのノードにバインドされます。IEEE 1275 バインド仕様の3.2.2.1 項で定義されているように、このドライバは互換名プロパティを使用して、各インタフェースに対してドライバのバインドを試みます。

Oracle Solaris USB アーキテクチャー (USBDA) は、USB 1.1、USB 2.0、および USB 3.0 の仕様に準拠しており、Oracle Solaris デバイスドライバインタフェース (DDI) の一部です。USBDA モデルは Oracle Common SCSI Architecture (SCSA) に似ています。次の図が示すように、USBDA は、汎用 USB トランスポート層という概念をクライアントドライバに提供する薄い層で、汎用 USB の主要な機能を実装するサービスをクライアントドライバに提供します。

図 5-1 Oracle Solaris USB アーキテクチャ (USBA)



USB バスの説明

USB 仕様は、ライセンス料を払わずに入手できます。USB 仕様は、バスとコネクタの電気的および機械的なインタフェースを定義します。

USB が採用するトポロジでは、ハブが USB デバイスに接続点を提供します。ホストコントローラには、システム内のすべての USB ポートの起点となるルートハブが含まれます。ハブの詳細は、[114 ページの「USB ホストコントローラとハブ」](#)を参照してください。

図 5-2 USB 物理デバイスの階層

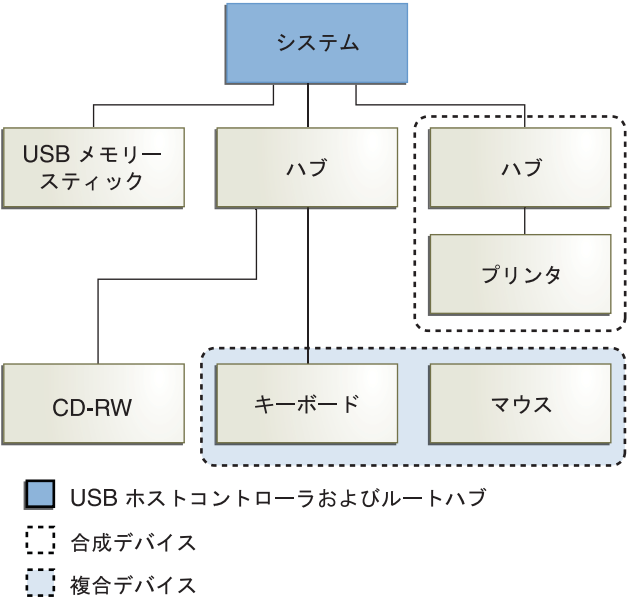


図 5-2 は、有効な USB ポートが 3 つ搭載されたシステムを示しています。1 番目の USB ポートは USB メモリースティックに接続されています。2 番目の USB ポートは外部ハブに接続されており、このハブには CD-RW デバイスと、キーボードとマウスの複合デバイスが接続されています。このキーボードは「複合デバイス」であるため USB コントローラが組み込まれており、このコントローラによって、キーボードとキーボードに接続されたマウスの両方が制御されます。キーボードとマウスは、同じ USB コントローラによって制御されるため、同一の USB バスアドレスを共有します。

図 5-2 は、ハブとプリンタの合成デバイスの例も示しています。このハブは外部ハブで、プリンタと同じケースに入っています。プリンタはこのハブに固定接続されます。このハブとプリンタは、それぞれ異なる USB バスアドレスを持ちます。

次の表に、図 5-2 に示したデバイスの一部について、デバイスツリーパス名を一覧表示します。

メモリースティック	/pci@1f,4000/usb@5/storage@1
キーボード	/pci@1f,4000/usb@5/hub@2/device@1/keyboard@0
マウス	/pci@1f,4000/usb@5/hub@2/device@1/mouse@1
cdrw デバイス	/pci@1f,4000/usb@5/hub@2/storage@3
プリンタ	/pci@1f,4000/usb@5/hub@3/printer@1

USB デバイスおよびドライバクラス

属性とサービスが似ている USB デバイスは、いくつかのデバイスクラスに分類されます。各デバイスクラスには対応するドライバが1つずつ存在しています。クラス内のデバイスは、同じ組み合わせのデバイスドライバによって管理されます。ただし、USB 仕様では、特定のクラスに属さないベンダー固有のデバイスも許可しています。

HID クラスには、ユーザーが制御する次のようなデバイスが含まれます。

- キーボード
- マウスデバイス
- ジョイスティック

Communication Device クラスには、次のデバイスが含まれます。

- モデム
- Ethernet アダプタ

その他にも、次のようなデバイスクラスがあります。

- オーディオ
- モニター
- プリンタ
- ストレージデバイス

各 USB デバイスはデバイスのクラスを表す記述子を持っています。デバイスクラスは、そのメンバーが構成とデータ転送についてどのように動作するかを指定します。追加のクラス情報を取得するには、<http://www.usb.org/home/> を参照してください。

Oracle Solaris リリースでサポートされる USB デバイスの詳細は、[usb\(7D\)](#) を参照してください。

USB 大容量ストレージデバイスの管理

このセクションでは次のタスクについて説明します。

- [122 ページの「USB フロッピーディスクデバイスの使用」](#)
- [123 ページの「USB 大容量ストレージデバイスのホットプラグ」](#)
- [126 ページの「USB 大容量ストレージデバイスを使用するための準備」](#)
- [127 ページの「USB デバイス情報を表示する方法」](#)
- [128 ページの「USB 大容量ストレージデバイス上にファイルシステムを作成する方法」](#)

- 130 ページの「USB 大容量ストレージデバイス上でパーティションを変更し、PCFS ファイルシステムを作成する方法」
- 133 ページの「USB 大容量ストレージデバイス上に Solaris パーティションを作成してスライスを変更する方法」
- 135 ページの「USB 大容量ストレージデバイスをマウントまたはマウント解除する方法」
- 137 ページの「USB 大容量ストレージデバイスの問題のトラブルシューティング」
- 137 ページの「特定の USB ドライバを無効にする」
- 138 ページの「特定の USB ドライバを無効にする方法」
- 138 ページの「使用されていない USB デバイスのリンクを削除する方法」

次の USB リムーバブル大容量ストレージデバイスがサポートされます。

- CD-RW
- ハードディスク
- DVD
- デジタルカメラ
- フロッピーディスクデバイス
- SmartMedia および CompactFlash デバイス

以前の Oracle Solaris リリースでは、すべての USB ストレージデバイスはリムーバブルメディアデバイスとして識別されていました。これにより、自動マウントをはじめ、次に示す利点の大部分が実現されていました。本リリースでは、USB 大容量ストレージデバイスがホットプラグ可能デバイスとして認識されますが、次のセクションで説明する USB リムーバブルデバイスの利点も得られます。ホットプラグ可能動作の詳細については、[123 ページの「USB 大容量ストレージデバイスのホットプラグ」](#)を参照してください。

USB 大容量ストレージデバイスを管理するためのガイドラインは次のとおりです。

- ホットプラグ対応デバイスは自動的にマウントされます。
- 標準の MS-DOS または Windows (FAT) ファイルシステムを使用する USB ストレージデバイスがサポートされます。
- 使いやすい `rmformat` コマンドを使用して、スライスを作成できます。`fdisk` コマンドを使用して USB デバイスのパーティションを作成できますが、`format` ユーティリティまたは `rmformat -F` コマンドを使用して USB デバイスを物理的にフォーマットしないでください。
- `rmformat` コマンドを使用して、メディアが挿入されているすべての USB デバイスを表示します。例については、[127 ページの「USB デバイス情報を表示する方法」](#)を参照してください。
- `mount` コマンドが必要なくなったため、ルート以外のユーザーでも USB ストレージデバイスにアクセスできます。デバイスは自動的にマウントされ、`/rmdisk` ディレクトリの下で利用できます。

- これらのデバイスは、リムーバブルメディアサービスを実行している場合でも実行していない場合でも管理できます。
- FAT ファイルシステムを持つディスクをマウントし、アクセスできるようになりました。例:

```
mount -F pcfs /dev/dsk/c2t0d0s0:c /mnt
```

- LOG SENSE ページをサポートするデバイスを除き、すべての USB ストレージデバイスの電源管理が行われます。LOG SENSE ページを使用するデバイスは通常、USB-SCSI ブリッジデバイスを介して接続する SCSI デバイスです。
- USB 大容量ストレージデバイスでは、アプリケーションの動作が異なる場合があります。USB ストレージデバイスでアプリケーションを使用するときには、次の点に考慮してください。
 - 以前のリリースではフロッピーディスクなどの小容量のデバイスだけがリムーバブルメディアとして認識されていたため、アプリケーションがメディアのサイズを正しく認識しないことがあります。
 - メディアを取り出すことができないデバイス(ハードディスクドライブなど)に対して、メディアを取り出す要求を行うと、要求は成功しますが、何も実行されません。
 - 以前の Oracle Solaris リリースの動作が必要な場合、つまり、すべての USB 大容量ストレージがリムーバブルメディアデバイスとして認識されるようにするには、`/etc/driver/drv/scsa2usb.conf` ファイルを更新すれば、以前の動作を強制的に適用できます。

USB 大容量ストレージデバイスの詳しい使用方法については、[scsa2usb\(7D\)](#) のマニュアルページを参照してください。

USB フロッピーディスクデバイスの使用

USB フロッピーディスクはリムーバブルメディアデバイスとして表示されます。USB フロッピーディスクデバイスは、`fd`(フロッピー)ドライバでは管理されません。アプリケーションから `fd`(ネイティブフロッピー)ドライバに対する `ioctl(2)` 呼び出しを実行すると、そのアプリケーションは失敗します。`read(2)` および `write(2)` 呼び出しのみを実行するアプリケーションは成功します。`SunPCI` や `rmformat` などのその他のアプリケーションも成功します。

USB フロッピーディスクデバイスは SCSI リムーバブルメディアデバイスとして認識されます。デバイスは `/rmdisk` ディレクトリの下で利用できます。

USB フロッピーディスクデバイスの使用方法の詳細については、[116 ページの「USB デバイスの概要」](#)を参照してください。

USB 大容量ストレージデバイスのホットプラグ

デバイスのホットプラグとは、オペレーティングシステムをシャットダウンすることなくあるいはシステムの電源を切ることなく、デバイスを追加したり取り外したりすることを指します。USB デバイスはすべてホットプラグ対応です。

hot-pluggable デバイス属性は、システムをリブートせずに接続または切り離しできたり、ユーザーの介入なしで自動的に構成および構成解除できたりするデバイスを識別します。すべての USB デバイスは、ホットプラグ可能デバイスとして識別され、それらの利点が得られます。また、リムーバブルメディアでない USB デバイスは、リムーバブルメディアデバイスとして識別されなくなり、*removable-media* 属性を持たなくなりました。

リムーバブルでない USB ストレージデバイスはドライバレベルでホットプラグ可能デバイスとして識別されます。この動作は、システムをリブートせずにこれらのデバイスを接続または取り外しできること、およびユーザーの介入なしで自動的に構成または構成解除できることを意味します。これらの変更は、カーネルレベルで行われているため、デバイスの使用には影響しません。たとえば、これらのデバイスのマウントおよびマウント解除は、リムーバブルメディア管理サービスによって制御されます。

これらのデバイスの使用方法の詳細については、[scsa2usb\(7D\)](#) を参照してください。

リムーバブルメディアマネージャーがホットプラグ対応のデバイスを認識するようになりました。デバイスをプラグインするだけで、数秒でマウントされます。何も起こらない場合はデバイスがマウントされているかどうかを確認してください。

リムーバブルメディアサービスが動作していることを確認します。

```
# svcs -a volfs
STATE      STIME      FMRI
online      Sep_11     svc:/system/filesystem/volfs:default
```

デバイスが有効で認識されている場合は、デバイスからファイルシステムをマウントできます。

マウントに失敗した場合は、`vold` を停止します。

```
# svcadm disable volfs
```

デバイス上のファイルシステムが自動的にマウントされない場合は、手動によるマウントを試みます。

デバイスをホットリムーブする前に、`eject -l` コマンドの別名からそのデバイスの名前を探します。次に、デバイスのメディアを取り出します。この処理を行わない場

合でも、デバイスが解放されてポートが使用できる状態に戻りますが、デバイス上のファイルシステムが破損する場合があります。

USB デバイスは、ホットプラグを実行するとすぐにシステムのデバイス階層に表示されます (prtconf コマンドで確認可能)。USB デバイスを使用していないときにそのデバイスを取り外すと、システムのデバイス階層からそのデバイスが削除されます。

デバイスを取り外したときにそのデバイスを使用していた場合、デバイスノードは残りますが、このデバイスを制御しているドライバはデバイス上のすべての動作を停止します。このデバイスに発行されるすべての新しい入出力動作はエラーを返します。

このような状況で、システムは元のデバイスを差し込むように要求します。デバイスが使用できない場合は、アプリケーションを停止してください。数秒後に、ポートが再び使用できるようになります。

注- 動作中の、つまり開いているデバイスを削除すると、データの整合性が損なわれる可能性があります。デバイスを取り外す前には、必ずデバイスを閉じるようにしてください。ただし、接続されているキーボードとマウスは例外で、動作中でも移動することができます。

▼ USB 大容量ストレージデバイスを追加する方法

- 1 スーパーユーザーになります。
- 2 USB 大容量ストレージデバイスを接続します。
- 3 USB デバイスが追加されたことを確認します。

例:

```
$ rmformat
Looking for devices...
  1. Logical Node: /dev/rdisk/c3t0d0p0
     Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
     Connected Device: SanDisk  Cruzer Micro      0.3
     Device Type: Removable
     Bus: USB
     Size: 245.0 MB
     Label: <None>
     Access permissions: Medium is not write protected.
```

- 4 デバイスが `/rmdisk` ディレクトリの下に自動的にマウントされていることを確認します。

例:

```
$ ls /rmdisk/noname
aa  bb
```

▼ USB カメラを追加する方法

カメラのメディアが PCFS ファイルシステムを使用している場合は、自動的にマウントされます。デバイスが `scsa2usb` ドライバにバインドされない場合は、`libusb` アプリケーションを使用して写真を転送します。詳細については、`/usr/share/doc/libusb/libusb.txt` ファイルを参照してください。

- 1 USB カメラを差し込んで電源を入れます。

システムによって、カメラ用の論理デバイスが作成されます。カメラが差し込まれると、`/var/adm/messages` ファイルにメッセージが出力され、デバイスの接続が確認されます。システムでは、カメラはストレージデバイスとして扱われます。

- 2 `/var/adm/messages` ファイルの出力を確認します。

```
# more /var/adm/messages
```

出力表示を確認すると、どの論理デバイスが作成されたかを確認でき、そのデバイスを使用してイメージにアクセスできます。出力表示は次のようになります。

```
Jul 15 09:53:35 buffy usba: [ID 349649 kern.info]    OLYMPUS, C-3040ZOOM,
000153719068
Jul 15 09:53:35 buffy genunix: [ID 936769 kern.info] scsa2usb1 is
/pci@0,0/pci925,1234@7,2/storage@2
Jul 15 09:53:36 buffy scsi: [ID 193665 kern.info] sd3 at scsa2usb1:
target 0 lun 0
```

次のコマンドを実行して、デバイスをマウント可能な `/dev/dsk` リンクエントリに関連付けます。

```
# ls -l /dev/dsk/c*0 | grep /pci@0,0/pci925,1234@7,2/storage@2
lrwxrwxrwx  1 root    root      58 Jun 14  2010 c3t0d0p0 ->
../../../../devices/pci@0,0/pci925,1234@7,2/storage@2/disk@0,0:a
```

- 3 USB カメラファイルシステムをマウントします。

ほとんどの場合、カメラのファイルシステムは PCFS ファイルシステムです。ファイルシステムが PCFS の場合は、自動的にマウントされます。

- x86 システム上でファイルシステムを手動でマウントするには、次のような構文を使用します。

```
# mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt
```

- SPARC システム上でファイルシステムを手動でマウントするには、次のような構文を使用します。

```
# mount -F pcfs /dev/dsk/c3t0d0s0:c /mnt
```

ファイルシステムのマウント方法については、[315 ページの「Oracle Solaris ファイルシステムのマウントおよびマウント解除」](#)を参照してください。

さまざまな PCFS ファイルシステムのマウント方法については、[mount_pcfs\(1M\)](#) のマニュアルページを参照してください。

- 4 (省略可能) イメージファイルが使用可能であることを確認します。

例:

```
# ls /mnt/DCIM/100OLYMP/  
P7220001.JPG* P7220003.JPG* P7220005.JPG*  
P7220002.JPG* P7220004.JPG* P7220006.JPG*
```

- 5 (省略可能) USB カメラが作成したイメージファイルを表示します。

例:

```
# /usr/dt/bin/sdtimage P7220001.JPG &
```

- 6 カメラを切り離す前に、ファイルシステムをマウント解除します。

例:

```
# umount /mnt
```

- 7 (省略可能) カメラの電源をオフにし、切り離します。

USB 大容量ストレージデバイスを使用するための準備

リムーバブルメディア上の情報にアクセスするには、リムーバブルメディアサービスを使用する方法と使用しない方法があります。GNOME のファイルマネージャーを使用してリムーバブルメディア上の情報にアクセスする方法については、GNOME デスクトップのドキュメントを参照してください。

USB デバイスがフォーマットされると、通常は `/rmdisk/label` ディレクトリの下にマウントされます。

デバイスノードは、文字型デバイスについては `/dev/rdisk` ディレクトリ、ブロック型デバイスについては `/dev/dsk` ディレクトリの下に作成されます。デバイスリンクは、デバイスのホットプラグを実行したときに作成されます。詳細は、[scsa2usb\(7D\)](#) のマニュアルページを参照してください。

リムーバブルメディアサービスでデバイスを認識できない場合は、デバイスを手動でマウントしてみます。最初に `rmformat` コマンドを使用してデバイスパスを特定してから、`mount` コマンドを使用して管理者としてデバイスを手動でマウントします。

デバイスをリムーバブルメディアサービスを使ってマウントした場合は、`rmumount` コマンドを使ってマウントを解除できます。デバイスを手動でマウントした場合は、`umount` コマンドを管理者として使ってマウントを解除します。

USB デバイスのマウントおよびマウント解除の方法については、[135 ページの「USB 大容量ストレージデバイスをマウントまたはマウント解除する方法」](#)を参照してください。

▼ USB デバイス情報を表示する方法

- USB デバイスに関する情報を表示します。

この例の `prtconf` の出力は、USB デバイス情報のみを表示するように省略されています。

```
$ prtconf
usb, instance #0
  hub, instance #2
    device, instance #8
      interface (driver not attached)
    printer (driver not attached)
    mouse, instance #14
    device, instance #9
      keyboard, instance #15
      mouse, instance #16
    storage, instance #7
      disk (driver not attached)
    communications, instance #10
      modem (driver not attached)
      data (driver not attached)
    storage, instance #0
      disk (driver not attached)
    storage, instance #1
      disk (driver not attached)
```

USB ストレージデバイス情報を表示するには、`rmformat` コマンドを使用します。

```
$ rmformat
Looking for devices...
1. Logical Node: /dev/rdisk/c3t0d0p0
   Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
   Connected Device: SanDisk  Cruzer Micro      0.3
   Device Type: Removable
   Bus: USB
   Size: 245.0 MB
   Label: <None>
   Access permissions: Medium is not write protected.
```


▼ USB 大容量ストレージデバイス上にファイルシステムを作成する方法

USB フロッピーディスクは、ファイルシステムを追加する前にフォーマットする必要があります。その他のすべての USB 大容量ストレージデバイスを使用するには、ファイルシステムが必要です。

USB デバイスをフォーマットするときには、次の点に注意してください。

- USB フロッピーディスク以外で `rmformat -F` を使用しないでください。
- デフォルトのスライスを使用しない場合は、`rmformat -s` コマンドを使用してスライスを作成します。必要に応じて、`fdisk` ユーティリティを使用して USB デバイスにパーティションを作成します。手順については、次を参照してください。
 - [130 ページの「USB 大容量ストレージデバイス上でパーティションを変更し、PCFS ファイルシステムを作成する方法」](#)
 - [133 ページの「USB 大容量ストレージデバイス上に Solaris パーティションを作成してスライスを変更する方法」](#)
- USB デバイスが自動的にマウントされている場合は、USB デバイス上にファイルシステムを作成する前に、マウント解除する必要があります。

注-手順4と5は、USB フロッピーディスクをフォーマットする必要がある場合のみ実行してください。

- 1 スーパーユーザーになります。
- 2 USB デバイスをシステムに追加します。USB デバイスのホットプラグを実行する方法については、次を参照してください。
 - [123 ページの「USB 大容量ストレージデバイスのホットプラグ」](#)
 - [143 ページの「`cfgadm` コマンドを使った USB デバイスのホットプラグ」](#)
- 3 (オプション)USB デバイスを特定します。

```
# rmformat
Looking for devices...
1. Logical Node: /dev/rdsd/c2t0d0p0
   Physical Node: /pci@0,0/pci108e,534a@2,1/hub7/floppy@1/disk@0,0
   Connected Device: MITSUMI   USB FDD           1039
   Device Type: Floppy drive
   Bus: USB
   Size: 1.4 MB
   Label: <None>
   Access permissions: Medium is not write protected.
```

この例では、フロッピーディスクデバイスは `c2t0d0p0` です。

- 4 必要に応じて、フロッピーディスクをフロッピーディスクドライブに挿入します。

- 5 必要に応じて、フロッピーディスクをフォーマットします。

```
# rmformat -F long raw-device
```

- 6 ファイルシステムのタイプを判別し、デバイスがマウント解除されていることを確認します。次のいずれかの手順に従います。

USB デバイスのマウント解除の方法については、[135 ページの「USB 大容量ストレージデバイスをマウントまたはマウント解除する方法」](#)を参照してください。

- データを別のシステムに転送する必要がある場合は、USB スティック上に ZFS プールおよびファイルシステムを作成します。

```
# zpool create c5t0d0 temp-pool
# zfs create temp-pool/data
```

- PCFS ファイルシステムを作成します。

```
# mkfs -F pcfs -o nofdisk,size=size raw-device
```

-size オプションを 512 バイトブロック単位で指定します。

次の例は、SPARC システムにおいて 1.4M バイトのフロッピーディスクに PCFS ファイルシステムを作成する方法を示しています。

```
# mkfs -F pcfs /dev/rdisk/c2t0d0p0
Construct a new FAT file system on /dev/rdisk/c2t0d0p0: (y/n)? y
```

次の例は、x86 システムにおいて 1.4M バイトのフロッピーディスクに PCFS ファイルシステムを作成する方法を示しています。

```
# mkfs -F pcfs /dev/rdisk/c2t0d0s2
Construct a new FAT file system on /dev/rdisk/c2t0d0s2: (y/n)? y
```

次の例は、SPARC システムにおいて 100M バイトの USB メモリースティックに PCFS ファイルシステムを作成する方法を示しています。

```
# mkfs -F pcfs /dev/rdisk/c5t0d0s2:c
```

次の例は、x86 システムにおいて 100M バイトの USB メモリースティックに PCFS ファイルシステムを作成する方法を示しています。

```
# mkfs -F pcfs /dev/rdisk/c5t0d0p0:c
```

このコマンドの実行には、数分かかることがあります。

- レガシー UFS ファイルシステムを作成します。

```
# newfs raw-device
```

大容量の USB ハードディスクの場合は、newfs -f 4096 オプションまたは newfs -T オプションの使用を検討してください。

注- フロッピーディスクの記憶容量は少ないので、UFS ファイルシステムのためにフロッピーディスクのかんりの容量が消費されます。

PCFS ファイルシステムの作成と、USB 大容量ストレージデバイス上のスライスの変更の詳細な例については、130 ページの「USB 大容量ストレージデバイス上でパーティションを変更し、PCFS ファイルシステムを作成する方法」および133 ページの「USB 大容量ストレージデバイス上に Solaris パーティションを作成してスライスを変更する方法」を参照してください。

▼ USB 大容量ストレージデバイス上でパーティションを変更し、PCFS ファイルシステムを作成する方法

次の手順では、USB デバイス上で既存のパーティションを削除し、新しいパーティションを作成してから PCFS ファイルシステムを作成する方法を説明します。このタスクを実行する前にデータをバックアップしてください。この手順には、ガイドラインとしてのみ使用できる特定の例が含まれています。指定する情報は、使用している特定のシステムに適用させてください。

- 1 スーパーユーザーになります。
- 2 **fdisk** ユーティリティを起動します。
`# fdisk /dev/rdisk/c3t0d0p0`
- 3 オプション 3 を選択してパーティションを削除します。

```
Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

      Cylinders
Partition  Status  Type          Start  End  Length  %
=====  =====  =====
      1      Active  Solaris2       1    28     28    97
```

```
SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)
Enter Selection: 3
```

4 削除するパーティションの番号を選択します。

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====
1	Active	Solaris2	1	28	28	97

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Specify the partition number to delete (or enter 0 to exit): 1

Partition deleted.

5 パーティションを作成します。

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 1

6 パーティションタイプとしてFAT32を選択します。

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)

```
6. Cancel (exit without updating disk configuration)
Select the partition type to create:
1=SOLARIS2  2=UNIX      3=PCIX0S    4=Other
5=DOS12     6=DOS16     7=DOSEXT    8=DOSBIG
9=DOS16LBA  A=x86 Boot   B=Diagnostic C=FAT32
D=FAT32LBA  E=DOSEXTLBA  F=EFI       0=Exit? c
```

7 このパーティションに使用するディスクの割合を指定します。

```
Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

Partition  Status  Type          Cylinders  Start  End  Length  %
=====  =====  =====
WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)
Select the partition type to create:
Specify the percentage of disk to use for this partition (or type "c" to
specify the size in cylinders). 100
```

8 新しいパーティションをアクティブなパーティションにするか非アクティブなパーティションにするかを選択します。

```
Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

Partition  Status  Type          Cylinders  Start  End  Length  %
=====  =====  =====
WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)
Select the partition type to create:
Should this become the active partition? If yes, it will be activated
each time the computer is reset or turned on.
Please type "y" or "n". n
```

9 ディスク構成を更新して終了します。

```
Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks
```

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====
1		Win95 FAT32	1	28	28	97

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 5

- 10 このパーティション上に **PCFS** ファイルシステムを作成します。

新しいファイルシステムを作成する前にデバイスがマウント解除されていることを確認します。USB デバイスのマウント解除の方法については、[135 ページの「USB 大容量ストレージデバイスをマウントまたはマウント解除する方法」](#)を参照してください。

```
# mkfs -F pcfs -o fat=32 /dev/rdisk/c3t0d0p0:c
Construct a new FAT file system on /dev/rdisk/c3t0d0p0:c: (y/n)? y
```

▼ USB 大容量ストレージデバイス上に **Solaris** パーティションを作成してスライスを変更する方法

次の手順では、Solaris パーティションを作成してスライスを変更する方法を示します。この手順には、ガイドラインとしてのみ使用できる特定の例が含まれていません。指定する情報は、使用している特定のシステムに適用させてください。

このタスクを実行する前にデータをバックアップしてください。

- 1 スーパーユーザーになります。
- 2 **fdisk** ユーティリティを起動します。

```
# fdisk /dev/rdisk/c5t0d0s2
No fdisk table exists. The default partition for the disk is:

    a 100% "SOLARIS System" partition

Type "y" to accept the default partition, otherwise type "n" to edit the
partition table.
y
```

3 現在のスライスを表示します。

例:

```
# prtvtoc /dev/rdisk/c5t0d0s2
* /dev/rdisk/c5t0d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   63 sectors/track
*   255 tracks/cylinder
*   16065 sectors/cylinder
*   5836 cylinders
*   5836 accessible cylinders
*
* Flags:
*   1: unmountable
*   10: read-only
*
*
* Partition Tag Flags First Sector Count Last Sector Mount Directory
*   0      0    00      0 93755340 93755339
*   2      0    00      0 93755340 93755339
```

4 スライスの情報を含むテキストファイルを作成します。

例:

```
slices: 0 = 0, 5GB, "wm", "home" :
        1 = 8225280000, 6GB :
        2 = 0, 44GB, "wm", "backup" :
        6 = 16450560000, 15GB
```

各スライスがシリンダ境界から始まっていることを確認します。たとえば、スライス 1 は 822280000 バイトから始まっています。この数は、シリンダサイズ (バイト数) の 1000 倍になります。

詳細は、[rmformat\(1\)](#) の `-s` オプションの説明を参照してください。

5 上記の手順で作成したスライスファイルを含むスライスを作成します。

例:

```
# rmformat -s slice_file /dev/rdisk/c5t0d0s2
```

6 新しいスライスの情報を表示します。

例:

```
# prtvtoc /dev/rdisk/c5t0d0s2
* /dev/rdisk/c5t0d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   63 sectors/track
*   255 tracks/cylinder
*   16065 sectors/cylinder
*   5836 cylinders
*   5836 accessible cylinders
```

```

*
* Flags:
*   1: unmountable
*  10: read-only
*
* Unallocated space:
*      First      Sector      Last
*      Sector      Count      Sector
*   10485760    5579240    16064999
*   28647912    3482088    32129999
*   63587280    30168060    93755339
*
*
* Partition Tag  Flags      First      Sector      Last
* Partition Tag  Flags      Sector      Count      Sector  Mount Directory
*   0         8    00         0    10485760    10485759
*   1         3    01    16065000    12582912    28647911
*   2         5    00         0    92274688    92274687
*   6         4    00    32130000    31457280    63587279

```

▼ USB 大容量ストレージデバイスをマウントまたはマウント解除する方法

次の手順では、USB 大容量ストレージデバイスをマウントおよびマウント解除する方法を示します。

- 1 スーパーユーザーになります。
- 2 (オプション)デバイスを特定します。

例:

```

$ rmformat
Looking for devices...
  1. Logical Node: /dev/rdisk/c3t0d0p0
     Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
     Connected Device: SanDisk  Cruzer Micro      0.3
     Device Type: Removable
     Bus: USB
     Size: 245.0 MB
     Label: <None>
     Access permissions: Medium is not write protected.

```

この例では、物理フロッピーディスクデバイスは `c2t0d0p0` です。

- 3 USB 大容量ストレージデバイスをマウントまたはマウント解除するには、次のいずれかの方法を選択します。
 - USB 大容量ストレージデバイスをコンソールユーザーとしてマウントします。
次のように、デバイスニックネーム、マウントポイント、またはデバイスパスを指定して `rmmount` コマンドを使用することもできます。

```
$ rmmount rmdisk0
$ rmmount NONAME
$ rmmount /dev/dsk/c3t0d0p0:1
```

例:

```
$ rmmount NONAME
NONAME /dev/dsk/c2t0d0p0 mounted
$ ls /media/NONAME
AA.TXT
```

- USB 大容量ストレージデバイスをコンソールユーザーとしてマウント解除します。

例:

```
$ rmumount NONAME
NONAME /dev/dsk/c2t0d0p0 unmounted
```

- USB 大容量ストレージデバイスをスーパーユーザーとしてマウントします。
次の例は、UFS ファイルシステムを使用しているデバイスのマウント方法を示しています。

```
$ mount /dev/dsk/c1t0d0s2 /mnt
```

次の例は、SPARC システムにおいて、PCFS ファイルシステムを使用しているデバイスをマウントする方法を示しています。

```
$ mount -F pcfs /dev/dsk/c3t0d0s2:c /mnt
```

次の例は、x86 システムにおいて、PCFS ファイルシステムを使用しているデバイスをマウントする方法を示しています。

```
$ mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt
```

次の例は、読み取り専用の HSFS ファイルシステムを使用している CD のマウント方法を示しています。

```
$ mount -F hsfs -o ro /dev/dsk/c1t0d0s2 /mnt
```

- USB 大容量ストレージデバイスをスーパーユーザーとしてマウント解除します。
デバイス上のファイルシステムを使用しているユーザーがいないことを最初に確認してください。

例:

```
$ fuser -c -u /mnt
$ umount /mnt
```

- 4 DVD、CD、またはフロッピーディスクデバイスの場合には、デバイスを取り出します。

例:

```
$ eject /dev/rdisk/c1t0d0s2
```


USB 大容量ストレージデバイスの問題のトラブルシューティング

USB 大容量ストレージデバイスの追加または取り外しで問題が発生した場合には、次のヒントを参考にしてください。

`/var/adm/messages` ファイルでデバイスの列挙の失敗を確認します。列挙の失敗の場合は、USB ハブを挿入するか、ハブを取り外してルート USB ハブに直接接続します。

- システムの稼働中に接続したデバイスにアクセスするときに、問題が発生した場合は、次のコマンドを実行してください。

`# devfsadm`

- 保存停止モードでシステムの電力消費を抑えている場合は、デバイスを移動しないでください。詳細は、[115 ページ](#)の「**SPARC: USB 電源管理**」を参照してください。
- アプリケーションがデバイスを使用しているときに、そのデバイスが取り外されて使用できなくなっている場合は、アプリケーションを停止してください。そのデバイスノードが削除されているかどうかを確認するには、`prtconf` コマンドを使用します。

特定の USB ドライバを無効にする

特定の種類の USB デバイスを無効にするには、対応するクライアントドライバを無効にします。たとえば、USB プリンタを無効にするには、そのプリンタを使用している `usbprn` ドライバを無効にします。`usbprn` を無効にしても、USB ストレージデバイスなどのほかのデバイスには影響しません。

次の表に、USB デバイスの種類の一部とそれらに対応するドライバを示します。

デバイスの種類	無効にするドライバ
オーディオ	<code>usb_ac</code> および <code>usb_as</code>
HID (通常はキーボードとマウス)	<code>hid</code>
ストレージ	<code>scsa2usb</code>
プリンタ	<code>usbprn</code>
シリアル	<code>usbser_edge</code>

システムに接続されている USB デバイスのドライバを無効にすると、次のようなコンソールメッセージが表示されます。

```
usb10: WARNING: usba:    no driver found for device name
```

▼ 特定の **USB** ドライバを無効にする方法

- 1 スーパーユーザーになります。
- 2 ドライバの別名エントリを `/etc/system` ファイルから除外します。
たとえば、次の `exclude` 文を追加して `usbprn` ドライバを除外します。

```
exclude: usbprn
```

- 3 システムをリブートします。

```
# init 6
```

▼ 使用されていない **USB** デバイスのリンクを削除する方法

システムの電源がオフのときに USB デバイスを取り外した場合には、次の手順を実行します。システムの電源が切断されているときに USB デバイスを取り外すと、存在しないデバイスへのリンクが残る場合があります。

- 1 スーパーユーザーになります。
- 2 そのデバイスにアクセスする可能性のあるアプリケーションをすべて閉じます。
- 3 特定の **USB** クラスの未使用のリンクを削除します。

例:

```
# devfsadm -C -c audio
```

または、関連するリンクをすべて削除します。

```
# devfsadm -C
```

USB オーディオデバイスの使用

このセクションでは次のタスクについて説明します。

- [140 ページの「複数の USB オーディオデバイスのホットプラグ」](#)
- [140 ページの「USB オーディオデバイスを追加する方法」](#)
- [141 ページの「システムのプライマリオードデバイスを識別する方法」](#)
- [142 ページの「プライマリ USB オーディオデバイスを変更する方法」](#)
- [142 ページの「USB オーディオデバイスに関する問題のトラブルシューティング」](#)

注 - USB 3.0 には、オーディオデバイスのサポートが含まれません。

特定の Oracle Solaris リリースでの USB オーディオサポートについては、[110 ページの「Oracle Solaris の USB サポートについて」](#)を参照してください。

Oracle Solaris USB オーディオのサポートは、連携するドライバ `usb_ac` と `usb_as` の組み合わせによって実装されています。オーディオコントロールドライバである `usb_ac` は Solaris USB Architecture 準拠のクライアントドライバで、ユーザーアプリケーションのインタフェースを制御します。オーディオストリーミングドライバである `usb_as` は、再生中および録音中にオーディオデータメッセージを処理します。また、サンプル周波数と精度を設定し、`usb_ac` ドライバからの要求をエンコードします。どちらのドライバも、USB オーディオクラス 1.0 仕様に準拠しています。

一部のオーディオデバイスでは、ソフトウェアが制御している音量を設定できません。この機能を管理するために、STREAMS モジュールの `usb_ah` が HID ドライバの先頭に置かれます。

Oracle Solaris では、再生専用、録音専用、録音および再生用の USB オーディオデバイスをサポートします。

USB オーディオデバイスのホットプラグは次のようにサポートされます。

- 完全にサポートされているオーディオデータ形式を確認するには、[usb_ac\(7D\)](#) のマニュアルページを参照してください。

プライマリオードデバイスは、`/dev/audio` です。次のコマンドを使用して、`/dev/audio` が USB オーディオを指しているかを確認できます。

```
%$ mixerctl
Device /dev/audioctl:
  Name      = USB Audio
  Version   = 1.0
  Config    = external

Audio mixer for /dev/audioctl is enabled
```

USB オーディオデバイスを接続した後、`audioplay` コマンドおよび `audiorecord` コマンドを使用し、`/dev/sound/N` デバイスリンクを介してデバイスにアクセスします。

`/dev/audio` および `/dev/sound/N` デバイスは、スピーカ、マイク、またはコンボデバイスを参照できます。不正なデバイスタイプを参照すると、そのコマンドは失敗します。たとえば、マイクに対して `audioplay` を使用しようとする、そのコマンドは失敗します。

ほとんどの Oracle オーディオアプリケーションでは、特定のデフォルトオーディオデバイスを選択できます。たとえば、`audioplay` や `audiorecord` の場合には、`AUDIODEV` シェル変数を設定するか、`-d` オプションを指定します。ただし、`/dev/audio` をオーディオファイルとしてハードコードしている他社製のアプリケーションでは `AUDIODEV` は動作しません。

USB オーディオデバイスを差し込むと、`/dev/audio` が使用中でない限り、自動的にそれが主オーディオデバイス `/dev/audio` になります。オンボードのオーディオから USB オーディオへ、および USB オーディオからオンボードのオーディオへ `/dev/audio` を変更する方法については、[142 ページの「プライマリ USB オーディオデバイスを変更する方法」](#) および [usb_ac\(7D\)](#) のマニュアルページを参照してください。

複数の USB オーディオデバイスのホットプラグ

USB オーディオデバイスがシステムに差し込まれると、それが主オーディオデバイス `/dev/audio` になります。そのデバイスは、システムをリブートした後でも、主オーディオデバイスのままです。USB オーディオデバイスが追加で差し込まれた場合、最後に差し込まれたデバイスが主オーディオデバイスになります。

USB オーディオデバイスに関する問題のトラブルシューティングについて、さらに詳しい情報を得るには、[usb_ac\(7D\)](#) のマニュアルページを参照してください。

▼ USB オーディオデバイスを追加する方法

- 1 USB スピーカを差し込みます。

プライマリオーディオデバイス `/dev/audio` は、USB スピーカを指します。

```
$ ls -l /dev/audio
lrwxrwxrwx  1 root    root    10 Feb 13 08:46 /dev/audio -> usb/audio0
```

- 2 (オプション) スピーカを取り外します。その後、再度差し込みます。

スピーカを取り外すと、`/dev/audio` デバイスがオンボードのオーディオに戻ります。

```
$ ls -l /dev/audio
lrwxrwxrwx  1 root    root    7 Feb 13 08:47 /dev/audio -> sound/0
```

3 USBマイクを追加します。

```
$ ls -l /dev/audio
lrwxrwxrwx 1 root root 10 Feb 13 08:54 /dev/audio -> usb/audio1
```

▼ システムのプライマリオードデバイスを識別する方法

この手順は、すでに USB オーディオデバイスが接続されていることを前提としています。

- システムの新しいオーディオリンクを調べます。

- ls コマンドを使用して、システムの新しいオーディオリンクを表示します。

例:

```
$ ls -lt /dev/audio*
lrwxrwxrwx 1 root root 7 Jul 23 15:46 /dev/audio -> usb/audio0
lrwxrwxrwx 1 root root 10 Jul 23 15:46 /dev/audiocctl ->
usb/audiocctl0/
% ls -lt /dev/sound/*
lrwxrwxrwx 1 root root 74 Jul 23 15:46 /dev/sound/1 ->
.././devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:...
lrwxrwxrwx 1 root root 77 Jul 23 15:46 /dev/sound/1ctl ->
.././devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:...
lrwxrwxrwx 1 root other 66 Jul 23 14:21 /dev/sound/0 ->
.././devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audio
lrwxrwxrwx 1 root other 69 Jul 23 14:21 /dev/sound/0ctl ->
.././devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audiocctl
$
```

プライマリオードデバイス /dev/audio が、新しく差し込まれた USB オーディオデバイスの /dev/usb/audio0 を指していることがわかります。

- prtconf コマンドを使用して USB デバイス情報を参照して、システム上の USB オーディオデバイスを調べることもできます。

```
$ prtconf
.
.
.
usb, instance #0
  hub, instance #0
    mouse, instance #0
    keyboard, instance #1
    device, instance #0
      sound-control, instance #0
      sound, instance #0
      input, instance #0
.
.
.
```

▼ プライマリ USB オーディオデバイスを変更する方法

- プライマリ USB オーディオデバイスを変更するには、次のいずれかの手順を選択します。
 - オンボードのオーディオデバイスをプライマリオーディオデバイスにするには、**USB** オーディオデバイスを取り外してください。すると、`/dev/audio` リンクは `/dev/sound/0` エントリを指すはずです。`/dev/sound/0` エントリがプライマリオーディオデバイスでない場合は、システムをシャットダウンして **boot -r** コマンドを実行するか、**devfsadm -i** コマンドをスーパーユーザーとして実行してください。
 - **USB** オーディオデバイスをプライマリオーディオデバイスにするには、そのデバイスを差し込んでデバイスリンクを確認するだけです。

USB オーディオデバイスに関する問題のトラブルシューティング

ドライバを適用し、音量も上げているのに、USB スピーカから音が出ないことがあります。デバイスのホットプラグを実行してもこの動作が変わらないことがあります。

この問題を解決するには、USB スピーカの電源を再投入します。

オーディオデバイスの所有権に関する注意事項

オーディオデバイスを操作するときは、オーディオデバイスの所有権に関する、次に挙げる点に注意してください。

- USB オーディオデバイスを差し込む時にコンソールにログインしていると、コンソールが `/dev/*` エントリの所有者になります。つまり、コンソールにログインしているかぎり、オーディオデバイスを使用できることになります。
- USB オーディオデバイスを差し込むときにコンソールにログインしていない場合、**root** がそのデバイスの所有者になります。ただし、その後にコンソールにログインして USB オーディオデバイスにアクセスしようすると、デバイスの所有権はコンソールに変更されます。詳細は、[logindevperm\(4\)](#) を参照してください。
- リモートから **rlogin** コマンドでログインして USB オーディオデバイスにアクセスしようとした場合は、所有権は変更されません。たとえば、権限のないユーザーが、ほかの人の所有するマイクを通して行われる会話を聞くことはできません。

cfgadm コマンドを使った USB デバイスのホットプラグ

このセクションでは次のタスクについて説明します。

- 144 ページの「USB バス情報を表示する方法(cfgadm)」
- 145 ページの「USB デバイスの構成を解除する方法」
- 145 ページの「USB デバイスの構成方法」
- 146 ページの「論理的に USB デバイスを接続解除する方法」
- 146 ページの「論理的に USB デバイスを接続する方法」
- 147 ページの「論理的に USB デバイスのサブツリーを接続解除する方法」
- 147 ページの「USB デバイスのリセット方法」
- 147 ページの「複数の構成を持つ USB デバイスのデフォルト構成を変更する方法」

cfgadm コマンドを使用せずに稼働中のシステムから USB デバイスを追加または削除できます。ただし、USB デバイスは、デバイスを物理的に削除しなくても「論理的に」ホットプラグを実行できます。この方法は、リモートで作業中に機能していない USB デバイスを無効にしたりリセットしたりする必要がある場合に便利です。cfgadm コマンドを使うと、製造元や製品情報を含む USB デバイスツリーを表示することもできます。

cfgadm コマンドは接続点についての情報を表示します。「接続点」とは、動的再構成を行うことができるシステム内の特定の場所のことです。

接続点は、次の要素から構成されています。

- 占有装置 (occupant)。USB デバイスなどの、システムに構成可能なハードウェアリソースのことです
- 受容体 (receptacle)。USB ポートなどの、占有装置を受け入れる場所のことです

接続点は、論理と物理の両方の接続点 ID (Ap_Id) で表現されます。物理 Ap_Id は接続点の物理的なパス名です。論理 Ap_Id は物理 Ap_Id に代わるユーザーに理解しやすい ID です。Ap_Id の詳細は、[cfgadm_usb\(1M\)](#) のマニュアルページを参照してください。

cfgadm コマンドを使用すると、USB デバイスステータス情報を取得できます。

受容体の状態	説明
empty/unconfigured	デバイスが物理的に接続されていません。
disconnected/unconfigured	デバイスは物理的に接続されているかもしれませんが、論理的に接続解除されており利用できません。
connected/unconfigured	デバイスは論理的に接続されていますが利用できません。このデバイスは、prtconf の出力に表示されます。

受容体の状態	説明
connected/configured	デバイスは接続されており利用可能です。

次のセクションでは、ソフトウェアから `cfgadm` コマンドを使用して USB デバイスのホットプラグを実行する方法について説明します。次のすべてのサンプル USB デバイス情報は、関連した情報に焦点を合わせるために一部省略されています。

▼ USB バス情報を表示する方法 (cfgadm)

`prtconf` コマンドを使用して USB 構成情報を表示する例については、[127 ページ](#)の「[USB デバイス情報を表示する方法](#)」を参照してください。

1 USB バス情報を表示します。

例:

```
$ cfgadm
Ap_Id                Type      Receptacle  Occupant    Condition
usb0/4.5             usb-hub   connected   configured  ok
usb0/4.5.1           usb-device connected   configured  ok
usb0/4.5.2           usb-printer connected   configured  ok
usb0/4.5.3           usb-mouse  connected   configured  ok
usb0/4.5.4           usb-device connected   configured  ok
usb0/4.5.5           usb-storage connected   configured  ok
usb0/4.5.6           usb-communi connected   configured  ok
usb0/4.5.7           unknown   empty       unconfigured ok
```

前の例で `usb0/4.5.1` は、第2レベルの外部ハブのポート1に接続されているデバイスを識別します。この第2レベルハブは第1レベルの外部ハブのポート5に接続されており、また第1レベルのハブは最初の USB コントローラのルートハブであるポート4に接続されています。

2 特定のUSB デバイス情報を表示します。

例:

```
$ cfgadm -l -s "cols=ap_id:info"
Ap_Id      Information
usb0/4.5.1  Mfg: Inside Out Networks Product: Edgeport/421 NConfigs: 1
Config: 0 : ...
usb0/4.5.2  Mfg: <undef> Product: <undef> NConfigs: 1 Config: 0 ...
usb0/4.5.3  Mfg: Mitsumi Product: Apple USB Mouse NConfigs: 1
Config: 0 ...
usb0/4.5.4  Mfg: NMB Product: NMB USB KB/PS2 M NConfigs: 1 Config: 0
usb0/4.5.5  Mfg: Hagiwara Sys-Com Product: SmartMedia R/W NConfigs: 1
Config: 0 : ...
usb0/4.5.6  Mfg: 3Com Inc. Product: U.S.Robotics 56000 Voice USB Modem
NConfigs: 2 ...
usb0/4.5.7
```


▼ USB デバイスの構成を解除する方法

システムに物理的に接続されている USB デバイスの構成を解除することはできません。しかし、そのデバイスにドライバを適用することはできません。USB デバイスの構成を解除しても、そのデバイスは `prtconf` 出力に表示されることに注意してください。

- 1 スーパーユーザーになります。
- 2 USB デバイスの構成を解除します。

例:

```
# cfgadm -c unconfigure usb0/4.7
Unconfigure the device: /devices/pci@8,7000000/usb@5,3/hub@4:4.7
This operation will suspend activity on the USB device
Continue (yes/no)? y
```

- 3 デバイスの構成が解除されていることを確認します。

例:

```
# cfgadm
Ap_Id                Type          Receptacle  Occupant    Condition
usb0/4.5             usb-hub       connected   configured  ok
usb0/4.5.1           usb-device    connected   configured  ok
usb0/4.5.2           usb-printer   connected   configured  ok
usb0/4.5.3           usb-mouse     connected   configured  ok
usb0/4.5.4           usb-device    connected   configured  ok
usb0/4.5.5           usb-storage   connected   configured  ok
usb0/4.5.6           usb-communi   connected   configured  ok
usb0/4.5.7           unknown      empty       unconfigured ok
usb0/4.6             usb-storage   connected   configured  ok
usb0/4.7             usb-storage   connected   unconfigured ok
```

▼ USB デバイスの構成方法

- 1 スーパーユーザーになります。
 - 2 USB デバイスを構成します。
- 例:
- ```
cfgadm -c configure usb0/4.7
```
- 3 USB デバイスが構成されていることを確認します。

例:

```
cfgadm usb0/4.7
Ap_Id Type Receptacle Occupant Condition
usb0/4.7 usb-storage connected configured ok
```

## ▼ 論理的に USB デバイスを接続解除する方法

物理的にシステムの近くにいないときに、システムから USB デバイスを取り外し、prtconf 出力を削除する場合、USB デバイスの接続を論理的に解除できます。デバイスは物理的に接続されたままです。しかし、論理的に接続解除され、使用できなくなり、システムにも表示されません。

- 1 スーパーユーザーになります。
- 2 USB デバイスを接続解除します。

例:

```
cfgadm -c disconnect -y usb0/4.7
```

- 3 デバイスが接続解除されていることを確認します。

例:

```
cfgadm usb0/4.7
Ap_Id Type Receptacle Occupant Condition
usb0/4.7 unknown disconnected unconfigured ok
```

## ▼ 論理的に USB デバイスを接続する方法

次の手順を使用して、以前に論理的に接続解除または構成解除された USB デバイスを論理的に接続します。

- 1 スーパーユーザーになります。
- 2 USB デバイスを接続します。

例:

```
cfgadm -c configure usb0/4.7
```

- 3 デバイスが接続されていることを確認します。

例:

```
cfgadm usb0/4.7
Ap_Id Type Receptacle Occupant Condition
usb0/4.7 usb-storage connected configured ok
```

デバイスを利用できるようになり、システムにも表示されるようになります。

## ▼ 論理的に **USB** デバイスのサブツリーを接続解除する方法

次の手順を使用して、USB デバイスのサブツリーを接続解除します。サブツリーは、ハブの下位デバイスの階層 (ツリー) です。

- 1 スーパーユーザーになります。
- 2 **USB** デバイスのサブツリーを接続解除します。  
例:
- 3 **USB** デバイスサブツリーの接続解除を確認します。  
例:

```
cfgadm usb0/4
Ap_Id Type Receptacle Occupant Condition
usb0/4 unknown disconnected unconfigured ok
```

## ▼ **USB** デバイスのリセット方法

USB デバイスでエラーが発生した場合は、`cfgadm` コマンドを使ってデバイスをリセットします。このコマンドを使うと、デバイスを論理的に削除し、再作成できます。

- 1 スーパーユーザーになります。
- 2 デバイスが使用中でないことを確認します。
- 3 デバイスをリセットします。  
例:
- 4 デバイスが接続されていることを確認します。  
例:

```
cfgadm -x usb_reset -y usb0/4.7
```

```
cfgadm usb0/4.7
Ap_Id Type Receptacle Occupant Condition
usb0/4.7 usb-storage connected configured ok
```

## ▼ 複数の構成を持つ **USB** デバイスのデフォルト構成を変更する方法

複数の構成を持つ USB デバイス进行操作するときには、次の点を考慮してください。

- USB デバイスの構成には、デバイス自体がどのようにオペレーティングシステムに表示されるかを定義します。この構成方法は、前のセクションで `cfgadm` について説明したシステムデバイスの構成方法とは異なります。
- 一部の USB デバイスでは、複数の構成がサポートされますが、一度に有効にできる構成は 1 つだけです。
- 複数の構成を持つデバイスを特定するには、`cfgadm -lv` の出力を確認します。Nconfigs は、1 より大きい値になります。
- デフォルトの USB 構成は configuration 1 です。現在の構成は、`cfgadm -lv` の出力に Config として反映されます。
- デフォルトの構成を変更しても、デバイスを同じポートに再接続している間は、デバイスのリブート、ホットリムーブ、および再構成を行なっても、構成の変更は適用されません。

1 デバイスが使用中でないことを確認します。

2 デフォルトの USB 構成を変更します。

例:

```
cfgadm -x usb_config -o config=2 usb0/4
Setting the device: /devices/pci@1f,0/usb@c,3:4
to USB configuration 2
This operation will suspend activity on the USB device
Continue (yes/no)? yes
```

3 デバイスの変更を確認します。

例:

```
cfgadm -lv usb0/4
Ap_Id Receptacle Occupant Condition Information When Type
 Busy Phys_Id
usb0/4 connected unconfigured ok Mfg: Sun 2000
Product: USB-B0B0 aka Robotech
With 6 EPPS High Clk Mode NConfigs: 7 Config: 2 : EVAL Board Setup
unavailable
usb-device n /devices/pci@1f,0/usb@c,3:4
```

Config が「2」になっていることに注目してください。

## InfiniBand デバイスの使用 (概要/タスク)

---

InfiniBand (IB) は、スイッチファブリックに基づく入出力テクノロジーです。この技術により、入出力デバイスとホストとの接続やホスト間の通信で、帯域幅が広く応答時間の短い相互接続が提供されます。

この章の内容は以下のとおりです。

- 149 ページの「[InfiniBand デバイスの概要](#)」
- 152 ページの「[IB デバイスの動的再構成 \(cfgadm\)](#)」

IB デバイスの使用手順については、次の節を参照してください。

- 151 ページの「[IB デバイスの動的再構成 \(タスクマップ\)](#)」
- 161 ページの「[InfiniBand デバイスでの uDAPL アプリケーションインタフェースの使用](#)」

動的再構成およびホットプラグについての一般的な情報については、[第4章「デバイスの動的構成 \(タスク\)」](#)を参照してください。

## InfiniBand デバイスの概要

IB デバイスは Solaris IB 連結ドライバによって管理されます。このドライバは、次の5種類のデバイスをサポートしています。

- IB Port デバイス
- IB 仮想物理接続点 (VPPA) デバイス
- IB HCA サービス (HCA\_SVC) デバイス
- 疑似デバイス
- 入出力コントローラ (IOC) デバイス

IB 連結ドライバは、Solaris IB デバイスマネージャー (IBDM) にサービス (このマニュアルでは「通信サービス」と呼ぶ) を照会して、IB Port、HCA\_SVC、および IB VPPA デバイスを列挙します。

Port デバイスは、Host Channel Adapter (HCA) の特定のポート番号に通信サービスをバインドします。これに対し、VPPA デバイスは、ポート番号とパーティションキー番号の組み合わせに通信サービスをバインドします。HCA\_SVC デバイスは、特定の HCA に通信サービスをバインドします。Port デバイスと HCA\_SVC デバイスでは、パーティションキー `p_key` の値として常に 0 が使用されます。Port、HCA\_SVC、および VPPA デバイスは、HCA の子であり、`ib.conf` ファイルから列挙されます。詳細は、[ib\(7D\)](#) のマニュアルページを参照してください。

IOC デバイスは、IB 連結ドライバの子であり、入出力ユニットの一部です。擬似デバイスも IB 連結ドライバの子です。独自の構成ファイルを持つほかのすべてのデバイスを参照して列挙されます。詳細は、[ib\(4\)](#) のマニュアルページを参照してください。

次の表に、IB デバイスツリーのパス名の形式を示します。

|                 |                                                                                                   |
|-----------------|---------------------------------------------------------------------------------------------------|
| IOC デバイス        | <code>/ib/ioc@1730000007F510C,1730000007F50</code>                                                |
| IB 擬似デバイス       | <code>/ib/&lt;driver&gt;@&lt;unit-address&gt;</code>                                              |
| IB VPPA デバイス    | <code>/pci@1f,2000/pci@1/pci15b3,5a44@0/ibport@&lt;port#&gt;,&lt;p_key&gt;,&lt;service&gt;</code> |
| IB HCA_SVC デバイス | <code>/pci@1f,2000/pci@1/pci15bc,5a44@0/ibport@0,0,&lt;service&gt;</code>                         |
| IB Port デバイス    | <code>/pci@1f,2000/pci@1/pci15b3,5a44@0/ibport@&lt;port#&gt;,0,&lt;service&gt;</code>             |
| HCA             | <code>/pci@1f,2000/pci@1/pci15b3,5a44@0</code>                                                    |

IB HCA\_SVC デバイスの `port#` および `p_key` は 0 です。

上記の表で、IB のコンポーネントはそれぞれ次のものを表します。

- `<services>` 通信サービスです。たとえば、`ipib` は、`ibd` カーネルクライアントドライバで使用される通信サービスです。
- `<p_key>` 使用されるパーティションキーの値です。
- `<port>` ポート番号です。
- `<unit-address>` IB カーネルクライアントドライバの `driver.conf` ファイルにこの名前前で指定されているプロパティを参照します。詳細は、[driver.conf\(4\)](#) のマニュアルページを参照してください。

# IB デバイスの動的再構成 (タスクマップ)

| タスク                                | 説明                                                                                       | 参照先                                                                                             |
|------------------------------------|------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| IB デバイス情報を表示します。                   | システム上の IB デバイスに関する情報を表示します。                                                              | 153 ページの「IB デバイス情報を表示する方法」                                                                      |
| IOC デバイスを構成または構成解除します。             | 次のいずれかの手順に従います。<br><br>IOC デバイスの構成を解除します。<br><br>IOC デバイスを構成します。                         | 154 ページの「IOC デバイスの構成を解除する方法」<br><br>155 ページの「IOC デバイスを構成する方法」                                   |
| ポートデバイスまたは VPPA デバイスを構成または構成解除します。 | 次のいずれかの手順に従います。<br><br>ポートデバイスまたは VPPA デバイスの構成を解除します。<br><br>ポートデバイスまたは VPPA デバイスを構成します。 | 155 ページの「IB Port、HCA_SVC、VPPA デバイスの構成を解除する方法」<br><br>156 ページの「IB Port、HCA_SVC、VPPA デバイスを構成する方法」 |
| IB 擬似デバイスを構成または構成解除します。            | 次のいずれかの手順に従います。<br><br>IB 擬似デバイスの構成を解除します。<br><br>IB 擬似デバイスを構成します。                       | 156 ページの「IB 擬似デバイスの構成を解除する方法」<br><br>157 ページの「IB 擬似デバイスを構成する方法」                                 |
| HCA のカーネル IB クライアントを表示します。         | HCA の構成を解除する場合などに、HCA のカーネル IB クライアントの情報を表示する必要があります。                                    | 157 ページの「HCA のカーネル IB クライアントを表示する方法」                                                            |
| IB HCA を構成または構成解除します。              | 次のいずれかの手順に従います。<br><br>HCA に接続されている IB デバイスの構成を解除します。<br><br>HCA に接続されている IB デバイスを構成します。 | 158 ページの「HCA に接続されている IB デバイスの構成を解除する方法」<br><br>159 ページの「IB HCA を構成する方法」                        |

| タスク                     | 説明                                                                            | 参照先                                                                                                                  |
|-------------------------|-------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| IB p_key テーブルを更新します。    | HCA ポートの p_key テーブルの情報が変更された場合は、IBTF と IBDM に通知して内部 p_key データベースを更新する必要があります。 | <a href="#">159 ページの「IB p_key テーブルを更新する方法」</a>                                                                       |
| IB 通信サービスを表示します。        | IBTF で現在使用されている IB 通信サービスを表示します。                                              | <a href="#">159 ページの「IB 通信サービスを表示する方法」</a>                                                                           |
| VPPA 通信サービスを追加または削除します。 | 次のいずれかの手順に従います。<br><br>VPPA 通信サービスを追加します。<br><br>VPPA 通信サービスを削除します。           | <a href="#">159 ページの「VPPA 通信サービスを追加する方法」</a><br><br><a href="#">160 ページの「既存の IB Port、HCA_SVC、VPPA 通信サービスを削除する方法」</a> |
| IOC の構成を更新します。          | すべての IOC デバイスノードのプロパティまたは特定の IOC Ap_Id を更新できます。                               | <a href="#">161 ページの「IOC の構成を更新する方法」</a>                                                                             |

## IB デバイスの動的再構成 (cfgadm)

cfgadm CLI だけを使用して、稼働中のシステムの IB デバイスを構成したり、構成を解除したりできます。このコマンドでは、IB ファブリックの表示、通信サービスの管理、および p\_key テーブルデータベースの更新を行うこともできます。詳細は、[cfgadm\\_ib\(1M\)](#) を参照してください。

cfgadm CLI は、ホストから見た IB ファブリック全体の動的再構成 (このガイドでは「DR」と呼ぶ) を管理します。cfgadm の操作は、Port、VPPA、HCA\_SVC、IOC、擬似デバイスなど、すべての IB デバイスでサポートされています。

cfgadm コマンドは接続点 (Ap\_Id) についての情報を表示します。「接続点」とは、DR 操作を行うことができるシステム内の特定の場所のことです。cfgadm でサポートされている Ap\_Id の詳細は、[cfgadm\\_ib\(1M\)](#) を参照してください。IB の Ap\_Id は、すべて connected と表示されます。

cfgadm コマンドを使用すると、IB デバイスステータス情報を取得できます。

| 受容体の状態                  | 説明                                    |
|-------------------------|---------------------------------------|
| connected/configured/ok | デバイスは接続されており利用可能です。devinfo ノードが存在します。 |



| 受容体の状態                         | 説明                                                                                                                            |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| connected/unconfigured/unknown | デバイスは利用不可で、このデバイスの devinfo ノードやデバイスドライバは存在しません。あるいは、このデバイスは ib 連結ドライバでできるように構成されませんでした。IB デバイスマネージャーではこのデバイスが認識されている可能性もあります。 |

次のセクションでは、cfgadm コマンドを使用して IB デバイスの動的再構成 (DR) を行う方法について説明します。次のすべてのサンプル IB デバイス情報は、関連する情報に焦点を合わせるために一部省略されています。

## ▼ IB デバイス情報を表示する方法

prtconf コマンドを使用して IB デバイスの一般的な情報を表示できます。たとえば、次のようになります。

```
$ prtconf
pci, instance #0
 pci15b3,5a44, instance #0
 ibport, instance #253
 ibport, instance #254
 ibport, instance #255
 .
 .
 .
ib, instance #0
 ioc, instance #243
 ioc, instance #244
 ioc, instance #245
 ioc, instance #246
 ioc, instance #247
 ioc, instance #248
 ibgen, instance #249
```

上記の例で、pci15b3,5a44 は IB HCA を指しています。

特定の IB デバイス情報を表示するには、次の手順を使用します。

- 1 スーパーユーザーになります。
- 2 IB ファブリック情報を表示します。

例:

```
cfgadm -a
Ap_Id Type Receptacle Occupant Condition
ib IB-Fabric connected configured ok
hca:1730000008070 IB-HCA connected configured ok
ib::1730000007F5198 IB-IOC connected configured ok
```

```

ib::1730000007F5199 IB-IOC connected configured ok
ib::1730000008070,0,hnfs IB-HCA_SVC connected configured ok
ib::1730000008071,0,sdp IB-PORT connected configured ok
ib::1730000008072,0,sdp IB-PORT connected configured ok
ib::1730000008071,8001,ipib IB-VPPA connected configured ok
ib::1730000008072,8001,ipib IB-VPPA connected configured ok
ib::ibgen,0 IB-PSEUDO connected configured ok

```

上記の出力例で、コンポーネントはそれぞれ次のものを表します。

|                                   |                                                                              |
|-----------------------------------|------------------------------------------------------------------------------|
| Ap_Id ib::1730000008072,0,sdp     | ポート 2 に接続され、sdp サービスにバインドされている IB Port デバイスを識別しています。                         |
| Ap_Id ib::1730000008072,8001,ipib | ポート 2 に接続され、p_key の値として 0x8001 を使用し、ibd サービスにバインドされている IB VPPA デバイスを識別しています。 |
| Ap_Id ib:: 1730000008070,0,hnfs   | hnfs サービスにバインドされている IB HCA_SVC デバイスを識別しています。                                 |
| Ap_Id ib::1730000007F5198         | IOC デバイスを識別しています。                                                            |
| Ap_Id ib::ibgen,0                 | 擬似デバイスを識別しています。                                                              |

### 3 特定の IB デバイス情報を表示します。

たとえば、IB VPPA デバイスの場合は次のようになります。

```

cfgadm -al -s "cols=ap_id:info" ib::1730000008072,8001,ipib
Ap_Id Information
ib::1730000008072,8001,ipib ipib

```

たとえば、IB HCA デバイスの場合は次のようになります。

```

cfgadm -al -s "cols=ap_id:info" hca::1730000008070
Ap_Id Information
hca::1730000008070 VID: 0x15b3, PID: 0x5a44, #ports: 0x2,
port1 GUID: 0x1730000008071, port2 GUID: 0x1730000008072

```

上記の出力には、ポート番号とその GUID がそれぞれ表示されています。

## ▼ IOC デバイスの構成を解除する方法

システムに物理的に接続されている IB デバイスの構成を解除することはできませんが、ドライバを適用することはできません。

### 1 スーパーユーザーになります。

- 2 IB デバイスの構成を解除します。

例:

```
cfgadm -c unconfigure ib::1730000007F5198
Unconfigure the device: /devices/ib:fabric::1730000007F5198
This operation will suspend activity on the IB device
Continue (yes/no)? y
```

- 3 デバイスの構成が解除されていることを確認します。

例:

```
cfgadm -a ib::1730000007F5198
ib::1730000007F5198 IB-IOC connected unconfigured unknown
```

## ▼ IOC デバイスを構成する方法

- 1 スーパーユーザーになります。

- 2 IB デバイスを構成します。

例:

```
cfgadm -yc configure ib::1730000007F5198
```

- 3 IB デバイスが構成されていることを確認します。

例:

```
cfgadm -al ib::1730000007F5198
Ap_Id Type Receptacle Occupant Condition
ib::1730000007F5198 IB-IOC connected configured ok
```

## ▼ IB Port、HCA\_SVC、VPPA デバイスの構成を解除する方法

IB Port、HCA\_SVC、または VPPA デバイスをシステムから削除するには、次の手順を使用します。

次の例では VPPA デバイスの構成を解除する方法を示しますが、Port デバイスや HCA\_SVC デバイスにも同じ手順を適用できます。

- 1 スーパーユーザーになります。
- 2 IBVPPA デバイスの構成を解除します。

例:

```
cfgadm -c unconfigure ib::1730000007F51,8001,ipib
Unconfigure the device: /devices/ib:fabric::1730000007F51,8001,ipib
```

```
This operation will suspend activity on the IB device
Continue (yes/no)? Y
```

- 3 デバイスが接続解除されていることを確認します。

例:

```
cfgadm -a ib::1730000007F51,8001,ipib
Ap_Id Type Receptacle Occupant Condition
ib::1730000007F51,8001,ipib IB-VPAA connected unconfigured unknown
```

## ▼ IB Port、HCA\_SVC、VPAA デバイスを構成する方法

IB Port、HCA\_SVC、または VPAA デバイスをシステム上で構成するには、次の手順を使用します。

次の例では VPAA デバイスを構成する方法を示しますが、Port デバイスや HCA\_SVC デバイスにも同様の手順を適用できます。

- 1 スーパーユーザーになります。

- 2 IBVPAA デバイスを構成します。

例:

```
cfgadm -c configure ib::1730000007F51,8001,ipib
```

- 3 デバイスが接続されていることを確認します。

例:

```
cfgadm -a ib::1730000007F51,8001,ipib
Ap_Id Type Receptacle Occupant Condition
ib::1730000007F51,8001,ipib IB-VPAA connected configured ok
```

---

注 - IB Port デバイスや HCA\_SVC デバイスの場合も、cfgadm による構成操作および構成解除操作は、前述の IB VPAA デバイスの例と同様です。

---

## ▼ IB 擬似デバイスの構成を解除する方法

IB 擬似デバイスをシステムから削除するには、次の手順を使用します。

- 1 スーパーユーザーになります。

- 2 IB 擬似デバイスの構成を解除します。

例:

```
cfgadm -c unconfigure ib::ibgen,0
Unconfigure the device: /devices/ib:fabric::ibgen,0
This operation will suspend activity on the IB device
Continue (yes/no)? Y
```

- 3 デバイスが接続解除されていることを確認します。

```
cfgadm -a ib::ibgen,0
Ap_Id Type Receptacle Occupant Condition
ib::ibgen,0 IB-PSEUDO connected unconfigured unknown
```

## ▼ IB 擬似デバイスを構成する方法

IB 擬似デバイスを構成するには、次の手順を使用します。

- 1 スーパーユーザーになります。

- 2 IB 擬似デバイスを構成します。

例:

```
cfgadm -yc configure ib::ibgen,0
```

- 3 デバイスが接続されていることを確認します。

例:

```
cfgadm -a ib::ibgen,0
Ap_Id Type Receptacle Occupant Condition
ib::ibgen,0 IB-PSEUDO connected configured ok
```

## ▼ HCA のカーネル IB クライアントを表示する方法

次の IB `cfgadm` プラグインコマンドを呼び出すと、この HCA を使用しているカーネル IB クライアントを一覧表示できます。カーネル IB クライアントが別の HCA を使用する場合は、最後の列に「yes」と表示されます。この HCA を使用しない IB マネージャーとカーネルクライアントは、`Ap_Id` の列に「-」と表示されます。

- HCA のカーネル IB クライアントを表示します。

例:

```
$ cfgadm -x list_clients hca:173000007F50
Ap_Id IB Client Alternate HCA
ib::173000007F51D0 ibgen no
ib::173000007F51D1 ibgen no
ib::173000007F51,8001,ipib ibd no
ib::ibgen,0 ibgen no
```

|   |        |    |
|---|--------|----|
| - | ibdm   | no |
| - | ibmf   | no |
| - | nfs/ib | no |

## ▼ HCA に接続されている IB デバイスの構成を解除する方法

IB `cfgadm` プラグインには、HCA の実際の DR を行う機能は含まれていません。HCA の DR を実際に行うには、基になるバスのプラグインを使用します。たとえば、PCI ベースの HCA の場合は、`cfgadm_pci` コマンドを使用できます。詳細は、[cfgadm\\_pci\(1M\)](#) のマニュアルページを参照してください。

ただし、IB `cfgadm` プラグインは、HCA のカーネル IB クライアントを一覧表示することで、HCA の DR を補助します。その手順を次に示します。

- 1 スーパーユーザーになります。
- 2 HCA のカーネル IB クライアントを一覧表示します。

例:

```
cfgadm -x list_clients hca:173000007F50
Ap_Id IB Client Alternate HCA
ib::1730000007F51D0 ibgen no
ib::1730000007F51D1 ibgen no
ib::1730000007F51,8001,ipib ibd no
ib::ibgen,0 ibgen no
- ibdm no
- ibmf no
- nfs/ib no
```

- 3 代替 HCA を持っていないカーネル IB クライアント (Port、VPPA、HCA\_SVC、IOC デバイスなど) の構成を解除します。

例:

```
cfgadm -x unconfig_clients hca:1730000008070
Unconfigure Clients of HCA /devices/ib:1730000008070
This operation will unconfigure IB clients of this HCA
Continue (yes/no)? y
```

- 4 HCA のカーネル IB クライアントの構成が解除されていることを確認します。

```
cfgadm -x list_clients hca:173000007F50
Ap_Id IB Client Alternate HCA
- ibdm no
- ibmf no
- nfs/ib no
```

## IB HCA を構成する方法

バス固有の `cfgadm` プラグインを呼び出して HCA を構成します。しかし、その詳細についてはこの章では記載していません。

### ▼ IB `p_key` テーブルを更新する方法

追加の `p_key` が有効にされた場合や無効にされた場合など、HCA ポートの `p_key` テーブルの情報が変更された場合は、InfiniBand Transport Framework (IBTF) と IBDM に通知して内部 `p_key` データベースを更新する必要があります。`cfgadm` コマンドは、IBTF および IBDM の `p_key` データベースの更新を補助します。詳細は、[ibtl\(7D\)](#) および [ibdm\(7D\)](#) のマニュアルページを参照してください。

- 1 スーパーユーザーになります。
- 2 `p_key` テーブルを更新します。

例:

```
cfgadm -x update_pkey_tbls -y ib
```

### ▼ IB 通信サービスを表示する方法

IBTF で現在使用されている IB 通信サービスを表示するには、次の手順を使用します。

- 1 スーパーユーザーになります。
- 2 IB 通信サービスを表示します。

例:

```
cfgadm -x list_services ib
Port communication services:
 srp
VPPA communication services:
 ibd
HCA_SVC communication services:
 hnfs
```

### ▼ VPPA 通信サービスを追加する方法

新しい VPPA 通信サービスを追加するには、次の手順を使用します。

新しい HCA\_SVC 通信サービスやポート通信サービスも、同様の手順で追加できます。

- 1 スーパーユーザーになります。
- 2 新しい VPPA 通信サービスを追加します。

例:

```
cfgadm -o comm=vppa,service=new -x add_service ib
```

- 3 新しいサービスが追加されていることを確認します。

例:

```
cfgadm -x list_services ib
Port communication services:
 srp
VPPA communication services:
 ibd
 new
HCA_SVC communication services:
 nfs_service
```

## ▼ 既存の IB Port、HCA\_SVC、VPPA 通信サービスを削除する方法

既存の IB Port、HCA\_SVC、VPPA 通信サービスを削除するには、次の手順を使用します。

- 1 スーパーユーザーになります。
- 2 VPPA 通信サービスを削除します。

例:

```
cfgadm -o comm=vppa,service=new -x delete_service ib
```

- 3 通信サービスが削除されていることを確認します。

例:

```
cfgadm -x list_services ib
Port communication services:
 srp
VPPA communication services:
 ibd
HCA_SVC communication services:
 hnfs
```



## ▼ IOC の構成を更新する方法

すべての IOC デバイスノードまたは特定の IOC Ap\_Id についてプロパティーを更新するには、次の手順を使用します。更新できるプロパティーは次のとおりです。

- port-list
- port-entries
- service-id
- service-name

これらのプロパティーの詳細は、[ib\(7D\)](#) のマニュアルページを参照してください。

構成がまったく変更されていない場合、これらのプロパティーは更新されません。次の例は、特定の IOC の構成を更新する方法を示しています。すべての IOC の構成を更新するには、特定の IOC Ap\_Id の代わりに、静的な ib Ap\_Id を指定します。

1 スーパーユーザーになります。

2 IOC の構成を更新します。

例:

```
cfgadm -x update_ioc_conf ib::1730000007F5198
This operation can update properties of IOC devices.
Continue (yes/no)? y
```

3 `prtconf -v` を実行して、プロパティーが更新されていることを確認します。

## InfiniBand デバイスでの uDAPL アプリケーションインタフェースの使用

uDAPL (User Direct Access Programming Library) は、RDMA (Remote Direct Memory Access) に対応した InfiniBand などの相互接続を介して行われるデータセンターアプリケーションのデータメッセージングのパフォーマンス、スケーラビリティ、および信頼性を向上させる標準 API です。uDAPL インタフェースは DAT Collaborative によって定義されています。DAT Collaborative の詳細は、次の Web サイトを参照してください。

<http://www.datcollaborative.org>

この Solaris リリースでは、次の uDAPL 機能がサポートされます。

- 標準の DAT レジストリライブラリ libdat。詳細は、[libdat\(3LIB\)](#) のマニュアルページを参照してください。

- 標準のサービスプロバイダ登録ファイル `dat.conf`。詳細は、[dat.conf\(4\)](#) のマニュアルページを参照してください。
- 複数のサービスプロバイダのサポート。これにより、各プロバイダが uDAPL ライブラリパスやバージョン番号などを独自の `service_provider.conf` ファイルで指定できます。詳細は、[service\\_provider.conf\(4\)](#) を参照してください。
- `datadm` コマンド。これは `dat.conf` を構成するための管理ツールです。詳細は、[datadm\(1M\)](#) のマニュアルページを参照してください。
- 新しいリソース制御プロパティー `project.max-device-locked-memory`。ロックされる物理メモリーの量を調節します。
- アドレス解決に IP インフラストラクチャーを活用する IPv4 アドレスまたは IPv6 アドレスを使った命名方式。IPv4 での ARP や IPv6 での近隣探索などが含まれます。Solaris uDAPL インタフェースアダプタは、IPoIB デバイスインスタンスに直接対応づけられます。
- DAT Collaborative コミュニティーによって使用されている標準のアドレス変換方式のサポート。
- Mellanox Tavor ホストチャネルアダプタをサポートするための uDAPL サービスプロバイダライブラリ。`dat.conf` 登録ファイルへの自動登録を提供します。
- SPARC プラットフォームと x86 プラットフォームの両方をサポートします。

## ▼ uDAPL を使用可能にする方法

- 1 スーパーユーザーになります。
- 2 次のパッケージがインストールされていることを確認します。必要に応じて、これらをインストールします。
  - `SUNWib` - Sun InfiniBand フレームワーク
  - `SUNWtavor` - Sun Tavor HCA ドライバ
  - `SUNWipoib` - Sun IP over InfiniBand
  - `SUNWudapl` - Sun User Direct Access Programming Library (ルート)
  - `SUNWudaplu` - Sun User Direct Access Programming Library (usr)
  - `SUNWudapltr` - Tavor 用 Sun uDAPL (ルート)
  - `SUNWudapltu` - Tavor 用 Sun uDAPL (usr)
- 3 IPoIB インタフェースを認識させるには、次のいずれかの手順を選択します。
  - `ifconfig` コマンドと `datadm` コマンドを使用して、インタフェースを手動で使えるようにします。例:

```
ifconfig ibd1 plumb
ifconfig ibd1 192.168.0.1/24 up
datadm -a /usr/share/dat/SUNWudaplt.conf
```

- 次の方法で、インタフェースを自動的に使えるようにします。
  - 次のファイルを作成し、適切な IP アドレスを指定します。
 

```
/etc/hostname.ibd1
```
  - システムをリブートします。

## DAT 静的レジストリの更新

datadm コマンドを使用して、DAT 静的レジストリ `dat.conf` ファイルを管理できます。このファイルの詳細は、[dat.conf\(4\)](#) のマニュアルページを参照してください。

datadm コマンドを使用して、サービスプロバイダを `dat.conf` ファイルに登録したり登録を解除したりすることもできます。詳細は、[datadm\(1M\)](#) のマニュアルページを参照してください。

IPoIB インタフェースアダプタの追加や削除を行なったときは、システムの現在の状態を反映させるために、`datadm` コマンドを実行して `dat.conf` ファイルを更新してください。現在インストールされているすべてのサービスプロバイダについて、インタフェースアダプタの新しいセットが再生成されます。

### ▼ DAT 静的レジストリを更新する方法

- 1 スーパーユーザーになります。
- 2 システムに **IPoIP** インタフェースアダプタを追加したり、システムから **IPoIP** インタフェースアダプタを削じたりしたあとに、**DAT** 静的レジストリを更新します。

```
datadm -u
```

- 3 更新された **DAT** 静的レジストリを表示します。

```
datadm
```

### ▼ DAT 静的レジストリにサービスプロバイダを登録する方法

- 1 スーパーユーザーになります。
- 2 **Mellanox Tavor** ホストチャネルアダプタ用の **Sun** のサービスプロバイダを追加したあとに、**DAT** 静的レジストリを更新します。

```
datadm -a /usr/share/dat/SUNWudaplt.conf
```

- 3 更新された **DAT** 静的レジストリを表示します。

```
datadm -v
```

## ▼ **DAT** 静的レジストリからサービスプロバイダの登録を解除する方法

- 1 スーパーユーザーになります。
- 2 **Mellanox Tavor** ホストチャネルアダプタ用の **Sun** のサービスプロバイダをシステムから削除したあとに、**DAT** 静的レジストリを更新します。

```
datadm -r /usr/share/dat/SUNWudaplt.conf
```

- 3 更新された **DAT** 静的レジストリを表示します。

```
datadm -v
```

## ディスクの管理 (概要)

---

この章では、Oracle Solaris ディスクスライスの概要および `format` ユーティリティについて説明します。

この章の内容は次のとおりです。

- 165 ページの「ディスク管理の新機能」
- 166 ページの「ディスク管理タスクについての参照先」
- 167 ページの「ディスク管理の概要」
- 176 ページの「ディスクをパーティションに分割する」

システムにディスクを追加する方法については、第 10 章「SPARC: ディスクの設定 (手順)」または 243 ページの「x86: ZFS ファイルシステム用のディスクの設定 (タスクマップ)」を参照してください。

## ディスク管理の新機能

Oracle Solaris リリースで追加されたディスク管理機能は次のとおりです。

- 165 ページの「Advanced Format ディスクのサポート」

### Advanced Format ディスクのサポート

以前の Oracle Solaris リリースでは、512 バイトの物理ブロックサイズおよび論理ブロックサイズのディスクをサポートしています。これは、業界標準である従来のディスクブロックサイズです。

現在、ディスクメーカーは、ブロックサイズが 512 バイトを超えるハードディスクドライブを示す一般的な用語である Advanced Format (AF) ディスクとも呼ばれる大容量ディスクを提供しています。

AFディスクは、通常4バイトの範囲のブロックサイズを利用しますが、次のように異なるものがあります。

- 4Kバイトネイティブディスク (4kn) – 4Kバイトの物理および論理ブロックサイズを利用します
- 512バイトエミュレーション (512e) – 4Kバイトの物理ブロックサイズを利用しますが、512バイトの論理ブロックサイズを報告します

比較のため、Oracle Solaris では、512バイトのネイティブ (512n) ディスクの用語を導入します。これは512バイトのブロックサイズの従来のディスクです。

Oracle Solaris 10 1/13 では、従来の512nディスクに加えて、次のように、Advanced Format ディスクをサポートしています。

- 非ルートUFSまたはZFSファイルシステムは、4kn および512e ディスクでサポートされます。
- SPARC: UFS または ZFS ルートファイルシステムは、512e ディスク上でブートおよびインストールできます。
- x86: UFS ルートファイルシステムは、512e ディスク上でブートおよびインストールできます。

Advanced Format ドライブを購入する前に、デバイス製造元に、512e デバイスが、データの転送中の電源障害発生後にデータの損失を避けるためのパワーセーフ機能を備えていることを確認してください。

# ディスク管理タスクについての参照先

ディスク管理の手順については、次を参照してください。

| ディスク管理タスク                       | 参照先                                                          |
|---------------------------------|--------------------------------------------------------------|
| ディスクをフォーマットしてディスクラベルを確認します。     | <a href="#">第9章「ディスクの管理(タスク)」</a>                            |
| SPARC システムに新しいディスクを追加します。       | <a href="#">第10章「SPARC: ディスクの設定(手順)」</a>                     |
| x86 システムに新しいディスクを追加します。         | <a href="#">243 ページの「x86: ZFS ファイルシステム用のディスクの設定(タスクマップ)」</a> |
| SCSI または PCI ディスクのホットプラグを実行します。 | <a href="#">第4章「デバイスの動的構成(タスク)」</a>                          |

# ディスク管理の概要

一般に、Oracle Solaris OS におけるディスクの管理とは、システムを設定し、Oracle Solaris インストールプログラムを実行し、適切なディスクスライスおよびファイルシステムを作成して Oracle Solaris OS をインストールすることを意味します。また、`format` ユーティリティーを使用して、新しいディスクドライブを追加したり、欠陥ディスクドライブを交換したりしなければならない場合もあります。

このセクションでは次の情報について説明します。

- [167 ページの「ディスク関連の用語」](#)
- [168 ページの「ディスクラベルについて」](#)
- [168 ページの「EFI \(GPT\) ディスクラベル」](#)
- [171 ページの「ディスクスライスについて」](#)
- [172 ページの「format ユーティリティー」](#)

## ディスク関連の用語

このセクションで説明している情報を効果的に利用するには、基本的なディスクアーキテクチャーを理解しておく必要があります。特に、次の表の用語を理解しておいてください。

| 用語         | 説明                                                     |
|------------|--------------------------------------------------------|
| トラック       | ディスクが回転するときに1つの静止したディスクヘッドの下を通過する同心リング。                |
| シリンダ       | ディスクが回転する軸から同じ距離にあるトラックの集まり。                           |
| セクター       | 各ディスクプラッタのセクション。                                       |
| ブロック       | ディスク上のデータ記憶領域。                                         |
| ディスクコントローラ | ディスクドライブを制御するチップおよび関連する回路。                             |
| ディスクラベル    | 通常は、ディスクのジオメトリおよびパーティション情報が格納されている先頭セクターから開始するディスクの一部。 |
| デバイスドライバ   | 物理 (ハードウェア) または仮想デバイスを制御するカーネルモジュール。                   |

詳細は、ディスク製造元の製品情報を参照してください。

## ディスクラベルについて

どのディスクにも、そのディスクのコントローラ、ジオメトリ、およびスライスに関する情報を格納する特殊な領域が確保されています。このような情報をディスクの「ラベル」と呼びます。VTOC ラベル付きのディスク上のディスクラベルを「VTOC (Volume Table of Contents)」と呼びます。「ディスクにラベルを付ける」とは、ディスクにスライス情報を書き込むことを意味します。通常は、ディスクのスライスやパーティションを変更した後にラベルを付けます。

Oracle Solaris リリースでは、次の2つのディスクラベルをサポートしています。

- SMI – 従来の VTOC ラベル。サイズが2T バイトに満たないディスク用です。
- EFI – 2 TB を超えるディスクをサポートします。EFI GPT (Extensible Firmware Interface GUID Partition Table) ディスクラベルは、2 TB 未満のディスクにも使用できます。

スライスを作成したあとでディスクにラベルを付けないと、OS はスライスを「認識」する方法がないので、そのスライスを利用できなくなります。

## EFI (GPT) ディスクラベル

EFI ラベルは、サイズが2T バイトを超える物理ディスクボリュームと仮想ディスクボリュームをサポートします。このリリースには、2T バイトを超えるサイズのディスクを管理するためのディスクユーティリティも付属しています。

次のファイルシステム製品では、1T バイトを超えるサイズのファイルシステムがサポートされています。

- Oracle Solaris ZFS ファイルシステムでは、1T バイトを超えるサイズのファイルシステムがサポートされています。
- 旧バージョンの Solaris Volume Manager ソフトウェアは、1T バイトを超えるディスクを管理するために使用することもできます。

システムでサポートされている Oracle Solaris リリースが実行されている場合、`format -e` コマンドを使用することで EFI ラベルをディスクに適用できます。ただし、EFI ラベルを適用する前に、[170 ページの「EFI ディスクラベルの制限」](#)に記載された重要な情報を確認することをお勧めします。

EFI ラベルが不要になった場合に、`format -e` コマンドを使って、VTOC ラベルを再度適用することもできます。例:

```
format -e
Specify disk (enter its number): 2
selecting c0t5d0
[disk formatted]
.
```



```
.
.
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Warning: This disk has an EFI label. Changing to SMI label will erase all
current partitions.
Continue? yes
Auto configuration via format.dat[no]?
Auto configuration via generic SCSI-2[no]?
format> quit
```



注意-ディスクラベルを変更すると、ディスク上のデータがすべて破棄されることに留意してください。

## EFI ラベルと VTOC ラベル

EFI ディスクラベルが VTOC ディスクラベルと異なる点は次のとおりです。

- サイズが 2T バイトを超えるディスクをサポートします。
- スライス 0-6 を使用できます (パーティション 2 はその他のスライス)。
- プライマリ (バックアップ) ラベルまたはその他のパーティションを使ってパーティションやスライスをオーバーラップすることはできません。EFI ラベルのサイズは通常 34 セクターなので、パーティションは通常セクター 34 で始まります。つまり、パーティションはセクターゼロ (0) では始められないことを意味します。
- EFI (GPT) ラベルにはジオメトリの概念は使用しません。パーティションは論理ブロックに基づいて定義されます。
- 代替シリンダ領域に格納されていた一部の情報が、ディスクまたは Solaris パーティションの最後の 2 つのシリンダに格納されるようになりました。
- `format` ユーティリティーを使ってパーティションサイズを変更する場合、サイズ 0 のパーティションには `unassigned` パーティションタグが割り当てられます。`format` ユーティリティーでは、0 より大きいサイズのパーティションには、デフォルトにより `usr` パーティションタグが割り当てられます。パーティションを変更したあと新しいパーティションタグを割り当てたい場合は、パーティション変更メニューを使用します。ただし、サイズ 0 以外のパーティションに `unassigned` パーティションタグを割り当て直すことはできません。

## EFI ディスクラベルの制限

サイズが2Tバイトを超えるディスクを使用することが現在の環境にとって適切かどうかを判断するときは、次の点を考慮してください。

- VTOC ラベル付きディスクを使用するシステム向けの階層化されたソフトウェア製品で、EFI ラベル付きディスクにアクセスできないことがあります。
- x86 ベースのシステムでは、EFI ラベル付きの2Tバイトを超えるディスク上で `fdisk` コマンドを使用できます。
- `format` ユーティリティーを使って EFI ラベル付きディスクをパーティションに分割します。
- EFI 仕様では、パーティションのオーバーラップは禁止されています。ディスク全体は `cx tydz` によって表されます。
- EFI ディスクラベルは、ディスクやパーティションのサイズ情報を提供します。使用可能な単位はセクターまたはブロックです。シリンダおよびヘッドは使用できません。
- 次の `format` オプションは、EFI ラベル付きディスクではサポートされていないか、不適切です。
  - EFI ラベル付きディスクは `format.dat` ファイルにエントリを必要としないため、`save` オプションはサポートされません。
  - `backup` オプションは適用できません。

## x86: EFI ラベル付きディスクのサポート

x86 システム上では、EFI ディスクラベルの Oracle Solaris サポートが利用可能です。x86 システムで EFI ラベルを追加するには、次のコマンドを使用します。

```
format -e
> [0] SMI Label
> [1] EFI Label
> Specify Label type[0]: 1
> WARNING: converting this device to EFI labels will erase all current
> fdisk partition information. Continue? yes
```

以前のラベル情報は EFI ディスクラベルに変換されません。

`format` コマンドを使ってラベルのパーティション情報を手動で作成し直す必要があります。EFI ラベル付きの2Tバイトのディスク上で `fdisk` コマンドを使用することはできません。サイズが2Tバイトを超えるディスクで `fdisk` コマンドを実行して Solaris パーティションを作成する場合、Solaris パーティションは2Tバイトに制限されます。EFI ディスクラベルの詳細は、前のセクションを参照してください。

## EFI ラベル付きディスクの管理

EFI ラベル付きディスクの管理方法は、次の表で確認できます。

| タスク                                                   | 参照先                                                                                          |
|-------------------------------------------------------|----------------------------------------------------------------------------------------------|
| システムがインストールされていない場合は、インストールします。                       | 『Oracle Solaris 10 1/13 インストールガイド: 基本インストール』                                                 |
| システムがすでにインストールされていますが、ルートプールディスクが破損しているか、交換する必要があります。 | 228 ページの「SPARC: ZFS ルートファイルシステム用のディスクを設定する方法」または 245 ページの「x86: ZFS ルートファイルシステム用のディスクを設定する方法」 |
| システムがすでにインストールされていますが、ルート以外のプールのディスクを設定する必要があります。     | 234 ページの「SPARC: ZFS ファイルシステム用のディスクを設定する方法」または 251 ページの「x86: ZFS ファイルシステム用のディスクを設定する方法」       |

## EFI ディスクラベルに関する問題のトラブルシューティング

EFI ラベル付きディスクに関する問題のトラブルシューティングを行うときは、次のエラーメッセージと解決方法を使用してください。

### エラーの発生原因

1T バイトを超えるディスクでは、SPARC または x86 カーネルを実行しているシステムをブートしてください。

### エラーメッセージ

```
Dec 3 09:12:17 holoship scsi: WARNING: /sbus@a,0/SUNW,socal@d,10000/
sf@1,0/ssd@w50020f23000002a4,0 (ssd1):
Dec 3 09:12:17 holoship corrupt label - wrong magic number
```

### エラーの発生原因

古い Solaris リリースで動作するシステムにディスクを追加しようとしました。

### 解決方法

このディスクは、EFI ディスクラベルをサポートしている Solaris リリースで動作するシステムに追加してください。

## ディスクスライスについて

ディスク上に格納されたファイルは、ファイルシステム中で管理されます。ディスク上の各ファイルシステムはスライス、つまりファイルシステム用に確保されたセクターセットのグループに割り当てられます。Oracle Solaris OS (および、システム管理者) からは、各ディスクスライスは別個のディスクドライブであるかのように見えます。

ファイルシステムの詳細は、[第 14 章「ファイルシステムの管理 \(概要\)」](#)を参照してください。

---

注- スライスを「パーティション」と呼ぶこともあります。format ユーティリティーなど、特定のインタフェースではスライスを「パーティション」と呼びます。

---

スライスを設定するときには、次の規則に注意してください。

- 各ディスクスライスは、ファイルシステムを1つしか持てない。
- ファイルシステムを複数のスライスにまたがって割り当てることはできない。

SPARC プラットフォームと x86 プラットフォームでは、スライスの設定が少し異なります。次の表に、主な相違点を示します。

## raw データスライスの使用

ディスクラベルは、各ディスクのブロック 0 に格納されます。つまり、他社製データベースアプリケーションを使って raw データスライスを作成するときは、ブロック 0 で開始してはいけません。そうすると、ディスクラベルが上書きされて、ディスク上のデータにアクセスできなくなります。

ディスク上の次の領域は、raw データスライス用に使用しないでください。raw データスライスは他社製のデータベースアプリケーションによって作成されることがあります。

- ブロック 0 (ディスクラベルが格納される領域)
- スライス 2 (VTOC ラベル付きディスク全体を表す)

## format ユーティリティー

手順や参照情報のセクションに進む前に、次の情報に目を通して format ユーティリティーの概要とその使用法を確認してください。

format ユーティリティーは、Oracle Solaris システム用にハードディスクドライブを用意するためのシステム管理ツールです。

次の表に、format ユーティリティーの機能とその利点を説明します。

表 7-1 format ユーティリティーの機能と利点

| 機能                           | 利点                                                                                                                                                                    |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| システム内で接続されている全ディスクドライブを検索します | 次の状態を報告します <ul style="list-style-type: none"> <li>■ ターゲットの位置</li> <li>■ ディスクのジオメトリ</li> <li>■ ディスクがフォーマット済みかどうか</li> <li>■ ディスク上にマウントされているパーティションが存在するかどうか</li> </ul> |
| ディスクラベルを検索します                | 修復処理に使用します                                                                                                                                                            |
| 欠陥セクターを修復します                 | 回復可能なエラーが発生したディスクドライブを製造元に返送しなくても、熟練した管理者なら修復できます                                                                                                                     |
| ディスクをフォーマットして、分析します          | ディスク上でセクターを作成し、検査します                                                                                                                                                  |
| ディスクをパーティションに分割します           | ディスクをスライスまたはパーティションに分割します。ZFS ルートプール以外の ZFS ファイルシステムは、ディスクスライスまたはパーティションに対応しません。                                                                                      |
| ディスクにラベルを付けます                | 後から検索できるように (通常は修復用)、ディスクにディスク名と構成情報を書き込みます                                                                                                                           |

format ユーティリティーのオプションについては、[第 13 章「format ユーティリティー \(参照情報\)」](#)を参照してください。

## format ユーティリティーを使用する場合

Oracle Solaris のインストール時に、Oracle Solaris インストールユーティリティーによってディスクドライブがパーティションに分割され、ラベルが付けられます。次のような場合に、format ユーティリティーを使用できます。

- スライスまたはパーティション情報を表示します
- ディスクをパーティションに分割する
- 既存のシステムにディスクドライブを追加する
- ディスクドライブをフォーマットする
- ディスクにラベルを付ける
- ディスクドライブを修復する
- ディスクのエラーを分析する

システム管理者が format ユーティリティーを使用するのは、主にディスクをパーティションに分割するためです。これらの手順については、[第 10 章「SPARC: ディスクの設定 \(手順\)」](#)および[243 ページの「x86: ZFS ファイルシステム用のディスクの設定 \(タスクマップ\)」](#)を参照してください。

format ユーティリティの使用上のガイドラインについては、次のセクションを参照してください。

## format ユーティリティ使用上のガイドライン

表 7-2 format ユーティリティのガイドライン

| タスク                                                           | ガイドライン                                                                                                                                                                                                                                                                                                                    | 参照先                                                                                                                                                                                                 |
|---------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ディスクをフォーマットします。                                               | <ul style="list-style-type: none"><li>■ ディスクをフォーマットし直すと、既存のデータが失われます。</li><li>■ ディスクドライブをフォーマットしてパーティションに分割した状態で出荷する製造元が増えているので、ディスクドライブをフォーマットする必要性は減少しています。既存システムにディスクドライブを追加したり、既存システムのディスクドライブを交換したりする場合は、format ユーティリティを使用しなくてもすむことがあります。</li><li>■ ディスクを配置し直して多数のディスクエラーが表示される場合は、そのラベルを付け直してみることをお勧めします。</li></ul> | <a href="#">201 ページの「ディスクをフォーマットする方法」</a> または <a href="#">205 ページの「ディスクラベルを作成する方法」</a>                                                                                                              |
| UFS または ZFS ルートファイルシステムを含むディスクを設定します。                         | 非冗長構成の場合、破損したディスクの ZFS ルートファイルシステムのデータをバックアップメディアから復元する必要があります。復元しなければ、インストールユーティリティを使用してシステムをもう一度インストールしなければなりません。                                                                                                                                                                                                       | <a href="#">228 ページの「SPARC: ZFS ルートファイルシステム用のディスクを設定する方法」</a> または <a href="#">245 ページの「x86: ZFS ルートファイルシステム用のディスクを設定する方法」</a> 。システムをインストールし直す必要がある場合は、『Oracle Solaris 10 1/13 インストールガイド: 基本インストール』。 |
| ZFS ルートプールディスクまたは UFS ルートファイルシステム用に VTOC ラベル付きディスクスライスを作成します。 | <ul style="list-style-type: none"><li>■ ZFS ストレージプールの最適な使用方法は、ディスク全体を使ってプールを作成することです。</li><li>■ ルートプールに使用するためのディスクでは、ディスクスライスを作成する必要があります。これは、長年にわたるブートの制限です。</li></ul>                                                                                                                                                   | <a href="#">229 ページの「SPARC: ZFS ルートファイルシステム用のディスクスライスを作成する方法」</a> または <a href="#">245 ページの「x86: ZFS ルートファイルシステム用のディスクを設定する方法」</a>                                                                   |

表 7-2 format ユーティリティのガイドライン (続き)

| タスク                                    | ガイドライン                                                                                                                       | 参照先                                                                                    |
|----------------------------------------|------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| UFS または ZFS 非ルートファイルシステムを含むディスクを設定します。 | ルート以外の UFS または ZFS ファイルシステムに使用されるディスクには通常、ユーザーファイルやデータファイル用の領域が含まれています。ルートプールやルート以外のプールに別のディスクを接続または追加すれば、ディスク容量を増やすことができます。 | 234 ページの「SPARC: ZFS ファイルシステム用のディスクを設定する方法」または 251 ページの「x86: ZFS ファイルシステム用のディスクを設定する方法」 |

## ディスクのフォーマット

ほとんどの場合、ディスクは製造元または再販業者によってフォーマットされています。このため、ドライブをインストールするときにフォーマットし直す必要はありません。ディスクがフォーマットされているかどうかを判別するには、`format` ユーティリティを使用します。詳細は、201 ページの「ディスクがフォーマット済みかを調べる方法」を参照してください。

ディスクがフォーマットされていない場合、`format` ユーティリティを使用してフォーマットしてください。

ディスクのフォーマットでは、次の 2 つのステップが行われます。

- ディスクメディアを使用できるようにする。
- 表面解析に基づいてディスクの欠陥リストを作成する。



注意-フォーマットは、ディスク上のデータを上書きする、破壊的なプロセスです。このため、通常は製造元や再販業者のみがディスクをフォーマットします。ディスクに欠陥があるために問題が再発していると思われる場合は、`format` ユーティリティを使用して表面解析を実行できますが、データを破壊しないコマンドだけを使用するように注意してください。詳細は、201 ページの「ディスクをフォーマットする方法」を参照してください。

データに利用できる合計ディスク容量のうち、ごくわずかな容量が欠陥情報とフォーマット情報の格納に使用されます。この容量はディスクのジオメトリによって異なり、使用年数がたち欠陥箇所が多くなるにつれて、少なくなります。

ディスクの種類とサイズに応じて、フォーマットは数分から数時間かかります。

# ディスクをパーティションに分割する

このセクションでは次の情報について説明します。

- [176 ページの「パーティションテーブル関連の用語」](#)
- [177 ページの「パーティションテーブル情報の表示」](#)
- [179 ページの「free hog スライスの使用方法」](#)

format ユーティリティーは、主にシステム管理者がディスクをパーティションに分割する場合に使われます。手順を次に示します。

- どのスライスが必要かを決定します。
- 各スライスまたはパーティションのサイズを決定します。
- format ユーティリティーを使用してディスクをパーティションに分割します。
- 新しいパーティション情報を使用してディスクにラベルを付けます。
- パーティションごとにファイルシステムを作成します。

ディスクをパーティションに分割するには、format ユーティリティーの partition メニューで modify コマンドを実行するのがもっとも簡単です。modify コマンドを使用すると、開始シリンダ境界を追跡しなくても、各パーティションのサイズを指定してパーティションを作成できます。modify コマンドを使用すると、「free hog」スライス内の残りのディスク領域も追跡できます。

## パーティションテーブル関連の用語

ディスクラベルのうち重要な部分は「パーティションテーブル」です。パーティションテーブルは、ディスクのスライス、スライスの境界(シリンダ単位)、スライスの合計サイズを表します。ディスクのパーティションテーブルは、format ユーティリティーを使用して表示できます。次の表に、パーティションテーブル関連の用語を示します。

表 7-3 パーティションテーブル関連の用語

| 用語  | 値                                                                               | 説明                                                                             |
|-----|---------------------------------------------------------------------------------|--------------------------------------------------------------------------------|
| 数値  | 0-7                                                                             | <b>VTOC</b> -0-7 の番号が付いたパーティションまたはスライス。<br><br><b>EFI</b> -0-6 の番号が付いたパーティション。 |
| タグ  | 0=UNASSIGNED 1=BOOT 2=ROOT<br>3=SWAP 4=USR 5=BACKUP<br>7=VAR 8=HOME 11=RESERVED | 一般にこのパーティションにマウントされたファイルシステムを記述する数値。                                           |
| フラグ | wm                                                                              | パーティションは書き込み可能でマウント可能です。                                                       |



表 7-3 パーティションテーブル関連の用語 (続き)

| 用語 | 値     | 説明                                                                                             |
|----|-------|------------------------------------------------------------------------------------------------|
|    | wu rm | パーティションは書き込み可能でマウントはできません。これは、スワップ領域専用のパーティションのデフォルト状態です。ただし、mount コマンドでは「マウント不可」のフラグは検査されません。 |
|    | rm    | パーティションは読み取り専用でマウント可能です。                                                                       |

パーティションのフラグとタグは必ず割り当てられるので、管理する必要はありません。

パーティションテーブルを表示する手順については、次の項目を参照してください。

- [177 ページの「パーティションテーブル情報の表示」](#)
- [203 ページの「ディスクスライス情報を表示する方法」](#)
- [210 ページの「ディスクラベルを検査する方法」](#)

## パーティションテーブル情報の表示

次の例は、format ユーティリティーを使って、74G バイトの VTOC ラベル付きディスクのパーティションテーブルを表示したものです。

Total disk cylinders available: 38756 + 2 (reserved cylinders)

| Part | Tag        | Flag | Cylinders    | Size    | Blocks                |
|------|------------|------|--------------|---------|-----------------------|
| 0    | root       | wm   | 3 - 2083     | 4.00GB  | (2081/0/0) 8390592    |
| 1    | swap       | wu   | 2084 - 3124  | 2.00GB  | (1041/0/0) 4197312    |
| 2    | backup     | wm   | 0 - 38755    | 74.51GB | (38756/0/0) 156264192 |
| 3    | unassigned | wm   | 0            | 0       | (0/0/0) 0             |
| 4    | unassigned | wm   | 0            | 0       | (0/0/0) 0             |
| 5    | unassigned | wm   | 0            | 0       | (0/0/0) 0             |
| 6    | unassigned | wm   | 0            | 0       | (0/0/0) 0             |
| 7    | home       | wm   | 3125 - 38755 | 68.50GB | (35631/0/0) 143664192 |
| 8    | boot       | wu   | 0 - 0        | 1.97MB  | (1/0/0) 4032          |
| 9    | alternates | wu   | 1 - 2        | 3.94MB  | (2/0/0) 8064          |

partition>

format ユーティリティーを使用して表示されるパーティションテーブルには、次の情報が含まれます。

| 列名           | 説明                                                      |
|--------------|---------------------------------------------------------|
| Part         | パーティションまたはスライスの番号。この列についての説明は、表 7-3 を参照してください。          |
| Tag          | パーティションのタグ。この列についての説明は、表 7-3 を参照してください。                 |
| Flag         | パーティションのフラグ。この列についての説明は、表 7-3 を参照してください。                |
| Cylinders    | スライスの開始シリンダ番号と終了シリンダ番号を示します。EFI ラベル付きディスクでは表示されません。     |
| Size         | スライスまたはパーティションのサイズ (MB)。                                |
| Blocks       | 合計シリンダ数と 1 スライス当たりの合計セクター数を示します。EFI ラベル付きディスクでは表示されません。 |
| First Sector | <b>EFI</b> – 開始ブロック番号。VTOC ラベル付きディスクでは表示されません。          |
| Last Sector  | <b>EFI</b> – 終了ブロック番号。VTOC ラベル付きディスクでは表示されません。          |

次に、prtvtoc コマンドを使用して EFI ディスクラベルを表示した例を示します。

```
prtvtoc /dev/rdisk/c4t1d0s0
* /dev/rdisk/c4t1d0s0 partition map
*
* Dimensions:
* 512 bytes/sector
* 2576941056 sectors
* 2576940989 accessible sectors
*
* Flags:
* 1: unmountable
* 10: read-only
*
*
* Partition Tag Flags First Sector Last
* Partition Tag Flags Sector Count Sector Mount Directory
* 0 2 00 34 629145600 629145633
* 1 4 00 629145634 629145600 1258291233
* 6 4 00 1258291234 1318633404 2576924637
* 8 11 00 2576924638 16384 2576941021
```

prtvtoc コマンドの出力では、次の 3 つのセクションに情報が表示されます。

- Dimensions
- フラグ
- Partition テーブル

| prtvtoc の列名     | 説明                                                              |
|-----------------|-----------------------------------------------------------------|
| Partition       | パーティションまたはスライスの番号。この列についての説明は、 <a href="#">表 7-3</a> を参照してください。 |
| Tag             | パーティションのタグ。この列についての説明は、 <a href="#">表 7-3</a> を参照してください。        |
| Flags           | パーティションのフラグ。この列についての説明は、 <a href="#">表 7-3</a> を参照してください。       |
| First Sector    | スライスまたはパーティションの最初のセクター。                                         |
| Sector Count    | スライスまたはパーティション内の合計セクター数。                                        |
| Last Sector     | スライスまたはパーティションの最後のセクター。                                         |
| Mount Directory | ファイルシステムの最後のマウントポイントのディレクトリを示します。                               |

## free hog スライスの使用方法

`format` ユーティリティを使用して 1 つまたは複数のディスクスライスのサイズを変更するときには、サイズ変更操作に対応して拡大縮小する一時スライスを指定します。

このスライスは、スライスを拡大すると領域を「解放 (free)」し、スライスを圧縮すると放棄された領域を「回収 (hog)」します。このため、提供側のスライスを「free hog」と呼びます。

free hog スライスは、インストール時または `format` ユーティリティの実行時にのみ存在します。日常の操作中に free hog スライスが継続して存在することはありません。

free hog スライスの使用方法については、[229 ページ](#)の「SPARC: ZFS ルートファイルシステム用のディスクスライスを作成する方法」または [246 ページ](#)の「x86: ZFS ルートファイルシステム用のディスクスライスを作成する方法」を参照してください。



## ディスク使用の管理 (タスク)

---

この章では、使用していないファイルや大きなディレクトリを見つけることにより、ディスク容量を最適化する方法を示します。

この章の内容は次のとおりです。

- 181 ページの「ディスク使用の管理 (タスクマップ)」
- 182 ページの「ファイルとディスク容量の情報の表示」
- 185 ページの「ファイルサイズの確認」
- 189 ページの「ディレクトリサイズの確認」
- 191 ページの「古いファイルまたは使用されていないファイルの検索と削除」

### ディスク使用の管理 (タスクマップ)

| タスク                       | 説明                                                                    | 参照先                                 |
|---------------------------|-----------------------------------------------------------------------|-------------------------------------|
| ファイルとディスク容量の情報を表示します。     | df コマンドを使用して、ディスク容量の利用状況に関する情報を表示します。                                 | 183 ページの「ファイルとディスク容量の情報を表示する方法」     |
| ファイルのサイズを表示します。           | ls コマンドと -lh オプションを使用して、ファイルのサイズに関する情報を表示します。                         | 186 ページの「ファイルサイズを表示する方法」            |
| サイズの大きなファイルを探します。         | ls -s コマンドを使用して、ファイルをサイズの降順にソートします。                                   | 187 ページの「サイズの大きなファイルを見つける方法」        |
| 指定されたサイズ制限を超えるファイルを見つけます。 | find コマンドと -size オプションを使用し、サイズ制限の値を指定して、そのサイズ制限を超えるファイルを見つけ、名前を表示します。 | 188 ページの「指定されたサイズ制限を超えるファイルを見つける方法」 |

| タスク                                | 説明                                                                                                                                                                    | 参照先                                                          |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| ディレクトリ、サブディレクトリ、およびファイルのサイズを表示します。 | <code>du</code> コマンドを使用して、1つ以上のディレクトリ、サブディレクトリ、およびファイルのサイズを表示する                                                                                                       | <a href="#">189 ページの「ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する方法」</a> |
| 最新ファイルのリストを表示します。                  | <code>ls -t</code> コマンドを使用して、もっとも新しく作成または変更されたファイルから順番にファイルのリストを表示します。                                                                                                | <a href="#">191 ページの「最新ファイルのリストを表示する方法」</a>                  |
| 古いファイルと使用されていないファイルを見つけて削除します。     | <code>find</code> コマンドと <code>-atime</code> および <code>-mtime</code> オプションを使用して、指定された日数アクセスされていないファイルを見つけます。これらのファイルは、 <code>rm 'cat filename'</code> コマンドを使用して削除できます。 | <a href="#">192 ページの「古いファイルと使用されていないファイルを見つけて削除する方法」</a>     |
| 一時ディレクトリを一度にクリアします。                | 一時ディレクトリを見つけ、 <code>rm -r *</code> コマンドを使用してディレクトリ全体を削除します。                                                                                                           | <a href="#">193 ページの「一時ディレクトリを一度にクリアする方法」</a>                |
| コアファイルを見つけて削除します。                  | <code>find . -name core -exec rm {} \;</code> コマンドを使用し、コアファイルを見つけて削除します。                                                                                              | <a href="#">194 ページの「コアファイルを見つけて削除する方法」</a>                  |
| クラッシュダンプファイルを削除します。                | <code>rm *</code> コマンドを使用して、 <code>/var/crash/</code> ディレクトリにあるクラッシュダンプファイルを削除します。                                                                                    | <a href="#">194 ページの「クラッシュダンプファイルを削除する方法」</a>                |

## ファイルとディスク容量の情報の表示

次の表に、ファイルのサイズとディスク容量の情報を表示するコマンドを示します。

| コマンド            | 説明                             | マニュアルページ               |
|-----------------|--------------------------------|------------------------|
| <code>df</code> | 空きディスクブロック数とファイル数を表示します。       | <a href="#">df(1M)</a> |
| <code>du</code> | 各サブディレクトリに割り当てられたディスク容量を表示します。 | <a href="#">du(1)</a>  |

| コマンド                    | 説明                                                                                                                  | マニュアルページ                 |
|-------------------------|---------------------------------------------------------------------------------------------------------------------|--------------------------|
| <code>find -size</code> | <code>-size</code> オプションで指定したサイズに基づいて、ディレクトリを再帰的に検索します。                                                             | <a href="#">find(1)</a>  |
| <code>ls -lh</code>     | ファイルのサイズを 1024 の累乗でリストします。                                                                                          | <a href="#">ls(1)</a>    |
| <code>zpool list</code> | プール領域サイズ、データセットや内部メタデータに割り当てられる領域量、割り当てられない領域量を表示します。                                                               | <a href="#">zpool.1m</a> |
| <code>zfs list</code>   | データセットや任意の子孫によって消費される領域量、データセットや任意の子孫が利用可能な領域量、このデータセットによって参照されるデータ量を表示します (プール内のほかのデータセットによって共有されることもされないこともあります)。 | <a href="#">zfs.1m</a>   |

利用可能な ZFS ストレージプールおよびファイルシステムの領域を判別する場合、`zpool list` および `zfs list` コマンドは、旧バージョンの `df` および `du` コマンドより優れています。旧バージョンのコマンドでは、プールおよびファイルシステムの領域を簡単に識別できず、下位のファイルシステムまたはスナップショットによって消費される領域の詳細を表示できません。

## ▼ ファイルとディスク容量の情報を表示する方法

### 1 ファイルシステムの種類を選択します。

- UFS
- ZFS

### 2 UFS ディスク領域の使用方法に関する情報を表示します。

```
$ df
/ (/dev/dsk/c0t0d0s0): 101294 blocks 105480 files
/devices (/devices): 0 blocks 0 files
/system/contract (ctfs): 0 blocks 2147483578 files
/proc (proc): 0 blocks 1871 files
/etc/mnttab (mnttab): 0 blocks 0 files
/etc/svc/volatile (swap): 992704 blocks 16964 files
/system/object (objfs): 0 blocks 2147483530 files
/usr (/dev/dsk/c0t0d0s6): 503774 blocks 299189 files
/dev/fd (fd): 0 blocks 0 files
/var/run (swap): 992704 blocks 16964 files
/tmp (swap): 992704 blocks 16964 files
/opt (/dev/dsk/c0t0d0s5): 23914 blocks 6947 files
/export/home (/dev/dsk/c0t0d0s7): 16810 blocks 7160 files
```

### 3 ZFS ディスク領域の使用方法に関する情報を表示します。

たとえば、次のルートプール(rpool)は、10.0GB が割り当て済みで、58.0GB は空き領域です。

```
zpool list rpool
NAME SIZE ALLOC FREE CAP HEALTH ALTROOT
rpool 68G 10.0G 58.0G 14% ONLINE -
```

個別のファイルシステムの USED 列からプールメタデータ用の一部領域を引いた値を確認することにより、プール領域の数値とファイルシステム領域の数値を比較すれば、プールの領域の詳細を確認できます。例:

```
zfs list -r rpool
NAME USED AVAIL REFER MOUNTPOINT
rpool 10.2G 56.8G 106K /rpool
rpool/ROOT 5.04G 56.8G 31K legacy
rpool/ROOT/solaris 5.04G 56.8G 5.04G /
rpool/dump 1.00G 56.8G 1.00G -
rpool/export 63K 56.8G 32K /export
rpool/export/home 31K 56.8G 31K /export/home
rpool/swap 4.13G 56.9G 4.00G -
```

#### 例 8-1 UFS ルートファイルシステムを使用したシステムのファイルサイズ情報を 1024 バイト単位で表示する

次の例では、UFS ルートファイルシステムを使用したシステムのファイルシステム情報が 1024 バイト単位で表示されています。

```
$ df -h
Filesystem size used avail capacity Mounted on
/dev/dsk/c0t0d0s0 249M 200M 25M 90% /
/devices 0K 0K 0K 0% /devices
ctfs 0K 0K 0K 0% /system/contract
proc 0K 0K 0K 0% /proc
mnttab 0K 0K 0K 0% /etc/mnttab
swap 485M 376K 485M 1% /etc/svc/volatile
objfs 0K 0K 0K 0% /system/object
/dev/dsk/c0t0d0s6 3.2G 2.9G 214M 94% /usr
fd 0K 0K 0K 0% /dev/fd
swap 485M 40K 485M 1% /var/run
swap 485M 40K 485M 1% /tmp
/dev/dsk/c0t0d0s5 13M 1.7M 10M 15% /opt
/dev/dsk/c0t0d0s7 9.2M 1.0M 7.3M 13% /export/home
```

/proc と /tmp は、ローカルファイルシステムですが、UFS ファイルシステムではありません。/proc は PROCFS ファイルシステムであり、/var/run と /tmp は TMPFS ファイルシステムであり、/etc/mnttab は MNTFS ファイルシステムです。



## 例 8-2 UFS ファイルシステムに割り当てられたブロックとファイルの合計数を表示する

次の例は、マウントされているすべてのファイルシステム、装置名、使用されている 512 バイトブロックの合計数、ファイル数を示しています。2 行構成の各エントリの 2 行目は、それぞれのファイルシステムに割り当てられているブロックの合計数とファイルの合計数を示します。

```
$ df -t
/ (/dev/dsk/c0t0d0s0): 101294 blocks 105480 files
 total: 509932 blocks 129024 files
/devices (/devices): 0 blocks 0 files
 total: 0 blocks 113 files
/system/contract (ctfs): 0 blocks 2147483578 files
 total: 0 blocks 69 files
/proc (proc): 0 blocks 1871 files
 total: 0 blocks 1916 files
/etc/mnttab (mnttab): 0 blocks 0 files
 total: 0 blocks 1 files
/etc/svc/volatile (swap): 992608 blocks 16964 files
 total: 993360 blocks 17025 files
/system/object (objfs): 0 blocks 2147483530 files
 total: 0 blocks 117 files
/usr (/dev/dsk/c0t0d0s6): 503774 blocks 299189 files
 total: 6650604 blocks 420480 files
/dev/fd (fd): 0 blocks 0 files
 total: 0 blocks 31 files
/var/run (swap): 992608 blocks 16964 files
 total: 992688 blocks 17025 files
/tmp (swap): 992608 blocks 16964 files
 total: 992688 blocks 17025 files
/opt (/dev/dsk/c0t0d0s5): 23914 blocks 6947 files
 total: 27404 blocks 7168 files
/export/home (/dev/dsk/c0t0d0s7): 16810 blocks 7160 files
 total: 18900 blocks 7168 files
```

## ファイルサイズの確認

ls コマンドを使用して、ファイルサイズを調べたりソートしたりできます。また、find コマンドを使用して、サイズの制限を超えているファイルを探することができます。詳細については、[ls\(1\)](#) および [find\(1\)](#) を参照してください。

---

注 - /var ディレクトリの容量がなくなった場合、/var ディレクトリに、ファイルシステム上の空き容量のあるディレクトリへのシンボリックリンクを設定しないでください。たとえそれが一時的な策であっても、デーモンプロセスおよびユーティリティの一部に問題が生じることがあります。

---

## ▼ ファイルサイズを表示する方法

1 確認したいファイルがあるディレクトリに移動します。

2 次のように入力して、ファイルのサイズを表示します。

```
$ ls [-lh] [-s]
```

-l 長形式でファイルとディレクトリのリストを表示し、それぞれのサイズをバイト単位で示します。(次の例を参照)

-h ファイルやディレクトリのサイズが 1024 バイトより大きい場合、ファイルとディレクトリのサイズを K バイト、M バイト、G バイト、または T バイト単位で示します。このオプションは、-o、-n、-@、および -g オプションによる出力も、ファイルやディレクトリのサイズを新しい形式で表示するよう変更します。詳細は、[ls\(1\)](#) のマニュアルページを参照してください。

-s ファイルとディレクトリのリストを表示し、それぞれのサイズをブロック単位で示します。

### 例 8-3 ファイルサイズを表示する

次の例は、lastlog と messages が /var/adm ディレクトリ内のその他のファイルよりも大きいことを示します。

```
$ cd /var/adm
$ ls -lh
total 148
drwxrwxr-x 5 adm adm 512 Nov 26 09:39 acct/
-rw----- 1 uucp bin 0 Nov 26 09:25 aculog
drwxr-xr-x 2 adm adm 512 Nov 26 09:25 exacct/
-r--r--r-- 1 root other 342K Nov 26 13:56 lastlog
drwxr-xr-x 2 adm adm 512 Nov 26 09:25 log/
-rw-r--r-- 1 root root 20K Nov 26 13:55 messages
drwxr-xr-x 2 adm adm 512 Nov 26 09:25 passwd/
drwxrwxr-x 2 adm sys 512 Nov 26 09:39 sa/
drwxr-xr-x 2 root sys 512 Nov 26 09:49 sm.bin/
-rw-rw-rw- 1 root bin 0 Nov 26 09:25 spellhist
drwxr-xr-x 2 root sys 512 Nov 26 09:25 streams/
-rw-r--r-- 1 root bin 3.3K Nov 26 13:56 utmpx
-rw-r--r-- 1 root root 0 Nov 26 10:17 vold.log
-rw-r--r-- 1 adm adm 19K Nov 26 13:56 wtmpx
```

次の例は、lpsched.1 ファイルが 2 ブロックを使用していることを示します。

```
$ cd /var/lp/logs
$ ls -s
total 2 0 lpsched 2 lpsched.1
```

## ▼ サイズの大きなファイルを見つける方法

- 1 検索したいディレクトリに移動します。
- 2 ファイルのサイズをブロック数でもっとも大きいものから降順に表示します。
  - ファイルの文字またはカラムが異なる場合、次のコマンドを使用して、ブロックサイズによりもっとも大きいものから降順で、ファイルのリストをソートします。

```
$ ls -l | sort +4rn | more
```

このコマンドは、左から4番目のフィールドにある文字により、リスト内のファイルをソートすることに注意してください。

- ファイルの文字またはカラムが同じである場合、次のコマンドを使用して、ブロックサイズによりもっとも大きいものから降順で、ファイルのリストをソートします。

```
$ ls -s | sort -nr | more
```

このコマンドは、もっとも左側の文字から始め、リスト内のファイルをソートすることに注意してください。

### 例8-4 サイズの大きなファイルを見つける (5番目のフィールドの文字によるソート)

```
$ cd /var/adm
$ ls -l | sort +4rn | more
-r--r--r-- 1 root root 4568368 Oct 17 08:36 lastlog
-rw-r--r-- 1 adm adm 697040 Oct 17 12:30 pacct.9
-rw-r--r-- 1 adm adm 280520 Oct 17 13:05 pacct.2
-rw-r--r-- 1 adm adm 277360 Oct 17 12:55 pacct.4
-rw-r--r-- 1 adm adm 264080 Oct 17 12:45 pacct.6
-rw-r--r-- 1 adm adm 255840 Oct 17 12:40 pacct.7
-rw-r--r-- 1 adm adm 254120 Oct 17 13:10 pacct.1
-rw-r--r-- 1 adm adm 250360 Oct 17 12:25 pacct.10
-rw-r--r-- 1 adm adm 248880 Oct 17 13:00 pacct.3
-rw-r--r-- 1 adm adm 247200 Oct 17 12:35 pacct.8
-rw-r--r-- 1 adm adm 246720 Oct 17 13:15 pacct.0
-rw-r--r-- 1 adm adm 245920 Oct 17 12:50 pacct.5
-rw-r--r-- 1 root root 190229 Oct 5 03:02 messages.1
-rw-r--r-- 1 adm adm 156800 Oct 17 13:17 pacct
-rw-r--r-- 1 adm adm 129084 Oct 17 08:36 wttmpx
```

### 例8-5 サイズの大きなファイルを見つける (もっとも左側の文字によるソート)

次の例では、lastlog と messages ファイルが /var/adm 内でもっとも大きなファイルです。

```
$ cd /var/adm
$ ls -s | sort -nr | more
48 lastlog
30 messages
24 wtmpx
18 pacct
8 utmpx
2 vold.log
2 sulog
2 sm.bin/
2 sa/
2 passwd/
2 pacct1
2 log/
2 acct/
0 spellhist
0 aculog
total 144
```

## ▼ 指定されたサイズ制限を超えるファイルを見つける方法

- 次のように **find** コマンドを使用して、指定したサイズを超えるファイルを見つけてファイル名を表示します。

```
$ find directory -size +nnn
```

*directory*          検索するディレクトリを指定します。

*-size +nnn*          512 バイトブロック数です。このサイズを上回るファイルがリストされます。

### 例 8-6 指定されたサイズ制限を超えるファイルを見つける

次の例は、作業中のカレントディレクトリ内で 400 ブロックを超えるファイルを見つける方法を示します。-print オプションは、find コマンドの出力を表示します。

```
$ find . -size +400 -print
./Howto/howto.doc
./Howto/howto.doc.backup
./Howto/howtotest.doc
./Routine/routineBackupconcepts.doc
./Routine/routineIntro.doc
./Routine/routineTroublefsck.doc
./.record
./Mail/pagination
./Config/configPrintadmin.doc
./Config/configPrintsetup.doc
./Config/configMailappx.doc
./Config/configMailconcepts.doc
./snapshot.rs
```

# ディレクトリサイズの確認

du コマンドとオプションを使用すると、ディレクトリのサイズを表示できます。さらに quot コマンドを使用すれば、ユーザーアカウントによって占められるローカル UFS ファイルシステム上のディスク容量のサイズを知ることができます。これらのコマンドの詳細については、[du\(1\)](#) および [quot\(1M\)](#) を参照してください。

## ▼ ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する方法

- du コマンドを使用して、1つ以上のディレクトリ、サブディレクトリ、およびファイルのサイズを表示するサイズは512バイトブロック単位で表示されます。

`$ du [-as] [directory ...]`

du 指定した各ディレクトリとそれらの下の各サブディレクトリのサイズを表示します。

-a 指定したディレクトリ内の各ファイルと各サブディレクトリのサイズと合計ブロック数を表示します。

-s 指定したディレクトリ内の合計ブロック数を表示します。

-h ディレクトリのサイズを 1024 バイト単位のブロック数で表示します。

-H ディレクトリのサイズを 1000 バイト単位のブロック数で表示します。

[directory ...] 調べたい1つ以上のディレクトリを指定します。複数のディレクトリを指定する場合は、コマンド行構文で空白を使用して区切ります。

### 例 8-7 ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する

次の例は、2つのディレクトリのサイズを示しています。

```
$ du -s /var/adm /var/spool/lp
130 /var/adm
40 /var/spool/lp
```

次の例は、2つのディレクトリとそれらのすべてのサブディレクトリとファイルのサイズを示しています。各ディレクトリ内の合計ブロック数も表示されます。

```
$ du /var/adm /var/spool/lp
2 /var/adm/exacct
```

```

2 /var/adm/log
2 /var/adm/streams
2 /var/adm/acct/fiscal
2 /var/adm/acct/nite
2 /var/adm/acct/sum
8 /var/adm/acct
2 /var/adm/sa
2 /var/adm/sm.bin
258 /var/adm
4 /var/spool/lp/admins
2 /var/spool/lp/requests/printing....
4 /var/spool/lp/requests
4 /var/spool/lp/system
2 /var/spool/lp/fifos
24 /var/spool/lp

```

次の例は、ディレクトリのサイズを 1024 バイト単位のブロック数で示しています。

```

$ du -h /usr/share/audio
796K /usr/share/audio/samples/au
797K /usr/share/audio/samples
798K /usr/share/audio

```

## ▼ ローカル **UFS** ファイルシステムのユーザーの所有状況を表示する方法

- 1 スーパーユーザーになります。
- 2 次のように入力して、ユーザー、ディレクトリまたはファイルシステム、1024 バイト単位のブロック数を表示します。

```
quot [-a] [filesystem ...]
```

**-a**            マウントされている各 UFS ファイルシステムの全ユーザーと 1024 バイト単位のブロック数を表示します。

**filesystem**    UFS ファイルシステムを指定します。このファイルシステムについて、ユーザーと使用されているブロック数が表示されます。

---

注 -quot コマンドは、ローカル UFS ファイルシステムに対してだけ使用できます。

---

### 例 8-8 ローカル UFS ファイルシステムのユーザーの所有状況を表示する

次の例では、ルート (/) ファイルシステムのユーザーが表示されています。その次に、マウントされた全 UFS ファイルシステムのユーザーが表示されています。

```
quot /
/dev/rdisk/c0t0d0s0:
```

```

43340 root
3142 rimmer
47 uucp
35 lp
30 adm
4 bin
4 daemon

quot -a
/dev/rdsk/c0t0d0s0 (/):
43340 root
3150 rimmer
47 uucp
35 lp
30 adm
4 bin
4 daemon
/dev/rdsk/c0t0d0s6 (/usr):
460651 root
206632 bin
791 uucp
46 lp
4 daemon
1 adm
/dev/rdsk/c0t0d0s7 (/export/home):
9 root

```

## 古いファイルまたは使用されていないファイルの検索と削除

負荷の大きいファイルシステムを整理する作業として、最近使用されていないファイルの特定と削除があります。使用されていないファイルは `ls` コマンドまたは `find` コマンドを使用して見つけることができます。詳細については、[ls\(1\)](#) および [find\(1\)](#) を参照してください。

ディスク容量を節約するには、`/var/tmp` または `/var/spool` 内にある一時ディレクトリを空にしたり、コアファイルやクラッシュダンプファイルを削除したりする方法もあります。クラッシュダンプファイルの詳細は、『[System Administration Guide: Advanced Administration](#)』の第 17 章「[Managing System Crash Information \(Tasks\)](#)」を参照してください。

### ▼ 最新ファイルのリストを表示する方法

- `ls -t` コマンドを使用して、もっとも新しく作成または変更されたファイルから順番にファイルのリストを表示します。

```
$ ls -t [directory]
```

-t            最新タイムスタンプのファイルを最初にしてソートします。

directory    検索するディレクトリを指定します。

### 例 8-9 最新ファイルのリストを表示する

次の例では、ls -tl コマンドを使用して、/var/adm ディレクトリ内でもっとも新しく作成または変更されたファイルを見つける方法を示します。sulog がもっとも新しく作成または変更されたファイルです。

```
$ ls -tl /var/adm
total 134
-rw----- 1 root root 315 Sep 24 14:00 sulog
-r--r--r-- 1 root other 350700 Sep 22 11:04 lastlog
-rw-r--r-- 1 root bin 4464 Sep 22 11:04 utmpx
-rw-r--r-- 1 adm adm 20088 Sep 22 11:04 wtmpx
-rw-r--r-- 1 root other 0 Sep 19 03:10 messages
-rw-r--r-- 1 root other 0 Sep 12 03:10 messages.0
-rw-r--r-- 1 root root 11510 Sep 10 16:13 messages.1
-rw-r--r-- 1 root root 0 Sep 10 16:12 vold.log
drwxr-xr-x 2 root sys 512 Sep 10 15:33 sm.bin
drwxrwxr-x 5 adm adm 512 Sep 10 15:19 acct
drwxrwxr-x 2 adm sys 512 Sep 10 15:19 sa
-rw----- 1 uucp bin 0 Sep 10 15:17 aculog
-rw-rw-rw- 1 root bin 0 Sep 10 15:17 spellhist
drwxr-xr-x 2 adm adm 512 Sep 10 15:17 log
drwxr-xr-x 2 adm adm 512 Sep 10 15:17 passwd
```

## ▼ 古いファイルと使用されていないファイルを見つけて削除する方法

- 1 スーパーユーザーになります。
- 2 指定した日数の間アクセスのないファイルを見つけて、ファイルにそれらのリストを書き込みます。

```
find directory -type f[-atime +nnn] [-mtime +nnn] -print > filename &
```

directory    検索するディレクトリを指定します。このディレクトリの下にあるディレクトリも検索します。

-atime +nnn    指定した日数 (nnn) の間アクセスのないファイルを見つけます。

-mtime +nnn    指定した日数 (nnn) の間変更のないファイルを見つけます。

filename      使用されないファイルリストを書き込むファイルを指定します。

- 3 上の手順でリストに書き込んだ使用されていないファイルを削除します。

```
rm 'cat filename'
```



*filename* は、前の手順で作成したファイルです。このファイルには、使用されていないファイルのリストが入っています。

### 例 8-10 古いファイルまたは使用されていないファイルの検索と削除

次の例は、`/var/adm` ディレクトリ内とそのサブディレクトリ内で過去 60 日にわたってアクセスされていないファイルを示しています。`/var/tmp/deadfiles` ファイルには、使用されていないファイルのリストが含まれます。`rm` コマンドは、これらの使用されていないファイルを削除します。

```
find /var/adm -type f -atime +60 -print > /var/tmp/deadfiles &
more /var/tmp/deadfiles
/var/adm/aculog
/var/adm/spellhist
/var/adm/wtmpx
/var/adm/sa/sa13
/var/adm/sa/sa27
/var/adm/sa/sa11
/var/adm/sa/sa23
/var/adm/sulog
/var/adm/vold.log
/var/adm/messages.1
/var/adm/messages.2
/var/adm/messages.3
rm 'cat /var/tmp/deadfiles'
#
```

## ▼ 一時ディレクトリを一度にクリアする方法

- 1 スーパーユーザーになります
- 2 整理するディレクトリに移動します。

```
cd directory
```



注意 - 手順 3 を実行する前に、正しいディレクトリにいることを確認してください。手順 3 はカレントディレクトリ内のすべてのファイルを削除します。

- 3 カレントディレクトリ内のファイルとサブディレクトリを削除します。  

```
rm -r *
```
- 4 サブディレクトリやファイルで、不要なもの、一時的なもの、または古くなったものがある他のディレクトリに移動します。
- 5 手順 3 を繰り返してこれらのサブディレクトリやファイルを削除します。

### 例 8-11 一時ディレクトリを一度にクリアする

次の例は、mywork ディレクトリを空にする方法、およびすべてのファイルとサブディレクトリが削除されたことを確認する方法を示しています。

```
cd mywork
ls
filea.000
fileb.000
filec.001
rm -r *
ls
#
```

## ▼ コアファイルを見つけて削除する方法

- 1 スーパーユーザーになります。  
コアファイルを検索したいディレクトリに移動します。
- 2 ディレクトリとサブディレクトリ内のすべてのコアファイルを見つけて削除します。

```
find . -name core -exec rm {} \;
```

### 例 8-12 コアファイルを見つけて削除する

次の例は、find コマンドを使用して jones のユーザーアカウントからコアファイルを見つけて削除する方法を示します。

```
cd /home/jones
find . -name core -exec rm {} \;
```

## ▼ クラッシュダンプファイルを削除する方法

クラッシュダンプファイルは非常に大きくなる可能性があります。これらのファイルを保存するようシステムで設定している場合は、必要以上に長期間保存しないでください。

- 1 次のように入力して、クラッシュダンプファイルが格納されているディレクトリに移動します。

```
cd /var/crash/system
```

*system* は、クラッシュダンプファイルを作成したシステムのことです。



注意-手順3を実行する前に、正しいディレクトリにいることを確認してください。手順3はカレントディレクトリ内のすべてのファイルを削除します。

- 2 クラッシュダンプファイルを削除します。

```
rm *
```

- 3 クラッシュダンプファイルが削除されたことを確認します。

```
ls
```

#### 例 8-13 クラッシュダンプファイルを削除する

次の例は、システム `venus` からクラッシュダンプファイルを削除する方法、およびクラッシュダンプファイルが削除されたことを確認する方法を示します。

```
cd /var/crash/venus
rm *
ls
```



## ディスクの管理 (タスク)

---

この章では、ディスク管理の手順について説明します。Oracle Solaris を実行するシステム上でディスクを管理する方法に精通している場合は、この章で説明する多くの内容を読み飛ばすことができます。

この章の内容は次のとおりです。

- 197 ページの「ディスクの管理 (タスクマップ)」
- 198 ページの「システム上のディスクの確認」
- 200 ページの「ディスクのフォーマット」
- 203 ページの「ディスクスライスの表示」
- 205 ページの「ディスクラベルの作成と検査」
- 212 ページの「破損したディスクラベルの復元」
- 214 ページの「他社製のディスクの追加」

ディスク管理の概要については、第7章「ディスクの管理 (概要)」を参照してください。

### ディスクの管理 (タスクマップ)

| タスク               | 説明                                                             | 参照先                                            |
|-------------------|----------------------------------------------------------------|------------------------------------------------|
| システム上のディスクを確認します。 | システム上のディスクの種類がわからない場合は、 <code>format</code> ユーティリティを使用して確認します。 | <a href="#">198 ページの「システム上のディスクを確認する方法」</a>    |
| ディスクをフォーマットします。   | <code>format</code> ユーティリティを使用して、ディスクがフォーマット済みかどうかを判断します。      | <a href="#">201 ページの「ディスクがフォーマット済みかを調べる方法」</a> |

| タスク                | 説明                                                                    | 参照先                                          |
|--------------------|-----------------------------------------------------------------------|----------------------------------------------|
|                    | ほとんどの場合、ディスクはフォーマット済みです。フォーマットする必要がある場合は、 <code>format</code> を使用します。 | <a href="#">201 ページの「ディスクをフォーマットする方法」</a>    |
| スライス情報を表示します。      | <code>format</code> ユーティリティを使用してスライス情報を表示します。                         | <a href="#">203 ページの「ディスクスライス情報を表示する方法」</a>  |
| ディスクラベルの作成         | <code>format</code> ユーティリティを使用してディスクラベルを作成します。                        | <a href="#">205 ページの「ディスクラベルを作成する方法」</a>     |
| ディスクラベルを検査します。     | <code>prtvtoc</code> コマンドを使用してディスクラベルを検査します。                          | <a href="#">210 ページの「ディスクラベルを検査する方法」</a>     |
| 破損したディスクラベルを復元します。 | システム障害または電源障害のために破損したディスクラベルの復元を試みます。                                 | <a href="#">212 ページの「破損したディスクラベルを復元する方法」</a> |

## システム上のディスクの確認

`format` ユーティリティを使用して、システムに接続されているディスクの種類を調べます。また、`format` ユーティリティを使用して、ディスクがシステムに認識されるかどうかを検査することもできます。`format` ユーティリティの使用の詳細は、[第 13 章「format ユーティリティ \(参照情報\)」](#) を参照してください。

### ▼ システム上のディスクを確認する方法

- 1 スーパーユーザーになります。
- 2 `format` ユーティリティを使用して、システム上で認識されるディスクを確認します。

# `format`

`format` ユーティリティは、AVAILABLE DISK SELECTIONS という見出しの下に、認識されるディスクのリストを表示します。

#### 例 9-1 システム上のディスクを確認する

次の例は、`format` コマンドの出力を示します。

```
format
AVAILABLE DISK SELECTIONS:
 0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
```

```

 /pci@1c,600000/scsi@2/sd@0,0
1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@1c,600000/scsi@2/sd@1,0
2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
 /pci@1c,600000/scsi@2/sd@2,0
3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
 /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number):

```

ディスクの物理デバイス名と論理デバイス名は、括弧 <> 内の商品名に対応しています。次の例を参照してください。この方法では、どの論理デバイス名がシステムに接続されたディスクを表しているかをすぐに識別できます。論理デバイス名と物理デバイス名については、69 ページの「**デバイス名の命名規則**」を参照してください。

次の例では、ワイルドカードを使用して、コントローラ 0 に接続された 4 つのディスクを表示します。

```

format /dev/rdisk/c0t6*
AVAILABLE DISK SELECTIONS:
0. /dev/rdisk/c0t600A0B800022024E000054AC4970A629d0p0 <...>
 /scsi_vhci/disk@g600a0b800022024e000054ac4970a629
1. /dev/rdisk/c0t600A0B800022024E000054AE4970A711d0p0 <...>
 /scsi_vhci/disk@g600a0b800022024e000054ae4970a711
2. /dev/rdisk/c0t600A0B800022028A000050444970A834d0p0 <...>
 /scsi_vhci/disk@g600a0b800022028a000050444970a834
3. /dev/rdisk/c0t600A0B800022028A000050454970A8EAd0p0 <...>
 /scsi_vhci/disk@g600a0b800022028a000050454970a8ea
Specify disk (enter its number):

```

次の例は、ディスク情報を確認する方法を示しています。

```

format
0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
 /pci@1c,600000/scsi@2/sd@0,0

```

出力から、ディスク 0 (ターゲット 0) が 2 番目の SCSI ホストアダプタ (scsi@2) に接続され、この 2 番目の SCSI ホストアダプタが 2 番目の PCI インタフェース (/pci@1c,600000/...) に接続されていることがわかります。物理デバイス名と論理デバイス名は、ディスクの商品名 SUN36G に対応しています。

ディスクの中には商品名を持たないものもあります。format の出力にディスクの商品名が含まれていない場合、次に示す手順に従って format ユーティリティの type および label 機能を使用して、ディスクの商品名を含めることができます。

次の手順は、ディスクまたはシステムがアクティブでない状態で (つまりインストール DVD またはネットワークからブートして) 実行する必要があります。ただし、ディスクが現在使用されておらず、Oracle Solaris リリースを格納しない場合は除きます。また、手順の最後でディスクにラベルを付け直すことで、既存パーティションの情報やデータをすべて削除できます。

```
format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
 0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
 /pci@1c,600000/scsi@2/sd@0,0
 1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
 /pci@1c,600000/scsi@2/sd@1,0
 2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
 /pci@1c,600000/scsi@2/sd@2,0
 3. c2t3d0 <drive type unknown>
 /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> type
AVAILABLE DRIVE TYPES:
 0. Auto configure
 1. other
Specify disk type (enter its number): 0
c2t3d0: configured with capacity of 33.92GB
<SEAGATE-ST336607LSUN36G-0507-33.92GB>
[disk formatted]
format> label
Ready to label disk, continue? yes
format> quit
```

参考 format ユーティリティーがディスクを認識しない場合

- 第10章「SPARC: ディスクの設定(手順)」または243ページの「x86: ZFS ファイルシステム用のディスクの設定(タスクマップ)」を参照してください。
- 205ページの「ディスクラベルを作成する方法」へ進みます。
- ディスクのハードウェアドキュメントを参照して、ディスクをシステムに接続します。

## ディスクのフォーマット

多くの場合、ディスクは製造元または再販業者によってフォーマットされています。通常は再フォーマットしなくてもドライブを取り付けることができます。

次の作業の前にディスクをフォーマットしておかなければなりません。

- データをディスクへ書き込みます。ただし、ほとんどのディスクはフォーマット済みです。
- Oracle Solaris インストールユーティリティーを使ってシステムをインストールします。





注意-フォーマットは、ディスク上のデータを上書きする、破壊的なプロセスです。このため、通常は製造元や再販業者のみがディスクをフォーマットします。ディスクに欠陥があるために問題が再発していると思われる場合は、**format** ユーティリティを使用して表面解析を実行できますが、データを破壊しないコマンドだけを使用するように注意してください。

## ▼ ディスクがフォーマット済みかを調べる方法

- 1 スーパーユーザーになります。
- 2 **format** ユーティリティを起動します。  

```
format
```

番号付きのディスクのリストが表示されます。
- 3 確認するディスクの番号を入力します。  
Specify disk (enter its number): 0
- 4 ディスクがフォーマット済みかを調べます。選択したディスクがフォーマット済みであれば、次のメッセージが表示されます。  
[disk formatted]

### 例9-2 ディスクがフォーマット済みかを調べる

次の例は、ディスク **c2t3d0** がフォーマット済みであることを示しています。

```
format
AVAILABLE DISK SELECTIONS:
 0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@lc,600000/scsi@2/sd@0,0
 1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@lc,600000/scsi@2/sd@1,0
 2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
 /pci@lc,600000/scsi@2/sd@2,0
 3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
 /pci@lc,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
```

## ▼ ディスクをフォーマットする方法

ディスクは製造元によってフォーマットされています。ディスクの再フォーマットはまれにしか発生しないはずです。この処理を行うと、終了までに長い時間がかかるうえ、すべてのデータがディスクから削除されてしまいます。

- 1 スーパーユーザーになります。
- 2 **format** ユーティリティーを起動します。  
**# format**  
番号付きのディスクのリストが表示されます。
- 3 フォーマットするディスクの番号を入力します。  
Specify disk (enter its number): **3**



---

注意-ルートファイルシステムを含むディスクを選択しないでください。ルート  
プールディスクをフォーマットすると、OSやこのディスク上のデータがすべて削除  
されます。

---

- 4 ディスクのフォーマットを開始するには、**format>**プロンプトで**format**と入力しま  
す。**y**と入力してコマンドを確認します。  
  
**format> format**  
The protection information is not enabled  
The disk will be formatted with protection type 0  
  
Ready to format. Formatting cannot be interrupted  
and takes 169 minutes (estimated). Continue? **yes**
- 5 フォーマットが正常に行われたことを、次のメッセージによって確認します。  
Beginning format. The current time is Fri Apr 1 ...  
  
Formatting...  
done  
  
Verifying media...  
pass 0 - pattern = 0xc6dec6de  
14086/23/734  
  
pass 1 - pattern = 0xdb6db6d  
14086/23/734  
  
Total of 0 defective blocks repaired.
- 6 **format** ユーティリティーを終了します。  
**format> quit**

# ディスクスライスの表示

ZFS ストレージプールを作成する最善の方法は、ディスクスライスの代わりにディスク全体を使用することです。というのも、ディスク全体のほうが管理しやすいからです。ディスクスライスを使用する必要があるのは、ディスクが ZFS ルートプールで使用するためのものである場合だけです。これは、長年にわたるブートの制限です。ルート以外のプールではディスク全体を使用してください。ディスク全体を使ってプールを作成すると、EFI ラベルが適用されます。以下の EFI ディスクラベルの例を参照してください。

ルートプールディスクとして使用するディスクを準備する必要がある場合には、以下の VTOC ディスクラベルの例に示すような、ディスクの全容量を含むスライス 0 を作成します。

ZFS ストレージプールで使用するディスクの設定については、[第 10 章「SPARC: ディスクの設定\(手順\)」](#)または[243 ページの「x86: ZFS ファイルシステム用のディスクの設定\(タスクマップ\)」](#)を参照してください。

---

注 - `format` ユーティリティでは、「スライス」ではなく「パーティション」という用語を使用します。

---

## ▼ ディスクスライス情報を表示する方法

ディスクが ZFS ルートプールで使用するためのものである場合、ディスクスライス情報を表示しなければならない可能性があります。また、SMI ラベルが含まれている必要もあります。

- 1 スーパーユーザーになります。
- 2 `format` ユーティリティを起動します。  
`# format`  
番号付きのディスクのリストが表示されます。
- 3 スライス情報を表示するディスクの番号を入力します。  
`Specify disk (enter its number):1`
- 4 `partition` メニューを選択します。  
`format> partition`
- 5 選択されたディスクのスライス情報が表示されます。  
`partition> print`

- 6 **format** ユーティリティーを終了します。
- partition> q  
format> q
- 7 特定のスライスのタグとサイズについてスライス情報が表示されることを確認します。
- 画面の出力に、スライスサイズが割り当てられていないことが示された場合は、ディスクにスライスがないものと思われます。

例9-3 ディスクスライス情報を表示する

次に示すのは、VTOC ラベル付きディスクのスライス情報を表示する例です。

```
format
Searching for disks...done
Specify disk (enter its number):3
Selecting c2t3d0
format> partition
partition> print
Current partition table (c2t3d0):
Total disk cylinders available: 14087 + 2 (reserved cylinders)
```

| Part | Tag        | Flag | Cylinders | Size     | Blocks                |
|------|------------|------|-----------|----------|-----------------------|
| 0    | root       | wm   | 0 - 14086 | 136.71GB | (14087/0/0) 286698624 |
| 1    | swap       | wu   | 0         | 0        | (0/0/0) 0             |
| 2    | backup     | wu   | 0 - 14086 | 136.71GB | (14087/0/0) 286698624 |
| 3    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 4    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 5    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 6    | usr        | wm   | 0         | 0        | (0/0/0) 0             |
| 7    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |

```
partition> q
format> q
```

これらの例のスライス情報の詳細は、第7章「ディスクの管理 (概要)」を参照してください。

次の例は、EFI ラベル付きディスクのスライス情報を示しています。

```
format
Searching for disks...done
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> partition
partition> print
Current partition table (default):
Total disk sectors available: 286722878 + 16384 (reserved sectors)
```

| Part | Tag        | Flag | First Sector | Size     | Last Sector |
|------|------------|------|--------------|----------|-------------|
| 0    | usr        | wm   | 34           | 136.72GB | 286722911   |
| 1    | unassigned | wm   | 0            | 0        | 0           |
| 2    | unassigned | wm   | 0            | 0        | 0           |

```

3 unassigned wm 0 0 0
4 unassigned wm 0 0 0
5 unassigned wm 0 0 0
6 unassigned wm 0 0 0
7 unassigned wm 0 0 0
8 reserved wm 286722912 8.00MB 286739295
partition> q
format> q

```

次の例は、ルートプールの EFI (GPT) ラベル付きディスクのスライス情報を示しています。

```

format
Searching for disks...done
Specify disk (enter its number): 0
selecting c2t0d0
[disk formatted]
format> partition
partition> print
Current partition table (original):
Total disk sectors available: 27246525 + 16384 (reserved sectors)
Part Tag Flag First Sector Size Last Sector
0 BIOS_ boot wm 256 256.00MB 524543
1 usr wm 524544 12.74GB 27246558
2 unassigned wm 0 0 0
3 unassigned wm 0 0 0
4 unassigned wm 0 0 0
5 unassigned wm 0 0 0
6 unassigned wm 0 0 0
8 reserved wm 27246559 8.00MB 27262942
partition> q
format> q

```

## ディスクラベルの作成と検査

一般に、ディスクにラベルを付ける操作は、システムのインストール時、または新しいディスクを使用するときに行います。ディスクラベルが破損したときは、新しくラベルを付け直す必要があります。ディスクラベルの破損は、電源障害が発生した場合などに起こります。

format ユーティリティーは、ラベルが付いていない SCSI ディスクを自動構成しようとし、format ユーティリティーがラベルの付いていないディスクを自動構成できる場合、次のようなメッセージが表示されます。

```
c2t3d0: configured with capacity of 136.73GB
```

### ▼ ディスクラベルを作成する方法

2T バイト未満のディスクに EFI ラベルを付ける場合は、[例 9-5](#) を参照してください。

- 1 スーパーユーザーになります。
- 2 **format** ユーティリティーを起動します。  
`# format`  
 番号付きのディスクのリストが表示されます。
- 3 ラベルを付けたいディスクの番号を入力します。  
`Specify disk (enter its number):1`  
**format** ユーティリティーでディスクタイプが認識されたら、次の手順はバックアップラベルを検索してディスクにラベルを付けることです。バックアップラベルを使用してディスクにラベルを付けると、ディスクタイプとジオメトリだけでなく、正しいパーティション情報を使用してディスクラベルが作成されます。
- 4 次のいずれかの方法を選択して、ディスクにラベルを付けます。
  - ディスクが正常に構成されていて、ラベルが付いていない場合は、手順5に進みます。  
`format` ユーティリティーにより、ディスクラベルを付けるかを尋ねるプロンプトが表示されます。
  - ラベル付きディスクのディスクタイプを変更する場合、またはこのディスクを `format` ユーティリティーで自動構成できなかった場合は、手順6に進んでディスクタイプを設定し、ラベルを付けます。
- 5 **Label it now?** プロンプトで **y** と入力して、ディスクにラベルを付けます。  
`Disk not labeled. Label it now? y`  
 これでディスクラベルが作成されました。手順10に進んで `format` ユーティリティーを終了します。
- 6 **format>** プロンプトで **type** と入力します。  
`format> type`  
 Available Drive Types メニューが表示されます。
- 7 ディスクタイプの候補のリストからディスクタイプを選択します。  
`Specify disk type (enter its number)[12]: 12`  
 または、**0** を選択して SCSI-2 ディスクを自動構成します。
- 8 ディスクラベルの作成ディスクにラベルが付いていない場合は、次のメッセージが表示されます。  
`Disk not labeled. Label it now? y`

ディスクラベルが付いている場合は、次のメッセージが表示されます。

```
Ready to label disk, continue? y
```

## 9 ディスクラベルを検査します。

```
format> verify
```

## 10 format ユーティリティを終了します。

```
format> q
#
```

### 例9-4 ディスクラベルを作成する

次の例では、36Gバイトのディスクを自動構成してラベルを付ける方法を示します。

```
format
AVAILABLE DISK SELECTIONS:
 0. c0t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
 /pci@1c,600000/scsi@2/sd@0,0
 1. c0t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
 /pci@1c,600000/scsi@2/sd@1,0
 2. c0t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
 /pci@1c,600000/scsi@2/sd@2,0
 3. c0t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
 /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
c0t3d0: configured with capacity of 33.92GB
Disk not labeled. Label it now? yes
format> verify
format> q
```

### 例9-5 ディスクにEFI ラベルを付ける

次の例は、`format -e` コマンドを使って、ディスクにEFI ラベルまたはSMI (VTOC) ラベルを付ける方法を示しています。階層化されたソフトウェア製品がEFI ラベル付きディスクのシステムでも動作することを確認しておいてください。EFI ラベルの一般的な制限事項については、[170 ページの「EFI ディスクラベルの制限」](#)を参照してください。

```
format -e
Searching for disks...done
AVAILABLE DISK SELECTIONS:
 0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@1c,600000/scsi@2/sd@0,0
 1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@1c,600000/scsi@2/sd@1,0
 2. c2t2d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
 /pci@1c,600000/scsi@2/sd@2,0
 3. c2t3d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
 /pci@1c,600000/scsi@2/sd@3,0
```

```
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[0]: 1
Ready to label disk, continue? yes
format> quit
```

例9-6 EFI ラベル付きディスクを SMI ラベル付きディスクに変更する

次の例は、format - e コマンドを使用して、EFI ラベル付きディスクを ZFS ルートプールで使用可能な SMI ラベル付きディスクに変更する方法を示したものです。

x86 ベースのシステムでは、最初に EFI fdisk パーティションを Solaris パーティションに変更する必要があります。例:

```
format -e
select disk ...
format> fdisk
FORMAT MENU:
 disk - select a disk
 type - select (define) a disk type
 Total disk size is 17833 cylinders
 Cylinder size is 16065 (512 byte) blocks

Partition Status Type Start End Length %
=====
1 =====
 EFI

 Cylinders
 Start End Length %
 =====
 0 17833 17834 100

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Edit/View extended partitions
6. Exit (update disk configuration and exit)
7. Cancel (exit without updating disk configuration)
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
This will make all files and
programs in this partition inaccessible (type "y" or "n"). y
Enter Selection: 1
Select the partition type to create:
1=SOLARIS2 2=UNIX 3=PCIXOS 4=Other 5=DOS12
6=DOS16 7=DOSEXT 8=DOSBIG 9=DOS16LBA A=x86 Boot
B=Diagnostic C=FAT32 D=FAT32LBA E=DOSEXTLBA F=EFI (Protective)
G=EFI_SYS 0=Exit? 1
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it will be activated
```



```

each time the computer is reset or turned on.
Please type "y" or "n". y
Enter Selection: 6
Partition 1 is now the active partition.

```

SPARC ベースのシステムでは、次の手順を実行します。

```

format -e
Searching for disks...done
AVAILABLE DISK SELECTIONS:
 0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@1c,600000/scsi@2/sd@0,0
 1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@1c,600000/scsi@2/sd@1,0
 2. c2t2d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
 /pci@1c,600000/scsi@2/sd@2,0
 3. c2t3d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
 /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t0d0
[disk formatted]
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Ready to label disk, continue? yes

```

さらに、デフォルトのパーティションテーブルをチェックし、テーブルがルート  
プールのスライスに最適なものになっていること、つまり、ディスク領域の全体がス  
ライス 0 に含まれていることを確認してください。スライス 0 のサイズを増やす方法  
については、次の手順を参照してください。

```

format> partition
partition> print
Current partition table (default):
Total disk cylinders available: 14085 + 2 (reserved cylinders)

```

| Part | Tag        | Flag | Cylinders  | Size     | Blocks                |
|------|------------|------|------------|----------|-----------------------|
| 0    | root       | wm   | 1 - 13     | 129.19MB | (13/0/0) 264576       |
| 1    | swap       | wu   | 14 - 26    | 129.19MB | (13/0/0) 264576       |
| 2    | backup     | wu   | 0 - 14086  | 136.71GB | (14087/0/0) 286698624 |
| 3    | unassigned | wm   | 0          | 0        | (0/0/0) 0             |
| 4    | unassigned | wm   | 0          | 0        | (0/0/0) 0             |
| 5    | unassigned | wm   | 0          | 0        | (0/0/0) 0             |
| 6    | usr        | wm   | 27 - 14084 | 136.43GB | (14058/0/0) 286108416 |
| 7    | unassigned | wm   | 0          | 0        | (0/0/0) 0             |
| 8    | boot       | wu   | 0 - 0      | 9.94MB   | (1/0/0) 20352         |
| 9    | alternates | wm   | 0          | 0        | (0/0/0) 0             |

```

partition> modify
Select partitioning base:
 0. Current partition table (default)
 1. All Free Hog
Choose base (enter number) [0]? 1

```

| Part | Tag | Flag | Cylinders | Size | Blocks |
|------|-----|------|-----------|------|--------|
|------|-----|------|-----------|------|--------|

```

0 root wm 0 0 (0/0/0) 0
1 swap wu 0 0 (0/0/0) 0
2 backup wu 0 - 14084 136.69GB (14085/0/0) 286657920
3 unassigned wm 0 0 (0/0/0) 0
4 unassigned wm 0 0 (0/0/0) 0
5 unassigned wm 0 0 (0/0/0) 0
6 usr wm 0 0 (0/0/0) 0
7 unassigned wm 0 0 (0/0/0) 0
8 boot wu 0 - 0 9.94MB (1/0/0) 20352
9 alternates wm 0 0 (0/0/0) 0

```

```

Do you wish to continue creating a new partition
table based on above table[yes]? yes
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

```

| Part | Tag        | Flag | Cylinders | Size     | Blocks                |
|------|------------|------|-----------|----------|-----------------------|
| 0    | root       | wm   | 1 - 14084 | 136.68GB | (14084/0/0) 286637568 |
| 1    | swap       | wu   | 0         | 0        | (0/0/0) 0             |
| 2    | backup     | wu   | 0 - 14084 | 136.69GB | (14085/0/0) 286657920 |
| 3    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 4    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 5    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 6    | usr        | wm   | 0         | 0        | (0/0/0) 0             |
| 7    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 8    | boot       | wu   | 0 - 0     | 9.94MB   | (1/0/0) 20352         |
| 9    | alternates | wm   | 0         | 0        | (0/0/0) 0             |

```

Okay to make this the current partition table[yes]? yes
Enter table name (remember quotes): "c2t0d0"
Ready to label disk, continue? yes
partition> quit
format> quit

```

## ▼ ディスクラベルを検査する方法

ディスクラベル情報の検査には、`prtvtoc` コマンドを使用します。ディスクラベルと `prtvtoc` コマンドの出力結果の詳しい説明については、[第7章「ディスクの管理\(概要\)」](#)を参照してください。

- 1 スーパーユーザーになります。
- 2 ディスクラベル情報を表示します。

```
prtvtoc /dev/rdsk/device-name
```

`device-name` には、検査する raw ディスクデバイスを指定してください。

## 例9-7 ディスクラベルを検査する

次の例は、VTOC ラベル付きディスクのディスクラベル情報を示しています。

```
prtvtoc /dev/rdisk/c2t3d0s0
* /dev/rdisk/c2t3d0s0 partition map
*
* Dimensions:
* 512 bytes/sector
* 848 sectors/track
* 24 tracks/cylinder
* 20352 sectors/cylinder
* 14089 cylinders
* 14087 accessible cylinders
*
* Flags:
* 1: unmountable
* 10: read-only
*
*
* Partition Tag Flags First Sector Last
* 0 2 00 0 286698624 286698623
* 2 5 01 0 286698624 286698623
```

次の例は、EFI ラベル付き非ルートプールディスクのディスクラベル情報を示しています。

```
prtvtoc /dev/dsk/c8t3d0
* /dev/dsk/c8t3d0 partition map
*
* Dimensions:
* 512 bytes/sector
* 143374738 sectors
* 143374671 accessible sectors
*
* Flags:
* 1: unmountable
* 10: read-only
*
* Unallocated space:
* First Sector Last
* Sector Count Sector
* 34 222 255
*
*
* Partition Tag Flags First Sector Last
* 0 4 00 256 143358065 143358320
* 8 11 00 143358321 16384 143374704
```

## 破損したディスクラベルの復元

電源障害やシステム障害が原因で、ディスクが認識されなくなることがあります。ただし、ディスクラベルが破損しても、スライス情報やディスクのデータを作り直したり、復元したりする必要がない場合もあります。

破損したディスクラベルを復元する作業の最初の手順は、正しいジオメトリとディスクタイプ情報を使用してディスクにラベルを付けることです。この作業は、通常のディスクラベル作成方法 (自動構成またはディスクタイプの手動指定) で実行できます。

`format` ユーティリティーでディスクタイプが認識されたら、次の手順はバックアップラベルを検索してディスクにラベルを付けることです。バックアップラベルを使用してディスクにラベルを付けると、ディスクタイプとジオメトリだけでなく、正しいパーティション情報を使用してディスクラベルが作成されます。

### ▼ 破損したディスクラベルを復元する方法

- 1 システムをシングルユーザーモードにブートします。

必要であれば、シングルユーザーモードでローカルの Oracle Solaris DVD またはネットワークからシステムをブートして、ディスクにアクセスします。

システムのブートについては、『[Oracle Solaris の管理: 基本管理](#)』を参照してください。

- 2 ディスクにラベルを付け直します。

**# format**

`format` ユーティリティーは、ラベルが付いていない SCSI ディスクを自動構成しようとしています。破損し、ラベルが付いていないディスクを自動構成できる場合は、次のメッセージが表示されます。

`cwtxdy: configured with capacity of abcMB`

次に、システム上のディスクの番号付きリストが表示されます。

- 3 復元するディスクの番号を入力します。

`Specify disk (enter its number): 1`

#### 4 次のいずれかを選択して、ディスクラベルの作成方法を決定します。

- ディスクが正常に構成された場合は、次を実行します。

##### a. バックアップラベルを検索します。

```
format> verify
Warning: Could not read primary label.
Warning: Check the current partitioning and 'label' the disk or
use the 'backup' command.
Backup label contents:
Volume name = < >
ascii name = <FUJITSU-MAP3147N SUN146G-0501>
pcyl = 14089
ncyl = 14087
acyl = 2
nhead = 24
nsect = 848
```

| Part | Tag        | Flag | Cylinders | Size     | Blocks                |
|------|------------|------|-----------|----------|-----------------------|
| 0    | root       | wm   | 0 - 14086 | 136.71GB | (14087/0/0) 286698624 |
| 1    | swap       | wu   | 0         | 0        | (0/0/0) 0             |
| 2    | backup     | wu   | 0 - 14086 | 136.71GB | (14087/0/0) 286698624 |
| 3    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 4    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 5    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |
| 6    | usr        | wm   | 0         | 0        | (0/0/0) 0             |
| 7    | unassigned | wm   | 0         | 0        | (0/0/0) 0             |

- ##### b. **format** ユーティリティーがバックアップラベルを検出することができ、その内容が適切に見える場合は、ディスクにバックアップラベルを付けます。

```
format> backup
Disk has a primary label, still continue? y

Searching for backup labels...found.
Restoring primary label

これで、ディスクラベルが復元されました
```

- ##### c. **format** ユーティリティーを終了します。

```
format> q
```

- ディスクが正常に構成されなかった場合は、次を実行します。

##### a. **type** コマンドを使用してディスクタイプを指定します。

```
format> type

Available Drives Type メニューが表示されます。
```

- ##### b. ディスクを自動構成するには、**0**を選択します。または、ディスクタイプの候補のリストからディスクタイプを選択します。

```
Specify disk type (enter its number)[12]: 12
```

- c. ディスクが正常に自動構成された場合は、ディスクラベルを作成するかどうか尋ねるプロンプトが表示されたときに **no** と応答します。

```
Disk not labeled. Label it now? no
```

- d. バックアップラベルを検索します。

```
format> verify
Warning: Could not read primary label.
Warning: Check the current partitioning and 'label' the disk
or use the 'backup' command.
.
.
.
```

- e. **format** ユーティリティーがバックアップラベルを検出することができ、その内容が適切に見える場合は、ディスクにバックアップラベルを付けます。

```
format> backup
Disk has a primary label, still continue? y
Searching for backup labels...found.
Restoring primary label
```

これで、ディスクラベルが復元されました。

- f. **format** ユーティリティーを終了します。

```
format> q
```

- 5 復元されたディスクのファイルシステムを検証します。

ZFS ファイルシステムで `zpool scrub` コマンドを使用する方法については、[zpool\(1M\)](#) を参照してください。

## 他社製のディスクの追加

Oracle Solaris では多くのサードパーティーディスクをサポートしています。ただし、デバイスドライバを提供しないとディスクが認識されない場合もあります。

ディスク追加時のその他のオプションを次に示します。

- SCSI ディスクを追加する場合、**format** ユーティリティーの自動構成機能の使用を試みることができます。
- PCI、SCSI、または USB ディスクのホットプラグを試みることもできます。詳細は、[第3章「デバイスの管理\(タスク\)」](#)を参照してください。

---

注 - Oracle の `format` ユーティリティーが他社製のどのディスクドライバでも正常に機能するとは限りません。ディスクドライバに `format` ユーティリティーとの互換性がない場合は、ディスクドライブのベンダーが独自のディスクフォーマットプログラムを提供しているはずです。

---

一般に、`format` ユーティリティーを起動し、ディスクタイプが認識されないなどという場合に、不足しているソフトウェアサポートがあることがわかります。

第10章「SPARC: ディスクの設定 (手順)」または243 ページの「x86: ZFS ファイルシステム用のディスクの設定 (タスクマップ)」で、システムディスクやセカンダリディスクを追加するための適切な構成手順を参照してください。





## SPARC: ディスクの設定 (手順)

---

この章では、SPARC システムでのディスクの設定方法について説明します。

SPARC システムでのディスクの設定に関する手順については、次のセクションを参照してください。

- 217 ページの「SPARC: UFS ファイルシステム用のディスクの設定 (タスクマップ)」
- 226 ページの「SPARC: ZFS ファイルシステム用のディスクの設定 (タスクマップ)」

ディスク管理の概要については、第 7 章「ディスクの管理 (概要)」を参照してください。x86 ベースシステムでディスクを設定する手順については、第 11 章「x86: ディスクの設定 (手順)」を参照してください。

### SPARC: UFS ファイルシステム用のディスクの設定 (タスクマップ)

次のタスクマップは、SPARC ベースシステム上で UFS ルートファイルシステム用のディスクまたは UFS ファイルシステム用のディスクを設定するための手順を記述したものです。

| タスク            | 説明                                                                                  | 参照先                                           |
|----------------|-------------------------------------------------------------------------------------|-----------------------------------------------|
| 1. ディスクを接続します。 | UFS ルートファイルシステム用のディスク<br><br>新しいディスクを接続して、ローカルまたはリモートの Oracle Solaris DVD からブートします。 | 219 ページの「SPARC: UFS ルートファイルシステム用のディスクを設定する方法」 |

| タスク                                      | 説明                                                                                                                      | 参照先                                                 |
|------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------|
|                                          | UFS ファイルシステム用のディスク<br>ディスクを接続します。                                                                                       | 219 ページの「SPARC: UFS ファイルシステム用のディスクを設定する方法」          |
| 2. UFS ファイルシステム用のスライスを作成し、ディスクにラベルを付けます。 | ディスクスライスを作成し、ディスクにラベルを付けます。                                                                                             | 220 ページの「SPARC: UFS ファイルシステム用のディスクスライスを作成する方法」      |
| 3. UFS ファイルシステムを作成します。                   | <code>newfs</code> コマンドを使用してディスクスライス上に UFS ファイルシステムを作成します。ルート (/) または <code>/usr</code> ファイルシステム、あるいはその両方を作成し直す必要があります。 | 225 ページの「SPARC: UFS ファイルシステムを作成する方法」                |
| 4. UFS ファイルシステムを復元します。                   | UFS のルート (/) または <code>/usr</code> ファイルシステム、あるいはその両方を復元します。必要に応じて、ほかのディスク上でファイルシステムを復元します。                              | 404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する (タスクマップ)」  |
| 5. UFS ルートファイルシステムのブートブロックをインストールします。    | システムをブートできるように、ルート (/) ファイルシステムにブートブロックをインストールします。                                                                      | 226 ページの「SPARC: UFS ルートファイルシステムのブートブロックをインストールする方法」 |

## SPARC: UFS ファイルシステム用のディスクの設定

ディスクには、UFS ルート (/) および `/usr` ファイルシステムを含めることも、ユーザーまたはデータファイルシステムを含めることも、あるいはそれらのファイルシステムの組み合わせを含めることもできます。UFS ルート (/) または `/usr` ファイルシステムを含むディスクを復元するには、次の 2 つの方法があります。

- Oracle Solaris OS 全体を再インストールします。
- ディスクを交換し、Oracle Solaris OS のファイルシステムをバックアップメディアから復元します。

ユーザーまたはデータファイルシステムを含むディスクの交換やそのファイルシステムの復元は必要に応じて、システムの再インストールやリブートを行わなくても実行することができます。

## ▼ SPARC: UFS ルートファイルシステム用のディスクを設定する方法

次の手順は、UFS ルートまたは `/usr` ファイルシステムが格納されていたディスクを交換する場合に使用できます。ディスクの交換方法については、ハードウェアの取り付けガイドを参照してください。

- 1 必要に応じて、損傷したシステムディスクをシステムから外します。
- 2 交換用のディスクをシステムに正しく接続します。
- 3 ローカルの **Oracle Solaris DVD** またはリモートの **Oracle Solaris DVD** (ネットワーク経由) のどちらでブートするかに応じて、次の表の手順で操作します。

| ブートの種類                       | アクション                                                                                                                 |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| ローカルドライブの Oracle Solaris DVD | <p>1. ドライブに Oracle Solaris DVD が入っていることを確認します。</p> <p>2. メディアからシングルユーザーモードでブートします。</p> <p><b>ok boot cdrom -s</b></p> |
| ネットワーク経由                     | <p>ネットワークからシングルユーザーモードでブートします。</p> <p><b>ok boot net -s</b></p>                                                       |

数分後に、スーパーユーザーのプロンプト (`#`) が表示されます。

参考 UFS ルートファイルシステム用のディスクを設定したあとの作業 ...

ディスク上で、UFS ルートおよび `/usr` ファイルシステム用のスライスとディスクラベルを作成します。[220 ページの「SPARC: UFS ファイルシステム用のディスクスライスを作成する方法」](#)を参照してください。

## ▼ SPARC: UFS ファイルシステム用のディスクを設定する方法

EFI ディスクラベル付きのディスクを追加する場合は、[168 ページの「EFI \(GPT\) ディスクラベル」](#)で詳細を参照してください。

一般に、最近のバスタイプのほとんどで、ホットプラグ機能がサポートされています。これは、ユーザーが空のスロットにディスクを挿入すれば、システムがその

ディスクを認識してくれることを意味します。ホットプラグ対応デバイスの詳細は、[第4章「デバイスの動的構成\(タスク\)」](#)を参照してください。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 (省略可能)ディスクタイプが **Solaris** ソフトウェアでサポートされていない場合は、ハードウェアに添付された説明書に従って、そのディスクのデバイスドライバを追加します。
- 3 ディスクをシステムに接続して、ディスクの物理的接続を確認します。  
詳細は、ディスクのハードウェアインストールガイドを参照してください。
- 4 新しいディスクが認識されていることを、**format** ユーティリティーで確認します。

参考 UFS ファイルシステム用のディスクを接続したあとの作業 ...

ディスクを接続し終わったら、そのディスク上でスライスとディスクラベルを作成します。[220 ページの「SPARC: UFS ファイルシステム用のディスクスライスを作成する方法」](#)を参照してください。

## ▼ **SPARC: UFS** ファイルシステム用のディスクスライスを作成する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 **format** ユーティリティーを起動します。  
**# format**  
利用可能なディスクの番号付きリストが表示されます。
- 3 パーティションの再分割を実行するディスクの番号を選択します。  
Specify disk (enter its number): *disk-number*  
*disk-number* は、パーティションの再分割を実行するディスクの番号です。
- 4 **partition** メニューを選択します。  
**format> partition**
- 5 現在のパーティション(スライス)テーブルを表示します。  
**partition> print**
- 6 変更作業を開始します。  
**partition> modify**

- 7 ディスクをすべて **free hog** に設定します。

Choose base (enter number) [0]? 1

**free hog** スライスの詳細は、179 ページの「**free hog** スライスの使用方法」を参照してください。

- 8 続行するかどうかを尋ねるプロンプトが表示されたら、**y** と応答して新しいパーティションテーブルを作成します。

Do you wish to continue creating a new partition table based on above table[yes]? y

- 9 プロンプトが表示されたら、**free hog** パーティション (スライス) と各スライスのサイズを指定します。

システムディスクを追加するときは、次のスライスを設定しなければなりません。

- ルート (スライス 0) およびスワップ (スライス 1) (必須)
- /usr (スライス 6)

スライスの情報を設定すると、新しいパーティションテーブルが表示されます。

ディスクスライスの作成例については、例 10-1 を参照してください。

- 10 プロンプトが表示されたら **y** と応答して、表示されたパーティションテーブルを現在のパーティションテーブルにします。

Okay to make this the current partition table[yes]? y

現在のパーティションテーブルが希望どおりでないために変更する場合は、**no** と応答して手順 6 に戻ります。

- 11 パーティションテーブルに名前を付けます。

Enter table name (remember quotes): "*partition-name*"

*partition-name* には、新しいパーティションテーブルの名前を指定します。

- 12 新しいディスク上でスライスの割り当てが終わったら、新しいパーティションテーブルを使用してディスクにラベルを付けます。

Ready to label disk, continue? **yes**

- 13 **partition** メニューを終了します。

partition> **q**

- 14 ディスクラベルを検査します。

format> **verify**

- 15 **format** ユーティリティーを終了します。

format> **q**

## 例 10-1 SPARC: UFS ルートファイルシステム用のディスクスライスの作成

次の例では、format ユーティリティーを使用して 18G バイトのディスクを 3 つのスライスに分割します。各スライスをルート (/) ファイルシステム、スワップ領域、/usr ファイルシステムに割り当てます。

```
format
AVAILABLE DISK SELECTIONS:
 0. /dev/rdisk/clt0d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@0,0
 1. /dev/rdisk/clt1d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@1,0
 2. /dev/rdisk/clt8d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@8,0
 3. /dev/rdisk/clt9d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@9,0

Specify disk (enter its number): 0
selecting clt0d0
[disk formatted]
format> partition
partition> print
partition> modify
Select partitioning base:
 0. Current partition table (original)
 1. All Free Hog

Part Tag Flag Cylinders Size Blocks
 0 root wm 0 0 (0/0/0) 0
 1 swap wu 0 0 (0/0/0) 0
 2 backup wu 0 - 7505 16.86GB (7506/0/0) 35368272
 3 unassigned wm 0 0 (0/0/0) 0
 4 unassigned wm 0 0 (0/0/0) 0
 5 unassigned wm 0 0 (0/0/0) 0
 6 usr wm 0 0 (0/0/0) 0
 7 unassigned wm 0 0 (0/0/0) 0

Choose base (enter number) [0]? 1
table based on above table[yes]? yes
Free Hog partition[6]? 6
Enter size of partition '0' [0b, 0c, 0.00mb, 0.00gb]: 4gb
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]: 4gb
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

Part Tag Flag Cylinders Size Blocks
 0 root wm 0 - 1780 4.00GB (1781/0/0) 8392072
 1 swap wu 1781 - 3561 4.00GB (1781/0/0) 8392072
 2 backup wu 0 - 7505 16.86GB (7506/0/0) 35368272
 3 unassigned wm 0 0 (0/0/0) 0
 4 unassigned wm 0 0 (0/0/0) 0
 5 unassigned wm 0 0 (0/0/0) 0
 6 usr wm 3562 - 7505 8.86GB (3944/0/0) 18584128
 7 unassigned wm 0 0 (0/0/0) 0

Okay to make this the current partition table[yes]? yes
Enter table name (remember quotes): "disk0"
Ready to label disk, continue? yes
```

```
partition> quit
format> verify
format> quit
```

## 例 10-2 SPARC: UFS ファイルシステム用のディスクスライスの作成

次の例では、format ユーティリティーを使用して 18G バイトのディスクを 1 つのスライスに分割し、それを /export/home ファイルシステムに割り当てます。

```
format /dev/rdisk/cl*
AVAILABLE DISK SELECTIONS:
 0. /dev/rdisk/clt0d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@0,0
 1. /dev/rdisk/clt1d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@1,0
 2. /dev/rdisk/clt8d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@8,0
 3. /dev/rdisk/clt9d0s0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
 /sbus@2,0/QLGC,isp@2,10000/sd@9,0
Specify disk (enter its number): 1
selecting clt1d0
[disk formatted]
format> partition
partition> print
partition> modify
Select partitioning base:
 0. Current partition table (original)
 1. All Free Hog
Choose base (enter number) [0]? 1
Part Tag Flag Cylinders Size Blocks
 0 root wm 0 0 (0/0/0) 0
 1 swap wu 0 0 (0/0/0) 0
 2 backup wu 0 - 7505 16.86GB (7506/0/0) 35368272
 3 unassigned wm 0 0 (0/0/0) 0
 4 unassigned wm 0 0 (0/0/0) 0
 5 unassigned wm 0 0 (0/0/0) 0
 6 usr wm 0 0 (0/0/0) 0
 7 unassigned wm 0 0 (0/0/0) 0

Do you wish to continue creating a new partition
table based on above table[yes]? y
Free Hog partition[6]? 7
Enter size of partition '0' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Part Tag Flag Cylinders Size Blocks
 0 root wm 0 0 (0/0/0) 0
 1 swap wu 0 0 (0/0/0) 0
 2 backup wu 0 - 7505 16.86GB (7506/0/0) 35368272
 3 unassigned wm 0 0 (0/0/0) 0
 4 unassigned wm 0 0 (0/0/0) 0
 5 unassigned wm 0 0 (0/0/0) 0
 6 usr wm 0 0 (0/0/0) 0
 7 unassigned wm 0 - 7505 16.86GB (7506/0/0) 35368272
```

```

Okay to make this the current partition table[yes]? yes
Enter table name (remember quotes): "home"
Ready to label disk, continue? y
partition> q
format> verify
format> q
#

```

次の例では、format ユーティリティーを使って、1.15T バイトの EFI ラベル付きディスクを 3 つのスライスに分割します。

```

format
.
.
.
partition> modify
Select partitioning base:
 0. Current partition table (original)
 1. All Free Hog
Choose base (enter number) [0]? 1

```

| Part | Tag        | Flag | First Sector | Size   | Last Sector |
|------|------------|------|--------------|--------|-------------|
| 0    | root       | wm   | 0            | 0      | 0           |
| 1    | usr        | wm   | 0            | 0      | 0           |
| 2    | unassigned | wm   | 0            | 0      | 0           |
| 3    | unassigned | wm   | 0            | 0      | 0           |
| 4    | unassigned | wm   | 0            | 0      | 0           |
| 5    | unassigned | wm   | 0            | 0      | 0           |
| 6    | usr        | wm   | 0            | 0      | 0           |
| 8    | reserved   | wm   | 2576924638   | 8.00MB | 2576941021  |

```

Do you wish to continue creating a new partition
table based on above table[yes]? y
Free Hog partition[6]? 4
Enter size of partition 0 [0b, 34e, 0mb, 0gb, 0tb]:
Enter size of partition 1 [0b, 34e, 0mb, 0gb, 0tb]:
Enter size of partition 2 [0b, 34e, 0mb, 0gb, 0tb]: 400gb
Enter size of partition 3 [0b, 838860834e, 0mb, 0gb, 0tb]: 400gb
Enter size of partition 5 [0b, 1677721634e, 0mb, 0gb, 0tb]:
Enter size of partition 6 [0b, 1677721634e, 0mb, 0gb, 0tb]:

```

| Part | Tag        | Flag | First Sector | Size     | Last Sector |
|------|------------|------|--------------|----------|-------------|
| 0    | unassigned | wm   | 0            | 0        | 0           |
| 1    | unassigned | wm   | 0            | 0        | 0           |
| 2    | usr        | wm   | 34           | 400.00GB | 838860833   |
| 3    | usr        | wm   | 838860834    | 400.00GB | 1677721633  |
| 4    | usr        | wm   | 1677721634   | 428.77GB | 2576924637  |
| 5    | unassigned | wm   | 0            | 0        | 0           |
| 6    | unassigned | wm   | 0            | 0        | 0           |
| 8    | reserved   | wm   | 2576924638   | 8.00MB   | 2576941021  |

```

Ready to label disk, continue? yes

partition> q

```

**参考** UFS ファイルシステム用のディスクスライスとディスクラベルを作成したあとの作業...

ディスクスライスとディスクラベルを作成し終わったら、ディスク上にファイルシステムを作成できます。[225 ページの「SPARC: UFS ファイルシステムを作成する方法」](#)を参照してください。



## ▼ SPARC: UFS ファイルシステムを作成する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。

- 2 スライスごとにファイルシステムを作成します。

```
newfs /dev/rdisk/cwtxdyisz
```

/dev/rdisk/cwtxdyisz は、作成するファイルシステムの raw デバイスです。

newfs コマンドの詳細は、[第 15 章「ファイルシステムの作成およびマウント \(タスク\)」](#) または [newfs\(1M\)](#) を参照してください。

- 3 新しいファイルシステムをマウントして、確認します。

```
mount /dev/dsk/cwtxdyisz /mnt
ls
lost+found
```

### 参考 UFS ファイルシステムを作成したあとの作業

- UFS ルートファイルシステム用のディスク – ディスク上でルート (/) および /usr ファイルシステムを復元する必要があります。
- [UFS ファイルおよびファイルシステムのバックアップを復元する \(タスクマップ\)](#)に進みます。
- ルート (/) と /usr ファイルシステムの復元後、ブートブロックをインストールします。[226 ページの「SPARC: UFS ルートファイルシステムのブートブロックをインストールする方法」](#)を参照してください。
- UFS ファイルシステム用のディスク – 新しいディスク上でファイルシステムを復元しなければいけない可能性があります。[404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する \(タスクマップ\)」](#)に進みます。新しいディスク上にファイルシステムを復元しない場合は、セカンダリディスクを追加する作業が終わったことになります。
- ユーザーが UFS ファイルシステムを利用できるようにする方法については、[315 ページの「Oracle Solaris ファイルシステムのマウントおよびマウント解除」](#)を参照してください。

▼ **SPARC: UFS ルートファイルシステムのブートブ  
ロックをインストールする方法**

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 UFS ルートファイルシステムのブートブロックをインストールします。  

```
installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk
/dev/rdisk/cwtxdys0
```

詳細は、[installboot\(1M\)](#)のマニュアルページを参照してください。
- 3 システムをリブートし、レベル3で実行することによって、ブートブロックがイン  
ストールされていることを確認します。  

```
init 6
```

例 10-3 SPARC: UFS ルートファイルシステムのブートブロックのインストール

次の例は、UFS ルートファイルシステムにブートブロックをインストールする方法  
を示しています。

```
installboot /usr/platform/sun4u/lib/fs/ufs/bootblk
/dev/rdisk/c0t0d0s0
```

**SPARC: ZFS ファイルシステム用のディスクの設定(タスク  
マップ)**

次のタスクマップは、SPARC ベースシステム上で ZFS ルートファイルシステム用の  
ZFS ルートプールディスクまたはルート以外の ZFS プールディスクを設定するた  
めの手順を記述したものです。

| タスク                                 | 説明                                                                                                                        | 参照先                                                  |
|-------------------------------------|---------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------|
| 1. ZFS ルートファイルシステム<br>用のディスクを設定します。 | ZFS ルートファイルシステム用<br>のディスク<br><br>新しいディスクを接続するか既<br>存のルートプールディスクを交<br>換し、ローカルまたはリモート<br>の Oracle Solaris DVD から<br>ブートします。 | 228 ページの「SPARC: ZFS<br>ルートファイルシステム用の<br>ディスクを設定する方法」 |

| タスク                                          | 説明                                                                                                                   | 参照先                                                                 |
|----------------------------------------------|----------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|
| 2. ZFS ルートファイルシステム用のディスクスライスを作成します。          | ZFS ルートプールで使用するためのディスクのディスクスライスを作成します。これは、長年にわたるブートの制限です。                                                            | <a href="#">229 ページの「SPARC: ZFS ルートファイルシステム用のディスクスライスを作成する方法」</a>   |
| 3. 必要に応じて、ZFS ルートファイルシステムのブートブロックをインストールします。 | ルートプールで使用するためのディスクを <code>zpool replace</code> コマンドを使用して交換する場合は、交換後のディスクからシステムをブートできるように、ブートブロックを手動でインストールする必要があります。 | <a href="#">233 ページの「SPARC: ZFS ルートファイルシステムのブートブロックをインストールする方法」</a> |
| 4. ZFS ファイルシステム用のディスクを設定します。                 | ZFS ファイルシステム用のディスク<br><br>ZFS ファイルシステム用のディスクを設定します。                                                                  | <a href="#">234 ページの「SPARC: ZFS ファイルシステム用のディスクを設定する方法」</a>          |

## SPARC: ZFS ファイルシステム用のディスクの設定

ZFS ファイルシステムでも、記述された手順に従ってディスクを設定できますが、ディスクやディスクスライスへの直接のマッピングは行われません。ZFS ファイルシステムを作成する前に、ZFS ストレージプールを作成する必要があります。詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』を参照してください。

ルートプールには、Oracle Solaris OS のブートに使用されるルートファイルシステムが含まれています。ミラー化されていないルートプールのディスクが破損すると、システムがブートしない可能性があります。ルートプールディスクが破損した場合の復旧方法としては、次の2つの方法が考えられます。

- Oracle Solaris OS 全体を再インストールします。
- ルートプールディスクを交換し、スナップショットまたはバックアップメディアからファイルシステムを復元します。ハードウェア障害によるシステムの停止時間を減らすには、冗長ルートプールを作成します。サポートされる冗長ルートプール構成は、ミラー化ルートプールだけです。

ルート以外のプールで使用されているディスクには通常、ユーザーファイルまたはデータファイル用の領域が含まれています。ルートプールやルート以外のプールに別のディスクを接続または追加すれば、ディスク容量を増やすことができます。あるいは、プール内の破損したディスクを次の方法で交換することもできます。

- 非冗長プール内のディスクは、すべてのデバイスが現在 ONLINE 状態になっている場合は交換可能です。

- 冗長プール内のディスクは、ほかのデバイスで十分な冗長性を確保できる場合は交換可能です。
- ミラー化ルートプールでは、ディスクを交換したりディスクを接続したりしたあと、故障したディスクやサイズの小さいディスクを切り離してプールのサイズを増やすことができます。

一般に、システム上でのディスクの設定方法はハードウェアごとに異なるため、システム上でディスクの追加や交換を行う際にはハードウェアのドキュメントを確認してください。ディスクを既存のコントローラに追加する必要がある場合、システムがホットプラグをサポートしていれば、空のスロットにディスクを挿入するだけで済む可能性があります。新しいコントローラを構成する必要がある場合は、[74 ページの「動的再構成とホットプラグ機能」](#)を参照してください。

## ▼ SPARC: ZFS ルートファイルシステム用のディスクを設定する方法

ディスクの交換方法については、ハードウェアの取り付けガイドを参照してください。

- 1 必要に応じて、損傷したディスクをシステムから外します。
- 2 交換用のディスクをシステムに接続し、必要に応じてディスクの物理的接続を確認します。
- 3 ローカルの **Oracle Solaris DVD** またはリモートの **Oracle Solaris DVD** (ネットワーク経由) のどちらでブートするかに応じて、次の表の手順で操作します。

| ブートの種類                       | アクション                                                                                                                             |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| ローカルドライブの Oracle Solaris DVD | <div>1. ドライブに Oracle Solaris DVD が入っていることを確認します。</div> <div>2. メディアからシングルユーザーモードでブートします。</div> <div><b>ok boot cdrom -s</b></div> |
| ネットワーク経由                     | <div>ネットワークからシングルユーザーモードでブートします。</div> <div><b>ok boot net -s</b></div>                                                           |

数分後に、スーパーユーザーのプロンプト (#) が表示されます。

参考 ZFS ルートファイルシステム用のディスクを設定したあとの作業 ...

ディスクの接続や交換が完了したら、スライスを作成してディスクラベルを更新することができます。229 ページの「[SPARC: ZFS ルートファイルシステム用のディスクスライスを作成する方法](#)」を参照してください。

## SPARC: ZFS ルートファイルシステム用のディスクスライスの作成

ZFS ルートプールで使用するためのディスクでは、ディスクスライスを作成する必要があります。これは、長年にわたるブートの制限です。ルートプールディスクの次の要件を確認してください。

- ディスクスライスと SMI (VTOC) ラベルが含まれている必要があります。
- ルートプールディスクでは、EFI ラベルはサポートされません。
- 単一のディスクであるか、あるいはミラー化構成の一部になっている必要があります。ルートプールでは、非冗長構成や RAIDZ 構成はサポートされません。
- ルートファイルシステムの、OS イメージの一部となっているサブディレクトリのうち、`/var` 以外のものはすべて、ルートファイルシステムと同じデータセット内に存在している必要があります。
- スワップデバイスとダンプデバイス以外の Solaris OS コンポーネントはすべて、ルートプール内に存在している必要があります。
- さらに、`/var` ディレクトリまたはデータセットは単一のデータセットでなければならない、という制限もあります。たとえば、Oracle Solaris Live Upgrade を使って ZFS BE の移行やパッチ適用を行ったり、このプールの ZFS フラッシュアーカイブを作成したりする必要もある場合には、`/var/tmp` のような `/var` の下位ファイルシステムを作成することはできません。

一般に、スライス 0 で、ディスク領域の全体を使ってディスクスライスを作成します。ディスク上の別のスライスを使用し、そのディスクを各種オペレーティングシステム間で共有したり、別の ZFS ストレージプールやストレージプールコンポーネントと共有したりしようとすることは、お勧めできません。

### ▼ SPARC: ZFS ルートファイルシステム用のディスクスライスを作成する方法

一般に、ルートプールディスクのインストールは、システムのインストール時に自動的に行われます。ルートプールディスクを交換したり、新しいディスクをミラー化ルートプールのディスクとして接続したりする必要がある場合には、次の手順を参照してください。

- 1 スーパーユーザーになります。
- 2 必要に応じて、故障したディスクをオフラインにして構成解除します。  
一部のハードウェアでは、故障したディスクを交換するための `zpool replace` 操作を試みる前に、ディスクをオフラインにして構成解除する必要があります。例:

```
zpool offline rpool c2t1d0s0
cfgadm -c unconfigure c2::disk/c2t1d0
```

- 3 必要に応じて、新しいディスクや交換用ディスクをシステムに物理的に接続します。

a. 故障したディスクを物理的に取り外します。

b. 交換用ディスクを物理的に挿入します。

c. 必要に応じて、交換用ディスクを構成します。例:

```
cfgadm -c configure c2::disk/c2t1d0
```

一部のハードウェアでは、交換用ディスクの装着後にそのディスクを再構成する必要があります。

- 4 `format` の出力を確認してディスクにアクセスできることを確認します。

たとえば、次の `format` コマンドでは、このシステムに4つのディスクが接続されていることがわかります。

```
format -e
AVAILABLE DISK SELECTIONS:
 0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
 /pci@1c,600000/scsi@2/sd@0,0
 1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
 /pci@1c,600000/scsi@2/sd@1,0
 2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
 /pci@1c,600000/scsi@2/sd@2,0
 3. c2t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
 /pci@1c,600000/scsi@2/sd@3,0
```

- 5 ZFS ルートプールに使用されるディスクを選択します。
- 6 パーティション(スライス)情報を出力することで、ディスクに **SMI** ラベルが含まれていることを確認します。

たとえば、`c2t1d0` のパーティション(スライス)出力から、このディスクに EFI ラベルが含まれていることがわかります。なぜなら、先頭と末尾のセクターが認識されているからです。

```
Specify disk (enter its number): 1
selecting c2t1d0
[disk formatted]
format> p
PARTITION MENU:
```

```

0 - change '0' partition
1 - change '1' partition
2 - change '2' partition
3 - change '3' partition
4 - change '4' partition
5 - change '5' partition
6 - change '6' partition
expand - expand label to use whole disk
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
!<cmd> - execute <cmd>, then return
quit
partition> p
Current partition table (original):
Total disk sectors available: 71116508 + 16384 (reserved sectors)

Part Tag Flag First Sector Size Last Sector
0 usr wm 256 33.91GB 71116541
1 unassigned wm 0 0 0
2 unassigned wm 0 0 0
3 unassigned wm 0 0 0
4 unassigned wm 0 0 0
5 unassigned wm 0 0 0
6 unassigned wm 0 0 0
8 reserved wm 71116542 8.00MB 71132925

partition>

```

- 7 ディスクにEFIラベルが含まれている場合には、ディスクにSMIラベルを付け直します。

次の例では、c2t1d0 ディスクにSMIラベルが付け直されていますが、デフォルトのパーティションテーブルは最適なスライス構成になっていません。

```

partition> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Auto configuration via format.dat[no]?
Auto configuration via generic SCSI-2[no]?
partition> p
Current partition table (default):
Total disk cylinders available: 24620 + 2 (reserved cylinders)

Part Tag Flag Cylinders Size Blocks
0 root wm 0 - 90 128.37MB (91/0/0) 262899
1 swap wu 91 - 181 128.37MB (91/0/0) 262899
2 backup wu 0 - 24619 33.92GB (24620/0/0) 71127180
3 unassigned wm 0 0 (0/0/0) 0
4 unassigned wm 0 0 (0/0/0) 0
5 unassigned wm 0 0 (0/0/0) 0
6 usr wm 182 - 24619 33.67GB (24438/0/0) 70601382
7 unassigned wm 0 0 (0/0/0) 0

partition>

```

**8 ZFS ルートプールディスクに最適なスライス構成を作成します。**

未割り当てのディスク領域がすべてスライス 0 に集められるように、free hog パーティションを設定します。次に、すべてのスライスサイズフィールドで Return キーを押すことで、1 つの大きなスライス 0 を作成します。

```
partition> modify
Select partitioning base:
 0. Current partition table (default)
 1. All Free Hog
Choose base (enter number) [0]? 1
```

| Part | Tag        | Flag | Cylinders | Size    | Blocks               |
|------|------------|------|-----------|---------|----------------------|
| 0    | root       | wm   | 0         | 0       | (0/0/0) 0            |
| 1    | swap       | wu   | 0         | 0       | (0/0/0) 0            |
| 2    | backup     | wu   | 0 - 24619 | 33.92GB | (24620/0/0) 71127180 |
| 3    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |
| 4    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |
| 5    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |
| 6    | usr        | wm   | 0         | 0       | (0/0/0) 0            |
| 7    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |

```
Do you wish to continue creating a new partition
table based on above table[yes]?
```

```
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:
```

| Part | Tag        | Flag | Cylinders | Size    | Blocks               |
|------|------------|------|-----------|---------|----------------------|
| 0    | root       | wm   | 0 - 24619 | 33.92GB | (24620/0/0) 71127180 |
| 1    | swap       | wu   | 0         | 0       | (0/0/0) 0            |
| 2    | backup     | wu   | 0 - 24619 | 33.92GB | (24620/0/0) 71127180 |
| 3    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |
| 4    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |
| 5    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |
| 6    | usr        | wm   | 0         | 0       | (0/0/0) 0            |
| 7    | unassigned | wm   | 0         | 0       | (0/0/0) 0            |

```
Okay to make this the current partition table[yes]?
Enter table name (remember quotes): "c2t1d0"
```

```
Ready to label disk, continue? yes
partition> quit
format> quit
```

**9 故障したディスクが交換されたことを ZFS に知らせます。**

```
zpool replace rpool c2t1d0s0
zpool online rpool c2t1d0s0
```

一部のハードウェアでは、交換用ディスクの装着後にそのディスクをオンラインにする必要がありません。



新しいディスクを接続してミラー化ルートプールを作成したり、サイズの大きなディスクを接続してサイズの小さいディスクと交換したりする場合には、次のような構文を使用します。

```
zpool attach rpool c0t0d0s0 c1t0d0s0
```

- 10 ルートプールディスクを新しいディスクと交換した場合には、その新しいディスクまたは交換用ディスクの再同期化が完了したあとで、ブートブロックを適用します。

例:

```
zpool status rpool
installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdisk/c2t1d0s0
```

- 11 新しいディスクからブートできることを確認します。
- 12 新しいディスクからシステムがブートした場合は、古いディスクを切り離します。この手順が必要となるのは、故障したディスクやサイズの小さいディスクを交換するために新しいディスクを接続した場合だけです。

```
zpool detach rpool c0t0d0s0
```

- 13 システムが新しいディスクから自動的にブートするように設定します。そのためには、`eeeprom` コマンドまたは SPARC ブート PROM の `setenv` コマンドを使用します。

参考 ZFS ルートファイルシステム用のディスクスライスを作成したあとの作業 ...

ZFS ルートファイルシステム用のディスクスライスを作成し終わったら、ルートプールのスナップショットを復元してルートプールを回復する必要があります。『Oracle Solaris ZFS 管理ガイド』の「ZFS ルートプールまたはルートプールのスナップショットを回復する」を参照してください。

## ▼ SPARC: ZFS ルートファイルシステムのブートブロックをインストールする方法

- 1 スーパーユーザーになります。
- 2 ZFS ルートファイルシステムのブートブロックをインストールします。

```
installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdisk/cwtxdys0
```

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

- 3 システムをリブートし、レベル3で実行することによって、ブートブロックがインストールされていることを確認します。

```
init 6
```

#### 例 10-4 SPARC: ZFS ルートファイルシステムのブートブロックのインストール

ルートプールで使用するためのディスクを物理的に交換してから Oracle Solaris OS を再インストールした場合や、ルートプール用の新しいディスクを接続した場合には、ブートブロックは自動的にインストールされます。ルートプールで使用するためのディスクを `zpool replace` コマンドを使用して交換する場合は、交換後のディスクからシステムをブートできるように、ブートブロックを手動でインストールする必要があります。

次の例は、ZFS ルートファイルシステムのブートブロックをインストールする方法を示しています。

```
installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdsd/c0t1d0s0
```

## ▼ SPARC: ZFS ファイルシステム用のディスクを設定する方法

ルート以外の ZFS ファイルシステムで使用されるディスクを設定する場合、そのディスクには、プールが作成されるときやプールにディスクが追加されるときに自動的にラベルが付けられます。ディスクの全体を使ってプールが作成される場合や、ZFS ストレージプールに1つのディスク全体が追加される場合には、EFI ラベルが適用されます。EFI ディスクラベルの詳細は、[168 ページの「EFI \(GPT\) ディスクラベル」](#)を参照してください。

一般に、最近のバスタイプのほとんどで、ホットプラグ機能がサポートされています。これは、ユーザーが空のスロットにディスクを挿入すれば、システムがそのディスクを認識してくれることを意味します。ホットプラグ対応デバイスの詳細は、[第4章「デバイスの動的構成 \(タスク\)」](#)を参照してください。

- 1 スーパーユーザーになります。
- 2 システムにディスクを接続し、ディスクが物理的に正しく接続されていることを確認します。  
詳細は、ディスクのハードウェアインストールガイドを参照してください。

- 3 必要に応じて、故障したディスクをオフラインにして構成解除します。

一部のハードウェアでは、故障したディスクを交換するための `zpool replace` 操作を試みる前に、ディスクをオフラインにして構成解除する必要があります。例:

```
zpool offline tank c1t1d0
cfgadm -c unconfigure c1::dsk/c1t1d0
<Physically remove failed disk c1t1d0>
<Physically insert replacement disk c1t1d0>
cfgadm -c configure c1::dsk/c1t1d0
```

一部のハードウェアでは、交換用ディスクの装着後にそのディスクを再構成する必要があります。

- 4 新しいディスクが認識されていることを確認します。

`format` ユーティリティーの出力を確認し、`AVAILABLE DISK SELECTIONS` の下にディスクが表示されているかチェックします。その後、`format` ユーティリティーを終了します。

```
format
```

- 5 必要に応じて、故障したディスクが交換されたことを **ZFS** に知らせます。

```
zpool replace tank c1t1d0
zpool online tank c1t1d0
```

新しいディスクの再同期化が実行されていることを確認します。

```
zpool status tank
```

- 6 必要に応じて、既存の **ZFS** ストレージプールに新しいディスクを接続します。

例:

```
zpool attach tank mirror c1t0d0 c2t0d0
```

新しいディスクの再同期化が実行されていることを確認します。

```
zpool status tank
```

詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』の第3章「[Oracle Solaris ZFS ストレージプールの管理](#)」を参照してください。



## x86: ディスクの設定 (手順)

---

この章では、x86 ベースのシステムでのディスクの設定方法について説明します。

x86 ベースのシステムでのディスクの設定に関する手順については、次のセクションを参照してください。

- 237 ページの「x86: UFS ファイルシステム用のディスクの設定 (タスクマップ)」
- 243 ページの「x86: ZFS ファイルシステム用のディスクの設定 (タスクマップ)」
- 253 ページの「Solaris fdisk パーティションの作成および変更」

ディスク管理の概要については、第7章「ディスクの管理 (概要)」を参照してください。SPARC ベースのシステムでのディスクの設定に関するステップ単位の手順については、第10章「SPARC: ディスクの設定 (手順)」を参照してください。

### x86: UFS ファイルシステム用のディスクの設定 (タスクマップ)

次のタスクマップは、x86 ベースのシステム上で UFS ルートファイルシステム用のディスクまたは UFS ファイルシステム用のディスクを設定するための手順を記述したものです。

タスク	説明	参照先
1. ディスクを接続します。	UFS ルートファイルシステム用のディスク  新しいディスクを接続して、ローカルまたはリモートの Oracle Solaris DVD からブートします。	239 ページの「x86: UFS ルートファイルシステム用のディスクを設定する方法」

タスク	説明	参照先
	UFS ファイルシステム用のディスク ディスクを接続します。	239 ページの「x86: UFS ファイルシステム用のディスクを接続する方法」
2. fdisk パーティションを作成または変更します。	x86 ベースシステムのディスクには fdisk パーティションが含まれている必要があります。	
3. スライスとディスクラベルを作成します。	ディスクスライスを作成してディスクにラベルを付けます (ディスク製造元により実行されていない場合)。	253 ページの「x86: Solaris fdisk パーティションを作成する方法」および 240 ページの「x86: UFS ファイルシステム用のディスクスライスを作成する方法」
4. UFS ファイルシステムを作成します。	newfs コマンドを使用してディスクスライス上に UFS ファイルシステムを作成します。ルート (/) または /usr ファイルシステム (あるいはその両方) を作成し直す必要があります。	242 ページの「x86: UFS ファイルシステムを作成する方法」
5. UFS ファイルシステムを復元します。	システムディスク上に UFS ルート (/) または /usr ファイルシステム (あるいは両方) を復元します。必要に応じて、ほかのディスク上でファイルシステムを復元します	404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する (タスクマップ)」
6. UFS ルートファイルシステムのブートブロックをインストールします。	システムをブートできるように、ルート (/) ファイルシステムにブートブロックをインストールします。	243 ページの「x86: UFS ルートファイルシステムのブートブロックをインストールする方法」

## x86: UFS ファイルシステム用のディスクの設定

ディスクには、UFS ルート (/) および /usr ファイルシステムを含めることも、ユーザーまたはデータファイルシステムを含めることも、あるいはそれらのファイルシステムの組み合わせを含めることもできます。UFS ルート (/) または /usr ファイルシステムを含むディスクを復元するには、次の 2 つの方法があります。

- Oracle Solaris OS 全体を再インストールします。
- ディスクを交換し、Oracle Solaris OS のファイルシステムをバックアップメディアから復元します。

ユーザーまたはデータファイルシステムを含むディスクの交換やそのファイルシステムの復元は必要に応じて、システムの再インストールやリブートを行わなくても実行することができます。

▼ **x86: UFS ルートファイルシステム用のディスクを設定する方法**  
次の手順は、UFS ルートまたは /usr ファイルシステムが格納されていたディスクを交換する場合に使用できます。ディスクの交換方法については、ハードウェアの取り付けガイドを参照してください。

- 1 必要に応じて、損傷したシステムディスクをシステムから外します。
- 2 交換用のディスクを接続して、ディスクの物理的接続を確認します。
- 3 ローカルの **Oracle Solaris DVD** またはリモートの **Oracle Solaris DVD** (ネットワーク経由) のどちらでブートするかに応じて、次の表の手順で操作します。

ブートの種類	アクション
ローカルドライブの Oracle Solaris DVD	1. ドライブに Oracle Solaris DVD が入っていることを確認します。  2. メディアからブートするオプションを選択します。
ネットワーク経由	3. ネットワークからブートするオプションを選択します。

参考    **UFS ルートファイルシステム用のディスクを設定したあとの作業 ...**  
必要に応じて、fdisk パーティションを作成します。253 ページの「[x86: Solaris fdisk パーティションを作成する方法](#)」を参照してください。

▼ **x86: UFS ファイルシステム用のディスクを接続する方法**  
EFI ディスクラベル付きのディスクを追加する場合は、168 ページの「[EFI \(GPT\) ディスクラベル](#)」で詳細を参照してください。

一般に、最近のバスタイプのほとんどで、ホットプラグ機能がサポートされています。これは、ユーザーが空のスロットにディスクを挿入すれば、システムがそのディスクを認識してくれることを意味します。ホットプラグ対応デバイスの詳細は、第4章「[デバイスの動的構成 \(タスク\)](#)」を参照してください。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。

- 2 (省略可能)ディスクが **Oracle Solaris** ソフトウェアでサポートされない場合は、ハードウェアに添付された説明書に従って、そのディスクのデバイスドライバを追加します。
- 3 ディスクをシステムに接続して、ディスクの物理的接続を確認します。  
詳細は、ディスクのハードウェアインストールガイドを参照してください。
- 4 新しいディスクが認識されていることを、**format** ユーティリティーで確認します。

参考 UFS ファイルシステム用のディスクを接続したあとの作業 ...

ディスクを接続し終わったら、**fdisk** パーティションを作成します。[253 ページ](#)の「[x86: Solaris fdisk パーティションを作成する方法](#)」を参照してください。

## ▼ x86: UFS ファイルシステム用のディスクスライスを作成する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 ディスクに有効な **Solaris fdisk** パーティションが含まれていることを確認します。  
**Solaris fdisk** パーティションを作成または変更する方法については、[253 ページ](#)の「[Solaris fdisk パーティションの作成および変更](#)」を参照してください。

- 3 **format** ユーティリティーを起動します。

```
format
```

番号付きのディスクのリストが表示されます。

- 4 パーティションの再分割を実行するディスクの番号を入力します。

```
Specify disk (enter its number): disk-number
```

*disk-number* は、パーティションの再分割を実行するディスクの番号です。

- 5 **partition** メニューを選択します。

```
format> partition
```

- 6 現在のパーティション(スライス)テーブルを表示します。

```
partition> print
```

- 7 変更作業を開始します。

```
partition> modify
```

- 8 ディスクをすべて **free hog** に設定します。

```
Choose base (enter number) [0]? 1
```



free hog スライスの詳細は、[179 ページの「free hog スライスの使用方法」](#)を参照してください。

- 9 続行するかどうかを尋ねるプロンプトが表示されたら、**yes** と応答して新しいパーティションテーブルを作成します。

```
Do you wish to continue creating a new partition
table based on above table[yes]? yes
```

- 10 プロンプトが表示されたら、**free hog** パーティション (スライス) と各スライスのサイズを指定します。

システムディスクを追加するときは、次のスライスを設定しなければなりません。

- ルート (スライス 0)、スワップ (スライス 1) (必須) および
- /usr (スライス 6)

スライスの情報を設定すると、新しいパーティションテーブルが表示されます。

- 11 プロンプトが表示されたら **yes** と応答して、表示されたパーティションテーブルを現在のパーティションテーブルにします。

```
Okay to make this the current partition table[yes]? yes
```

表示されたパーティションテーブルが希望どおりでないために変更する場合は、no と応答して[手順 7](#)に戻ります。

- 12 パーティションテーブルに名前を付けます。

```
Enter table name (remember quotes): "partition-name"
```

*partition-name* には、新しいパーティションテーブルの名前を指定します。

- 13 新しいディスク上でスライスの割り当てが終わったら、新しいパーティションテーブルを使用してディスクにラベルを付けます。

```
Ready to label disk, continue? yes
```

- 14 **partition** メニューを終了します。

```
partition> quit
```

- 15 新しいディスクラベルを検査します。

```
format> verify
```

- 16 **format** ユーティリティーを終了します。

```
format> quit
```

## 参考 ディスクスライスとディスクラベルを作成したあとの作業

ディスクスライスとディスクラベルを作成し終わったら、ディスク上にファイルシステムを作成できます。[242 ページの「x86: UFS ファイルシステムを作成する方法」](#)を参照してください。

## ▼ x86: UFS ファイルシステムを作成する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 スライスごとにファイルシステムを作成します。

```
newfs /dev/rdsk/cwtxdysz
```

/dev/rdsk/cwtxdysz は、作成するファイルシステム用の raw デバイスです。

newfs コマンドの詳細は、[第 15 章「ファイルシステムの作成およびマウント \(タスク\)」](#) または [newfs\(1M\)](#) を参照してください。

- 3 新しいファイルシステムをマウントして、確認します。

```
mount /dev/dsk/cwtxdysz /mnt
ls /mnt
lost+found
```

## 参考 UFS ファイルシステムを作成したあとの作業

- UFS ルートファイルシステム用のディスク - ディスク上でルート (/) および /usr ファイルシステムを復元する必要があります。
  - [404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する \(タスクマップ\)」](#)に進みます。
  - ルート (/) と /usr ファイルシステムの復元後、ブートブロックをインストールします。[243 ページの「x86: UFS ルートファイルシステムのブートブロックをインストールする方法」](#)を参照してください。
- UFS ファイルシステム用のディスク - 新しいディスク上でファイルシステムを復元しなければいけない可能性があります。[404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する \(タスクマップ\)」](#)に進みます。新しいディスク上にファイルシステムを復元しない場合は、セカンダリディスクを追加する作業が終わったことになります。
- ユーザーがファイルシステムを利用できるようにする方法については、[315 ページの「Oracle Solaris ファイルシステムのマウントおよびマウント解除」](#)を参照してください。

▼ **x86: UFS ルートファイルシステムのブートブロックをインストールする方法**

- 1    スーパーユーザーになるか、同等の役割を引き受けます。
- 2    **UFS ルートファイルシステムのブートブロックをインストールします。**  
      `# /sbin/installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/cwtxdysz`  
      詳細は、[installgrub\(1M\)](#)のマニュアルページを参照してください。
- 3    システムをリブートし、レベル3で実行することによって、ブートブロックがインストールされていることを確認します。  
      `# init 6`

例 11-1    x86: UFS ルートファイルシステムのブートブロックのインストール

次の例は、x86 システム上で UFS ルートファイルシステムのブートブロックをインストールする方法を示したものです。

```
/sbin/installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c1d0s0
stage1 written to partition 0 sector 0 (abs 2016)
stage2 written to to partition 0, 227 sectors starting 50 (abs 2066)
```

# **x86: ZFS ファイルシステム用のディスクの設定 (タスクマップ)**

次のタスクマップに、x86 ベースシステム上で ZFS ルートファイルシステム用の ZFS ルートプールディスクを設定する手順を示します。

タスク	説明	参照先
1. ZFS ルートファイルシステム用のディスクを設定します。	ZFS ルートファイルシステム用のディスク  新しいディスクを接続するか既存のルートプールディスクを交換し、ローカルまたはリモートの Oracle Solaris DVD からブートします。	245 ページの「x86: ZFS ルートファイルシステム用のディスクを設定する方法」
2. 必要に応じて、fdisk パーティションを作成または変更します。	有効な Solaris fdisk パーティションがディスクに含まれている必要があります。	253 ページの「Solaris fdisk パーティションの作成および変更」

タスク	説明	参照先
3. ZFS ルートファイルシステム用のディスクスライスを作成します。	ZFS ルートプールで使用するためのディスクのディスクスライスを作成します。これは、長年にわたるブートの制限です。	253 ページの「 <a href="#">x86: Solaris fdisk パーティションを作成する方法</a> 」および 246 ページの「 <a href="#">x86: ZFS ルートファイルシステム用のディスクスライスを作成する方法</a> 」
4. ZFS ルートファイルシステムのブートブロックをインストールします。	ルートプールで使用するためのディスクを <code>zpool replace</code> コマンドを使用して交換する場合は、交換後のディスクからシステムをブートできるように、ブートブロックを手動でインストールする必要があります。	251 ページの「 <a href="#">x86: ZFS ルートファイルシステムのブートブロックをインストールする方法</a> 」
5. ZFS ファイルシステム用のディスクを設定します。	ZFS ファイルシステム用のディスク ディスクを接続します。	251 ページの「 <a href="#">x86: ZFS ファイルシステム用のディスクを設定する方法</a> 」

## x86: ZFS ファイルシステム用のディスクの設定

ZFS ファイルシステムでも、記述された手順に従ってディスクを設定したり、`fdisk` パーティションを作成したりできますが、ディスクやディスクスライスへの直接のマッピングは行われません。ZFS ファイルシステムを作成する前に、ZFS ストレージプールを作成する必要があります。詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』を参照してください。

ルートプールには、Oracle Solaris OS のブートに使用されるルートファイルシステムが含まれています。ミラー化されていないルートプールのディスクが破損すると、システムがブートしない可能性があります。ルートプールディスクが破損した場合の復旧方法としては、次の 2 つの方法が考えられます。

- Oracle Solaris OS 全体を再インストールします。
- ルートプールディスクを交換し、スナップショットまたはバックアップメディアからファイルシステムを復元します。ハードウェア障害によるシステムの停止時間を減らすには、冗長ルートプールを作成します。サポートされる冗長ルートプール構成は、ミラー化ルートプールだけです。

ルート以外のプールで使用されるディスクには通常、ユーザーファイルやデータファイル用の領域が含まれます。ルートプールやルート以外のプールに別のディスクを接続または追加すれば、ディスク容量を増やすことができます。あるいは、プール内の破損したディスクを次の方法で交換することもできます。

- 非冗長プール内のディスクは、すべてのデバイスが現在 ONLINE 状態になっている場合は交換可能です。
- 冗長プール内のディスクは、ほかのデバイスで十分な冗長性を確保できる場合は交換可能です。
- ミラー化ルートプールでは、ディスクを交換したりディスクを接続したりしたあと、故障したディスクやサイズの小さいディスクを切り離してプールのサイズを増やすことができます。

一般に、システムへのディスクの設定方法はハードウェアごとに異なるため、システム上でディスクの追加や交換を行う際にはハードウェアのドキュメントを確認してください。ディスクを既存のコントローラに追加する必要がある場合、システムがホットプラグをサポートしていれば、空のスロットにディスクを挿入するだけで済む可能性があります。新しいコントローラを構成する必要がある場合は、[74 ページの「動的再構成とホットプラグ機能」](#)を参照してください。

▼ **x86: ZFS ルートファイルシステム用のディスクを設定する方法**

ディスクの交換方法については、ハードウェアの取り付けガイドを参照してください。

- 1 必要に応じて、損傷したディスクをシステムから外します。
- 2 交換用のディスクを接続して、ディスクの物理的接続を確認します。
- 3 ローカルの **Oracle Solaris DVD** またはリモートの **Oracle Solaris DVD** (ネットワーク経由) のどちらでブートするかに応じて、次の表の手順で操作します。

ブートの種類	アクション
ローカルドライブの Oracle Solaris DVD	1. ドライブに Oracle Solaris DVD が入っていることを確認します。  2. メディアからブートするオプションを選択します。
ネットワーク経由	3. ネットワークからブートするオプションを選択します。

参考 **ZFS ルートファイルシステム用のディスクを設定したあとの作業...**

ディスクの接続または交換が完了したら、`fdisk` パーティションを作成します。[253 ページの「x86: Solaris fdisk パーティションを作成する方法」](#)を参照してください。

## x86: ZFS ルートファイルシステム用のディスクスライスの作成

ZFS ルートプールで使用するためのディスクでは、ディスクスライスを作成する必要があります。これは、長年にわたるブートの制限です。ルートプールディスクの次の要件を確認してください。

- ディスクスライスと SMI (VTOC) ラベルが含まれている必要があります。
- ルートプールディスクでは、EFI ラベルはサポートされません。
- x86 システムのルートプールディスクには fdisk パーティションが含まれている必要があります。
- 単一のディスクであるか、あるいはミラー化構成の一部になっている必要があります。ルートプールでは、非冗長構成や RAIDZ 構成はサポートされません。
- ルートファイルシステムの、OS イメージの一部となっているサブディレクトリのうち、/var 以外のものはすべて、ルートファイルシステムと同じデータセット内に存在している必要があります。
- スワップデバイスとダンプデバイス以外の Solaris OS コンポーネントはすべて、ルートプール内に存在している必要があります。
- さらに、/var ディレクトリまたはデータセットは単一のデータセットでなければならない、という制限もあります。たとえば、Oracle Solaris Live Upgrade を使って ZFS BE の移行やパッチ適用を行ったり、このプールの ZFS フラッシュアーカイブを作成したりする必要もある場合には、/var/tmp のような /var の下位ファイルシステムを作成することはできません。

x86 ベースシステムでは、まず fdisk パーティションを作成する必要があります。その後、スライス 0 で、ディスク領域の全体を使ってディスクスライスを作成します。

ディスク上の別のスライスを使用し、そのディスクを各種オペレーティングシステム間で共有したり、別の ZFS ストレージプールやストレージプールコンポーネントと共有したりしようとすることは、お勧めできません。

### ▼ x86: ZFS ルートファイルシステム用のディスクスライスを作成する方法

一般に、ルートプールディスクのインストールは、システムのインストール時に自動的に行われます。ルートプールディスクを交換したり、新しいディスクをミラー化ルートプールのディスクとして接続したりする必要がある場合には、次の手順を参照してください。

fdisk パーティションの完全な説明については、[253 ページの「x86: fdisk パーティションの作成上のガイドライン」](#)を参照してください。

- 1 スーパーユーザーになります。

- 2 必要に応じて、故障したディスクをオフラインにして構成解除します。  
一部のハードウェアでは、故障したディスクを交換するための `zpool replace` 操作を試みる前に、ディスクをオフラインにして構成解除する必要があります。例:

```
zpool offline rpool c2t1d0s0
cfgadm -c unconfigure c2::dsk/c2t1d0
```

- 3 必要に応じて、新しいディスクや交換用ディスクをシステムに物理的に接続します。

- a. 故障したディスクを物理的に取り外します。
- b. 交換用ディスクを物理的に挿入します。
- c. 必要に応じて、交換用ディスクを構成します。例:

```
cfgadm -c configure c2::dsk/c2t1d0
```

一部のハードウェアでは、交換用ディスクの装着後にそのディスクを再構成する必要があります。

- 4 `format` の出力を確認してディスクにアクセスできることを確認します。

たとえば、次の `format` コマンドでは、このシステムに4つのディスクが接続されていることがわかります。

```
format -e
AVAILABLE DISK SELECTIONS:
 1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
 /pci@0,0/pci10de,375ef/pci108e,286e0/disk@0,0
 2. c8t1d0 <Sun-STK RAID INT-V1.0-136.61GB>
 /pci@0,0/pci10de,375ef/pci108e,286e0/disk@1,0
 3. c8t2d0 <Sun-STK RAID INT-V1.0-136.61GB>
 /pci@0,0/pci10de,375ef/pci108e,286e0/disk@2,0
 4. c8t3d0 <Sun-STK RAID INT-V1.0-136.61GB>
 /pci@0,0/pci10de,375ef/pci108e,286e0/disk@3,0
```

- 5 ZFS ルートプールに使用されるディスクを選択します。

```
Specify disk (enter its number): 1
selecting c8t0d0
[disk formatted]
.
.
.
format>
```

- 6 `fdisk` パーティションのステータスを確認します。

- ディスクに `fdisk` パーティションが含まれていない場合は、次のようなメッセージが表示されます。

```
format> fdisk
No Solaris fdisk partition found.
```

その場合は手順 4 に進み、**fdisk** パーティションを作成します。

- ディスクに **EFI fdisk** やその他のパーティションタイプが含まれている場合は手順 5 に進み、**Solaris fdisk** パーティションを作成します。
- ディスクに **Solaris fdisk** パーティションが含まれている場合は手順 6 に進み、ルートプール用のディスクスライスを作成します。

**7 必要に応じて、**fdisk** オプションを選択して **Solaris fdisk** パーティションを作成します。**

```
format> fdisk
No fdisk table exists. The default partition for the disk is:
```

```
a 100% "SOLARIS System" partition
```

```
Type "y" to accept the default partition, otherwise type "n" to edit the
partition table. y
```

続いて手順 6 に進み、ルートプール用のディスクスライスを作成します。

**8 ディスクに **EFI fdisk** パーティションが含まれている場合は、**Solaris fdisk** パーティションを作成する必要があります。**

**format** ユーティリティーでディスクのパーティションテーブルを出力し、そのテーブル内に先頭セクターとサイズが含まれていた場合、これは **EFI** パーティションになります。**Solaris fdisk** パーティションを次のようにして作成する必要があります。

- **format** のオプションから **fdisk** を選択します。

```
format -e c8t0d0
selecting c8t0d0
[disk formatted]
format> fdisk
```

- オプション 3 「Delete a partition」を選択して既存の **EFI** パーティションを削除します。

```
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
Are you sure you want to delete partition 1? This will make all files and
programs in this partition inaccessible (type "y" or "n"). y
```

```
Partition 1 has been deleted.
```

- オプション 1 「Create a partition」を選択して新しい **Solaris** パーティションを作成します。

```
Enter Selection: 1
Select the partition type to create: 1
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it will be activated
each time the computer is reset or turned on.
Please type "y" or "n". y
Partition 1 is now the active partition.
```



- ディスク構成を更新して終了します。

```
Enter Selection: 6
format>
```

- SMI パーティションテーブルを表示します。デフォルトのパーティションテーブルが適用されていた場合、スライス0のサイズが0であるか非常に小さい値である可能性があります。次の手順を参照してください。

```
format> partition
partition> print
```

- 9 パーティション(スライス)情報を出力することでディスクにSMIラベルが含まれていることを確認し、スライス0のサイズ情報を確認します。

未割り当てのディスク領域がすべてスライス0に集められるように、free hog パーティションを設定します。次に、すべてのスライスサイズフィールドで Return キーを押すことで、1つの大きなスライス0を作成します。

```
partition> modify
Select partitioning base:
 0. Current partition table (default)
 1. All Free Hog
Choose base (enter number) [0]? 1
Part Tag Flag Cylinders Size Blocks
 0 root wm 0 0 (0/0/0) 0
 1 swap wu 0 0 (0/0/0) 0
 2 backup wu 0 - 17829 136.58GB (17830/0/0) 286438950
 3 unassigned wm 0 0 (0/0/0) 0
 4 unassigned wm 0 0 (0/0/0) 0
 5 unassigned wm 0 0 (0/0/0) 0
 6 usr wm 0 0 (0/0/0) 0
 7 unassigned wm 0 0 (0/0/0) 0
 8 boot wu 0 - 0 7.84MB (1/0/0) 16065
 9 alternates wm 0 0 (0/0/0) 0
```

Do you wish to continue creating a new partition table based on above table[yes]?

```
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:
```

```
Part Tag Flag Cylinders Size Blocks
 0 root wm 1 - 17829 136.58GB (17829/0/0) 286422885
 1 swap wu 0 0 (0/0/0) 0
 2 backup wu 0 - 17829 136.58GB (17830/0/0) 286438950
 3 unassigned wm 0 0 (0/0/0) 0
 4 unassigned wm 0 0 (0/0/0) 0
 5 unassigned wm 0 0 (0/0/0) 0
 6 usr wm 0 0 (0/0/0) 0
 7 unassigned wm 0 0 (0/0/0) 0
 8 boot wu 0 - 0 7.84MB (1/0/0) 16065
 9 alternates wm 0 0 (0/0/0) 0
```

Do you wish to continue creating a new partition

```
table based on above table[yes]? yes
Enter table name (remember quotes): "c8t0d0"
```

```
Ready to label disk, continue? yes
```

- 10 故障したディスクが交換されたことを ZFS に知らせます。

```
zpool replace rpool c2t1d0s0
zpool online rpool c2t1d0s0
```

一部のハードウェアでは、交換用ディスクの装着後にそのディスクをオンラインにする必要があります。

新しいディスクを接続してミラー化ルートプールを作成したり、サイズの大きなディスクを接続してサイズの小さいディスクと交換したりする場合には、次のような構文を使用します。

```
zpool attach rpool c0t0d0s0 c1t0d0s0
```

- 11 ルートプールディスクを新しいディスクと交換した場合には、ブートブロックを適用します。

例:

```
installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/c8t0d0s0
```

- 12 新しいディスクからブートできることを確認します。

- 13 新しいディスクからシステムがブートした場合は、古いディスクを切り離します。  
この手順が必要となるのは、故障したディスクやサイズの小さいディスクを交換するために新しいディスクを接続した場合だけです。

```
zpool detach rpool c0t0d0s0
```

- 14 システムの BIOS を再構成することにより、新しいディスクから自動的にブートするようにシステムを設定します。

参考 ZFS ルートファイルシステム用のディスクスライスを作成したあとの作業 ...

ZFS ルートファイルシステム用のディスクスライスを作成し終わったら、ルートプールのスナップショットを復元してルートプールを回復する必要があります。『Oracle Solaris ZFS 管理ガイド』の「ZFS ルートプールまたはルートプールのスナップショットを回復する」を参照してください。

## ▼ x86: ZFS ルートファイルシステムのブートブロックをインストールする方法

- 1 スーパーユーザーになります。
- 2 システムディスクにブートブロックをインストールします。  

```
installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/cwtxdysz
```

 詳細は、[installgrub\(1M\)](#)のマニュアルページを参照してください。
- 3 システムをリブートし、レベル3で実行することによって、ブートブロックがインストールされていることを確認します。  

```
init 6
```

### 例 11-2 x86: ZFS ルートファイルシステムのブートブロックのインストール

ルートプールで使用するためのディスクを物理的に交換してから Oracle Solaris OS を再インストールした場合や、ルートプール用の新しいディスクを接続した場合には、ブートブロックは自動的にインストールされます。ルートプールで使用するためのディスクを `zpool replace` コマンドを使用して交換する場合は、交換後のディスクからシステムをブートできるように、ブートブロックを手動でインストールする必要があります。

次の例は、ZFS ルートファイルシステムのブートブロックをインストールする方法を示しています。

```
installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/c1d0s0
stage2 written to partition 0, 277 sectors starting at 50 (abs 16115)
stage1 written to partition 0 sector 0 (abs 16065)
```

## ▼ x86: ZFS ファイルシステム用のディスクを設定する方法

ルート以外の ZFS ファイルシステムで使用されるディスクを設定する場合、そのディスクには、プールが作成されるときやプールにディスクが追加されるときに自動的にラベルが付けられます。ディスクの全体を使ってプールが作成される場合や、ZFS ストレージプールに1つのディスク全体が追加される場合には、EFI ラベルが適用されます。EFI ディスクラベルの詳細は、[168 ページの「EFI \(GPT\) ディスクラベル」](#)を参照してください。

一般に、最近のバスタイプのほとんどで、ホットプラグ機能がサポートされています。これは、ユーザーが空のスロットにディスクを挿入すれば、システムがそのディスクを認識してくれることを意味します。ホットプラグ対応デバイスの詳細は、[第4章「デバイスの動的構成 \(タスク\)」](#)を参照してください。

- 1 スーパーユーザーになります。

- 2 システムにディスクを接続し、ディスクが物理的に正しく接続されていることを確認します。

詳細は、ディスクのハードウェアインストールガイドを参照してください。

- 3 必要に応じて、故障したディスクをオフラインにして構成解除します。  
一部のハードウェアでは、故障したディスクを交換するための `zpool replace` 操作を試みる前に、ディスクをオフラインにして構成解除する必要があります。例:

```
zpool offline tank c1t1d0
cfgadm -c unconfigure c1::dsk/c1t1d0
<Physically remove failed disk c1t1d0>
<Physically insert replacement disk c1t1d0>
cfgadm -c configure c1::dsk/c1t1d0
```

一部のハードウェアでは、交換用ディスクの装着後にそのディスクを再構成する必要があります。

- 4 新しいディスクが認識されていることを確認します。  
`format` ユーティリティの出力を確認し、AVAILABLE DISK SELECTIONS の下にディスクが表示されているかチェックします。その後、`format` ユーティリティを終了します。

```
format
```

- 5 必要に応じて、故障したディスクが交換されたことを ZFS に知らせます。

```
zpool replace tank c1t1d0
zpool online tank c1t1d0
```

新しいディスクの再同期化が実行されていることを確認します。

```
zpool status tank
```

- 6 必要に応じて、既存の ZFS ストレージプールに新しいディスクを接続します。  
例:

```
zpool attach tank mirror c1t0d0 c2t0d0
```

新しいディスクの再同期化が実行されていることを確認します。

```
zpool status tank
```

詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』の第3章「[Oracle Solaris ZFS ストレージプールの管理](#)」を参照してください。

# Solaris fdisk パーティションの作成および変更

Solaris fdisk パーティションの作成または変更に関するガイドラインや例については、次の各セクションを確認してください。

## x86: fdisk パーティションの作成上のガイドライン

次のガイドラインに従って1つまたは複数の fdisk パーティションを設定してください。

- ディスクは最大4つの fdisk パーティションに分割できます。いずれか1つのパーティションを Solaris パーティションにする必要があります。
- Solaris パーティションをディスク上でアクティブなパーティションにする必要があります。アクティブなパーティションとは、システム起動時にデフォルトでオペレーティングシステムがブートされるパーティションのことです。
- Solaris の fdisk パーティションは、シリンダ境界から開始する必要があります。
- 最初のディスクの先頭のセクターには、ブート情報 (マスターブートレコードを含む) が書き込まれるので、最初のディスクの1番目の fdisk パーティションとして、Solaris の fdisk パーティションを作成する場合は、ディスクのシリンダ0ではなくシリンダ1から開始しなければなりません。
- ディスク全体を Solaris fdisk パーティションにすることもできます。または、もっと小さいサイズにして、DOS パーティションに多くの容量を割り当てることもできます。十分な空き容量がある場合は、既存のパーティションに影響を与えずに、ディスク上に新しい fdisk パーティションを作成することもできます。

---

**x86のみ** – Solaris スライスとはパーティションと呼ばれることがあります。インタフェースによっては、「パーティション」の代わりに「スライス」という用語を使用します。

fdisk パーティションは、x86 システムでのみサポートされます。混乱を避けるため、Oracle Solaris のドキュメントでは、fdisk パーティションと Solaris fdisk パーティション内のエンティティを区別しています。「スライス」または「パーティション」と呼ばれるのは、後者です。

---

## ▼ x86: Solaris fdisk パーティションを作成する方法

始める前に fdisk パーティションについては、[253 ページの「x86: fdisk パーティションの作成上のガイドライン」](#)を参照してください。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。

- 2 **format** ユーティリティーを起動します。

```
format
```

番号付きのディスクのリストが表示されます。

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

- 3 **Solaris fdisk** パーティションを作成するディスクの番号を入力します。

```
Specify disk (enter its number): disk-number
```

*disk-number* は、Solaris fdisk パーティションを作成するディスクの番号です。

- 4 **fdisk** メニューを選択します。

```
format> fdisk
```

表示される fdisk メニューは、fdisk パーティションがすでにディスク上に存在しているかどうかによって異なります。

- 5 ディスク全体にまたがる **Solaris fdisk** パーティションを作成してそれをアクティブにするには、プロンプトで **y** を入力します。次に、手順 13 に進みます。

```
No fdisk table exists. The default partition for the disk is:
```

```
a 100% "SOLARIS System" partition
```

```
Type "y" to accept the default partition, otherwise type "n" to edit the
partition table.
```

```
y
```

- 6 ディスク全体にまたがる **Solaris fdisk** パーティションを作成しない場合は、プロンプトで **n** を入力します。

```
Type "y" to accept the default partition, otherwise type "n" to edit the
partition table.
```

```
n
```

```
Total disk size is 17848 cylinders
Cylinder size is 16065 (512 byte) blocks
```

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====

```
SELECT ONE OF THE FOLLOWING:
```

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Edit/View extended partitions
6. Exit (update disk configuration and exit)
7. Cancel (exit without updating disk configuration)

```
Enter Selection:
```

- 7 「1. Create a partition」を選択し、**fdisk** パーティションを作成します。

Enter Selection: 1

- 8 「1(=Solaris2)」を選択して、**Solaris fdisk** パーティションを作成します。

Select the partition type to create:

1=SOLARIS2	2=UNIX	3=PCIXOS	4=Other
5=DOS12	6=DOS16	7=DOSEXT	8=DOSBIG
9=DOS16LBA	A=x86 Boot	B=Diagnostic	C=FAT32
D=FAT32LBA	E=DOSEXTLBA	F=EFI	0=Exit? 1

- 9 **Solaris fdisk** パーティション用に割り当てるディスクのパーセントを指定します。このパーセントを計算するときには、既存の **fdisk** パーティションのサイズを考慮してください。

Specify the percentage of disk to use for this partition  
(or type "c" to specify the size in cylinders). *nn*

- 10 プロンプトで **y** を入力して、**Solaris fdisk** パーティションをアクティブにします。

Should this become the active partition? If yes, it will be  
activated each time the computer is reset or turned on.  
Please type "y" or "n". **y**

**fdisk** パーティションがアクティブになったあと、Enter Selection: プロンプトが表示されます。

- 11 別の **fdisk** パーティションを作成する場合は、「1. Create a partition」を選択します。

**fdisk** パーティションの作成方法については、手順 8 - 10 を参照してください。

- 12 ディスク構成を更新し、**Selection** メニューから **fdisk** メニューを終了します。

Selection: 6

- 13 **label** コマンドを使用して、ディスクに新しいラベルを付けます。

```
format> label
Ready to label disk, continue? yes
format>
```

- 14 **format** ユーティリティーを終了します。

```
format> quit
```

### 例 11-3 x86: ディスク全体を占有する Solaris fdisk パーティションを作成する

次の例は、**format** ユーティリティーの **fdisk** オプションを使用して、ディスク全体にまたがる **Solaris fdisk** パーティションを作成する方法を説明しています。

```
format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
```

```

0. c0d0 <DEFAULT cyl 2466 alt 2 hd 16 sec 63>
 /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0
1. c0d1 <DEFAULT cyl 522 alt 2 hd 32 sec 63>
 /pci@0,0/pci-ide@7,1/ide@0/cmdk@1,0
2. c1d0 <DEFAULT cyl 13102 alt 2 hd 16 sec 63>
 /pci@0,0/pci-ide@7,1/ide@1/cmdk@0,0
Specify disk (enter its number): 0
selecting c0d0
Controller working list found
[disk formatted]
format> fdisk
No fdisk table exists. The default partitioning for your disk is:

 a 100% "SOLARIS System" partition.

Type "y" to accept the default partition, otherwise type "n" to edit the
partition table. y

format> label
Ready to label disk, continue? yes
format> quit

```

#### 参考 Solaris fdisk パーティションを作成したあとの作業

ディスク上で Solaris fdisk パーティションを作成し終わったら、そのディスクをルートプールディスクまたはルート以外のプールディスクとして使用できます。詳細については、[218 ページの「SPARC: UFS ファイルシステム用のディスクの設定」](#)を参照してください。

## fdisk パーティション識別子の変更

x86 システム版 Solaris の fdisk パーティション識別子が、130 (0x82) から 191 (0xbf) に変更されました。Oracle Solaris のすべてのコマンド、ユーティリティー、およびドライバは、どちらの fdisk 識別子を使用しても機能するように更新されました。fdisk の機能に変更はありません。

### ▼ Solaris fdisk 識別子を変更する方法

fdisk の新しいメニュー項目を使用すると、新しい識別子と古い識別子を切り替えられます。fdisk の識別子は、パーティションに含まれるファイルシステムがマウントされている場合でも変更できます。

fdisk メニューでは、次の 2 つの type の値によって、古い識別子と新しい識別子が表示されます。

- Solaris identifies 0x82
- Solaris2 identifies 0xbf

- 1 スーパーユーザーになります。



- 2 現在の **fdisk** 識別子を表示します。

例:

```
Total disk size is 39890 cylinders
Cylinder size is 4032 (512 byte) blocks
```

Partition	Status	Type	Cylinders		Length	%
=====	=====	=====	=====	=====	=====	=====
1	Active	x86 Boot	1	6	6	0
2		Solaris2	7	39889	39883	100

- 3 **fdisk** メニューのオプション 4 を選択し、**fdisk** パーティション識別子を **0x82** に変更します。

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 4

- 4 オプション 5 を選択し、ディスク構成を更新して終了します。

- 5 必要であれば、**fdisk** メニューのオプション 4 を選択し、**fdisk** パーティション識別子を **0xbf** に変更します。

例:

```
Total disk size is 39890 cylinders
Cylinder size is 4032 (512 byte) blocks
```

Partition	Status	Type	Cylinders		Length	%
=====	=====	=====	=====	=====	=====	=====
1	Active	x86 Boot	1	6	6	0
2		Solaris	7	39889	39883	100

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 4

- 6 オプション 5 を選択し、ディスク構成を更新して終了します。



## Oracle Solaris iSCSI ターゲットの構成 (タスク)

---

この章では、Oracle Solaris リリースで iSCSI ターゲットを構成する方法について説明します。iSCSI ターゲットの構成に関連する手順については、261 ページの「[Solaris iSCSI ターゲットデバイスの構成 \(タスク\)](#)」を参照してください。

iSCSI イニシエータの構成については、『[Oracle Solaris SAN Configuration and Multipathing Guide](#)』の第 4 章「[Configuring Oracle Solaris iSCSI Initiators](#)」を参照してください。

Solaris iSCSI の構成の問題のトラブルシューティングについては、281 ページの「[iSCSI 構成に関する問題のトラブルシューティング](#)」を参照してください。

## Oracle Solaris iSCSI の技術 (概要)

iSCSI は Internet SCSI (Small Computer System Interface) の略語であり、データストレージサブシステムを結合するための、インターネットプロトコル (IP) ベースのストレージネットワークワーキング標準です。このネットワークワーキング標準は、IETF (Internet Engineering Task Force) によって開発されました。iSCSI 技術の詳細は、次の RFC 3720 を参照してください。

<http://www.ietf.org/rfc/rfc3720.txt>

iSCSI プロトコルを使用すると、SCSI コマンドが IP ネットワーク経由で転送されるため、ユーザーはあたかもブロックデバイスがローカルシステムに接続されているかのようにネットワーク経由でブロックデバイスにアクセスできます。

既存の TCP/IP ネットワーク内のストレージデバイスを使用する場合、次の解決法が利用できます。

- iSCSI ブロックデバイスまたはテープ - SCSI コマンドとデータをブロックレベルから IP パケットへと変換します。あるシステムと、テープデバイスやデータベースなどのターゲットデバイスとの間で、ブロックレベルのアクセスが必要に

なる場合には、ネットワーク内で iSCSI を使用することをお勧めします。ブロックレベルデバイスへのアクセスにはロックがかからないため、iSCSI ターゲットデバイスなどのブロックレベルデバイスに複数のユーザーやシステムがアクセスすることができます。

- NFS – ファイルデータを IP 経由で転送します。ネットワーク内で NFS を使用する利点は、ファイルデータを複数のシステム間で共有できることにあります。NFS 環境で利用可能なデータに多数のユーザーがアクセスする場合、必要に応じてファイルデータへのアクセスにロックがかかります。

Solaris iSCSI ターゲットおよびイニシエータを使用する利点を次に示します。

- iSCSI プロトコルは、既存の Ethernet ネットワーク上で動作します。
  - サポートされている任意のネットワークインタフェースカード (NIC)、Ethernet ハブ、または Ethernet スイッチを使用できます。
  - 1 つの IP ポートから複数の iSCSI ターゲットデバイスを処理できます。
  - IP ネットワークの既存のインフラストラクチャーや管理ツールを使用できます。
- クライアントに接続可能な既存のファイバチャネルデバイスを利用でき、ファイバチャネル HBA の費用がかかりません。また、専用のアレイを持つシステムが、複製されたストレージを ZFS または UFS ファイルシステムでエクスポートすることも可能になりました。
- 構成可能な iSCSI ターゲットデバイスの最大数に制限はありません。
- このプロトコルは、適切なハードウェアを備えたファイバチャネル SAN (Storage Area Network) 環境または iSCSI SAN 環境への接続に使用できます。

Solaris iSCSI イニシエータソフトウェア使用時の現時点における制限や制約を、次に示します。

- SLP を使用する iSCSI デバイスは、現時点ではサポートされていません。
- iSCSI デバイスのブートは、現時点ではサポートされていません。
- iSCSI ターゲットをダンプデバイスとして構成することはできません。
- iSCSI は 1 つのセッションで複数の接続をサポートしますが、現在の Solaris 実装は 1 つのセッションで 1 つの接続しかサポートしません。

詳細は、RFC 3720 を参照してください。

- 既存のネットワーク経由で大量のデータを転送すると、パフォーマンスに影響する可能性があります。

## Solaris iSCSI のソフトウェア要件およびハードウェア要件の識別

- Solaris iSCSI ソフトウェアおよびデバイス

- Solaris iSCSI イニシエータソフトウェアの場合は Solaris 10 リリース (1/06 リリース以降)
- Solaris iSCSI ターゲットソフトウェアの場合は Solaris 10 リリース (8/07 リリース以降)
- 次の Solaris 10 ソフトウェアパッケージ:
  - SUNWiscsir – Sun iSCSI デバイスドライバ (root)
  - SUNWiscsiu – Sun iSCSI (usr)
  - SUNWiscsitgtr – Sun iSCSI ターゲット (root)
  - SUNWiscsitgtu – Sun iSCSI ターゲット管理ユーティリティ (usr)
- サポートされている NIC

## Solaris iSCSI ターゲットデバイスの構成 (タスク)

ここでは、Solaris iSCSI ターゲットデバイスの構成に関連する一般的なタスクのリストを示します。一部のタスクはネットワーク構成の必要に応じたオプションです。下の一部のリンクは、ネットワーク構成とイニシエータの構成について説明する個別のドキュメントに移動します。

- [260 ページの「Solaris iSCSI のソフトウェア要件およびハードウェア要件の識別」](#)
- [『Oracle Solaris 管理: ネットワークインタフェースとネットワーク仮想化』](#)
- [263 ページの「Solaris iSCSI ターゲットデバイスの構成と管理」](#)
- [『Oracle Solaris SAN Configuration and Multipathing Guide』の「How to Configure an iSCSI Initiator and Target Discovery」](#)
- [266 ページの「iSCSI ディスクにアクセスする方法」](#)
- [268 ページの「iSCSI ベースのストレージネットワークにおける認証の構成」](#)
- [272 ページの「Solaris iSCSI マルチパスデバイスの設定」](#)
- [276 ページの「iSCSI 構成の監視」](#)

## Oracle Solaris の iSCSI 用語

iSCSI ターゲットおよびイニシエータを構成する前に、次の用語を確認してください。

用語	説明
イニシエータ	iSCSI ターゲットに対する SCSI 要求を発行するドライバ。
ターゲットデバイス	iSCSI ストレージコンポーネント。

用語	説明
検出	利用可能なターゲットの一覧をイニシエータに提供するプロセス。
検出方法	<p>iSCSI ターゲットを発見するための方法。現時点では次の3つの方法を使用できます。</p> <ul style="list-style-type: none"> <li>■ Internet Storage Name Service (iSNS) - 1 つ以上の iSNS サーバーと対話することで可能性のあるターゲットを発見します。</li> <li>■ SendTargets - <i>discovery-address</i> を使用することによって、可能性のあるターゲットが発見されます。</li> <li>■ 静的 - 静的なターゲットアドレスが構成されます。</li> </ul>

## 動的または静的ターゲット発見の構成

デバイス発見を実行するために、動的デバイス発見方式のいずれかを構成するか、または静的 iSCSI イニシエータターゲットを使用するかを決定します。

- 動的デバイス発見 - ファイバチャネルブリッジへの iSCSI などのように、iSCSI ノードが多数のターゲットを公開している場合、その iSCSI ノードに IP アドレスとポートの組み合わせを提供すると、その iSCSI イニシエータは SendTargets 機能を使ってデバイス発見を実行できるようになります。

次の2つの動的デバイス発見方式を使用できます。

- SendTargets - ファイバチャネルブリッジへの iSCSI などのように、iSCSI ノードが多数のターゲットを公開している場合、その iSCSI ノードに IP アドレスとポートの組み合わせを提供すると、その iSCSI イニシエータは SendTargets 機能を使ってデバイス発見を実行できるようになります。
- iSNS - iSNS (Internet Storage Name Service) を使用すると、できる限り少ない構成情報で、iSCSI イニシエータがアクセス権を持つターゲットを発見できます。また、ストレージノードの動作状態が変更されたときに iSCSI イニシエータに通知する状態変更通知機能もあります。iSNS 発見方式を使用するために、iSNS サーバーのアドレスとポートの組み合わせを指定して、デバイス発見を実行するために指定した iSNS サーバーを iSCSI イニシエータで照会できるようにすることができます。iSNS サーバーのデフォルトポートは 3205 です。iSNS の詳細については、RFC 4171 を参照してください。  
<http://www.ietf.org/rfc/rfc4171.txt>  
iSNS 発見サービスは、ネットワーク内のすべてのターゲットを発見するための管理モデルを提供します。
- 静的デバイス発見 - iSCSI ノードのターゲットが少数である場合や、イニシエータのアクセス対象となるターゲットを制限する場合には、次の静的ターゲットアドレス命名規則を使用して、*target-name* を静的に構成できます。

`target,target-address[:port-number]`

また、アレイの管理ツールから静的ターゲットアドレスを決定できます。

---

注 - 単一の iSCSI ターゲットが静的デバイス発見方式と動的デバイス発見方式の両方によって発見されるように構成しないでください。発見方式を重複して使用すると、イニシエータが iSCSI ターゲットデバイスと通信するときのパフォーマンスが低下する可能性があります。

---

## Solaris iSCSI ターゲットデバイスの構成と管理

`iscsitadm` コマンドを使って、Solaris iSCSI ターゲットデバイスを設定および管理できます。Solaris iSCSI ターゲットデバイスは、ディスクまたはテープデバイスです。iSCSI ターゲットとして選択したデバイスでは、iSCSI デーモン用のバッキングストアとして、同サイズの ZFS または UFS ファイルシステムを提供する必要があります。

ZFS を使用した Solaris iSCSI ターゲットデバイスの設定については、『[Oracle Solaris ZFS 管理ガイド](#)』の「[ZFS ボリュームを Solaris iSCSI ターゲットとして使用する](#)」を参照してください。

ターゲットデバイスの設定後に、`iscsiadm` コマンドを使って iSCSI ターゲットを識別します。これにより、iSCSI ターゲットデバイスが検出および使用されます。

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

基本的な操作を次に示します。

- バッキングストアディレクトリの識別 - iSCSI デーモンは、作成されたターゲットおよび論理ユニットごとに情報を格納する必要があります。デフォルトでは、このデバイス用のバッキングストアはベースディレクトリ内にも配置されます。このため、ホストシステムが使用する ZFS プールが大規模なものである場合、もっとも簡単な方法は、その場所にあるすべてをデーモンで格納できるように許可することです。バッキングストアを分散する必要がある場合は、各論理ユニットを作成するときにバッキングストアの場所を指定できます。
- iSCSI ターゲットの作成 - CLI はデフォルトで、要求されたデバイスの種類が論理ユニット 0 の LBA であるとみなします。文字型デバイスのためにパスルールモードが必要な場合には、`-raw` オプションを使用する必要があります。最初の LUN を作成したあとで、`-lun number` を指定して、同じ iSCSI ターゲットのほかの LUN を作成できます。

デーモンによりバックグラウンドタスクが開始され、LUN がゼロに初期化されます。初期化中に基盤となるファイルシステムの使用率が 100 パーセントになった場合、デーモンによりターゲットが削除されます。この初期化中、LUN はオフライン

インとしてマーク付けされるため、イニシエータから使用することはできません。ただしこのとき、イニシエータからこの LUN を発見することはできます。待機中の Solaris イニシエータは、Inventory Change 通知を受信するとデバイスを自動的にオンラインにします。

## ▼ iSCSI ターゲットの作成方法

この手順では、iSCSI ターゲットのあるローカルシステムにユーザーがログインしているものとします。

- 1 スーパーユーザーになります。

- 2 ZFS プールとボリュームを作成します。

```
target# zpool create sanpool mirror c1t2d0 c1t3d0
target# zfs create sanpool/vol1
```

- 3 ターゲットのバックイングストアディレクトリを識別します。

例:

```
target# iscsitadm modify admin -d /sanpool/vol1
```

- 4 iSCSI ターゲットを作成します。

例:

```
target# iscsitadm create target --size 2g sanpool
```

- 5 iSCSI ターゲットに関する情報を表示します。

例:

```
target# iscsitadm list target -v sanpool
```

- 6 このターゲットを検出および使用するよう、iSCSI イニシエータを設定します。詳細は、[264 ページの「iSCSI ターゲット発見を構成する方法」](#)を参照してください。

## ▼ iSCSI ターゲット発見を構成する方法

この手順では、ユーザーが現在ログインしているローカルシステムから iSCSI ターゲットデバイスへのアクセスを構成するものとします。

- 1 スーパーユーザーになります。

- 2 次のいずれかの方式を使って、ターゲットデバイスが動的または静的に発見されるように構成します。

- デバイスが動的に発見されるように構成します (SendTargets)。

例:



```
initiator# iscsiadm add discovery-address 10.0.0.1:3260
```

- デバイスが動的に発見されるように構成します (iSNS)。

例:

```
initiator# iscsiadm add iSNS-server 10.0.0.1:3205
```

- デバイスが静的に発見されるように構成します。

例:

```
initiator# iscsiadm add static-config eui.5000ABCD78945E2B,10.0.0.1
```

発見方式が有効化されるまで iSCSI 接続は起動されません。次の手順を参照してください。

### 3 次のいずれかを使って iSCSI ターゲット発見方式を有効にします。

- 動的に発見される (SendTargets) デバイスを構成した場合は、SendTargets 発見方式を有効にします。

```
initiator# iscsiadm modify discovery --sendtargets enable
```

- 動的に発見される (iSNS) デバイスを構成した場合は、iSNS 発見方式を有効にします。

```
initiator# iscsiadm modify discovery --iSNS enable
```

- 静的ターゲットを構成した場合は、静的ターゲット発見方式を有効にします。

```
initiator# iscsiadm modify discovery --static enable
```

### 4 ローカルシステム用の iSCSI デバイスリンクを作成します。

```
initiator# devfsadm -i iscsi
```

## ▼ Solaris iSCSI ターゲットの iSNS 発見を構成する方法

ネットワークに他社製の iSNS サーバーまたは Sun iSNS サーバーが含まれている場合は、Solaris iSCSI ターゲットの iSNS 発見を設定できます。

この手順では、ユーザーが現在ログインしているローカルシステムから構成済みの iSCSI ターゲットデバイスにアクセスするものとします。

- 1 スーパーユーザーになります。
- 2 iSNS サーバーの情報を追加します。

例:

```
initiator# iscsitadm modify admin --isns-server ip-address or hostname[:port]
```

ネットワーク内の iSNS サーバーの *ip-address* を特定します。

この手順により、iSNS サーバーの情報がすべての Solaris iSCSI ターゲットに追加されます。

### 3 iSNS サーバー検出を有効にします。

例:

```
initiator# iscsitadm modify admin --isns-access enable
```

この手順により、すべての Solaris iSCSI ターゲットで iSNS 発見が有効になります。

## iSCSI ディスクにアクセスする方法

リブート時に iSCSI ディスクにアクセスするには、ディスク上にファイルシステムを作成し、SCSI デバイス上の UFS ファイルシステムの場合と同じ方法で、`/etc/vfstab` エントリを追加します。次に、iSCSI イニシエータサービスに依存する、iSCSI ディスクをマウントするための新しい SMF サービスを作成します。詳細は、[266 ページの「リブート時に iSCSI ディスクにアクセスする方法」](#)を参照してください。

Solaris iSCSI イニシエータによってデバイスが発見されると、ログインネゴシエーションが自動的に発生します。Solaris iSCSI ドライバは、利用可能な LUN の個数を判断し、デバイスノードを作成します。この時点で、iSCSI デバイスをほかのすべての SCSI デバイスと同様に扱えます。

ローカルシステム上で iSCSI ディスクを表示するには、`format` ユーティリティを使用します。

次の `format` の出力で、ディスク 2 および 3 は MPxIO の制御下でない iSCSI LUN です。ディスク 21 および 22 は MPxIO の制御下にある iSCSI LUN です。

```
initiator# format
AVAILABLE DISK SELECTIONS:
 0. c0t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@8,600000/SUNW,qlc@4/fp@0,0/ssd@w500000e010685cf1,0
 1. c0t2d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
 /pci@8,600000/SUNW,qlc@4/fp@0,0/ssd@w500000e0106e3ba1,0
 2. c3t0d0 <ABCSTORAGE-100E-00-2.2 cyl 20813 alt 2 hd 16 sec 63>
 /iscsi/disk@0000iqn.2001-05.com.abcstorage%3A6-8a0900-477d70401-
 b0fff044352423a2-hostname-020000,0
 3. c3t1d0 <ABCSTORAGE-100E-00-2.2 cyl 20813 alt 2 hd 16 sec 63>
 /iscsi/disk@0000iqn.2001-05.com.abcstorage%3A6-8a0900-3fcd70401-
 -085ff04434f423a2-hostname-010000,0
 .
 .
 .
 21. c4t60A98000686F694B2F59775733426B77d0 <ABCSTORAGE-LUN-0.2 cyl
 4606 alt 2 hd 16 sec 256>
 /scsi_vhci/ssd@g60a98000686f694b2f59775733426b77
 22. c4t60A98000686F694B2F59775733434C41d0 <ABCSTORAGE-LUN-0.2 cyl
 4606 alt 2 hd 16 sec 256>
 /scsi_vhci/ssd@g60a98000686f694b2f59775733434c41
```

## ▼ リブート時に iSCSI ディスクにアクセスする方法

システムのリブート後に iSCSI ディスクにアクセスするには、次の手順に従います。

### 1 スーパーユーザーになるか、同等の役割を引き受けます。

- 2 iSCSI LUN のエントリを `/etc/vfstab` ファイルに追加します。 `mount at boot` オプションを `iscsi` に設定します。

```
initiator# vi /etc/vfstab
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
/dev/dsk/c3t600144F04B555F370000093D00495B00d0s0 - /mnt ufs - no -
```

## ▼ 発見された iSCSI ターゲットを削除する方法

発見アドレス、iSNS サーバー、または静的構成を削除するか、あるいは発見方式を無効にしたあとで、関連付けられたターゲットがログアウトします。たとえば、これらの関連付けられたターゲットが引き続き使用され、ファイルシステムがマウントされている場合は、これらのデバイスのログアウトが失敗し、アクティブターゲットリスト上に残ります。

この省略可能な手順では、ユーザーが現在ログインしているローカルシステム上で、iSCSI ターゲットデバイスへのアクセスがすでに構成されているものとします。

- 1 スーパーユーザーになります。
- 2 (省略可能) 次のいずれかを使って iSCSI ターゲット発見方式を無効にします。
  - SendTargets 発見方式を無効にする必要がある場合は、次のコマンドを使用します。
 

```
initiator# iscsiadm modify discovery --sendtargets disable
```
  - iSNS 発見方式を無効にする必要がある場合は、次のコマンドを使用します。
 

```
initiator# iscsiadm modify discovery --iSNS disable
```
  - 静的ターゲット発見方式を無効にする必要がある場合は、次のコマンドを使用します。
 

```
initiator# iscsiadm modify discovery --static disable
```
- 3 次のいずれかを使って iSCSI デバイス発見エントリを削除します。
  - iSCSI SendTargets 発見エントリを削除します。  
例:
 

```
initiator# iscsiadm remove discovery-address 10.0.0.1:3260
```
  - iSCSI iSNS 発見エントリを削除します。  
例:
 

```
iscsiadm remove isns-server 10.0.0.1:3205
```
  - 静的 iSCSI 発見エントリを削除します。  
例:
 

```
initiator# iscsiadm remove static-config eui.5000ABCD78945E2B,10.0.0.1
```

---

注-使用中の論理ユニットが関連付けられた発見エントリを無効化または削除しようとすると、次のメッセージが表示され、無効化または削除が失敗します。

logical unit in use

このエラーが発生した場合は、論理ユニット上の関連付けられたすべての入出力を停止し、ファイルシステムのマウント解除などを行います。そのあとで無効化または削除の操作を再び実行します。

---

#### 4 iSCSI ターゲットデバイスを削除します。

論理ユニット番号 (LUN) を指定して、ターゲットを削除します。ターゲットの作成時に LUN を指定しなかった場合、値 0 が使用されます。ターゲットに複数の LUN が関連付けられている場合は、LUN 0 を最後に削除する必要があります。

例:

```
initiator# iscsitadm delete target --lun 0 sandbox
```

## iSCSI ベースのストレージネットワークにおける認証の構成

iSCSI デバイスの認証設定はオプションです。

セキュリティ保護された環境では、信頼できるイニシエータだけがターゲットにアクセスできるため、認証は必要ありません。

セキュリティ保護の不十分な環境では、ターゲットは、接続要求が本当に指定されたホストからのものなのかを判断できません。そのような場合、ターゲットは、チャレンジハンドシェーク認証プロトコル (CHAP) を使ってイニシエータを認証できます。

CHAP 認証では「チャレンジ」と「応答」の概念が使用され、つまり、ターゲットがイニシエータに対して身元の証明を要求します。このチャレンジ/応答方式が機能するには、ターゲットがイニシエータの秘密鍵を知っており、かつイニシエータがチャレンジに応答するように設定されている必要があります。秘密鍵をアレイ上に設定する手順については、アレイのベンダーのドキュメントを参照してください。

iSCSI は単方向認証と双方向認証をサポートします。

- 「単方向認証」では、ターゲットがイニシエータの身元を認証できます。
- 「双方向認証」では、イニシエータがターゲットの身元を認証できるようにすることで、二次レベルのセキュリティを追加します。

## ▼ iSCSI イニシエータの CHAP 認証を構成する方法

この手順では、ユーザーが現在ログインしているローカルシステムから構成済みの iSCSI ターゲットデバイスに安全にアクセスするものとします。

- 1 スーパーユーザーになります。
- 2 単方向 CHAP または双方向 CHAP のどちらを構成するかを決定します。
  - 単方向認証 (デフォルトの方式) では、ターゲットがイニシエータを検証できます。手順 3 から 5 のみを完了してください。
  - 双方向認証では、二次レベルのセキュリティを追加する目的で、イニシエータがターゲットを認証することができます。手順 3 から 9 を完了してください。

- 3 単方向 CHAP - イニシエータ上で秘密鍵を設定します。

たとえば、次のコマンドを実行すると、CHAP の秘密鍵を定義するためのダイアログが起動されます。

```
initiator# iscsiadm modify initiator-node --CHAP-secret
```

---

注 - CHAP シークレットの長さは 12 文字 - 16 文字である必要があります。

---

- 4 (省略可能) 単方向 CHAP - イニシエータ上で CHAP 名を設定します。

デフォルトではイニシエータの CHAP 名は、イニシエータのノード名に設定されます。

次のコマンドを使用して、イニシエータの CHAP 名を変更できます。

```
initiator# iscsiadm modify initiator-node --CHAP-name new-CHAP-name
```

Solaris 環境では、CHAP 名はデフォルトで常にイニシエータノード名に設定されます。CHAP 名は、512 バイト未満の任意の長さのテキストに設定できます。512 バイトの長さの制限は Solaris の制限です。ただし、CHAP 名を設定しない場合は、初期化のときにイニシエータノード名に設定されます。

- 5 単方向 CHAP - シークレットの設定完了後にイニシエータ上で CHAP 認証を有効にします。

```
initiator# iscsiadm modify initiator-node --authentication CHAP
```

CHAP 認証では、イニシエータノードにユーザー名とパスワードが必要です。ユーザー名は通常、渡されたユーザー名のシークレットをターゲットが検索するために使用されます。

- 6 次のいずれかを選択して双方向 CHAP を有効または無効にします。

- 双方向 CHAP - ターゲットの双方向認証パラメータを有効にします。

例:

```
initiator# iscsiadm modify target-param -B enable eui.5000ABCD78945E2B
```

- 双方向 CHAP を無効にします。例:

```
initiator# iscsiadm modify target-param -B disable eui.5000ABCD78945E2B
```

- 7 双方向 CHAP - ターゲット上で認証方法を CHAP に設定します。

例:

```
initiator# iscsiadm modify target-param --authentication CHAP eui.5000ABCD78945E2B
```

- 8 双方向 CHAP - ターゲット上でターゲットデバイスの秘密鍵を設定します。  
たとえば、次のコマンドを実行すると、CHAP の秘密鍵を定義するためのダイアログが起動されます。

```
initiator# iscsiadm modify target-param --CHAP-secret eui.5000ABCD78945E2B
```

- 9 双方向 CHAP - ターゲット上で CHAP 名を設定します。  
デフォルトでは、ターゲットの CHAP 名はターゲット名に設定されます。  
次のコマンドを使用して、ターゲットの CHAP 名を変更できます。

```
initiator# iscsiadm modify target-param --CHAP-name target-CHAP-name
```

## ▼ iSCSI ターゲットの CHAP 認証を構成する方法

この手順では、iSCSI ターゲットのあるローカルシステムにユーザーがログインしているものとします。

- 1 スーパーユーザーになります。
- 2 ターゲットの CHAP シークレット名を設定します。  
規則では、ホスト名をシークレット名として使用します。例:

```
target# iscsitadm modify admin -H stormpike
```

- 3 CHAP シークレットを指定します。  
CHAP シークレットは、12 - 16 文字にする必要があります。例:

```
target# iscsitadm modify admin -C
Enter secret: xxxxxx
Re-enter secret: xxxxxx
```

- 4 1 つ以上のターゲットに関連付けられるイニシエータオブジェクトを作成します。  
この手順は、毎回 IQN 値を入力しなくても済むように、わかりやすい名前 (通常はホスト名、この場合は monster620) を IQN 値に関連付けるために行います。例:

```
iscsitadm create initiator -n iqn.1986-03.com.sun: 01:00e081553307.4399f40e monster620
```

- 5 イニシエータで使ったのと同じ **CHAP** 名を指定します。  
この名前は、イニシエータオブジェクトに使ったわかりやすい名前でもかまいません。例:  

```
target# iscsitadm modify initiator -H monster620 monster620
```
- 6 イニシエータで使ったのと同じ **CHAP** シークレットを指定します。  
例:  

```
target# iscsitadm modify initiator -C monster620
Enter secret: xxxxxx
Re-enter secret: xxxxxx
```
- 7 イニシエータオブジェクトを1つ以上のターゲットに関連付けます。  
例:  

```
target# iscsitadm modify target -l monster620 sandbox
```

## 他社製の **RADIUS** サーバーを使用して **iSCSI** 構成内の **CHAP** 管理を単純化する

他社製の **RADIUS** サーバーを使用すると、**CHAP** シークレット管理を単純化できます。**RADIUS** サーバーは集中管理認証サービスです。**RADIUS** サーバーを使って双方向認証を行う場合、イニシエータの **CHAP** シークレットは依然として指定する必要がありますが、各イニシエータ上で各ターゲットの **CHAP** シークレットを指定する必要はなくなります。

詳細については、次を参照してください。

- **CHAP** – <http://www.ietf.org/rfc/rfc1994.txt>
- **RADIUS** – <http://www.ietf.org/rfc/rfc2865.txt>

### ▼ **iSCSI** 構成の **RADIUS** を構成する方法

この手順では、ユーザーが現在ログインしているローカルシステムから構成済みの **iSCSI** ターゲットデバイスに安全にアクセスするものとします。

- 1 スーパーユーザーになります。
- 2 **RADIUS** サーバーの IP アドレスとポート (デフォルトのポートは **1812**) を、イニシエータノードに構成します。  
例:  

```
initiator# iscsiadm modify initiator-node --radius-server 10.0.0.72:1812
```
- 3 **RADIUS** サーバーの共有鍵をイニシエータノードに構成します。  

```
initiator# iscsiadm modify initiator-node --radius-shared-secret
```



---

注 - Solaris iSCSI 実装の場合、RADIUS サーバーに共有シークレットが構成されていないと、Solaris iSCSI ソフトウェアは RADIUS サーバーと通信できません。

---

#### 4 RADIUS サーバーを有効にします。

```
initiator# iscsiadm modify initiator-node --radius-access enable
```

### Solaris iSCSI と RADIUS サーバーに関するエラーメッセージ

このセクションでは、Solaris iSCSI と RADIUS サーバーの構成に関するエラーメッセージとその考えられる解決法について説明します。

empty RADIUS shared secret

原因: イニシエータ上で RADIUS サーバーが有効になっているにもかかわらず、RADIUS の共有シークレットが設定されていません。

対処方法: RADIUS の共有シークレットをイニシエータに構成します。詳細は、[271 ページの「iSCSI 構成の RADIUS を構成する方法」](#)を参照してください。

WARNING: RADIUS packet authentication failed

原因: イニシエータによる RADIUS データパケットの認証が失敗しました。このエラーが発生する可能性があるのは、イニシエータノード上に構成された共有シークレットが RADIUS サーバー上の共有シークレットと異なっている場合です。

正しい RADIUS 共有シークレットをイニシエータに構成し直します。詳細は、[271 ページの「iSCSI 構成の RADIUS を構成する方法」](#)を参照してください。

## Solaris iSCSI マルチパスデバイスの設定

Solaris iSCSI マルチパス (MPxIO) デバイスを使用するときは、次のガイドラインを考慮してください。

- **Solaris iSCSI** および **MPxIO** - MPxIO は、iSCSI イニシエータの MS/T (Multiple Sessions per Target) を構成する Solaris iSCSI 構成での、ターゲットポート集約と可用性をサポートします。
  - 複数の NIC を集約およびフェイルオーバーする場合は、IPMP を使用してください。
  - iSCSI ホストの基本的な構成は、iSCSI トラフィック専用の 2 つの NIC を備えたサーバーです。NIC は IPMP を使用して構成されます。パフォーマンスを最適化するために、iSCSI 以外のトラフィック用に追加の NIC が用意されます。
  - アクティブマルチパスは、Solaris iSCSI MS/T 機能、および IPMP 構成のフェイルオーバーと冗長性を使用する場合のみ実現できます。



- IPMP 構成で 1 つの NIC に障害が発生すると、IPMP はフェイルオーバーの処理を行います。MPxIO ドライバは障害を検出しません。IPMP 以外の構成では、MPxIO ドライバは障害状態になり、バスがオフラインになります。
- IPMP 構成で 1 つのターゲットポートに障害が発生すると、MPxIO ドライバは障害を検出し、フェイルオーバーの処理を行います。IPMP 以外の構成では、MPxIO ドライバは障害を検出し、フェイルオーバーの処理を行います。
- MS/T の構成方法については、273 ページの「ターゲットの複数の iSCSI セッションを有効にする方法」を参照してください。IPMP の構成については、『Oracle Solaris の管理: IP サービス』のパート V 「IPMP」を参照してください。
- **Solaris iSCSI**、ファイバチャネル (FC)、および **MPxIO** – より複雑な iSCSI/FC 構成では、MPxIO ドライバは次のように動作します。
  - FC SAN にデュアル iSCSI-FC ブリッジが存在する場合、iSCSI はターゲットバスを MPxIO に提示します。MPxIO は、一意の SCSI/LUN 識別子を照合し、それらが同一である場合は 1 つのバスを iSCSI ドライバに提示します。
  - iSCSI と FC の両方を使用してターゲットを接続する構成の場合、MPxIO ドライバは同じデバイスに対して異なるトランスポートを提供できます。この構成では、MPxIO は両方のバスを使用します。
  - iSCSI と FC を MPxIO と組み合わせて使用する場合は、`/kernel/drv/fp.conf` ファイルおよび `/kernel/drv/iscsi.conf` ファイルの MPxIO の設定が、サポートしたい MPxIO 構成と一致していることを確認します。たとえば、`fp.conf` では、MPxIO を HBA 全体で有効にするのか、ポート単位で有効にするのかを決定できます。
- 他社製ハードウェアに関する注意事項 – 他社製の HBA が Solaris iSCSI および MPxIO での使用に適しているかどうかを確認します。  
 他社製の HBA を使用する場合は、その他社製 HBA のベンダーに `/kernel/drv/scsi_vhci.conf` ファイルの対称オプション情報を問い合わせる必要がある場合があります。

## ▼ ターゲットの複数の iSCSI セッションを有効にする方法

この手順を使用して、単一のターゲットに接続する複数の iSCSI セッションを作成できます。このシナリオは、ログインのリダイレクションをサポートするか、または同じターゲットポータルグループに含まれている複数のターゲットのポータルを使用する iSCSI ターゲットデバイスの場合に役に立ちます。Solaris SCSI マルチパス

(MPxIO) を利用して、ターゲットごとに複数の iSCSI セッションを使用します。また、ホスト側の複数の NIC を使用して同じターゲット上の複数のポータルに接続すると、より広い帯域幅を実現できます。

MS/T 機能は、イニシエータのセッション ID (ISID) を変えることで、ターゲット上に 2 つ以上のセッションを作成します。この機能を有効にすると、ネットワークに 2 つの SCSI レイヤーパスが作成され、複数のターゲットを iSCSI レイヤーから Solaris I/O レイヤーまで公開できるようになります。MPxIO ドライバは、これらのパスに対する予約を処理します。

iSCSI と MPxIO パスの相互動作の仕組みについては、[272 ページの「Solaris iSCSI マルチパスデバイスの設定」](#)を参照してください。

iSCSI ターゲットの複数セッションを構成する前に、次の項目を確認してください。

- 通常の MS/T 構成には、2 つ以上の構成済みセッションがあります。  
ただし、ストレージが複数の TPGT をサポートしていて、ホストシステム上で SendTarget 発見を使用している場合には、構成済みセッションの数を 1 に設定できます。SendTarget 発見によって、複数のパスが存在することが自動的に検出されて複数のターゲットセッションが作成されます。
- /kernel/drv/iscsi.conf ファイルで mxpio 構成パラメータが有効になっていることを確認します。
- IPMP を使用して複数のネットワーク接続が構成されていることを確認します。
- 複数のネットワーク接続が使用可能であることを確認します。

```
ifconfig -a
```

- 1 スーパーユーザーになります。
- 2 iSCSI イニシエータとターゲットの現在のパラメータを一覧表示します。

a. iSCSI イニシエータの現在のパラメータを一覧表示します。例:

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
.
.
.
Configured Sessions: 1
```

b. iSCSI ターゲットデバイスの現在のパラメータを一覧表示します。例:

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
```

```

.
.
.
Configured Sessions: 1

```

Configured Sessions (構成済みのセッション) の値は、ターゲットポータルグループ内の各ターゲット名用に作成される構成済みの iSCSI セッションの数です。

- 3 次のいずれかを選択し、構成済みのセッションの数をイニシエータノードで変更してすべてのターゲットに適用するか、ターゲットレベルで変更して特定のターゲットに適用します。

ターゲットのセッション数は1から4の間である必要があります。

- iSCSI イニシエータノードにパラメータを適用します。

例:

```
initiator# iscsiadm modify initiator-node -c 2
```

- iSCSI ターゲットにパラメータを適用します。

例:

```
initiator# iscsiadm modify target-param -c 2 iqn.1992-08.com.abcstorage:sn.84186266
```

- 構成済みセッションを1つ以上のローカル IP アドレスにバインドします。

構成済みのセッションは、特定のローカル IP アドレスにバインドすることもできます。この方法を使用する場合は、コンマ区切りのリストで1つ以上のローカル IP アドレスを指定します。各 IP アドレスは iSCSI セッションを表します。この方法は、initiator-node または target-param のレベルでも実行できます。例:

```
initiator# iscsiadm modify initiator-node -c 10.0.0.1,10.0.0.2
```

---

注-指定した IP アドレスがルーティング可能ではない場合、アドレスが無視され、デフォルトの Solaris ルートおよび IP アドレスがこのセッションで使用されます。

---

- 4 パラメータが変更されたことを確認します。

- a. イニシエータノードの更新された情報を表示します。例:

```

initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
.
.
.
Configured Sessions: 2

```

- b. ターゲットノードの更新された情報を表示します。例:

```

initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -

```

```

.
.
.
Configured Sessions: 2

```

- 5 **mpathadm list lu** コマンドを使用して複数パスを一覧表示し、OS デバイス名が **iscsiadm list** の出力に一致していることと、パスの数が 2 以上であることを確認します。

## iSCSI 構成の監視

**iscsiadm list** コマンドを使用して、iSCSI イニシエータとターゲットデバイスに関する情報を表示できます。

### ▼ iSCSI 構成の監視

- 1 スーパーユーザーになります。
- 2 iSCSI イニシエータに関する情報を表示します。

例:

```

iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzrl200
 Login Parameters (Default/Configured):
 Header Digest: NONE/-
 Data Digest: NONE/-
 Authentication Type: NONE
 RADIUS Server: NONE
 RADIUS access: unknown
 Configured Sessions: 1

```

- 3 使用中の発見方式に関する情報を表示します。

例:

```

iscsiadm list discovery
Discovery:
 Static: enabled
 Send Targets: enabled
 iSNS: enabled

```

#### 例 12-1 iSCSI ターゲットの情報の表示

次の例では、特定の iSCSI ターゲットのパラメータ設定を表示する方法を示します。

```

iscsiadm list target-param iqn.1992-08.com.abcstorage:sn.33592219
Target: iqn.1992-08.com.abcstorage:sn.33592219

```

iscsiadm list target-param -v コマンドによって次の情報が表示されます。

- ターゲットの認証設定
- ターゲットのログインパラメータのデフォルト設定
- 各ログインパラメータに構成されている値

iscsiadm list target-param -v コマンドでは、/ 指定子の前にデフォルトのパラメータ値が表示され、/ 指定子の後に構成後のパラメータ値が表示されます。パラメータを構成していない場合は、パラメータ値がハイフン (-) で表示されます。詳細は、次の例を参照してください。

```
iscsiadm list target-param -v eui.50060e8004275511 Target: eui.50060e8004275511
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
Login Parameters (Default/Configured):
 Data Sequence In Order: yes/-
 Data PDU In Order: yes/-
 Default Time To Retain: 20/-
 Default Time To Wait: 2/-
 Error Recovery Level: 0/-
 First Burst Length: 65536/-
 Immediate Data: yes/-
 Initial Ready To Transfer (R2T): yes/-
 Max Burst Length: 262144/-
 Max Outstanding R2T: 1/-
 Max Receive Data Segment Length: 65536/-
 Max Connections: 1/-
 Header Digest: NONE/-
 Data Digest: NONE/-
Configured Sessions: 1
```

次の例では、ターゲットとイニシエータの間でネゴシエートされたあとのパラメータが出力されています。

```
iscsiadm list target -v eui.50060e8004275511
Target: eui.50060e8004275511
 TPGT: 1
 ISID: 4000002a0000
 Connections: 1
 CID: 0
 IP address (Local): 172.90.101.71:32813
 IP address (Peer): 172.90.101.40:3260
 Discovery Method: Static
 Login Parameters (Negotiated):
 Data Sequence In Order: yes
 Data PDU In Order: yes
 Default Time To Retain: 0
 Default Time To Wait: 3
 Error Recovery Level: 0
 First Burst Length: 65536
 Immediate Data: yes
 Initial Ready To Transfer (R2T): yes
 Max Burst Length: 262144
 Max Outstanding R2T: 1
```

Max Receive Data Segment Length: 65536  
Max Connections: 1  
Header Digest: NONE  
Data Digest: NONE

## iSCSI イニシエータと iSCSI ターゲットのパラメータの変更

iSCSI イニシエータおよび iSCSI ターゲットデバイスのどちらでもパラメータは変更可能です。ただし、iSCSI イニシエータ上で変更可能なパラメータは、次のものだけです。

- iSCSI イニシエータのノード名 – イニシエータのノード名を別の名前に変更できます。イニシエータのノード名を変更した場合は、名前を変更したときの iSNS サーバー上の発見ドメインインストールの構成に応じて、iSNS によって発見されたターゲットがイニシエータのターゲットリストから削除されることがあります。詳細は、[279 ページの「iSCSI イニシエータと iSCSI ターゲットのパラメータを変更する方法」](#)を参照してください。
- ヘッダーダイジェスト – NONE、デフォルト値、または CRC32。
- データダイジェスト – NONE、デフォルト値、または CRC32。
- 認証と CHAP シークレット – 認証の設定の詳細は、[270 ページの「iSCSI ターゲットの CHAP 認証を構成する方法」](#)を参照してください。
- 構成済みセッション – 複数セッションの構成方法については、[273 ページの「ターゲットの複数の iSCSI セッションを有効にする方法」](#)を参照してください。

iSCSI ドライバは、iSCSI イニシエータと iSCSI ターゲットデバイスのパラメータのデフォルト値を提供します。iSCSI イニシエータのパラメータを変更すると、iSCSI ターゲットデバイスにすでに別の値が設定されている場合以外、その変更後のパラメータが iSCSI ターゲットデバイスに継承されます。



注意 – 変更対象のパラメータがターゲットソフトウェアによってサポートされていることを確認してください。サポートされていない場合、iSCSI ターゲットデバイスにログインできない可能性があります。サポートされているパラメータの一覧については、使用するアレイのドキュメントを参照してください。

---

iSCSI パラメータの変更は、イニシエータとターゲット間の I/O が完了したあとで行うようにしてください。iscsiadm modify コマンドを使って変更が実行されると、iSCSI ドライバはセッションを接続し直します。

## ▼ iSCSI イニシエータと iSCSI ターゲットのパラメータを変更する方法

この手順の前半では、iSCSI イニシエータの変更されたパラメータが iSCSI ターゲットデバイスにどのように継承されるかを示します。この手順の後半では、iSCSI ターゲットデバイス上のパラメータを実際に変更する方法を示します。

この省略可能な手順では、ユーザーが現在ログインしているローカルシステム上で、iSCSI ターゲットデバイスへのアクセスがすでに構成されているものとします。

- 1 スーパーユーザーになります。
- 2 iSCSI イニシエータとターゲットデバイスの現在のパラメータを一覧表示します。

a. iSCSI イニシエータの現在のパラメータを一覧表示します。例:

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzrl200
 Login Parameters (Default/Configured):
 Header Digest: NONE/-
 Data Digest: NONE/-
 Authentication Type: NONE
 RADIUS Server: NONE
 RADIUS access: unknown
 Configured Sessions: 1
```

b. iSCSI ターゲットデバイスの現在のパラメータを一覧表示します。例:

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
 Login Parameters (Default/Configured):
 Data Sequence In Order: yes/-
 Data PDU In Order: yes/-
 Default Time To Retain: 20/-
 Default Time To Wait: 2/-
 Error Recovery Level: 0/-
 First Burst Length: 65536/-
 Immediate Data: yes/-
 Initial Ready To Transfer (R2T): yes/-
 Max Burst Length: 262144/-
 Max Outstanding R2T: 1/-
 Max Receive Data Segment Length: 65536/-
 Max Connections: 1/-
 Header Digest: NONE/-
 Data Digest: NONE/-
 Configured Sessions: 1
```

iSCSI イニシエータと iSCSI ターゲットデバイスの両方で、ヘッダーダイジェストとデータダイジェストのパラメータがどちらも現在 NONE に設定されています。

iSCSI ターゲットデバイスのデフォルトパラメータを確認するには、Example 12-1 の例 12-1 出力を参照してください。

**3 iSCSI イニシエータのパラメータを変更します。**

たとえば、ヘッダーダイジェストを CRC32 に設定します。

```
initiator# iscsiadm modify initiator-node -h CRC32
```

イニシエータのノード名を変更した場合は、新しい名前がターゲットと同じ発見ドメインに属していないと、iSNS によって発見されたターゲットがログアウトおよび削除されることがあります。ただし、ターゲットが使用中の場合は、削除されません。たとえば、これらのターゲット上でファイルが開いている場合またはファイルシステムがマウントされている場合は、それらのターゲットは削除されません。

これらのターゲットと新しいイニシエータノードの名前が同じ発見ドメインに属している場合は、名前の変更後に新しいターゲットも表示されることがあります。

**4 パラメータが変更されたことを確認します。**

**a. iSCSI イニシエータの更新済みパラメータ情報を表示します。例:**

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzrl200
 Login Parameters (Default/Configured):
 Header Digest: NONE/CRC32
 Data Digest: NONE/-
 Authentication Type: NONE
 RADIUS Server: NONE
 RADIUS access: unknown
 Configured Sessions: 1
```

ヘッダーダイジェストは CRC32 に設定されています。

**b. iSCSI ターゲットデバイスの更新済みパラメータ情報を表示します。例:**

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
Login Parameters (Default/Configured):
 Data Sequence In Order: yes/-
 Data PDU In Order: yes/-
 Default Time To Retain: 20/-
 Default Time To Wait: 2/-
 Error Recovery Level: 0/-
 First Burst Length: 65536/-
 Immediate Data: yes/-
 Initial Ready To Transfer (R2T): yes/-
 Max Burst Length: 262144/-
 Max Outstanding R2T: 1/-
 Max Receive Data Segment Length: 65536/-
 Max Connections: 1/-
 Header Digest: CRC32/-
```



```
Data Digest: NONE/-
Configured Sessions: 1
```

ヘッダーダイジェストはCRC32に設定されています。

- 5 iSCSI イニシエータが iSCSI ターゲットに接続し直されたことを確認します。例:

```
initiator# iscsiadm list target -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
TPGT: 2
ISID: 4000002a0000
Connections: 1
CID: 0
IP address (Local): nnn.nn.nn.nnn:64369
IP address (Peer): nnn.nn.nn.nnn:3260
Discovery Method: SendTargets
Login Parameters (Negotiated):
.
.
.
Header Digest: CRC32
Data Digest: NONE
```

- 6 (オプション) iSCSI イニシエータまたは iSCSI ターゲットデバイスのパラメータを設定解除します。

`iscsiadm modify` コマンドを使用して、パラメータをそのデフォルト設定に戻すことで設定解除できます。または、`iscsiadm remove` を使用して、ターゲットのプロパティをすべてデフォルト設定にリセットすることができます。

`iscsiadm modify target-param` コマンドは、コマンド行で指定されたパラメータのみを変更します。

次の例では、ヘッダーダイジェストを NONE にリセットする方法を示します。

```
initiator# iscsiadm modify target-param -h none iqn.1992-08.com.abcstorage:sn...
```

`iscsiadm remove target-param` コマンドについては、[iscsiadm\(1M\)](#) のマニュアルページを参照してください。

## iSCSI 構成に関する問題のトラブルシューティング

一般的な iSCSI 構成に関する問題のトラブルシューティングを行う際に利用可能なツールを、次に示します。

- snoop – このツールは更新され、iSCSI パケットをサポートするようになりました。
- Wireshark – この製品は <http://www.wireshark.org/> から入手できます。

どちらのツールも、ポート 3260 上の iSCSI パケットをフィルタできます。

以降のセクションでは、iSCSI のさまざまなトラブルシューティングやエラーメッセージ解決シナリオについて説明します。

# ローカルシステムから iSCSI ターゲットに接続できない

## ▼ iSCSI の接続に関する問題のトラブルシューティングを行う方法

- 1 スーパーユーザーになります。
- 2 iSCSI ターゲットの情報を一覧表示します。

例:

```
initiator# iscsiadm list target
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
TPGT: default
ISID: 4000002a0000
Connections: 0
```

- 3 **iscsiadm list target** の出力結果に接続が表示されていない場合は、接続が失敗したことについて、**/var/adm/messages** ファイル内で考えられる理由を調査します。

また、ping コマンドを使用するか、または、telnet コマンドを使ってストレージデバイスの iSCSI ポートに接続し、iSCSI サービスが利用可能かどうかを確認することによって、接続がアクセス可能かどうかを確認することもできます。デフォルトのポートは 3260 です。

さらに、ストレージデバイスのログファイルでエラーを確認します。

- 4 **iscsiadm list target** の出力結果に目的のターゲットが表示されていない場合は、**/var/adm/messages** ファイル内でその考えられる原因を調査します。

SendTargets を発見方式として使用している場合は、**-v** オプションを使って **discovery-address** の一覧表示を試みると、ホストが目的のターゲットを認識しているかどうかわかります。例:

```
initiator# iscsiadm list discovery-address -v 10.0.0.1
Discovery Address: 10.0.0.1:3260
 Target name: eui.210000203787dfc0
 Target address: 10.0.0.1:11824
 Target name: eui.210000203787e07b
 Target address: 10.0.0.1:11824
```

発見方式として iSNS を使用している場合は、iSNS 発見方式を有効にし、**-v** オプションを使用して **-isns-server** を一覧表示することで、期待されるターゲットがホストに認識されることを確認します。例:

```
initiator# iscsiadm list isns-server -v
iSNS Server IP Address: 10.20.56.56:3205
 Target name: iqn.1992-08.com.xyz:sn.1234566
 Target address: 10.20.57.161:3260, 1
 Target name: iqn.2003-10.com.abc:group-0:154:abc-65-01
 Target address: 10.20.56.206:3260, 1
```

```
Target name: iqn.2003-10.com.abc:group-0:154:abc-65-02
Target address: 10.20.56.206:3260, 1
```

## ローカルシステム上で iSCSI デバイスまたは iSCSI ディスクが利用できない

### ▼ iSCSI デバイスまたは iSCSI ディスクが利用できない問題のトラブルシューティングを行う方法

- 1 スーパーユーザーになります。
- 2 列挙中にこのターゲット上で発見された LUN を確認します。

例:

```
iscsiadm list target -S
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
TPGT: default
ISID: 4000002a0000
Connections: 1
LUN: 0
 Vendor: ABCSTOR
 Product: 0010
 OS Device Name: /dev/rdisk/c3t34d0s2
```

-s オプションを指定すると、列挙中にこのターゲット上で発見された LUN が表示されます。表示されるはずの LUN が表示されない場合は、/var/adm/messages ファイルを確認して、エラーが報告されたかどうかを調べます。ストレージデバイスのログファイルにエラーが記録されていないか確認します。また、ストレージデバイスのすべての LUN マスクが正しく構成されていることも確認します。

## iSNS 発見方式を使用する場合の LUN マスクの使用

特定のイニシエータに対するストレージの認証を制御する手段として iSNS 発見ドメインを使用しないでください。承認されたイニシエータだけが LUN にアクセスできるようにするには、代わりに LUN マスクを使用します。

ターゲットが使用中のときに発見ドメインからターゲットを削除しようとした場合、iSCSI イニシエータはこのターゲットからログアウトしません。このイニシエータがこのターゲット(および関連付けられた LUN)にアクセスしないようにする場合は、LUN マスクを使用する必要があります。ターゲットを発見ドメインから削除するだけでは不十分です。

## iSCSI の一般的なエラーメッセージ

このセクションでは、`/var/adm/messages` ファイルに見つかる可能性のある iSCSI メッセージと、回復のために適用できる解決方法について説明します。

メッセージの形式は次のとおりです。

`iscsi TYPE (OID) STRING (STATUS-CLASS#/STATUS-DETAIL#)`

**TYPE** 接続 (connection)、セッション (session) のいずれかです。

**OID** 接続またはセッションのオブジェクト ID です。この ID は、OS インスタンスに一意です。

**STRING** 状態の説明です。

**STATUS-CLASS#/STATUS-DETAIL#** これらの値は、RFC 3720 で定義されている iSCSI ログイン応答として返されます。

`iscsi connection(OID) login failed - Miscellaneous iSCSI initiator errors.`

原因: 何らかのイニシエータエラーにより、デバイスログインが失敗しました。

`iscsi connection(OID) login failed - Initiator could not be successfully authenticated.`

原因: デバイスによるイニシエータの認証が成功しませんでした。

対処方法: 該当する場合には、CHAP 名、CHAP パスワード、または RADIUS サーバーの設定が正しいことを確認します。

`iscsi connection(OID) login failed - Initiator is not allowed access to the given target.`

原因: イニシエータが iSCSI ターゲットデバイスにアクセスすることを、デバイスが許可できません。

対処方法: イニシエータ名を確認するとともに、その名前がストレージデバイスによって正しくマスクまたはプロビジョニングされていることを確認します。

`iscsi connection(OID) login failed - Requested ITN does not exist at this address.`

原因: ユーザーが要求する iSCSI ターゲット名 (ITN) へのアクセスを、デバイスが提供しません。

対処方法: イニシエータの発見情報が正しく指定されており、ストレージデバイスが正しく構成されていることを確認します。

`iscsi connection(OID) login failed - Requested ITN has been removed and no forwarding address is provided.`

原因: ユーザーが要求する iSCSI ターゲット名 (ITN) へのアクセスを、デバイスが提供しなくなりました。

対処方法: イニシエータの発見情報が正しく入力されており、ストレージデバイスが正しく構成されていることを確認します。

`iscsi connection(OID) login failed - Requested iSCSI version range is not supported by the target.`

原因: イニシエータの iSCSI バージョンがストレージデバイスによってサポートされていません。

`iscsi connection(OID) login failed - No more connections can be accepted on this Session ID (SSID).`

原因: ストレージデバイスは、このイニシエータノードから iSCSI ターゲットデバイスへの接続をこれ以上受け入れることができません。

`iscsi connection(OID) login failed - Missing parameters (e.g., iSCSI initiator and/or target name).`

原因: ストレージデバイスが、イニシエータ名またはターゲット名が正しく指定されていないと報告しています。

対処方法: iSCSI のイニシエータ名またはターゲット名を正しく指定します。

`iscsi connection(OID) login failed - Target hardware or software error.`

原因: ストレージデバイスでハードウェアエラーまたはソフトウェアエラーが発生しました。

対処方法: ストレージのドキュメントを参照するか、ストレージのベンダーに連絡して必要なサポートを受けます。

`iscsi connection(OID) login failed - iSCSI service or target is not currently operational.`

原因: ストレージデバイスが現在動作していません。

対処方法: ストレージのドキュメントを参照するか、ストレージのベンダーに連絡して必要なサポートを受けます。

`iscsi connection(OID) login failed - Target has insufficient session, connection or other resources.`

原因: ストレージデバイスのリソースが不足しています。

対処方法: ストレージのドキュメントを参照するか、ストレージのベンダーに連絡して必要なサポートを受けます。

`iscsi connection(OID) login failed - unable to initialize authentication`

`iscsi connection(OID) login failed - unable to set authentication`

`iscsi connection(OID) login failed - unable to set username`

`iscsi connection(OID) login failed - unable to set password`

`iscsi connection(OID) login failed - unable to set ipsec`

`iscsi connection(OID) login failed - unable to set remote authentication`

原因: イニシエータが認証の初期化または設定を正しく行えませんでした。

対処方法: イニシエータの認証が正しく構成されていることを確認します。

`iscsi connection(OID) login failed - unable to make login pdu`

原因: イニシエータが、イニシエータまたはストレージデバイスの設定に基づいてログインのペイロードデータユニット (PDU) を作成できませんでした。

対処方法: 任意のターゲットログインパラメータやその他のデフォルト以外の設定値のリセットを試みます。

`iscsi connection(OID) login failed - failed to transfer login`

`iscsi connection(OID) login failed - failed to receive login response`

原因: イニシエータが、ログインのペイロードデータユニット (PDU) をネットワーク接続経由で転送または受信できませんでした。

対処方法: ネットワーク接続が到達可能であることを確認します。

`iscsi connection(OID) login failed - received invalid login response (OP CODE)`

原因: ストレージデバイスがログインに対して予想外の応答を返しました。

`iscsi connection(OID) login failed - login failed to authenticate with target`

原因: イニシエータがストレージデバイスを認証できませんでした。

対処方法: イニシエータの認証が正しく構成されていることを確認します。

`iscsi connection(OID) login failed - initiator name is required`

原因: どのアクションを実行する場合も、イニシエータ名が構成されている必要があります。

対処方法: イニシエータ名が構成されていることを確認します。

`iscsi connection(OID) login failed - authentication receive failed`

`iscsi connection(OID) login failed - authentication transmit failed`

原因: イニシエータが認証情報を転送または受信できませんでした。

対処方法: 状況に応じて、ストレージデバイスまたはRADIUSサーバーとのネットワーク接続を確認します。

`iscsi connection(OID) login failed - login redirection invalid`

原因: ストレージデバイスが、イニシエータを無効な宛先にリダイレクトしようとしました。

対処方法: ストレージのドキュメントを参照するか、ストレージのベンダーに連絡して必要なサポートを受けます。

`iscsi connection(OID) login failed - target protocol group tag mismatch, expected <TPGT>, received <TPGT>`

原因: イニシエータとターゲットのTPGT(ターゲットポータルグループタグ、Target Portal Group Tag)が一致しません。

対処方法: イニシエータ上またはストレージデバイス上のTPGT発見設定を確認します。

`iscsi connection(OID) login failed - can't accept PARAMETER in security stage`

原因: ログインのセキュリティーフェーズで、デバイスがサポートされていないログインパラメータで応答しました。

対処方法: パラメータ名が参考のため記載されています。ストレージのドキュメントを参照するか、ストレージのベンダーに連絡して必要なサポートを受けます。

`iscsi connection(OID) login failed - HeaderDigest=CRC32 is required, can't accept VALUE`

`iscsi connection(OID) login failed - DataDigest=CRC32 is required, can't accept VALUE`

原因: このターゲットに対して、CRC32に設定されたHeaderDigestまたはDataDigestのみを受け入れるようにイニシエータが構成されています。デバイスは値VALUEを返しました。

対処方法: イニシエータとデバイスのダイジェスト設定に互換性があることを確認します。

`iscsi connection(OID) login failed - HeaderDigest=None is required, can't accept VALUE`

`iscsi connection(OID) login failed - DataDigest=None is required, can't accept VALUE`

原因: このターゲットに対して、NONEに設定されたHeaderDigestまたはDataDigestのみを受け入れるようにイニシエータが構成されています。デバイスは値VALUEを返しました。

対処方法: イニシエータとデバイスのダイジェスト設定に互換性があることを確認します。

`iscsi connection(OID) login failed - can't accept PARAMETER`

原因: イニシエータはこのパラメータをサポートしません。

`iscsi connection(OID) login failed - can't accept MaxOutstandingR2T VALUE`

原因: イニシエータは、記載された *VALUE* の MaxOutstandingR2T を受け入れません。

`iscsi connection(OID) login failed - can't accept MaxConnections VALUE`

原因: イニシエータは、記載された *VALUE* の最大接続数を受け入れません。

`iscsi connection(OID) login failed - can't accept ErrorRecoveryLevel VALUE`

原因: イニシエータは、記載された *VALUE* のエラー回復レベルを受け入れません。

`iscsi session(OID) NAME offline`

原因: このターゲット *NAME* のすべての接続が、削除されたか、または失敗しました。

`iscsi connection(OID) failure - unable to schedule enumeration`

原因: イニシエータがこのターゲットの LUN を列挙できませんでした。

対処方法: LUN の列挙を強制実行するには、`devfsadm -i iscsi` コマンドを実行します。詳細は、[devfsadm\(1M\)](#) のマニュアルページを参照してください。

`iscsi connection(OID) unable to connect to target NAME (errno: ERRNO)`

原因: イニシエータによるネットワーク接続の確立が失敗しました。

対処方法: 接続エラーに関する特定の *ERRNO* については、`/usr/include/sys/errno.h` ファイルを参照してください。



## format ユーティリティー (参照情報)

---

この章では、format ユーティリティーのメニューとコマンドについて説明します。

この章の内容は次のとおりです。

- 289 ページの「format ユーティリティーを使用する上での推奨事項および要件」
- 290 ページの「format のメニューとコマンドの説明」
- 296 ページの「format コマンドへの入力規則」
- 297 ページの「format ユーティリティーのヘルプを利用する」

format ユーティリティーの使用方法の概要については、172 ページの「format ユーティリティー」を参照してください。

### format ユーティリティーを使用する上での推奨事項および要件

format ユーティリティーを使用するには、スーパーユーザーになる必要があります。そうしないと、format ユーティリティーを使用しようとしたときに次のようなエラーメッセージが表示されます。

```
$ format
Searching for disks...done
No permission (or no disks found)!
```

# format のメニューとコマンドの説明

format メニューの内容が次のように表示されます:

```
FORMAT MENU:
 disk - select a disk
 type - select (define) a disk type
 partition - select (define) a partition table
 current - describe the current disk
 format - format and analyze the disk
 fdisk - run the fdisk program (x86 only)
 repair - repair a defective sector
 label - write label to the disk
 analyze - surface analysis
 defect - defect list management
 backup - search for backup labels
 verify - read and display labels
 save - save new disk/partition definitions
 inquiry - show vendor, product and revision
 volname - set 8-character volume name
 !<cmd> - execute <cmd>, then return
 quit
```

format>

次の表に、format ユーティリティーのメインメニューの項目を示します。

表 13-1 format ユーティリティーのメインメニュー項目の説明

メニュー項目	コマンド/メニュー	説明
disk	コマンド	システムのドライブをすべて表示します。あとの操作で使用するディスクを選択することもできます。このディスクは、「現在のディスク」と呼ばれます。
type	コマンド	現在のディスクの製造元とモデルを表示します。認識されているドライブタイプのリストも表示します。SCSI-2 対応ディスクドライブの場合は必ず Auto configure オプションを選択します。
partition	メニュー	スライスの作成および変更を行います。詳細は、 <a href="#">292 ページの「partition メニュー」</a> を参照してください。
current	コマンド	現在のディスクに関する次の情報を表示します。 <ul style="list-style-type: none"><li>■ デバイス名とデバイスタイプ</li><li>■ シリンダ数、代替シリンダ数、ヘッド数、セクター数</li><li>■ 物理デバイス名</li></ul>

表 13-1 format ユーティリティのメインメニュー項目の説明 (続き)

メニュー項目	コマンド/メニュー	説明
format	コマンド	<p>次のいずれかの情報源をこの順番に使用して、現在のディスクをフォーマットします。</p> <ol style="list-style-type: none"> <li>1. format.dat ファイル内の情報</li> <li>2. 自動構成プロセスからの情報</li> <li>3. format.dat エントリが存在しない場合に、プロンプトへの応答として入力する情報</li> </ol> <p>このコマンドは、IDE ディスクには適用できません。IDE ディスクは、あらかじめ製造元でフォーマットされます。</p>
fdisk	メニュー	<p>x86 プラットフォームのみ: fdisk プログラムを実行し、Solaris fdisk パーティションを作成します。</p> <p>fdisk コマンドは、1T バイトを超えるサイズの EFI ラベル付きディスクで使用することはできません。</p>
repair	コマンド	現在のディスク上で特定のブロックを修復します。
label	コマンド	現在のディスクに新しいラベルを書き込みます。
analyze	メニュー	読み取り、書き込み、および比較テストを実行します。詳細は、 <a href="#">294 ページの「analyze メニュー」</a> を参照してください。
defect	メニュー	欠陥リストを検索して表示します。詳細は、 <a href="#">295 ページの「defect メニュー」</a> を参照してください。この機能は、IDE ディスクには適用できません。IDE ディスクは欠陥の検出を自動的に行います。
backup	コマンド	<p><b>VTOC</b> - バックアップラベルを検索します。</p> <p><b>EFI</b> - サポートされません。</p>
verify	コマンド	<p>現在のディスクに関する次の情報を表示します。</p> <ul style="list-style-type: none"> <li>■ デバイス名とデバイスタイプ</li> <li>■ シリンダ数、代替シリンダ数、ヘッド数、セクター数</li> <li>■ パーティションテーブル</li> </ul>
save	コマンド	<p><b>VTOC</b> - 新しいディスク情報およびパーティション情報を保存します。</p> <p><b>EFI</b> - 適用できません。</p>
inquiry	コマンド	<b>SCSI</b> ディスクのみ - 現在のドライブのベンダー、製品名、リビジョンレベルを表示します。
volname	コマンド	8 文字のボリューム名を新規に指定して、ディスクラベルを作成します。

表 13-1 format ユーティリティのメインメニュー項目の説明 (続き)

メニュー項目	コマンド/メニュー	説明
quit	コマンド	format メニューを終了します。

## partition メニュー

partition メニューの内容が次のように表示されます:

```
format> partition
PARTITION MENU:
 0 - change '0' partition
 1 - change '1' partition
 2 - change '2' partition
 3 - change '3' partition
 4 - change '4' partition
 5 - change '5' partition
 6 - change '6' partition
 7 - change '7' partition
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
quit
```

次の表に、partition メニューの項目を示します。

表 13-2 partition メニュー項目の説明

サブコマンド	説明
change 'n' partition	新しいパーティションに次の情報を指定できます。 <ul style="list-style-type: none"><li>■ 識別タグ</li><li>■ アクセス権フラグ</li><li>■ 開始シリンダ</li><li>■ サイズ</li></ul>
select	事前定義済みのパーティションテーブルを選択できます。
modify	パーティションテーブル内のすべてのスライスを変更できます。個々のスライスに対して実行する change 'x' partition コマンドよりも、このコマンドが多く使用されます。
name	現在のパーティションテーブルの名前を指定できます。
print	現在のパーティションテーブルを表示します。
label	パーティションマップとラベルを現在のディスクに書き込みます。

表 13-2 partition メニュー項目の説明 (続き)

サブコマンド	説明
quit	partition メニューを終了します。

## x86: fdisk メニュー

fdisk メニューは x86 ベースのシステムでのみ使用でき、次のように表示されます。

```
format> fdisk
 Total disk size is 8924 cylinders
 Cylinder size is 16065 (512 byte) blocks

 Cylinders
 Partition Status Type Start End Length %
 ===== ===== =
 1 EFI 0 8924 8925 100

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Edit/View extended partitions
6. Exit (update disk configuration and exit)
7. Cancel (exit without updating disk configuration)
Enter Selection:
```

次の表に、fdisk メニューの項目を示します。

表 13-3 x86:fdisk メニュー項目の説明

メニュー項目	説明
Create a partition	fdisk パーティションを作成します。Oracle Solaris や DOS など、OS ごとに別々のパーティションを作成しなければなりません。1 台のディスクの最大パーティション数は 4 です。fdisk のパーティションのサイズをパーセンテージで入力するように促すプロンプトが表示されます。
Specify the active partition	ブートに使用するパーティションを指定できます。このメニュー項目により、第 1 段階のブートプログラムが第 2 段階のブートプログラムを検索する場所を指定します。
Delete a partition	以前に作成したパーティションを削除します。このコマンドを実行すると、パーティション内のすべてのデータが失われます。
Change between Solaris and Solaris2 Partition IDs	パーティション識別子を 130 (0x82) から 191 (0xbf) に、または 191 (0xbf) から 130 (0x82) に変更します。

表 13-3 x86: fdisk メニュー項目の説明 (続き)

メニュー項目	説明
Edit/View extended partitions	通常ブートに使用されるパーティション情報を管理します。
Exit (update disk configuration and exit)	新しいパーティションテーブルを書き込んで fdisk メニューを終了します。
Cancel (exit without updating disk configuration)	パーティションテーブルを変更せずに fdisk メニューを終了します。

## analyze メニュー

analyze メニューの内容が次のように表示されます:

```
format> analyze

ANALYZE MENU:
 read - read only test (doesn't harm SunOS)
 refresh - read then write (doesn't harm data)
 test - pattern testing (doesn't harm data)
 write - write then read (corrupts data)
 compare - write, read, compare (corrupts data)
 purge - write, read, write (corrupts data)
 verify - write entire disk, then verify (corrupts data)
 print - display data buffer
 setup - set analysis parameters
 config - show analysis parameters
 quit
analyze>
```

次の表に、analyze メニューの項目を示します。

表 13-4 analyze メニュー項目の説明

サブコマンド	説明
read	現在のディスクの各セクターを読み込みます。デフォルトで欠陥ブロックを修復します。
refresh	データを損なわずに、現在のディスク上で読み取りおよび書き込みを実行します。デフォルトで欠陥ブロックを修復します。
test	データを損なわずに一連のパターンをディスクに書き込みます。デフォルトで欠陥ブロックを修復します。
write	一連のパターンをディスクに書き込んでから、そのデータをディスクから読み込みます。ディスク上の既存のデータは破壊されます。デフォルトで欠陥ブロックを修復します。
compare	ディスクに一連のパターンを書き込み、そのデータを読み込み、書き込みバッファ内のデータと比較します。ディスク上の既存のデータは破壊されます。デフォルトで欠陥ブロックを修復します。

表 13-4 analyze メニュー項目の説明 (続き)

サブコマンド	説明
purge	ディスク上のデータをすべて削除し、いかなる手段でも取り出せないようにします。ディスク全体(またはディスクのセクション)に 3 種類のパターンを書き込むことにより、データを削除します。検査に合格すると16進のビットパターンがディスク全体(またはディスクのセクション)に上書きされます。  デフォルトで欠陥ブロックを修復します。
verify	最初の検査に合格すると、ディスク全体の各ブロックに固有のデータを書き込みます。次の検査に合格すると、データを読み取り、検証します。ディスク上の既存のデータは破壊されます。デフォルトで欠陥ブロックを修復します。
print	読み取り / 書き込みバッファ内のデータを表示します。
setup	次の解析パラメータを指定できます。  Analyze entire disk? yes Starting block number: <i>depends on drive</i> Ending block number: <i>depends on drive</i> Loop continuously? no Number of passes: 2 Repair defective blocks? yes Stop after first error? no Use random bit patterns? no Number of blocks per transfer: 126 (0/n/nn) Verify media after formatting? yes Enable extended messages? no Restore defect list? yes Restore disk label? yes
config	現在の解析パラメータを表示します。
quit	analyze メニューを終了します。

## defect メニュー

defect メニューの内容が次のように表示されます。

```
format> defect

DEFECT MENU:
 primary - extract manufacturer's defect list
 grown - extract manufacturer's and repaired defects lists
 both - extract both primary and grown defects lists
 print - display working list
 dump - dump working list to file
 quit
defect>
```

次の表に、defect メニューの項目を示します。

表 13-5 defect メニュー項目の説明

サブコマンド	説明
プライマリ	製造元の欠陥リストをディスクドライブから読み込み、メモリー内の欠陥リストを更新します。
grown	増分の欠陥リストを読み取り、メモリー内の欠陥リストを更新します。「増分の欠陥」とは、解析中に検出された欠陥のことです。
both	製造元の欠陥リストと増分の欠陥リストの両方を読み取ります。その後、メモリー内欠陥リストを更新します。
print	メモリー内の欠陥リストを表示します。
dump	メモリー内の欠陥リストをファイルに保存します。
quit	defect メニューを終了します。

## format コマンドへの入力規則

format ユーティリティーを使用する場合は、さまざまな情報を入力する必要があります。このセクションでは、入力する情報に関する規則について説明します。データ指定時に format のヘルプ機能を使用する方法については、[297 ページ](#)の「[format ユーティリティーのヘルプを利用する](#)」を参照してください。

## format コマンドへ番号を指定する

format ユーティリティーを使用する際、数値を入力する必要があります。入力方法には、適切なデータを指定する方法と、選択肢のリストから番号を選択する方法があります。どちらの場合も、ヘルプ機能を使用すると、format は期待する数値の上限と下限を表示します。適切な数値を入力するだけで済みます。数値は、その一部として底を明示的に指定しない限り (16 進数を表す 0x など)、10 進数と見なされます。

次の例は、整数の入力を示しています。

```
Enter number of passes [2]: 34
Enter number of passes [34] 0xf
```

## format のコマンド名を指定する

format ユーティリティーでメニュープロンプトを表示する際、入力としてコマンド名が必要になります。コマンド名は、目的のコマンドとして区別できる長さまで省略できます。



たとえば、`p` (artition) を使用して `format` メニューから `partition` メニューにアクセスできます。次に、`p(rint)` を使用して現在のスライステーブルを表示できます。

```
format> p
PARTITION MENU:
 0 - change '0' partition
 1 - change '1' partition
 2 - change '2' partition
 3 - change '3' partition
 4 - change '4' partition
 5 - change '5' partition
 6 - change '6' partition
 7 - change '7' partition
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
quit
partition> p
```

## format コマンドへディスク名を指定する

`format` ユーティリティでは、名前を指定しなければならない場合があります。このような場合は、名前に使用する文字列を自由に指定できます。空白を含む名前は、二重引用符 (") で囲まなければなりません。二重引用符で囲まなければ、名前の最初の語だけが使用されます。

たとえば、ディスクの特定のパーティションテーブルを指定する場合、`partition` メニューの `name` サブコマンドを使用できます。

```
partition> name
Enter table name (remember quotes): "new disk3"
```

## format ユーティリティのヘルプを利用する

`format` ユーティリティにはヘルプ機能が組み込まれており、`format` ユーティリティが入力待ちの状態であればいつでも使用できます。疑問符 (?) を入力するだけで必要な入力に関するヘルプが表示されます。`format` ユーティリティでは、どんなタイプの入力が必要かについて簡潔な説明が表示されます。

メニュープロンプトから ? と入力すると、利用できるコマンドのリストが表示されます。

`format` ユーティリティに関連するマニュアルページには、次が含まれます。

- **format(1M)** – `format` ユーティリティの基本機能およびコマンド行で使用可能なすべての変数について説明します。

- [format.dat\(4\)](#) – format ユーティリティで使用するディスクドライブ構成に関する情報を提供します。

## ファイルシステムの管理 (概要)

---

ファイルシステムの管理は、もっとも重要なシステム管理タスクの1つです。

この章の内容は以下のとおりです。

- 299 ページの「ファイルシステム管理タスクについての参照先」
- 300 ページの「ファイルシステムの概要」
- 306 ページの「デフォルトの Oracle Solaris ファイルシステム」
- 307 ページの「ファイルシステムのマウントおよびマウント解除の概要」
- 311 ページの「ファイルシステムのタイプを調べる」

## ファイルシステム管理タスクについての参照先

ファイルシステム管理の手順については、次を参照してください。

ファイルシステム管理タスク	参照先
新しいディスクデバイスを接続して構成します。	第7章「ディスクの管理 (概要)」
新規ファイルシステムを作成およびマウントします。	第15章「ファイルシステムの作成およびマウント (タスク)」
リモートファイルをユーザーが利用できるようにします。	『System Administration Guide: Network Services』の第5章「Network File System Administration (Tasks)」

# ファイルシステムの概要

ファイルシステムは、ファイルを編成して格納するためのディレクトリ構造です。

「ファイルシステム」という用語は、次のような場合に使用されます。

- 特定の種類のファイルシステム (ディスクベース、ネットワークベース、または仮想) を指す場合
- ルートディレクトリ (/) から始まるファイルツリー全体を指す場合
- ディスクスライスやほかの記憶メディアデバイスのデータ構造を指す場合
- ファイルツリー構造のうち、ファイルがアクセスできるように主なファイルツリー上のマウントポイントに接続されている部分を指す場合

通常、その意味は状況に応じて判断できます。

Solaris オペレーティングシステムは、各種ファイルシステムへの標準インタフェースを提供する「仮想ファイルシステム」(VFS) アーキテクチャーを使用します。VFS アーキテクチャーによって、カーネルはファイルの読み取り、書き込み、一覧表示などの基本操作を処理できます。また、VFS アーキテクチャーにより新しいファイルシステムの追加も簡単になります。

## Oracle Solaris ファイルシステムのタイプ

Oracle Solaris OS では、次の3種類のファイルシステムがサポートされます。

- ディスクベースのファイルシステム
- ネットワークベースのファイルシステム
- 仮想ファイルシステム

ファイルシステムのタイプを確認するには、[311 ページの「ファイルシステムのタイプを調べる」](#)を参照してください。

## Oracle Solaris のディスクベースのファイルシステム

「ディスクベースのファイルシステム」は、ハードディスクやDVDなどの物理メディアに格納されます。ディスクベースのファイルシステムは、さまざまな形式で作成できます。次の表で、作成できる形式について説明します。

ディスクベースのファイルシステム	形式の説明
ZFS	ZFSはディスクベースのファイルシステムです。詳細については、『Oracle Solaris ZFS 管理ガイド』を参照してください。
UFS	レガシー UNIX ファイルシステム (4.3 Tahoe リリースで提供されていた BSD Fat Fast ファイルシステムに基づく)。

ディスクベースのファイルシステム	形式の説明
HSFS	High Sierra、Rock Ridge、および ISO 9660 のファイルシステム。High Sierra は、はじめての CD-ROM ファイルシステムです。ISO 9660 は、High Sierra ファイルシステムの公式の標準バージョンです。HSFS ファイルシステムは CD-ROM 上で使用される読み取り専用ファイルシステムです。Oracle Solaris HSFS では ISO 9660 の Rock Ridge 拡張がサポートされます。CD-ROM 上に存在する場合、これらの拡張機能は、書き込み可能性およびハードリンクを除くすべてのファイルシステム機能およびファイルタイプを提供します。
PCFS	PC ファイルシステム。DOS ベースのパーソナルコンピュータ用に作成された DOS フォーマットのディスク上のデータとプログラムに読み取りと書き込みのアクセスが可能です。
UDFS	UDFS (Universal Disk Format) ファイルシステム。DVD (Digital Versatile Disc または Digital Video Disc) と呼ばれる光学式メディアテクノロジーに情報を格納するための業界標準形式です。

ディスクベースの各種ファイルシステムは、次のように特定のメディアのタイプに対応しています。

- ZFS または UFS とハードディスク
- HSFS と CD-ROM
- PCFS と USB フロッピーディスク
- UDF と DVD

ただし、これらの対応関係は制限的なものではありません。たとえば、DVD 上に ZFS または UFS ファイルシステムを作成できます。

## UDFS (Universal Disk Format) ファイルシステム

リムーバブルメディア上で UDFS ファイルシステムを作成する方法については、[29 ページの「リムーバブルメディア上にファイルシステムを作成する方法」](#)を参照してください。

UDF ファイルシステムは、「DVD」(Digital Versatile Disc または Digital Video Disc) 光学式メディアに情報を格納するための業界標準形式です。

UDF ファイルシステムは、SPARC と x86 の両方のプラットフォームにおいて、動的に読み込み可能な 32 ビットと 64 ビットのモジュールとして提供されます。また、ファイルシステムを作成、マウント、および検査するシステム管理ユーティリティも同時に提供されます。Oracle Solaris の UDF ファイルシステムは、サポートされている ATAPI と SCSI の DVD ドライブ、CD-ROM デバイス、およびディスクドライブで機能します。さらに、Oracle Solaris の UDF ファイルシステムは UDF 1.50 仕様に完全に準拠しています。

UDF ファイルシステムには次のような機能があります。

- UDF ファイルシステムが入っている業界標準の CD-ROM や DVD-ROM のメディアにアクセスできます
- さまざまなプラットフォームやオペレーティングシステムで情報を交換できます
- UDF 形式に基づく DVD ビデオ仕様を使用することで、放送品質並みの映像、高品質のサウンド、すぐれた対話性という特長を備えた新しいアプリケーションを実装できます

次の機能は、UDF ファイルシステムにはありません。

- 書き込み可能なメディア (CD-RW) へのディスクアットワンス (Disk At Once) 記録、およびインクリメンタル記録

次に、UDF ファイルシステムの要件を示します。

- SPARC または x86 プラットフォームがサポートされていること
- CD-ROM または DVD-ROM ドライブがサポートされていること

Oracle Solaris で実装された UDF ファイルシステムには、次のような互換性があります。

- 業界標準の読み取り / 書き込み UDF Version 1.50 のサポート
- 完全に国際化されたファイルシステムのユーティリティ

## ネットワークベースのファイルシステム

「ネットワークベースのファイルシステム」は、ネットワークからアクセスされるファイルシステムです。ネットワークベースのファイルシステムは通常、1つのシステム (通常はサーバー) 上にあり、ほかのシステムからネットワーク経由でアクセスされます。

NFS サービスで分散されたリソース (ファイルやディレクトリ) を提供するには、サーバーからそれらのリソースを共有して個々のクライアントでマウントします。詳細は、[309 ページの「NFS 環境」](#)を参照してください。

## 仮想ファイルシステム

「仮想ファイルシステム」は、特殊なカーネル情報と機能へのアクセスを提供するメモリーベースのファイルシステムです。ほとんどの仮想ファイルシステムは、ディスク領域を使用しません。また、一時ファイルシステム (TMPFS) などの一部の仮想化ファイルシステムは、ディスク上のスワップ空間を使用します。

## 一時ファイルシステム

一時ファイルシステム (TMPFS) は、ファイルシステムの読み取りと書き込みにローカルメモリーを使用します。TMPFS を使用すると、ローカルディスク上で、あるいはネットワーク経由で一時ファイルの読み書きを行う際のオーバーヘッドを軽減

でき、システムパフォーマンスを改善できます。一時ファイルは、たとえば、プログラムのコンパイル時に作成されます。OSは、一時ファイルを操作しているとき、多くのディスク処理またはネットワーク処理を行います。TMPFSを使ってこれらの一時ファイルを管理することで、それらの作成、操作、および削除の効率を大幅に向上できます。

TMPFS ファイルシステムのファイルは、永続的に保存されるわけではありません。これらのファイルは、ファイルシステムのマウントが解除されるときと、システムがシャットダウンまたはリブートされるときに削除されます。

TMPFS は、Oracle Solaris OS 内の `/tmp` ディレクトリのデフォルトのファイルシステムです。ZFS または UFS ファイルシステムの場合と同様に、`/tmp` ディレクトリとの間でファイルをコピーまたは移動できます。

TMPFS ファイルシステムは、一時的な退避場所としてスワップ空間を使用します。

TMPFS ファイルシステムがマウントされたシステムのスワップ空間が足りないと、次の2つの問題が発生する可能性があります。

- TMPFS ファイルシステムは、通常のファイルシステムと同様に容量不足になる可能性があります。
- TMPFS はスワップ空間を割り当ててファイルのデータを保存するので (必要な場合)、一部のプログラムがスワップ空間不足のために実行できなくなる可能性があります。

TMPFS ファイルシステムの作成方法については、[第 15 章「ファイルシステムの作成およびマウント\(タスク\)」](#)を参照してください。スワップ空間を追加する方法については、[第 16 章「追加スワップ空間の構成\(タスク\)」](#)を参照してください。

## ループバックファイルシステム

ループバックファイルシステム (LOFS) を使用すると、代替パス名を使用してファイルにアクセスできるように、新しい仮想ファイルシステムを作成できます。たとえば、`/tmp/newroot` 上にルート (`/`) ディレクトリのループバックマウントを作成できます。このループバックマウントでは、NFS サーバーからマウントされたファイルシステムを含むファイルシステム階層全体が、`/tmp/newroot` の下に複製されたように見えます。どのファイルにも、ルート (`/`) で始まるパス名または `/tmp/newroot` で始まるパス名を使用してアクセスできます。

LOFS ファイルシステムの作成方法については、[第 15 章「ファイルシステムの作成およびマウント\(タスク\)」](#)を参照してください。

## プロセスファイルシステム

プロセスファイルシステム (PROCFS) はメモリー内に存在し、`/proc` ディレクトリ内にアクティブなプロセスのプロセス番号別リストが格納されます。`/proc` ディレクト

りの内容は、ps などのコマンドで使用されます。デバッガやほかの開発ツールも、ファイルシステムコールを使用して、プロセスのアドレス空間にアクセスできます。



注意 - /proc ディレクトリ内のファイルは削除しないでください。/proc ディレクトリからプロセスを削除しても、そのプロセスは強制終了されません。/proc ファイルはディスク容量を消費しないため、このディレクトリからファイルを削除しても無意味です。

/proc ディレクトリは、管理が不要です。

その他の仮想ファイルシステム

次のタイプの仮想ファイルシステムは、参考のために掲載してあります。管理は不要です。

仮想ファイルシステム	説明
CTFS	CTFS (契約ファイルシステム) は、契約の作成、制御、および監視のためのインタフェースです。契約は、豊富なエラー報告機能とリソースの削除を延期する手段 (オプション) を提供することにより、プロセスと、このプロセスが依存するシステムとの関係を拡張します。  サービス管理機能 (SMF) は、プロセス契約 (契約の一種) を使用して、サービスを構成するプロセス群を追跡します。このため、マルチプロセスサービスの一部分での障害をそのサービスの障害として識別できます。
FIFOFS (先入れ先出し)	プロセスにデータへの共通アクセス権を与える名前付きパイプのファイル。
FDFS (ファイル記述子)	開いているファイルに、ファイル記述子を使用して名前を明示的に与えます。
MNTFS	ローカルシステムに、マウント済みファイルシステムのテーブルへの読み取り専用アクセスを提供します。
NAMEFS	ほとんどの場合、ファイル記述子をファイルの先頭に動的にマウントするために STREAMS に使用されます。
OBJFS	OBJFS (オブジェクト) ファイルシステムは、現在カーネルによってロードされているすべてのモジュールの状態を説明します。デバッガは、この仮想ファイルシステムを使って、カーネルに直接アクセスすることなくカーネルシンボルの情報にアクセスできます。



仮想ファイルシステム	説明
SHAREFS	ローカルシステムの共有ファイルシステムのテーブルに対する読み取り専用アクセス権を提供します。
SPECFS (特殊)	キャラクタ型特殊デバイスとブロック型デバイスへのアクセスを提供します。
SWAPFS	カーネルがスワッピングに使用するファイルシステム。

## 拡張ファイル属性

ZFS、UFS、NFS、およびTMPFS ファイルシステムは、拡張ファイル属性を含むように機能拡張されました。アプリケーション開発者は、拡張ファイル属性を使って、ファイルに特定の属性を関連付けることができます。たとえば、ウィンドウシステムの管理アプリケーションの開発者は、表示アイコンとファイルを関連付けることができます。拡張ファイル属性は、論理的には、ターゲットファイルに関連付けられている隠しディレクトリ内のファイルとして表されます。

属性を追加し、拡張属性の名前空間内に入っているシェルコマンドを実行するには、`runat` コマンドを使用します。拡張属性の名前空間とは、特定のファイルに関連付けられた、非表示の属性ディレクトリです。

`runat` コマンドを使用して属性をファイルに追加するには、最初に属性ファイルを作成する必要があります。

```
$ runat filea cp /tmp/attrdata attr.1
```

次に、`runat` コマンドを使用して、ファイルの属性をリストに表示します。

```
$ runat filea ls -l
```

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

属性認識オプションの追加により、多くの Oracle Solaris ファイルシステムコマンドがファイルシステム属性をサポートするようになりました。属性認識オプションを使って、ファイル属性を照会したり、コピーしたり、検索したりできます。詳細は、各ファイルシステムコマンドのマニュアルページを参照してください。

## スワップ空間

Oracle Solaris OSは、一部のディスクスライスをファイルシステムではなく一時記憶域として使用します。これらのスライスを「スワップスライス」または「スワップ空間」と呼びます。スワップ空間は、現在のプロセスを処理するだけの十分な物理メモリーがシステムにない場合に、仮想メモリー記憶域として使用されます。

多くのアプリケーションは十分なスワップ空間が使用できることを前提に作成されているため、スワップ空間を割り当て、その使われ方を監視して、必要に応じてス

ワップ空間を追加する方法を知っておく必要があります。スワップ空間の概要とスワップ空間の追加方法については、第 16 章「追加スワップ空間の構成(タスク)」を参照してください。

## デフォルトの Oracle Solaris ファイルシステム

ZFS ファイルシステムは階層構造になっており、ルートディレクトリ (/) から始まり、下位に多数のディレクトリが続いています。Oracle Solaris のインストールプロセスは、デフォルトのディレクトリセットをインストールし、一連の規則を適用して類似するタイプのファイルをグループ化します。

Oracle Solaris のファイルシステムやディレクトリの簡単な概要については、filesystem(5) を参照してください。

次の表で、デフォルトの Oracle Solaris ファイルシステムのサマリーを提供します。

表 14-1 デフォルトの Oracle Solaris ファイルシステム

ファイルシステムまたはディレクトリ	ファイルシステムのタイプ	説明
ルート (/)	UFS または ZFS	階層ファイルツリーの最上位。ルート (/) ディレクトリには、カーネル、デバイスドライバ、システムのブートに使用されるプログラムなど、システム処理に欠かせないディレクトリとファイルが入っています。また、ローカルとリモートのファイルシステムをファイルツリーに接続できるマウントポイントディレクトリも入っています。
/usr	UFS または ZFS	ほかのユーザーと共有できるシステムファイルとディレクトリ。特定のタイプのシステム上でのみ実行できるファイルは、/usr ディレクトリにあります (SPARC 実行可能ファイルなど)。どのタイプのシステム上でも使用できるファイル(マニュアルページなど)は、/usr/share ディレクトリに配置できます。
/export/home または /home	NFS、UFS、または ZFS	ユーザーのホームディレクトリのマウントポイント。ホームディレクトリには、そのユーザーの作業ファイルが格納されます。デフォルトでは、/home ディレクトリは自動マウントされるファイルシステムです。
/var	UFS または ZFS	ローカルシステムの使用中に変化または拡大する可能性のあるシステムファイルとディレクトリ。これには、システムログ (vi や ex のバックアップファイルなど) が含まれます。

表 14-1 デフォルトの Oracle Solaris ファイルシステム (続き)

ファイルシステムまたはディレクトリ	ファイルシステムのタイプ	説明
/opt	NFS、UFS、または ZFS	オプションの他社製のソフトウェア製品のマウントポイント。一部のシステムでは、/opt ディレクトリが UFS ファイルシステムまたは ZFS ファイルシステムである場合があります。
/tmp	TMPFS	システムがブートされるか、/tmp ファイルシステムがマウント解除されるたびに削除される一時ファイル。
/proc	PROCFS	アクティブなプロセスのプロセス番号別リスト。
/etc/mnttab	MNTFS	ローカルシステムに、マウント済みファイルシステムのテーブルへの読み取り専用アクセスを提供する仮想ファイルシステム。
/system/volatile	TMPFS	システムのブート後は不要になる一時ファイルを格納するメモリーベースのファイルシステム。
/system/contract	CTFS	契約情報を保持する仮想ファイルシステム。
/system/object	OBJFS	カーネルに直接アクセスすることなくカーネルシンボルの情報にアクセスする際にデバッグによって使用される仮想ファイルシステム。

## ファイルシステムのマウントおよびマウント解除の概要

ファイルシステム上のファイルにアクセスするには、ファイルシステムをマウントする必要があります。ファイルシステムのマウントとは、ファイルシステムを特定のディレクトリ(マウントポイント)に接続し、システムで使用可能にすることです。ルート(/)ファイルシステムは、常にマウントされています。ほかのファイルシステムは、ルート(/)ファイルシステムに接続したり切り離したりできます。

ほとんどのファイルシステムは、システムブート時に SMF サービスにより自動的にマウントされます。一般に、ファイルシステムのマウントやマウント解除を手動で行う必要はありません。さまざまなファイルシステムタイプをマウントする方法の詳細については、[315 ページの「Oracle Solaris ファイルシステムのマウントおよびマウント解除」](#)を参照してください。

ファイルシステムをマウントすると、そのファイルシステムがマウントされている間は、マウントポイントのディレクトリ内に実際に存在しているファイルやディレクトリは使用できなくなります。これらのファイルは、永続的にマウントプロセスの影響を受け続けるわけではありません。ファイルシステムのマウントが解除されると、再び使用可能な状態になります。ただし、通常は存在するがアクセスできないファイルは混乱の原因となるので、通常マウントディレクトリは空になっています。

ファイルシステムのマウント手順については、[315 ページの「Oracle Solaris ファイルシステムのマウントおよびマウント解除」](#)を参照してください。

## マウントされたファイルシステムテーブル

ファイルシステムをマウントまたはマウント解除すると、現在マウントされているファイルシステムのリストを使用して、`/etc/mnttab` (マウントテーブル) ファイルが変更されます。このファイルの内容を表示するには、`cat` または `more` コマンドを使用します。ただし、このファイルを編集することはできません。次にマウントテーブルファイル `/etc/mnttab` の例を示します。

```
$ more /etc/mnttab
rpool/ROOT/zfsBE / zfs dev=3390002 0
/devices /devices devfs dev=8580000 1337114941
/dev /dev dev dev=85c0000 1337114941
ctfs /system/contract ctfs dev=8680001 1337114941
proc /proc proc dev=8600000 1337114941
mnttab /etc/mnttab mntfs dev=86c0001 1337114941
swap /system/volatile tmpfs xattr,dev=8700001 1337114941
objfs /system/object objfs dev=8740001 1337114941
sharefs /etc/dfs/sharetab sharefs dev=8780001 1337114941
/usr/lib/libc/libc_hwcapi2.so.1 /lib/libc.so.1 lofs dev=3390002 13371149
fd /dev/fd fd rw,dev=8880001 1337114969
rpool/ROOT/zfsBE/var /var zfs rw,devices,
setuid,nonbmand,exec,
rstchown,xattr,atime,dev=3390003 1337114969
swap /tmp tmpfs xattr,dev=8700002 1337114969
rpool/VARSHARE /var/share zfs rw,devices,setuid,nonbmand,exec,
rstchown,xattr,atime,dev=3390004 1337114969
```

## 仮想ファイルシステムテーブル

ほとんどのファイルシステムは、システムブート時に SMF サービスにより自動的にマウントされます。

レガシーまたはリモートファイルシステムをマウントしたり、ZFS スワップポリュームを変更したりする場合は、`/etc/vfstab` ファイルの編集が必要なことがあります。ZFS スワップポリュームの変更については、[第 16 章「追加スワップ空間の構成\(タスク\)」](#)を参照してください。

レガシーまたはリモートファイルシステムをマウントするためのエントリを追加するには、次の情報を指定する必要があります。

- ファイルシステムが存在するデバイスまたは NFS サーバー
- ファイルシステムのマウントポイント
- ファイルシステムのタイプ
- システムのブート時に、`mountall` コマンドを使ってファイルシステムを自動的にマウントするかどうか

■ マウントオプション

次に、ZFS ルートファイルシステムがあるシステムの `vfstab` の例を示します。さらに、システムは、NFS サーバー `neo` からリモートファイルシステム `/users/data` をマウントしています。

```
cat /etc/vfstab
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
fd - /dev/fd fd - no -
/proc - /proc proc - no -
/dev/zvol/dsk/rpool/swap - - swap - no -
/devices - /devices devfs - no -
sharefs - /etc/dfs/sharetabsharefs - no -
ctfs - /system/contract ctfs - no -
objfs - /system/object objfs - no -
swap - /tmp tmpfs - yes -
neo:/users/data - /data nfs - yes -
```

ZFS ファイルシステムは、ブート時に SMF サービスにより自動的にマウントされます。レガシーのマウント機能を使用し、`vfstab` に従って ZFS ファイルシステムをマウントできます。詳細については、『Oracle Solaris ZFS 管理ガイド』を参照してください。

`/etc/vfstab` の各フィールドの説明、およびこのファイルの編集方法と使用方法については、[vfstab\(4\)](#) を参照してください。

## NFS 環境

NFS は、1 つのシステム (通常はサーバー) のリソース (ファイルやディレクトリ) をネットワーク上のほかのシステムと共有するための分散型ファイルシステムサービスです。たとえば、他社製のアプリケーションやソースファイルをほかのシステム上のユーザーと共有できます。

NFS は、リソースの実際の物理的な位置をユーザーが意識しなくてすむようにします。よく使用されるファイルのコピーをシステムごとに配置しなくても、あるシステムのディスク上にコピーを 1 つ配置することによって NFS は、ほかのすべてのシステムがそのコピーにネットワークからアクセスできるようにします。NFS の環境では、リモートファイルはローカルファイルと区別がつきません。

詳細については、『[Solaris のシステム管理 \(ネットワークサービス\)](#)』の第 4 章「[ネットワークファイルシステムの管理 \(概要\)](#)」を参照してください。

システムは、ネットワーク上で共有するリソースがあるときに、NFS サーバーになります。サーバーは、現在共有されているリソースとそのアクセス制限 (読み取り / 書き込み、読み取り専用アクセスなど) のリストを管理します。

リソースを共有する場合は、リモートシステムにマウントできるように、そのリソースを使用可能な状態にします。

リソースを共有するには、次の方法があります。

- ZFSのプロパティを設定してZFS共有を作成します。例:
- `share` コマンドを使用してレガシー共有を作成する。

```
share -F nfs /ufsfs
```

NFSの詳細については、『[Solaris のシステム管理 \(ネットワークサービス\)](#)』の第4章「[ネットワークファイルシステムの管理 \(概要\)](#)」を参照してください。

## NFS Version 4

Oracle Solaris リリースには、Oracle の実装の NFS version 4 分散ファイルアクセスプロトコルが含まれています。

NFS version 4 では、ファイルアクセス、ファイルロック、およびマウントプロトコルが1つのプロトコルに統合されるので、ファイアウォールの通過が容易になり、セキュリティが向上します。Oracle Solaris の NFS version 4 実装は、SEAM としても知られている Kerberos V5 と完全に統合されていますので、認証、整合性、およびプライバシーの機能を備えています。NFS version 4 を使用して、クライアントとサーバーとの間で使用するセキュリティの種類交渉を行うこともできます。NFS version 4 を実装しているサーバーは、さまざまなセキュリティ種類とファイルシステムに対応できます。

NFS Version 4 の機能については、『[Solaris のシステム管理 \(ネットワークサービス\)](#)』の「[NFS サービスの新機能](#)」を参照してください。

## 自動マウント (autofs)

NFS ファイルシステムリソースをマウントするには、「自動マウント」(*autofs*) というクライアント側のサービスを使用します。*autofs* サービスにより、システムから NFS リソースにアクセスするたびに、これらのリソースを自動的にマウントしたりマウント解除したりできます。ユーザーがこのディレクトリ内で、このディレクトリに格納されているファイルを使用している間、ファイルシステムリソースはマウントされたままになります。リソースが一定の時間アクセスされなかった場合、リソースは自動的にマウント解除されます。

*autofs* サービスには、次の機能があります。

- システムブート時に NFS リソースをマウントする必要がないために、ブート時間が短くなります。
- NFS リソースをマウントまたはマウント解除するために、スーパーユーザーのパスワードを知っている必要はありません。
- NFS リソースは使用されるときにだけマウントされるために、ネットワークトラフィックが軽減されます。

autofs サービスは automount ユーティリティーによって初期化されます。このコマンドは、システムのブート時に自動的に実行されます。automountd デーモンは永続的に動作し、必要に応じて NFS ファイルシステムをマウントまたはマウント解除します。デフォルトでは、/home ファイルシステムは automount デーモンによってマウントされます。

autofs では、同じファイルシステムを提供するサーバーを複数指定できます。このような方法では、1つのサーバーがダウンしても、autofs がその他のマシンからファイルシステムのマウントを試みることができます。

autofs を設定および管理する方法の完全な情報については、『[System Administration Guide: Network Services](#)』の第5章「[Network File System Administration \(Tasks\)](#)」を参照してください。

## ファイルシステムのタイプを調べる

ファイルシステムのタイプは、次のいずれかによって判定できます。

- 仮想ファイルシステムテーブル (/etc/vfstab ファイル) 内の FS type フィールド
- ローカルファイルシステムの /etc/default/fs ファイル
- NFS ファイルシステムの /etc/dfs/fstypes ファイル

## ファイルシステムのタイプを調べる方法

これらのコマンドは、ファイルシステムがマウントされているかどうかにかかわらず、使用できます。

ディスクスライスの raw デバイス名がわかる場合、fstyp コマンドで、ファイルシステムのタイプを調べることができます (そのディスクスライスにファイルシステムが含まれている場合)。詳細は、[fstyp\(1M\)](#) のマニュアルページを参照してください。

例 14-1 ファイルシステムのタイプを調べる

次の例は、fstyp コマンドを使ってファイルシステムのタイプを確認しています。

```
fstyp /dev/rdisk/c0t0d0s0
zfs
```

次の例では、/etc/vfstab ファイルを使用して、/legacy ファイルシステムのタイプを調べます。

```
$ grep /legacy /etc/vfstab
/dev/dsk/c0t3d0s6 /dev/rdisk/c0t3d0s6 /legacy ufs 2 yes -
```





# ファイルシステムの作成およびマウント (タスク)

---

この章の内容は次のとおりです。

- [313 ページの「Oracle Solaris ファイルシステムの作成」](#)
- [315 ページの「Oracle Solaris ファイルシステムのマウントおよびマウント解除」](#)
- [319 ページの「Oracle Solaris ファイルシステムの作成およびマウント」](#)

## Oracle Solaris ファイルシステムの作成

このセクションでは、Oracle Solaris ファイルシステムを作成する方法の概要を説明します。

### ZFS ファイルシステムの作成

ZFS ファイルシステムは、特定のディスクパーティションに直接関連付けられていません。ZFS ファイルシステムで使用される ZFS ストレージプールには、多数のデバイスを含めることができます。ディスク全体またはディスクスライスを ZFS ストレージプールに追加できます。プールの中に、さらに別のファイルシステムを作成することをお勧めします。ファイルシステムに基づいてプールを管理すれば、プールに含まれるさまざまなデータを管理しやすくなります。

ZFS ファイルシステムの作成には、`zfs create` コマンドを使用します。ZFS ファイルシステムは作成時に自動的にマウントされます。詳細については、[319 ページの「ZFS ファイルシステムの作成方法」](#)を参照してください。

ZFS ストレージプールおよびファイルシステムの作成方法の詳細は、『[Oracle Solaris ZFS 管理ガイド](#)』を参照してください。

## 一時ファイルシステムの作成

一時ファイルシステム (TMPFS) は、ファイルシステムの読み取りと書き込みにローカルメモリーを使用します。TMPFS ファイルシステムを使用すると、ローカルディスク上で、あるいはネットワーク経由で一時ファイルの読み書きを行う際のオーバーヘッドが軽減されるのでシステムのパフォーマンスを向上できます。TMPFS ファイルシステム内のファイルは、リブートまたはマウント解除すると削除されます。

複数の TMPFS ファイルシステムを作成した場合は、すべてのファイルシステムが同じシステムリソースを使用するという点に注意してください。mount コマンドの `-o size` オプションを使用して TMPFS のサイズを制限しないと、ある TMPFS ファイルシステムで作成されたファイルが、ほかの TMPFS ファイルシステムのための領域を使い切ってしまう可能性があります。

TMPFS ファイルシステムを作成およびマウントする方法の詳細については、[321 ページの「TMPFS ファイルシステムを作成およびマウントする方法」](#)を参照してください。

一般的な情報については、[tmpfs\(7FS\)](#) を参照してください。

## LOFS ファイルシステムの作成

LOFS ファイルシステムは、既存のファイルシステムに代替パスを提供する仮想ファイルシステムです。ほかのファイルシステムを LOFS ファイルシステムにマウントしても、元のファイルシステムは変化しません。

LOFS ファイルシステムを作成およびマウントする方法の詳細については、[322 ページの「LOFS ファイルシステムを作成およびマウントする方法」](#)を参照してください。

一般的な情報については、[lofs\(7FS\)](#) を参照してください。

---

注- LOFS ファイルシステムは慎重に作成してください。LOFS ファイルシステムは仮想ファイルシステムなので、ユーザーやアプリケーションを混乱させる可能性があります。

---

# Oracle Solaris ファイルシステムのマウントおよびマウント解除

ZFS ファイルシステムのマウントやマウント解除は、自動的に行われます。レガシー UFS ファイルシステムを使用できるようにするには、マウントします。マウントしたファイルシステムは、システムのディレクトリツリー内の指定したマウントポイントに接続されます。ルート (/) ファイルシステムは、常にマウントされています。

次の表に、ファイルシステムをその用途に応じてマウントするためのガイドラインを示します。

必要なマウントの種類	推奨されるマウント方法
ローカルまたはリモートのファイルシステムをときどきマウントする。	コマンド行から手で <code>mount</code> コマンドを入力します。
ローカルの UFS ファイルシステムを頻繁にマウントする。ローカルの ZFS ファイルシステムは、SMF サービスにより自動的にマウントされます。	<code>/etc/vfstab</code> ファイルを使用して、システムがマルチユーザー状態でブートされたときに、自動的にファイルシステムをマウントします。
ホームディレクトリなどのリモートのレガシー UFS ファイルシステムを頻繁にマウントする。	<ul style="list-style-type: none"> <li>■ <code>/etc/vfstab</code> ファイルを使用して、システムがマルチユーザーモードでブートされたときに、自動的にファイルシステムをマウントします。</li> <li>■ <code>autofs</code> を使用して、ユーザーがアクセスするときに自動的にマウントします。その後、ファイルシステムから別のディレクトリに移動するときに自動的にマウント解除します。</li> </ul>

リムーバブルメディアのマウント方法については、[21 ページの「リムーバブルメディアの管理 \(概要\)」](#)を参照してください。

どのファイルシステムがすでにマウント済みであるかを調べるには、`mount` コマンドを使用します。

```
$ mount [-v]
```

`-v` は、マウントされているファイルシステムのリストを冗長モードで表示します。

例 15-1 どのファイルシステムがマウントされているかを調べる

この例は、`mount` コマンドを使用して、現在マウントされているファイルシステムに関する情報を表示する方法を示しています。

## 例 15-1 どのファイルシステムがマウントされているかを調べる (続き)

```
$ mount
/ on rpool/ROOT/zfsBE read/write/setuid/devices/rstchown/dev=3390002 on Tue ...
/devices on /devices read/write/setuid/devices/rstchown/dev=8580000 on Tue May 15 ...
/dev on /dev read/write/setuid/devices/rstchown/dev=85c0000 on Tue May 15 14:49:01 2012
/system/contract on ctfs read/write/setuid/devices/rstchown/dev=8680001 on Tue May 15 ...
/proc on proc read/write/setuid/devices/rstchown/dev=8600000 on Tue May 15 14:49:01 2012
/etc/mnttab on mnttab read/write/setuid/devices/rstchown/dev=86c0001 on Tue May 15 14:49:01 ...
/system/volatile on swap read/write/setuid/devices/rstchown/xattr/dev=8700001 on Tue May 15 ...
/system/object on objfs read/write/setuid/devices/rstchown/dev=8740001 on Tue May 15 ...
/etc/dfs/sharetab on sharefs read/write/setuid/devices/rstchown/dev=8780001 on Tue May 15 ...
/lib/libc.so.1 on /usr/lib/libc/libc_hwcapi2.so.1 read/write/setuid/devices/rstchown/dev ...
/dev/fd on fd read/write/setuid/devices/rstchown/dev=8880001 on Tue May 15 14:49:29 2012
/var on rpool/ROOT/zfsBE/var read/write/setuid/devices/rstchown/nonbmand/exec/xattr/atime/ ...
/tmp on swap read/write/setuid/devices/rstchown/xattr/dev=8700002 on Tue May 15 14:49:29 2012
/var/share on rpool/VARSHARE read/write/setuid/devices/rstchown/nonbmand/exec/xattr/atime/ ...
/home/rimmer on pluto:/export/home/rimmer remote/read/write/setuid/xattr/...
```

この例は、zfs mount コマンドを使用して、現在マウントされている ZFS ファイルシステムに関する情報を表示する方法を示しています。

```
$ zfs mount
rpool/ROOT/zfsBE /
rpool/ROOT/zfsBE /var
rpool/VARSHARE /var/share
tank/home /tank/home
```

## /etc/vfstab ファイルのフィールドの説明

/etc/vfstab ファイル内のエントリには、次の表に示すように7つのフィールドがあります。

表 15-1 /etc/vfstab ファイルのフィールドの説明

フィールド名	説明
device to mount	<p>このフィールドは、次のいずれかを指定します。</p> <ul style="list-style-type: none"> <li>ローカルのレガシー UFS ファイルシステム用のブロック型デバイス名 (/dev/dsk/c8t1d0s7 など)。</li> <li>リモートファイルシステム用のリソース名 (myserver:/export/home など)。 リモートシステムリソースにエントリを追加した後に、次のサービスが有効になっていることを確認してください。</li> </ul> <pre># svcs -a   grep nfs/client disabled      May_14   svc:/network/nfs/client:default # svcadm enable svc:/network/nfs/client:default</pre> <p>NFS の詳細については、ネットワークサービスに関するシステム管理ガイドを参照してください。</p> <ul style="list-style-type: none"> <li>スワップボリューム (/dev/zvol/dsk/rpool/swap など)。</li> <li>仮想ファイルシステム用のディレクトリ。</li> </ul>
device to fsck	<p>「device to mount」フィールドで識別したレガシー UFS ファイルシステムに対応する raw (キャラクタ型) デバイス名 (/dev/rdisk/c8t1d0s7 など)。このフィールドによって、fsck コマンドが使用する raw インタフェースが決まります。読み取り専用ファイルシステムやリモートファイルシステムなど、適用できるデバイスがない場合は、ダッシュ (-) を使用します。</p>
mount point	<p>レガシーまたはリモートファイルシステムをマウントする場所 (/data など) を指定します。</p>
FS type	<p>ファイルシステムのタイプを指定します。</p>
fsck pass	<p>fsck コマンドがレガシー UFS ファイルシステムを検査するか決めるために使用するパス番号。このフィールドでダッシュ (-) を指定すると、ファイルシステムは検査されません。現在、/etc/vfstab ファイルの fsck pass 値はブートプロセスで無視されます。</p> <p>このフィールドにゼロが含まれるときは、レガシー UFS ファイルシステムは検査されません。このフィールドに 0 より大きい値が指定されている場合、UFS ファイルシステムは常に検査されます。</p> <p>このフィールドに 1 が指定されている場合、すべてのレガシー UFS ファイルシステムは vfstab ファイル内の順番どおりに 1 つずつ検査されます。このフィールドに 1 より大きな値が指定され、さらに preen (修復) オプション (-o p) が指定されている UFS ファイルシステムが複数ある場合、効率を最大限に高めるために、fsck コマンドは複数のディスク上のファイルシステムを自動的に並行して検査します。それ以外の場合、このフィールドの値は意味を持ちません。</p>

表 15-1 /etc/vfstab ファイルのフィールドの説明 (続き)

フィールド名	説明
mount at boot	システムのブート時にファイルシステムが <code>mountall</code> コマンドによって自動的にマウントされるかどうかを <code>yes</code> または <code>no</code> で設定します。このフィールドは <code>autofs</code> とは連動していません。 <code>/proc</code> や <code>/dev/fd</code> などの仮想ファイルシステムの場合、このフィールドは常に <code>no</code> に設定するべきです。
mount options	ファイルシステムのマウントに使用されるオプションを (空白を入れずに) コンマで区切ったリスト。オプションなしを示すにはダッシュ (-) を使用します。詳細は、 <a href="#">vfstab(4)</a> を参照してください。

注 - /etc/vfstab ファイル内の各フィールドには必ずエントリが必要です。フィールドに値を指定しない場合は、必ずダッシュ (-) を入力してください。ダッシュを入力しないと、システムが正常にブートしない可能性があります。同様に、フィールドの値に空白文字を使用しないでください。

## Oracle Solaris ファイルシステムをマウント解除するための前提条件

ファイルシステムをマウント解除する場合の前提条件は次のとおりです。

- 管理者である必要があります。
- 使用中のファイルシステムはマウント解除できません。ユーザーがそのファイルシステム内のディレクトリにアクセスしているとき、プログラムがそのファイルシステム上のファイルを開いているとき、またはファイルシステムが共有されているときには、ファイルシステムは使用中とみなされます。

次の方法でファイルシステムをマウント解除が可能な状態にできます。

- 別のファイルシステム内のディレクトリにカレントディレクトリを変更する。
- システムからログアウトする。
- `fuser` コマンドを使用して、そのファイルシステムを使用中のすべてのプロセスを表示し、必要に応じて終了させる。詳細は、[327 ページの「ファイルシステムを使用中のすべてのプロセスを終了させる方法」](#)を参照してください。

ほかのユーザーが使用しているファイルシステムをマウント解除する必要があるときは、各ユーザーに通知します。

- ファイルシステムの共有を解除する。
  - 例:

```
zfs set share.nfs=off tank/fs1
```
  - レガシー `unshare` メソッドを使用します。詳細は、[unshare\(1M\)](#) を参照してください。

ファイルシステムをマウント解除したことを確認するには、`mount` コマンドからの出力を調べます。

```
$ mount | grep unmounted-file-system
```

## Oracle Solaris ファイルシステムの作成およびマウント

このセクションでは、Oracle Solaris ファイルシステムを作成およびマウントする方法の具体例を示します。

### ▼ ZFS ファイルシステムの作成方法

- 1 スーパーユーザーになります。

- 2 ZFS ストレージプールを作成します。

次の例は、`tank` という名前の単純なミラー化ストレージプールと `tank` という名前の ZFS ファイルシステムを、1つのコマンドで作成する方法を示しています。`/dev/dsk/c1t0d0` ディスク全体と `/dev/dsk/c2t0d0` ディスク全体を使用することを前提としています。

```
zpool create tank mirror c1t0d0 c2t0d0
```

- 3 ZFS ファイルシステムを作成します。

```
zfs create tank/fs
```

この新規 ZFS ファイルシステム `tank/fs` では、ディスク領域を必要なだけ使用でき、`/tank/fs` に自動的にマウントされます。

- 4 ファイルシステムが作成されたことを確認します。

```
zfs list -r tank
NAME USED AVAIL REFER MOUNTPOINT
tank 117K 268G 21K /tank
tank/fs 21K 268G 21K /tank/fs
```

### ▼ レガシー UFS ファイルシステムを作成およびマウントする方法

始める前に 次の前提条件を満たしているかどうかを確認します。

- ディスクがフォーマットされ、スライスに分割されている。
- 既存のレガシー UFS ファイルシステムを再作成する場合は、そのマウントを解除します。

- ファイルシステムを格納するスライスのデバイス名がわかっている。

ディスクおよびディスクスライス番号の検索方法については、[第9章「ディスクの管理\(タスク\)」](#)を参照してください。

ディスクのフォーマットとスライスへの分割方法については、[第7章「ディスクの管理\(概要\)」](#)を参照してください。

- 1 スーパーユーザーになります。
- 2 レガシー UFS ファイルシステムを作成します。

```
newfs [-N] [-b size] [-i bytes] /dev/rdisk/device-name
```

システムから、確認を促すプロンプトが表示されます。



注意-この手順を実行する前に、スライスのデバイス名が正しく指定されていることを確認してください。間違ったスライスを指定すると、その内容は新しいファイルシステムの作成時に消去されます。そして、システムがパニックを起こす原因となる可能性があります。

- 3 レガシー UFS ファイルシステムが作成されていることを確認するには、新しいファイルシステムを検査します。

```
fsck /dev/rdisk/device-name
```

*device-name* 引数は、新しいファイルシステムを格納するディスクデバイスの名前を指定します。

fsck コマンドは、新しいファイルシステムの整合性を検査して、問題があれば通知し、問題を修復する前にプロンプトを表示します。fsck コマンドの詳細は、[fsck\(1m\)](#)を参照してください。

- 4 レガシー UFS ファイルシステムをマウントします。

```
mkdir /directory-name
```

```
mount /dev/dsk/device-name /directory-name
```

## 例 15-2 レガシー UFS ファイルシステムの作成およびマウント

次の例は、UFS ファイルシステム `/dev/rdisk/c0t1d0s0` を作成して `/legacy` にマウントする方法を示しています。

```
newfs /dev/rdisk/c0t1d0s0
newfs: construct a new file system /dev/rdisk/c0t1d0s0: (y/n)? y
/dev/rdisk/c0t1d0s0: 286722656 sectors in 46668 cylinders of 48 tracks, 128 sectors
 140001.3MB in 2917 cyl groups (16 c/g, 48.00MB/g, 5824 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 98464, 196896, 295328, 393760, 492192, 590624, 689056, 787488, 885920,
Initializing cylinder groups:
```



```

.....
super-block backups for last 10 cylinder groups at:
 285773216, 285871648, 285970080, 286068512, 286166944, 286265376, 286363808,
 286462240, 286560672, 286659104
fsck /dev/rdisk/c0t1d0s0
mkdir /legacy
mount /dev/dsk/c0t1d0s0 /legacy

```

参考 レガシー UFS ファイルシステムを作成したあとの作業...

ブート時にレガシー UFS ファイルシステムを自動的にマウントする場合は、[323 ページの「/etc/vfstab ファイルにエントリを追加する方法」](#)に進んでください。

## ▼ TMPFS ファイルシステムを作成およびマウントする方法

- 1 スーパーユーザーになります。
- 2 必要に応じて、TMPFS ファイルシステムとしてマウントするディレクトリを作成します。

```
mkdir /mount-point
```

/mount-point は、TMPFS ファイルシステムがマウントされるディレクトリです。

- 3 TMPFS ファイルシステムをマウントします。

```
mount -F tmpfs [-o size=number] swap mount-point
```

-o size=number      TMPFS ファイルシステムのサイズ制限を M バイト単位で示します。

mount-point      TMPFS ファイルシステムがマウントされるディレクトリを示します。

ブート時に TMPFS ファイルシステムが自動的にマウントされるようにシステムを設定する方法については、[例 15-4](#)を参照してください。

- 4 TMPFS ファイルシステムが作成されていることを確認します。

```
mount -v
```

### 例 15-3 TMPFS ファイルシステムを作成およびマウントする

次の例は、TMPFS ファイルシステム /export/reports を作成およびマウントし、そのサイズを 50M バイトに制限する方法を示しています。

```
mkdir /export/reports
chmod 777 /export/reports
```

```
mount -F tmpfs -o size=50m swap /export/reports
mount -v
```

#### 例 15-4 ブート時にTMPFS ファイルシステムをマウントする

ブート時にシステムが自動的に TMPFS ファイルシステムをマウントするように設定するには、`/etc/vfstab` のエントリを追加します。次の例は、ブート時に `/export/test` を TMPFS ファイルシステムとしてマウントする `/etc/vfstab` ファイルのエントリを示しています。 `size=number` オプションを指定していないため、`/export/test` の TMPFS ファイルシステムのサイズは利用できるシステムリソースによってのみ制限されます。

```
swap - /export/test tmpfs - yes -
```

## ▼ LOFS ファイルシステムを作成およびマウントする方法

- 1 スーパーユーザーになります。
- 2 必要に応じて、LOFS ファイルシステムとしてマウントするディレクトリを作成します。  

```
mkdir loopback-directory
```
- 3 新しく作成したディレクトリに対して、適切なアクセス権と所有権を設定します。
- 4 必要に応じて、LOFS ファイルシステムをマウントするマウントポイントを作成します。  

```
mkdir /mount-point
```
- 5 LOFS ファイルシステムをマウントします。  

```
mount -F lofs loopback-directory /mount-point
```

`loopback-directory` ループバックマウントポイントにマウントするファイルシステムを指定します。

`/mount-point` LOFS ファイルシステムをマウントするディレクトリを指定します。
- 6 LOFS ファイルシステムがマウントされていることを確認します。  

```
mount -v
```

**例 15-5 LOFS ファイルシステムを作成およびマウントする**

次の例は、新しいソフトウェアを、実際にはインストールしないで、ループバックファイルシステムとして `/new/dist` ディレクトリに作成、マウント、およびテストする方法を示しています。

```
mkdir /tmp/newroot
mount -F lofs /new/dist /tmp/newroot
chroot /tmp/newroot newcommand
```

**例 15-6 ブート時に LOFS ファイルシステムをマウントする**

ブート時にシステムが自動的に LOFS ファイルシステムをマウントするように設定するには、`/etc/vfstab` ファイルの最後にエントリを追加します。次の例は、ルート (`/`) ファイルシステムの LOFS ファイルシステムを `/tmp/newroot` にマウントする `/etc/vfstab` ファイルのエントリを示しています。

```
/ - /tmp/newroot lofs - yes -
```

ループバックファイルシステムのエントリは、`/etc/vfstab` ファイル内の最後のエントリでなければなりません。ループバックファイルシステムの `/etc/vfstab` エントリが、そこに組み込まれるファイルシステムよりも前にあると、ループバックファイルシステムをマウントできません。

**▼ /etc/vfstab ファイルにエントリを追加する方法**

レガシーのマウント動作が必要な ZFS ファイルシステムが存在しない場合は、ここに示す手順に従ってブート時に非 ZFS ファイルシステムをマウントします。ZFS ファイルシステムをマウントする方法の詳細は、『[Oracle Solaris ZFS 管理ガイド](#)』を参照してください。

- 1 スーパーユーザーになります。
- 2 必要に応じて、マウントするファイルシステムのマウントポイントを作成します。

```
mkdir /mount-point
```

ファイルシステムをマウントするには、ローカルシステム上にマウントポイントを作成する必要があります。「マウントポイント」とは、マウントされるファイルシステムが接続されるディレクトリのことです。

- 3 `/etc/vfstab` ファイルを編集し、エントリを追加します。次のことを確認してください。
  - a. 各フィールドを空白 (空白文字またはタブ) で区切ります。

- b. フィールドで値を指定しない場合はダッシュ (-) を入力します。
- c. 変更を保存します。

注-ルート (/) ファイルシステムは、ブートプロセスの過程でカーネルによって読み取り専用としてマウントされます。そのため、remount オプション (および、remount と一緒に使用できるオプション) だけが /etc/vfstab ファイルのルート (/) エントリでは有効です。

例 15-7 /etc/vfstab ファイルにエントリを追加する

次の例は、ディスクスライス /dev/dsk/c0t3d0s7 をレガシー UFS ファイルシステムとして、マウントポイント /files1 にマウントする方法を示しています。「device to fsck」として raw キャラクタ型デバイス /dev/rdsk/c0t3d0s7 を指定します。「fsck pass」の値が 2 なので、ファイルシステムは順不同で検査されます。

#device	device	mount	FS	fsck	mount	mount
#to mount	to fsck	point	type	pass	at boot	options
#						
/dev/dsk/c0t3d0s7	/dev/rdsk/c0t3d0s7	/files1	ufs	2	yes	-

次の例は、システム pluto 上のディレクトリ /export/man を、NFS ファイルシステムとしてマウントポイント /usr/man にマウントする方法を示しています。ファイルシステムが NFS であるため、「device to fsck」や「fsck pass」は指定されません。この例では、「mount options」は ro (読み取り専用) と soft になっています。

#device	device	mount	FS	fsck	mount	mount
#to mount	to fsck	point	type	pass	at boot	options
pluto:/export/man	-	/usr/man	nfs	-	yes	ro,soft

▼ 1つのファイルシステムをマウントする方法 (/etc/vfstab ファイル)

- 1 スーパーユーザーになります。
- 2 /etc/vfstab ファイル内に列挙されているファイルシステムをマウントします。

# mount /mount-point

/mount-point は、/etc/vfstab ファイル内の「mount point」または「device to mount」フィールドのエントリを指定します。通常は、マウントポイントを指定するほうが簡単です。

**例 15-8 1つのファイルシステムをマウントする (/etc/vfstab ファイル)**

次の例は、/etc/vfstab ファイル内に列挙されているローカル /legacy ファイルシステムをマウントする方法を示しています。

```
mount /legacy
```

**例 15-9 すべてのファイルシステムをマウントする (/etc/vfstab ファイル)**

次に、ファイルシステムがすでにマウントされている状態で mountall コマンドを使用したとき表示されるメッセージの例を示します。

```
mountall
mount: /tmp is already mounted or swap is busy
```

次の例は、/etc/vfstab ファイル内に列挙されているすべてのローカルシステムをマウントする方法を示しています。

```
mountall -l
```

次の例は、使用可能なすべての ZFS ファイルシステムをマウントする方法を示します。

```
zfs mount -a
```

次の例は、/etc/vfstab ファイル内に列挙されているすべてのリモートファイルシステムをマウントする方法を示しています。

```
mountall -r
```

## ▼ NFS ファイルシステムのマウント方法 (mount コマンド)

- 1 スーパーユーザーになります。
- 2 必要に応じて、マウントするファイルシステムのマウントポイントを作成します。

```
mkdir /mount-point
```

ファイルシステムをマウントするには、ローカルシステム上にマウントポイントを作成する必要があります。「マウントポイント」とは、マウントされるファイルシステムが接続されるディレクトリのことです。

- 3 リソース(ファイルまたはディレクトリ)がサーバーから使用可能かどうかを確認します。

NFS ファイルシステムをマウントするには、`share` コマンドを使用し、サーバー上のリソースを使用可能にしておかなければなりません。リソースを共有する方法については、『[Solaris のシステム管理 \(ネットワークサービス\)](#)』の「[NFS サービスについて](#)」を参照してください。

- 4 NFS ファイルシステムをマウントします。

```
mount -F nfs [-o mount-options] server:/directory /mount-point
```

#### 例 15-10 NFS ファイルシステムをマウントする (mount コマンド)

次の例は、サーバー `pluto` の `/export/packages` ディレクトリを `/mnt` にマウントする方法を示しています。

```
mount -F nfs pluto:/export/packages /mnt
```

## ▼ x86: ハードディスクから PCFS (DOS) ファイルシステムをマウントする方法 (mount コマンド)

次の手順で、PCFS (DOS) ファイルシステムをハードディスクからマウントします。

- 1 スーパーユーザーになります。
- 2 必要に応じて、マウントするファイルシステムのマウントポイントを作成します。

```
mkdir /mount-point
```

ファイルシステムをマウントするには、ローカルシステム上にマウントポイントを作成する必要があります。「マウントポイント」とは、マウントされるファイルシステムが接続されるディレクトリのことです。

- 3 PCFS ファイルシステムをマウントします。

```
mount -F pcfs [-o rw | ro] /dev/dsk/device-name:logical-drive /mount-point
```

`-o rw | ro` PCFS ファイルシステムを読み取り / 書き込み (rw) または読み取り専用 (ro) にマウントできることを指定します。このオプションを指定しない場合のデフォルトは `rw` です。

`/dev/dsk/device-name` ディスク全体のデバイス名を指定します (`/dev/dsk/c0t0d0p0` など)。

`logical-drive` DOS の論理ドライブ名 (c から z) またはドライブ番号 (1 から 24) を指定します。ドライブ c はドライブ 1 に相当し、ディスク上のプライマリ DOS スライスを表します。ほ

かのすべてのドライブ名やドライブ番号は、拡張 DOS スライス内の DOS 論理ドライブを表します。

*/mount-point*

ファイルシステムをマウントするディレクトリを指定します。

「device-name」と「logical-drive」は、コロンの必要があります。

#### 例 15-11 x86: ハードディスクから PCFS (DOS) ファイルシステムをマウントする (mount コマンド)

次の例は、プライマリ DOS スライス内の論理ドライブを */pcfs/c* ディレクトリにマウントする方法を示しています。

```
mount -F pcfs /dev/dsk/c0t0d0p0:c /pcfs/c
```

次の例では、*/mnt* ディレクトリの拡張 DOS スライスに含まれる最初の論理ドライブを読み取り専用でマウントする方法を示します。

```
mount -F pcfs -o ro /dev/dsk/c0t0d0p0:2 /mnt
```

## ▼ ファイルシステムを使用中のすべてのプロセスを終了させる方法

- 1 スーパーユーザーになります。
- 2 停止対象のプロセスを確認するために、ファイルシステムにアクセスしているすべてのプロセスを一覧表示します。

```
fuser -c [-u] /mount-point
```

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

*-u*                    プロセス ID ごとにユーザーのログイン名が表示されます。

*/mount-point*       プロセスを終了させるファイルシステムの名前を指定します。

- 3 ファイルシステムを使用しているすべてのプロセスを終了させます。

```
fuser -c -k /mount-point
```

ファイルシステムを使用している各プロセスに SIGKILL が送信されます。

---

注- ユーザーのプロセスを終了させるときには、必ず事前に警告してください。

---

- 4 ファイルシステムを使用しているプロセスがないことを確認します。

```
fuser -c /mount-point
```

#### 例 15-12 ファイルシステムを使用中のすべてのプロセスを終了させる

次の例は、/export/home ファイルシステムを使用中のプロセス 4006c を終了させる方法を示しています。

```
fuser -c /export/home
/export/home: 4006c
fuser -c -k /export/home
/export/home: 4006c
fuser -c /export/home
/export/home:
```

## ▼ 1つのファイルシステムをマウント解除する方法

次の手順を使って、ファイルシステムをマウント解除します。

- 1 [318 ページの「Oracle Solaris ファイルシステムをマウント解除するための前提条件」](#)の前提条件を満たしているかどうかを確認します。
- 2 ファイルシステムをマウント解除します。

```
umount /mount-point
```

/mount-point は、マウント解除するファイルシステムの名前を示します。

次のいずれかを指定できます。

- ファイルシステムがマウントされているディレクトリ名
- ファイルシステムのデバイス名パス
- NFS ファイルシステムのリソース
- LOFS ファイルシステムのループバックディレクトリ

#### 例 15-13 ファイルシステムをアンマウントする

次の例は、レガシー UFS ファイルシステムをマウント解除する方法を示しています。

```
umount /legacy
```



次の例は、UFS の `/legacy` ファイルシステムを強制的にマウント解除する方法を示しています。

```
umount -f /legacy
#
```

次の例は、すべての ZFS ファイルシステムをマウント解除する方法を示しています。

```
zfs umount -a
```

使用中のファイルシステムを除く、すべてのファイルシステムがマウント解除されます。



## 追加スワップ空間の構成 (タスク)

---

この章では、Oracle Solaris OS のインストール後に、ZFS ルートファイルシステムで追加スワップ空間を構成するためのガイドラインや手順について説明します。

この章の内容は次のとおりです。

- 331 ページの「スワップ空間について」
- 334 ページの「スワップ空間の追加が必要かどうかを調べる方法」
- 335 ページの「スワップ空間の割り当て方法」
- 335 ページの「スワップ空間の計画」
- 337 ページの「スワップリソースの監視」
- 338 ページの「Oracle Solaris ZFS ルート環境でのスワップ空間の追加または変更」

## スワップ空間について

次を決定するには、Oracle Solaris のスワップメカニズムの機能を理解してください。

- スワップ空間の要件
- スワップ空間と TMPFS ファイルシステムとの関係
- スワップ空間に関連するエラーメッセージから回復する方法

## スワップ空間と仮想メモリー

Oracle Solaris OS ソフトウェアやアプリケーションソフトウェアは、一部のディスク領域を、ファイルシステムとしてではなく一時記憶域として使用できます。ディスク上に予約される領域は、スワップ空間と呼ばれます。スワップ空間は、現在のプロセスを処理するだけの十分な物理メモリーがシステムにない場合に、仮想メモリー記憶域として使用されます。ZFS ルートファイルシステムの場合、スワップ用に予約されるディスク領域は、ZFS ボリュームになります。

仮想メモリーシステムは、ディスク上のファイルの物理コピーをメモリー内の仮想アドレスに対応付けます。これらのマッピングに関するデータが入った物理メモリーページは、ファイルシステム内の通常ファイルまたはスワップ空間から読み直されます。メモリーをバックアップしているディスク空間に割り当てられるIDはわからないため、スワップ空間から読み直されたメモリーは「匿名メモリー」と呼ばれます。

Oracle Solaris OS には、「仮想スワップ空間」という概念が導入されています。これは、匿名メモリーページとこれらのページを実際にバックアップする物理記憶域(またはディスク上にとられたバックアップ用のスワップ空間)の間に位置する層です。システムの仮想スワップ空間は、すべての物理(ディスク上にとられたバックアップ用のスワップ空間)スワップ空間と現在使用可能な物理メモリーの一部の合計に等しくなります。

仮想スワップ空間の長所は次のとおりです。

- 仮想スワップ空間が物理(ディスク)記憶域に対応していなくてもかまわないので、大きな物理スワップ空間を確保する必要がなくなります。
- SWAPFS という疑似ファイルシステムが、匿名メモリーページのアドレスを提供します。SWAPFS はメモリーページの割り当てを制御するので、ページに対する処理を柔軟に決定できます。たとえば、ディスク上にとられたバックアップ用のスワップ記憶域のページ要件を変更できます。

## スワップ空間と TMPFS ファイルシステム

Oracle Solaris 環境では、TMPFS ファイルシステムは `/etc/vfstab` ファイル内のエントリによって自動的に稼働されます。TMPFS ファイルシステムは、ファイルとそれに関連付けられた情報をディスクではなくメモリー(`/tmp` ディレクトリ)に格納するので、これらのファイルへのアクセスが高速になります。この機能によって、コンパイラや DBMS 製品のように `/tmp` の使用量の大きいアプリケーションの場合は、パフォーマンスが大幅に改善されます。

TMPFS ファイルシステムは、システムのスワップリソースから `/tmp` ディレクトリ内の領域を割り当てます。つまり、`/tmp` ディレクトリ内の領域を使い果たすと、スワップ空間も使い果たしたことになります。したがって、`/tmp` ディレクトリの使用量が大きいアプリケーションの場合は、スワップ空間の使用状況を監視しなければ、システムがスワップ空間を使い果たす可能性があります。

スワップリソースが限られているときに TMPFS を使用する場合は、次の方法を使用してください。

- サイズオプション (-o size) を指定して TMPFS ファイルシステムをマウントし、TMPFS が使用できるスワップリソースを制御します。
- より大きな別のディレクトリをポイントするように、コンパイラの TMPDIR 環境変数の設定を変更します。

コンパイラの TMPDIR 変数を使用すると、コンパイラが /tmp ディレクトリを使用するかどうかだけが制御されます。この変数は、ほかのプログラムによる /tmp ディレクトリの使用には影響しません。

## スワップ空間と動的再構成

動的再構成時に CPU やシステムボードで障害が発生した場合に対応できるよう、十分なスワップ空間を確保することをお勧めします。スワップ空間が不足していると、CPU やシステムボードで障害が発生した際に、ホストまたはドメインはより少ないメモリーでリブートしなければなりません。

この追加スワップ空間を使用できない場合、メモリー不足のためアプリケーションの起動に失敗する可能性があります。この問題が発生した場合は、ユーザーが介入して、スワップ空間を追加するか、起動に失敗したアプリケーションのメモリー使用の構成を変更する必要があります。

リブート時のメモリー破損に備えて十分な追加スワップ空間が確保されていた場合、メモリーを大量に消費するアプリケーションのすべてが、通常どおりに起動します。したがって、スワップの多発によりシステムの動作は多少遅くなるにしても、ユーザーは通常どおりにシステムを利用できます。

詳細は、使用するハードウェアの動的構成ガイドを参照してください。

## SAN 環境でのスワップ空間の構成

SAN 環境などで、ネットワークで接続されたディスクにスワップ空間を構成するかどうかを判断するには、次の点を確認してください。

- ローカルに接続されたディスクのスワップ空間の問題を診断する方が、ネットワークで接続されたディスクのスワップ空間の問題を診断するより簡単です。
- SAN 経由のスワップ空間のパフォーマンスは、ローカルに接続されたディスクに構成されているスワップ空間と同等になります。
- システムのパフォーマンスに問題がある場合は、パフォーマンスデータを分析したあとでシステムにメモリーを追加する方が、ローカルに接続されたディスクにスワップを移動するよりも、SAN 経由のスワップのパフォーマンス問題をより適切に解決できる可能性があります。

## スワップ空間の追加が必要かどうかを調べる方法

スワップ空間の追加が必要かどうかを調べるには、`swap -l` コマンドを使用します。

たとえば、次の `swap -l` コマンドの出力は、このシステムのスワップ空間がほぼ完全に使い果たされたか、割り当て率が 100% に達していることを示しています。

```
% swap -l
swapfile dev swaplo blocks free
/dev/dsk/c0t0d0s1 136,1 16 1638608 88
```

システムのスワップ空間の割り当て率が 100% に達すると、アプリケーションのメモリーページが一時的にロックしてしまいます。アプリケーションエラーが発生しない場合でも、システムパフォーマンスが低下する可能性があります。

## スワップ関連のエラーメッセージ

次のメッセージは、アプリケーションに匿名メモリーを追加する必要がある場合に表示されます。ただし、バックアップ用のスワップ空間はもう余っていません。

```
application is out of memory
```

```
malloc error 0
```

```
messages.1:Sep 21 20:52:11 mars genunix: [ID 470503 kern.warning]
WARNING: Sorry, no swap space to grow stack for pid 100295 (myprog)
```

## TMPFS 関連のエラーメッセージ

次のメッセージは、ファイルに書き込むときにページを割り当てることができない場合に表示されます。この問題は、TMPFS が許容限度を超えて書き込もうとしたときや、現在実行されているプログラムが大量のメモリーを使用している場合に発生することがあります。

```
directory: File system full, swap space limit exceeded
```

次のメッセージは、TMPFS が新しいファイルやディレクトリの作成中に物理メモリーを使い果たしたことを意味します。

```
directory: File system full, memory allocation failed
```

TMPFS 関連のエラーメッセージから回復する方法については、[tmpfs\(7FS\)](#) のマニュアルページを参照してください。

## スワップ空間の割り当て方法

最初に、スワップ空間は Oracle Solaris インストールプロセスの一部として割り当てられます。ZFS ルート環境では、デフォルトのスワップサイズは通常 512M バイトから 2G バイトの範囲です。

スワップ空間の割り当て方法については、[335 ページの「スワップ空間の計画」](#)を参照してください。

## スワップ空間と `/etc/vfstab` ファイル

システムのインストールが完了すると、`/etc/vfstab` ファイルにスワップ領域とスワップファイルが一覧表示されます。スワップスライスとスワップファイルは、システムのブート時に `/sbin/swapadd` スクリプトによって有効になります。

`/etc/vfstab` ファイル内のスワップデバイスエントリには、次の情報が入っています。

- UFS ルートファイルシステムを備えるシステム上のスワップスライスまたはスワップファイルのフルパス名、あるいは ZFS ルートファイルシステムを備えるシステム上のスワップボリュームのパス名
- スワップスライスまたはスワップファイルのファイルシステムタイプ

スワップファイルが入っているファイルシステムは、スワップファイルが有効になる前にマウントしておかなければなりません。このため、`/etc/vfstab` ファイル内で、ファイルシステムをマウントするエントリが、スワップファイルを有効にするエントリより前に入っていることを確認してください。

## スワップ空間の計画

スワップ空間のサイズを決定する上でもっとも重要な要素は、システムのソフトウェアアプリケーションの要件です。たとえば、コンピュータ支援設計シミュレータ、データベース管理製品、トランザクションモニター、地質分析システムなどの大型アプリケーションは、200 - 1000M バイトのスワップ空間を消費することがあります。

アプリケーションのスワップ空間の要件については、アプリケーションベンダーに問い合わせてください。

アプリケーションベンダーに問い合わせてもスワップ空間の要件を判断できない場合は、次のシステムタイプ別のガイドラインに従ってスワップ空間を割り当ててください。

注-クラッシュダンプの内容は圧縮されるため、ダンプデバイスを物理メモリーと同じサイズにする必要はありません。デフォルトでは、ダンプ内容の値はカーネルページに設定されます。ただし、すべてのメモリーページをダンプするようにダンプ内容の値が設定されている場合は、ダンプサイズを物理メモリーの半分以上に増やすことを検討してください。

表 16-1 ZFS ファイルシステムのスワップボリュームとダンプボリュームのサイズ

システムタイプ	スワップボリュームのサイズ	ダンプボリュームのサイズ
約 4G バイトの物理メモリーを備えたシステム	1G バイト	2G バイト
約 8G バイトの物理メモリーを備えた中型のサーバー	2G バイト	4G バイト
約 16 - 128G バイトの物理メモリーを備えたハイエンドサーバー	4G バイト	8 - 64G バイト
128G バイトを超える物理メモリーを備えたハイエンドサーバー	物理メモリーサイズの 1/4	物理メモリーサイズの 1/2

注-アクティブな ZFS ファイルシステムを多数備えたビジー状態のシステムでは、物理メモリーの 1/2 から 3/4 のサイズがダンプデバイスのサイズとして使用される可能性があります。

ZFS ルートファイルシステムを含むシステムで、サイズが小さすぎてシステムクラッシュを保持できないようなダンプデバイスを `dumpadm -d` コマンドで指定しようとすると、次のようなメッセージが表示されます。

```
dumpadm: dump device /dev/zvol/dsk/rpool/dump is too small to hold a
system dump dump size 43467329536 bytes, device size 42949672960 bytes
```

## ZFS ベースのシステムでのスワップ空間の割り当て

ZFS ルートファイルシステムの初期インストール時に、スワップ領域が ZFS ルートプール内の ZFS ボリュームに自動的に作成されます。

ZFS ルートプールでは、スワップデバイスが固定サイズのスライスに事前に割り当てられることはないため、あとでスワップサイズを変更するのは非常に簡単です。



アプリケーションのスワップ要件を評価してから、初期インストール時またはインストール後に必要に応じて、デフォルトのスワップサイズを使用することも、スワップボリュームのサイズを調整することもできます。

初期インストール時のデフォルトのダンプボリュームサイズは、`dumpadm` の情報と物理メモリーのサイズに基づいて、カーネルにより計算されます。

ZFS 環境では、ファイルシステムはプールの領域を消費するため、`/var/crash` ディレクトリは、保存するクラッシュダンプの数に応じて必要な容量を消費します。

## スワップリソースの監視

`/usr/sbin/swap` コマンドを使用してスワップ領域を管理します。2つのオプション `-l` と `-s` は、スワップリソースに関する情報を表示します。

`swap -l` コマンドを使用すると、システムのスワップ領域を確認できます。有効になっているスワップデバイスやファイルは、`swapfile` カラムの下に表示されます。例:

```
swap -l
swapfile dev swaplo blocks free
/dev/dsk/c0t0d0s1 136,1 16 1638608 1600528
```

ZFS ルートファイルシステムを備えるシステムでは、`swap -l` コマンドにより、ZFS ボリュームパス名が出力されることを除いて、類似の出力が得られます。例:

```
swap -l
swapfile dev swaplo blocks free
/dev/zvol/dsk/rpool/swap 256,1 16 1058800 1058800
```

`swap -s` コマンドを使用すると、スワップリソースを監視できます。

```
swap -s
total: 57416k bytes allocated + 10480k reserved = 67896k used,
833128k available
```

`used` 値と `available` 値の合計は、システム上の合計スワップ空間に等しくなります。これには、物理メモリーの一部とスワップデバイス(またはファイル)が含まれます。

使用可能なスワップ空間と使用済みスワップ空間の容量 (`swap -s` の出力内) を使用して、時間経過に伴うスワップ空間の使用状況を監視できます。システムパフォーマンスが適正であれば、`swap -s` を使用するとどの程度のスワップ空間が使用可能であるかがわかります。システムパフォーマンスが低下したときは、使用可能なスワップ空間の容量を検査して減少していないかどうかを調べてください。これによって、システムに対するどのような変更が原因でスワップ空間の使用量が増大したかを識別できます。

このコマンドを使用するときには、カーネルとユーザープロセスが物理メモリーをロックして解除するたびに、スワップに使用できる物理メモリーの容量が動的に変化するので注意してください。

注 – `swap -l` コマンドでは、スワップ空間が 512 バイトのブロック数として表示されます。`swap -s` コマンドでは、スワップ空間が 1024 バイトのブロック数として表示されます。`swap -l` で表示されたブロック数を合計して K バイトに変換すると、結果は「used + available」(`swap -s` の出力) より小さくなります。これは、`swap -l` では、スワップ空間の計算に物理メモリーが算入されないからです。

次の表に、`swap -s` コマンドの出力とその説明を示します。

表 16-2 `swap -s` コマンドの出力

キーワード	説明
bytes allocated	現在バックアップ用の記憶域(ディスク上にとられたバックアップ用のスワップ空間)として使用可能なスワップ空間の合計容量を表す 1024 バイトのブロック数。
reserved	現在は割り当てられていないが、あとから使用できるようにメモリーによって回収されるスワップ空間の合計容量を表す 1024 バイトのブロック数。
used	割り当て済みまたは予約済みのスワップ空間の合計容量を表す 1024 バイトのブロック数。
available	あとから予約や割り当てに使用可能なスワップ空間の合計容量を表す 1024 バイトのブロック数。

# Oracle Solaris ZFS ルート環境でのスワップ空間の追加または変更

次のセクションでは、ZFS ルート環境でスワップ空間を追加または変更する方法を説明します。システムまたはアプリケーションで追加のスワップ空間が必要かどうかを確認する方法については、前述のセクションを参照してください。

ZFS スワップまたはダンプデバイスのカスタマイズについては、『Oracle Solaris ZFS 管理ガイド』の「ZFS スワップボリュームとダンプボリュームをカスタマイズする」を参照してください。

## ▼ Oracle Solaris ZFS ルート環境でスワップ空間を追加する方法

- 1 スーパーユーザーになります。

- 2 現在のスワップボリュームを確認します。

使用中のスワップボリュームは削除できません。現在のスワップボリュームが使用中かどうかは、blocks 列に示されるブロックと free 列に示されるブロックを比較して確認できます。2つの列のブロックが等しい場合、スワップ領域は使用中ではありません。例:

```
swap -l
swapfile dev swaplo blocks free
/dev/zvol/dsk/rpool/swap 256,1 16 1058800 1058800
```

- 3 次のいずれかを選択して、スワップボリュームのサイズを変更します。

- a. 現在のスワップ領域が使用中でなければ、現在のスワップボリュームのサイズを変更できますが、増加したスワップ空間を確認するには、システムをリブートする必要があります。

例:

```
zfs get volsize rpool/swap
NAME PROPERTY VALUE SOURCE
rpool/swap volsize 517M -
zfs set volsize=2g rpool/swap
zfs get volsize rpool/swap
NAME PROPERTY VALUE SOURCE
rpool/swap volsize 2G -
init 6
```

- b. システムをリブートできない場合、別のスワップボリュームを追加して、総スワップ空間を増やします。

例:

```
zfs create -V 2G rpool/swap2
```

2 番目のスワップボリュームをアクティブにします。

```
swap -a /dev/zvol/dsk/rpool/swap2
swap -l
swapfile dev swaplo blocks free
/dev/zvol/dsk/rpool/swap 256,1 16 1058800 1058800
/dev/zvol/dsk/rpool/swap2 256,3 16 4194288 4194288
```

- 4 必要に応じて、`/etc/vfstab` ファイルに2つ目のスワップボリュームのエントリを追加します。

例:

```
/dev/zvol/dsk/rpool/swap2 - - swap - no -
```



## UFS ファイルシステムの整合性検査 (手順)

---

この章では、UFS ファイルシステムの整合性検査に関する概要と手順について説明します。

この章で説明する手順は次のとおりです。

- 358 ページの「代替ブートデバイスから UFS ルート (/) ファイルシステム、/usr ファイルシステム、または /var ファイルシステムを検査する方法」
- 360 ページの「ルート (/)、/usr、または /var 以外の UFS ファイルシステムを検査する方法」
- 362 ページの「UFS ファイルシステムを修復する方法」
- 368 ページの「不正なスーパーブロックを復元する方法 (Solaris 8、9、および 10 リリース)」
- 365 ページの「不正な UFS スーパーブロックを復元する方法 (Solaris 10 6/06 リリース)」

この章の内容は以下のとおりです。

- 342 ページの「UFS ファイルシステムの整合性」
- 349 ページの「UFS ファイルシステムの状態はどのように記録されるか」
- 350 ページの「fsck コマンドで検査して修復される内容」
- 357 ページの「UFS ファイルシステムを対話式で検査して修復する」
- 364 ページの「不正な UFS スーパーブロックの復元」
- 369 ページの「fsck コマンドの構文とオプション」

fsck のエラーメッセージについては、『Solaris のシステム管理 (上級編)』の第 20 章「UFS ファイルシステムの不整合解決 (手順)」を参照してください。

この章で取り上げる UFS ファイルシステム構造のバックグラウンド情報については、第 18 章「UFS ファイルシステム (参照情報)」を参照してください。

## UFS ファイルシステムの整合性

UFS ファイルシステムは、一連の内部テーブルを基にして使用済み i ノード、使用可能ブロックを特定します。これらの内部テーブルがディスク上のデータと正しく同期していないと、整合性が失われ、ファイルシステムの修復が必要になります。

次のような原因でオペレーティングシステムが異常終了すると、ファイルシステムの整合性が失われることがあります。

- 電源障害
- 不注意によるシステム電源の切断
- 正しいシャットダウン手順以外の方法によるシステム電源の切断
- カーネル内のソフトウェアエラー

ファイルシステムの整合性が失われることは重大ですが、頻繁に起きるものではありません。システムをブートすると、ファイルシステムの整合性検査が (`fsck` コマンドを使用して) 自動的に実行されます。ほとんどの場合は、このファイルシステムの検査によって問題が修復されます。

`fsck` コマンドは、ファイルシステム上に配置されているが参照不可能なファイルとディレクトリを `lost+found` ディレクトリに入れます。参照不可能なファイルとディレクトリの名前として i ノード番号が割り当てられます。`lost+found` ディレクトリが存在しない場合は、`fsck` コマンドによって作成されます。`lost+found` ディレクトリ内の領域が足りない場合は、`fsck` コマンドによってそのサイズが拡張されます。

i ノードについては、[372 ページの「i ノード」](#)を参照してください。

## UFS ファイルシステムユーティリティー (`fsck`、`mkfs`、および `newfs`) の拡張機能

ファイルシステム検査ユーティリティー `fsck` は、FreeBSD 4.9 バージョンの `fsck` プログラム内の機能やその他の拡張機能が含まれるように拡張されています。

この Oracle Solaris リリースの `fsck` ユーティリティーには次の改善が加えられています。

- ファイルシステムをより詳しく検査して修復し、改善されたエラーメッセージを表示します。たとえばいくつかのシナリオにおいて、`fsck` は失われている構造を特定し、それらを適切に置き換えます。
- バックアップスーパーブロックを自動的に検索します。
- `fsck` を再実行する必要があるタイミングを報告します。
- ディレクトリが消去された場合は、`fsck` が直ちにディレクトリの内容の復元を試みるので、このユーティリティーの再実行時間が短縮されます。

- `fsck` が重複するブロックを検出し、重複するブロックを参照するすべてのファイルがクリアされなかった場合、`fsck` は、`fsck` の実行の最後に `i` ノード番号を報告します。そのあとで、`find` コマンドを使用して破損した `i` ノードを確認できます。
- デバイスファイルや ACL エントリなどの拡張属性およびほかの特別なファイルのステータスに関連する改善されたエラーメッセージが含まれています。
- より詳細なメッセージを使用できるようにする `-v` オプションが用意されています。

さらに、`newfs` コマンドと `mkfs` コマンドが更新され、ファイルシステムのスーパーブロック情報をテキストで表示したり、スーパーブロック情報をバイナリ形式でダンプしたりするための新しいオプションが追加されました。

`newfs [ -S or -B ] /dev/rdsk/...`

`-S`    ファイルシステムのスーパーブロックをテキスト形式で表示します。

`-B`    ファイルシステムのスーパーブロックをバイナリ形式でダンプします。

`mkfs [ -o calcsb or -o calcbinsb ] /dev/rdsk/... size`

`-o calcsb`      ファイルシステムのスーパーブロックをテキスト形式で表示します。

`-o calcbinsb`    ファイルシステムのスーパーブロックをバイナリ形式でダンプします。

`fsck` ユーティリティーは、このスーパーブロック情報を使用して、バックアップスーパーブロックを検索します。

次のセクションでは、具体的な `fsck` の機能拡張および対応するエラーメッセージについて説明します。`fsck` ユーティリティーを使用して破損したスーパーブロックを修復する手順については、[365 ページの「不正な UFS スーパーブロックを復元する方法 \(Solaris 10 6/06 リリース\)」](#)を参照してください。

## バックアップスーパーブロックの自動検索

次の `fsck` エラーメッセージの例は、バックアップスーパーブロックの自動発見の失敗を示しています。



注意 - ファイルシステムに破損したスーパーブロックが含まれ、`ntrack` や `nsect` などの `newfs` または `mkfs` のカスタマイズされたパラメータを使用してファイルシステムが作成されている場合、修復処理のために自動的に発見されたスーパーブロックがファイルシステムを破損させる可能性があります。

カスタマイズされたパラメータを使用して作成されたファイルシステムで、ファイルシステムに不正なスーパーブロックが含まれている場合、`fsck` は、`fsck` セッションを取り消すための次のようなプロンプトを表示します。

---

#### CANCEL FILESYSTEM CHECK?

---

- ファイルシステムが `newfs` コマンドを使用して作成され、`fsck` がプライマリスーパーブロックのみが破損していることを報告する場合、`fsck` でスーパーブロックを復元することを検討してください。

```
fsck /dev/dsk/c1t2d0s0
** /dev/rdisk/c1t2d0s0
BAD SUPERBLOCK AT BLOCK 16: BLOCK SIZE LARGER THAN MAXIMUM SUPPORTED

LOOK FOR ALTERNATE SUPERBLOCKS WITH MKFS? no

LOOK FOR ALTERNATE SUPERBLOCKS WITH NEWFS? yes

FOUND ALTERNATE SUPERBLOCK 32 WITH NEWFS

USE ALTERNATE SUPERBLOCK? yes

FOUND ALTERNATE SUPERBLOCK AT 32 USING NEWFS
If filesystem was created with manually-specified geometry, using
auto-discovered superblock may result in irrecoverable damage to
filesystem and user data.

CANCEL FILESYSTEM CHECK? no

** Last Mounted on
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3a - Check Connectivity
** Phase 3b - Verify Shadows/ACLs
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cylinder Groups
CORRECT GLOBAL SUMMARY
SALVAGE? y

UPDATE STANDARD SUPERBLOCK? y

81 files, 3609 used, 244678 free (6 frags, 30584 blocks, 0.0%
fragmentation)

***** FILE SYSTEM WAS MODIFIED *****
```



- ファイルシステムが `mkfs` コマンドを使用して作成され、`fsck` がプライマリスーパーブロックのみが破損していることを報告する場合、`fsck` でスーパーブロックを復元することを検討してください。

```
fsck /dev/dsk/c1t2d0s0
** /dev/rdisk/c1t2d0s0
BAD SUPERBLOCK AT BLOCK 16: BLOCK SIZE LARGER THAN MAXIMUM SUPPORTED

LOOK FOR ALTERNATE SUPERBLOCKS WITH MKFS? yes

FOUND ALTERNATE SUPERBLOCK 32 WITH MKFS

USE ALTERNATE SUPERBLOCK? yes

FOUND ALTERNATE SUPERBLOCK AT 32 USING MKFS
If filesystem was created with manually-specified geometry, using
auto-discovered superblock may result in irrecoverable damage to
filesystem and user data.

CANCEL FILESYSTEM CHECK? no

** Last Mounted on
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3a - Check Connectivity
** Phase 3b - Verify Shadows/ACLs
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cylinder Groups
CORRECT GLOBAL SUMMARY
SALVAGE? y

UPDATE STANDARD SUPERBLOCK? y

81 files, 3609 used, 243605 free (117 frags, 30436 blocks, 0.0%
fragmentation)

***** FILE SYSTEM WAS MODIFIED *****
```

次の例は、破損したスーパーブロックのシナリオで `fsck` の `-y` オプションを指定した場合に何が起こるかを示しています。`fsck` セッションが自動的に終了します。代替のスーパーブロックを使用してセッションを再実行するためのメッセージが表示されます。

```
fsck -y /dev/dsk/c1t2d0s0
#
** /dev/rdisk/c1t2d0s0
BAD SUPERBLOCK AT BLOCK 16: BLOCK SIZE LARGER THAN MAXIMUM SUPPORTED

LOOK FOR ALTERNATE SUPERBLOCKS WITH MKFS? yes

LOOK FOR ALTERNATE SUPERBLOCKS WITH NEWFS? yes

SEARCH FOR ALTERNATE SUPERBLOCKS FAILED.
```

USE GENERIC SUPERBLOCK FROM MKFS? yes

CALCULATED GENERIC SUPERBLOCK WITH MKFS

If filesystem was created with manually-specified geometry, using auto-discovered superblock may result in irrecoverable damage to filesystem and user data.

CANCEL FILESYSTEM CHECK? yes

Please verify that the indicated block contains a proper superblock for the filesystem (see fsdb(1M)).

FSCK was running in YES mode. If you wish to run in that mode using the alternate superblock, run 'fsck -y -o b=453920 /dev/rdisk/c1t2d0s0'.

次の fsck エラーメッセージのシナリオは、バックアップスーパーブロックの新しいプロンプトを示していますが、この例では、fsck の実行は取り消されています。ファイルシステムがカスタマイズされたパラメータを使用して作成されている場合またはこのシステム上での fsck の実行に関する他の心配がある場合は、fsck セッションの取り消しが適切な応答です。

次のように、さまざまなスーパーブロックのエラー状況が斜体で表示されます。

# fsck /dev/rdisk/c0t1d0s0

\*\* /dev/rdisk/c0t1d0s0

*BAD SUPERBLOCK AT BLOCK 16: BLOCK SIZE LARGER THAN MAXIMUM SUPPORTED  
BAD SUPERBLOCK AT BLOCK 16: NUMBER OF DATA BLOCKS OUT OF RANGE  
BAD SUPERBLOCK AT BLOCK 16: INODES PER GROUP OUT OF RANGE  
BAD SUPERBLOCK AT BLOCK 16: MAGIC NUMBER WRONG  
BAD SUPERBLOCK AT BLOCK 16: BAD VALUES IN SUPER BLOCK  
BAD SUPERBLOCK AT BLOCK 16: NCG OUT OF RANGE  
BAD SUPERBLOCK AT BLOCK 16: CPG OUT OF RANGE  
BAD SUPERBLOCK AT BLOCK 16: NCYL IS INCONSISTENT WITH NCG\*CPG  
BAD SUPERBLOCK AT BLOCK 16: SIZE OUT OF RANGE  
BAD SUPERBLOCK AT BLOCK 16: NUMBER OF DIRECTORIES OUT OF RANGE  
BAD SUPERBLOCK AT BLOCK 16: ROTATIONAL POSITION TABLE SIZE OUT OF RANGE  
BAD SUPERBLOCK AT BLOCK 16: SIZE OF CYLINDER GROUP SUMMARY AREA WRONG  
BAD SUPERBLOCK AT BLOCK 16: INOPB NONSENSICAL RELATIVE TO BSIZE*

LOOK FOR ALTERNATE SUPERBLOCKS WITH MKFS? yes

FOUND ALTERNATE SUPERBLOCK 32 WITH MKFS

USE ALTERNATE SUPERBLOCK? yes

FOUND ALTERNATE SUPERBLOCK AT 32 USING MKFS

If filesystem was created with manually-specified geometry, using auto-discovered superblock may result in irrecoverable damage to filesystem and user data.

CANCEL FILESYSTEM CHECK? no

\*\* Last Mounted on

\*\* Phase 1 - Check Blocks and Sizes

```

** Phase 2a - Check Duplicated Names
** Phase 2b - Check Pathnames
** Phase 3a - Check Connectivity
** Phase 3b - Verify Shadows/ACLs
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cylinder Groups

```

SALVAGE? **yes**

UPDATE STANDARD SUPERBLOCK? **yes**

82 files, 3649 used, 244894 free (6 frags, 30611 blocks, 0.0% fragmentation)

\*\*\*\*\* FILE SYSTEM WAS MODIFIED \*\*\*\*\*

## fsck による再実行が必要なタイミングの報告

fsck によって再実行が必要なタイミングの報告が改善されるので、特に実行に時間がかかる大きなファイルシステムの場合などに、このコマンドを何度も実行するための時間が節約され、実行する必要性も少なくなります。

次の新しいメッセージは、エラーシナリオの最後に fsck ユーティリティを再実行するように要求します。

\*\*\*\*\* PLEASE RERUN FSCK \*\*\*\*\*

または

Please rerun fsck(1M) to correct this.

これらの新しいプロンプトは、fsck を再実行すべきかどうかの判断が難しいという以前の問題を解決します。

上のメッセージのように fsck の再実行を要求されない場合は、次のようなメッセージが表示されても fsck を再実行する必要はありません。

\*\*\*\*\* FILE SYSTEM WAS MODIFIED \*\*\*\*\*

ただし、このメッセージのあとに fsck を再実行してもファイルシステムは損傷しません。このメッセージは、fsck の修正処置に関する情報を示すだけのものです。

## 拡張属性に関連する新しい fsck メッセージ

拡張属性を含むファイルを報告および修復する新しい fsck メッセージが用意されています。例:

BAD ATTRIBUTE REFERENCE TO I=1 FROM I=96

```
Attribute directory I=97 not attached to file I=96
I=96 OWNER=root MODE=40755
SIZE=512 MTIME=Jul 12 10:55:10 2010
DIR= <xattr>
```

FIX? **yes**

```
ZERO LENGTH ATTR DIR I=12 OWNER=root MODE=160755
SIZE=0 MTIME=Jul 12 10:56:10 2010
CLEAR? yes
```

```
File should BE marked as extended attribute
I=22 OWNER=root MODE=100644
SIZE=0 MTIME=Jul 12 10:57:10 2010
FILE= <xattr>
```

FIX? **yes**

```
UNREF ATTR DIR I=106 OWNER=root MODE=160755
SIZE=512 MTIME=Jul 12 10:58:10 2010
RECONNECT? yes
```

```
File I=107 should NOT be marked as extended attribute
I=107 OWNER=root MODE=100644
SIZE=0 MTIME=Jul 12 10:59:10 2010
FILE=?/attfsdir-7-att
```

FIX? **yes**  
DIR I=106 CONNECTED.

## 重複するブロックまたはフラグメントの処理の改善

fsck エラーメッセージは、ブロック、フラグメント、または LFN に関する情報を報告するようになりました。LFN は、ファイルの先頭からの論理フラグメント番号です。たとえば、次のような出力が表示されます。

```
** Phase 1 - Check Blocks and Sizes
FRAGMENT 784 DUP I=38 LFN 0
FRAGMENT 785 DUP I=38 LFN 1
FRAGMENT 786 DUP I=38 LFN 2
.
.
.
```

fsck はオブジェクトをフラグメントとして処理しますが、以前の Solaris リリースでは、オブジェクト情報をブロックとして報告するだけでした。現在は、フラグメントとして正しく報告します。

fsck が重複するブロックまたはフラグメントに関連するエラーを検出した場合、fsck は、fsck の出力の最後にクリアされないファイルを表示します。たとえば、次のような出力が表示されます。

```
LIST REMAINING DUPS? yes

Some blocks that were found to be in multiple files are still
assigned to file(s).
Fragments sorted by inode and logical offsets:
Inode 38:
 Logical Offset 0x00000000 Physical Fragment 784
 Logical Offset 0x00000800 Physical Fragment 786
 Logical Offset 0x00001000 Physical Fragment 788
 Logical Offset 0x00001800 Physical Fragment 790
```

この例では、そのあとで、`find -i inode-number` コマンドを使用して i ノード 38 を識別します。

## UFS ファイルシステムの状態はどのように記録されるか

`fsck` コマンドは、スーパーブロックに保存される状態タグを使用してファイルシステムの状況を記録します。また、このフラグを使用して、ファイルシステムの整合性を検査する必要があるか判断します。このフラグは、ブート時に `/sbin/rcS` スクリプトによって使用されるか、`fsck -m` コマンドによって使用されます。`fsck -m` コマンドの結果を無視する場合は、状態フラグの設定に関係なく、すべてのファイルシステムを検査できます。

スーパーブロックについては、[372 ページの「スーパーブロック」](#)を参照してください。

次の表に、状態フラグが取り得る値とその説明を示します。

表 17-1 UFS ファイルシステムの状態フラグの値

状態フラグの値	説明
FSACTIVE	このファイルシステムがマウント済みであり、メモリー内のデータが変更されたことを示します。この状態フラグを持つマウント済みファイルシステムでは、システムの電源が切断されると、ユーザーデータまたはメタデータが失われます。
FSBAD	このファイルシステムに不整合なファイルシステムデータが含まれていることを示します。
FSCLEAN	このファイルシステムが正常にマウント解除されており、破損していないことを示します。
FSLOG	このファイルシステムのロギングが有効になっていることを示します。このフラグが設定されたファイルシステムは、マウントされている場合も、マウント解除されている場合もあります。ロギングが有効になっているファイルシステムに設定できるフラグは、FSLOG、FSBAD のいずれかです。ロギングが無効になっているファイルシステムに設定できるフラグは、FSACTIVE、FSSTABLE、FSCLEAN のいずれかです。

表 17-1 UFS ファイルシステムの状態フラグの値 (続き)

状態フラグの値	説明
FSSTABLE	このファイルシステムがマウント済みであり、アイドル状態であることを示します。この状態フラグを持つマウント済みファイルシステムでは、システムの電源が切断されても、ユーザーデータやメタデータが失われることはありません。

## fsck コマンドで検査して修復される内容

このセクションでは、ファイルシステムの通常の処理中に発生する問題、原因、fsck コマンド (検査および修復ユーティリティ) で検出される問題、およびそれらの修正方法について説明します。

### UFS ファイルシステムの不整合が発生する理由

就業日には毎日多数のファイルが作成、変更、または削除されます。ファイルが変更されるたびに、オペレーティングシステムは一連のファイルシステムの更新処理を実行します。これらの更新処理がディスクに確実に書き込まれると、ファイルシステムの整合性が保たれます。

ユーザープログラムが書き込みなどの、ファイルシステムを変更する処理を実行すると、書き込まれるデータはまずカーネルのインコアバッファーにコピーされます。一般に、ディスクの更新は非同期に処理されます。このため、ユーザープロセスは、書き込みシステムコールが値を返したあとすぐに処理を続けることができますが、実際のデータの書き込みは、ずいぶんあとに実行されることもあります。したがって、ディスク上にあるファイルシステムは、インコア情報で表されるファイルシステムの状態から常に遅延することになります。

別の目的にバッファーが必要になったり、カーネルが `fsflush` デーモンを自動的に (30 秒間隔で) 実行すると、インコア情報を反映するようにディスク情報が更新されます。システムがインコア情報を書き込まずに停止すると、ディスク上のファイルシステムの整合性がなくなります。

ファイルシステムの整合性は、さまざまな原因で失われることがあります。もっとも一般的な原因は、オペレータのエラーとハードウェア障害です。

システムを正しくシャットダウンしなかったり、マウントされているファイルシステムが正しくオフラインにされないと、「クリーンでない停止」が原因で問題が発生することがあります。クリーンでないシャットダウンを防ぐには、システムをシャットダウンしたり、ディスクをドライブから物理的に取り出したり、ディスクをオフライン状態にしたりする前に、ファイルシステムの現在の状態をディスクに書き込まなければなりません。つまり、「同期」させなければなりません。

このほか、ハードウェアの欠陥や、ディスクまたはコントローラのファームウェアの問題が原因で整合性が失われることもあります。ディスクドライブ上ではいつでもブロックが損傷を受ける可能性があります。あるいは、ディスクコントローラが正常に機能しなくなることもあります。

## 整合性が検査される UFS コンポーネント

このセクションでは、UFS ファイルシステムのコンポーネント、つまりスーパーブロック、シリンダグループブロック、i ノード、間接ブロック、データブロックに fsck コマンドが適用する整合性検査の種類について説明します。

UFS ファイルシステム構造については、[371 ページの「UFS ファイルシステムのシリンダグループの構造」](#)を参照してください。

### スーパーブロックの検査

スーパーブロックにはサマリー情報が格納されており、UFS ファイルシステム内でもっとも破損しがちなコンポーネントです。ファイルシステムの i ノードやデータブロックが変更されるたびに、スーパーブロックも変更されます。CPU が停止した場合、直前のコマンドが sync コマンドでなければ、スーパーブロックはほぼ確実に破損しています。

スーパーブロックの不整合は、次の面から検査されます。

- ファイルシステムのサイズ
- i ノード数
- 空きブロック数
- 空き i ノード数

### ファイルシステムのサイズと i ノードリストのサイズの検査

ファイルシステムのサイズは、スーパーブロックと i ノードリストに使用されるブロック数よりも大きくなければなりません。i ノード数は、ファイルシステムの最大許容数よりも小さくなければなりません。i ノードは、ファイルに関するすべての情報を表します。ファイルシステムのサイズとレイアウト情報は、fsck コマンドにとってもっとも重要な情報部分です。これらのサイズはファイルシステムの作成時に静的に決められるため、実際に検査する方法はありません。ただし、fsck コマンドを使用してサイズが妥当な範囲内にあるかどうかは検査できます。ファイルシステムのほかのすべての検査を行うには、これらのサイズが正確でなければなりません。fsck コマンドがプライマリスーパーブロックの静的パラメータ内に不正な情報を検出すると、オペレータに代替スーパーブロックの位置を指定するように促します。

UFS ファイルシステム構造の詳細は、[371 ページの「UFS ファイルシステムのシリンダグループの構造」](#)を参照してください。

## 空きブロック数の検査

空きブロック数は、シリンダグループのブロックマップに格納されます。fsck コマンドは、空きマーク付きのすべてのブロックがファイルによって使用されていないかどうかを検査します。すべてのブロックを検査し終わると、fsck コマンドは空きブロック数とi ノードによって使用されるブロック数の合計がファイルシステム内の合計ブロック数に等しくなるかどうかを検査します。ブロックマップ内に間違いがあると、fsck コマンドはブロックが割り当てられている状態のままで構築し直します。

スーパーブロック内のサマリー情報には、ファイルシステム内の空きブロックの合計数のカウントが入っています。fsck コマンドは、このブロック数をファイルシステム内で見つかった空きブロック数と比較します。数が一致しなければ、fsck コマンドはスーパーブロック内の空きブロック数を実際の空きブロック数で置き換えます。

## 空きi ノード数の検査

スーパーブロック内のサマリー情報には、ファイルシステム内の空きi ノード数が入っています。fsck コマンドは、このi ノード数をファイルシステム内で見つかった空きi ノード数と比較します。数が一致しなければ、fsck はスーパーブロック内の空きi ノード数を実際の空きi ノード数で置き換えます。

## i ノード

i ノードリストは、i ノード2から順番に検査されます。(i ノード0とi ノード1は予約ノード)。各i ノードの不整合は、次の面から検査されます。

- 形式とタイプ
- リンク数
- 重複ブロック
- 不正なブロック番号
- i ノードのサイズ

## i ノードの形式とタイプ

各i ノードには、そのタイプと状態を記述するモードのワードが入っています。i ノードには、次の9つのタイプがあります。

- 通常
- ディレクトリ
- ブロック型特殊ファイル
- キャラクタ型特殊ファイル
- FIFO(名前付きパイプ)
- シンボリックリンク
- シャドウ(ACLで使用される)



- 属性ディレクトリ
- ソケット

i ノードの状態は、次の3つのうちのいずれかです。

- 割り当て済み
- 未割り当て
- 部分的に割り当て済み

ファイルシステムの作成時、決まった数のi ノードは無効になっています。これらは、必要なときが来るまで割り当てられません。「割り当て済みのi ノード」とは、ファイルを指すi ノードです。「未割り当てのi ノード」は、ファイルを指さないで空のはずです。「部分的に割り当て済み」の状態は、i ノードが正しくフォーマットされていないことを意味します。たとえば、ハードウェア障害が原因でi ノードリストに不正なデータが書き込まれると、i ノードは「部分的に割り当て済み」の状態になることがあります。fsck コマンドが実行できる唯一の修正動作は、そのi ノードを消去することです。

## リンク数の検査

各i ノードには、そこにリンクされているディレクトリエントリ数が入っています。fsck コマンドは、ルート (/) ディレクトリから順番にディレクトリ構造全体を検査し、i ノードごとに実際のリンク数を計算して、各i ノードのリンク数を検証します。

i ノードに格納されているリンク数が fsck コマンドによって判断された実際のリンク数と一致しない場合は、次の3つの状況が考えられます。

- 格納されたリンク数が0でなく、実際のリンク数が0の場合。  
この状況は、i ノードにリンクされているディレクトリエントリが存在しない場合に発生することがあります。この場合、fsck コマンドはリンクされていないファイルを `lost+found` ディレクトリに入れます。
- 格納されたリンク数が0でなく、実際のリンク数も0でない。しかし、それらの数が等しくない場合。  
この状況は、ディレクトリエントリが追加または削除されたが、i ノードが更新されていない場合に発生することがあります。この場合、fsck コマンドは格納されたリンク数を実際のリンク数で置き換えます。
- 格納されたリンク数が0で実際のリンク数が0でない場合。  
この場合、fsck コマンドはi ノードのリンク数を実際のリンク数に変更します。

## 重複ブロックの検査

各 i ノードには、それが使用するすべてのブロックのリスト、またはリストを指すポインタ (間接ブロック) が入っています。間接ブロックは i ノードによって所有されるので、間接ブロックの整合性が失われると、それを所有する i ノードが直接影響を受けます。

fsck コマンドは、i ノードから使用される各ブロック番号を、割り当て済みブロックのリストと比較します。別の i ノードからすでにブロック番号が使用されていると、そのブロック番号は重複ブロックのリストに入れられます。それ以外の場合は、割り当て済みブロックのリストが更新され、ブロック番号が追加されます。

重複ブロックが見つかり、fsck コマンドは再び i ノードリストを調べて、各重複ブロックを使用するほかの i ノードを検索します。fsck コマンドでは、どの i ノードでエラーが発生しているか、正確に判断することはできません。このため、保持する i ノードと消去する i ノードを選択するように促すプロンプトが表示されます。i ノード内に多数の重複ブロックが入っている場合は、ファイルシステムに書き込まれていない間接ブロックの影響を受けている可能性があります。

## 不正なブロック番号の検査

fsck コマンドは、i ノードから使用される各ブロック番号を検査して、その値がファイルシステム内の最初のデータブロック番号の値よりも大きく、最後のデータブロック番号の値より小さいかどうかを調べます。ブロック番号がこの範囲に含まれない場合は、不正なブロック番号と見なされます。

間接ブロックがファイルシステムに正しく書き込まれていないことが原因で、i ノード内に不正なブロック番号が発見されることがあります。fsck コマンドはその i ノードの消去を促すプロンプトを表示します。

## i ノードサイズの検査

各 i ノードには、参照するデータブロック数が入っています。実際のデータブロック数は、割り当て済みのデータブロック数と間接ブロック数の合計です。fsck コマンドはデータブロック数を計算し、そのブロック数を i ノードから使用されるブロック数と比較します。i ノードに不正なブロック数が入っていると、fsck コマンドはその修正を促すプロンプトを表示します。

各 i ノードには、64 ビットのサイズフィールドがあります。このフィールドは、i ノードに関連付けられたファイル内の文字数 (データバイト数) を示します。i ノードのサイズフィールドに整合性があるかどうかは、サイズフィールド内の文字数を使用して、i ノードに関連付けるべきブロック数を計算し、その結果を i ノードから使用される実際のブロック数と比較して概算で検査されます。

## 間接ブロック

間接ブロックは*i* ノードによって所有されます。したがって、間接ブロック内の整合性が失われると、それを所有する*i* ノードが影響を受けます。不整合は、次の面から検査されます。

- すでに別の*i* ノードから使用されているブロック
- ファイルシステムの範囲に含まれないブロック番号

直接ブロックに対しても整合性検査が実行されます。

## データブロック

*i* ノードは、3 種類のデータブロックを直接または間接に参照できます。参照されるブロックは、すべて同じ種類でなければなりません。次の 3 種類のデータブロックがあります。

- プレーンデータブロック
- シンボリックリンクデータブロック
- ディレクトリデータブロック

プレーンデータブロックには、ファイルに格納される情報が入っています。シンボリックリンクデータブロックには、シンボリックリンクに格納されるパス名が入っています。ディレクトリデータブロックには、ディレクトリエントリが入っています。fsck コマンドはディレクトリデータブロックの妥当性しか検査できません。

ディレクトリは、*i* ノードの `mode` フィールド内のエントリによって通常ファイルと区別されます。ディレクトリに関連付けられたデータブロックには、ディレクトリエントリが入っています。ディレクトリデータブロックの不整合は、次の面から検査されます。

- 未割り当ての*i* ノードを指すディレクトリ内の*i* ノード番号
- ファイルシステム内の*i* ノード番号より大きいディレクトリ内の*i* ノード番号
- 「`.`」と「`..`」ディレクトリには許されないディレクトリ内の*i* ノード番号
- ファイルシステムから切り離されたディレクトリ

## 未割り当てディレクトリの検査

ディレクトリデータブロック内の*i* ノード番号が未割り当て*i* ノードを指す場合、fsck コマンドはそのディレクトリエントリを削除します。この状況は、新しいディレクトリエントリが入っているデータブロックが変更されて書き出されたが、*i* ノードが書き込まれていない場合に発生します。また、警告なしに CPU が停止された場合にも発生します。

## 不正な i ノード番号の検査

ディレクトリエントリの i ノード番号が i ノードリストの最後を超える位置を指す場合、fsck コマンドはそのディレクトリエントリを削除します。この状況は、不正なデータがディレクトリのデータブロックに書き込まれると発生します。

## 不正な「.」と「..」 エントリのチェック

「.」ディレクトリの i ノード番号のエントリは、ディレクトリデータブロックの最初のエントリでなければなりません。また、それ自体を参照しなければなりません。つまり、その値はディレクトリデータブロックの i ノード番号に等しくなければなりません。

「..」ディレクトリの i ノード番号は、ディレクトリデータブロックの第2のエントリでなければなりません。その値は、親ディレクトリの i ノード番号(または、ディレクトリがルート(/)ディレクトリの場合は、それ自体の i ノード番号)に等しくなければなりません。

「.」と「..」ディレクトリの i ノード番号が不正であれば、fsck コマンドはそれらを正しい値で置き換えます。ディレクトリへのハードリンクが複数個ある場合は、最初に見つかったハードリンクが「..」が指す実際の親であると見なされます。この場合、fsck コマンドはほかの名前を削除するように促すプロンプトを表示します。

## 切り離されたディレクトリ

fsck コマンドは、ファイルシステム全体で参照関係を検査します。ファイルシステムにリンクされていないディレクトリが見つかったら、fsck コマンドはそのディレクトリをファイルシステムの lost+found ディレクトリにリンクします。この条件は、ファイルシステムに i ノードは書き込まれたが、対応するディレクトリデータブロックは書き込まれていない場合に発生する可能性があります。

## 通常データブロック

通常ファイルに関連付けられたデータブロックには、ファイルの内容が入っています。fsck コマンドは、通常ファイルのデータブロックの内容が有効かどうかは検査しません。

## fsck サマリーメッセージ

fsck コマンドを対話式で実行して正常に終了すると、次のようなメッセージが表示されます。

```
fsck /dev/rdsd/c0t0d0s7
** /dev/rdsd/c0t0d0s7
** Last Mounted on /export/home
```

```

** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3a - Check Connectivity
** Phase 3b - Verify Shadows/ACLs
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cylinder Groups
2 files, 9 used, 2833540 free (20 frags, 354190 blocks, 0.0% fragmentation)
#

```

fsck 出力の最後の行は、ファイルシステムについて次のような情報を記述します。

# files	使用中の i ノード数
# used	使用中のフラグメント数
# free	未使用のフラグメント数
# frags	未使用の非ブロックフラグメント数
# blocks	未使用の完全ブロック数
% fragmentation	断片化の比率。ファイルシステム内の空きフラグメント × 100 / 全フラグメント

フラグメントについては、[376 ページ](#)の「[フラグメントサイズ](#)」を参照してください。

## UFS ファイルシステムを対話式で検査して修復する

次の場合には、ファイルシステムを対話式で検査する必要があります。

- マウントできない場合
- 使用中に不整合が発生する場合

使用中のファイルシステムの整合性が失われると、コンソールウィンドウやシステムメッセージファイルにエラーメッセージが出力されたり、システムがクラッシュしたりすることがあります。たとえば、システムメッセージファイル `/var/adm/messages` に次のようなメッセージが出力されることがあります。

```

Sep 5 13:42:40 hostname ufs: [ID 879645 kern.notice] NOTICE: /: unexpected
free inode 630916, run fsck(1M)

```

`hostname` は、エラーを報告したシステムです。

fsck コマンドを実行する前に、これらの参照情報を利用して、fsck エラーメッセージの内容を解決する場合があります。

- [369 ページ](#)の「[fsck コマンドの構文とオプション](#)」
- 『Solaris のシステム管理 (上級編)』の第 20 章「[UFS ファイルシステムの不整合解決 \(手順\)](#)」

fsck コマンドを実行して UFS ファイルシステムを検査するときには、次の点を考慮してください。

- fsck を使用して検査するファイルシステムは、非アクティブでなければなりません。ディスクにまだフラッシュされていないファイルシステムの変更がある場合、または fsck による検査の進行中にファイルシステムを変更した場合、ファイルシステムの破損とみなされることがあります。この場合、問題の報告内容に信頼性があるとは言えません。
- fsck を使用して修復するファイルシステムは、非アクティブでなければなりません。ディスクにまだフラッシュされていないファイルシステムの変更がある場合、または fsck による修復の進行中にファイルシステムを変更した場合、ファイルシステムが破損したり、システムがクラッシュしたりすることがあります。
- fsck を使用する前に、そのファイルシステムのマウントを解除します。これにより、ファイルシステムのデータ構造の整合性をできるだけ確保できます。ただし、ルート (/) ファイルシステム、/usr ファイルシステム、および /var ファイルシステムだけは、アクティブにします。これらのファイルシステムは、fsck を実行するためにマウントする必要があります。
- ルート (/) ファイルシステム、/usr ファイルシステム、および /var ファイルシステムを修復する必要がある場合には、可能であれば代替デバイスからシステムをブートすると、マウント解除されて非アクティブになります。

ルート (/) ファイルシステム、/usr ファイルシステム、または /var ファイルシステムに対して fsck を実行する手順については、[358 ページ](#)の「代替ブートデバイスから UFS ルート (/) ファイルシステム、/usr ファイルシステム、または /var ファイルシステムを検査する方法」を参照してください。

## ▼ 代替ブートデバイスから **UFS** ルート (/) ファイルシステム、/usr ファイルシステム、または /var ファイルシステムを検査する方法

次のメッセージが表示される場合は、fsck を再実行する必要はありません。

```
***** FILE SYSTEM WAS MODIFIED *****
```

ただし、このメッセージのあとに fsck を再実行してもファイルシステムは損傷しません。このメッセージは、fsck の修正処置に関する情報を示すだけのものです。

この手順では、ローカルの Solaris DVD またはネットワークブートサーバーを利用でき、代替デバイスからシステムをブートできることを前提とします。

不正なスーパーブロックの復元については、365 ページの「不正な UFS スーパーブロックを復元する方法 (Solaris 10 6/06 リリース)」または 368 ページの「不正なスーパーブロックを復元する方法 (Solaris 8、9、および 10 リリース)」を参照してください。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 ルート (/) ファイルシステムがミラー化されているシステムの場合のみ: ルート (/) ミラーを切り離れたあと、代替デバイスからブートします。先にブートしてしまうと、ファイルシステムが破損するおそれがあります。  
ルート (/) ミラーの切り離しについては、『[Solaris Volume Manager 管理ガイド](#)』の「サブミラーに関する作業」を参照してください。
- 3 検査の必要なルート (/) ファイルシステム、`/usr` ファイルシステム、または `/var` ファイルシステムのデバイス (`/dev/dsk/c0t0d0s0` など) を特定します。  
このデバイス名は、代替デバイスからブートしたあとで指定する必要があります。代替デバイスからブートしたあとでは、このデバイスの特定がより難しくなります。
- 4 ローカルの Solaris DVD やネットワークなどの代替デバイスから検査する必要があるルート (/) ファイルシステム、`/usr` ファイルシステム、または `/var` ファイルシステムのあるシステムを、シングルユーザーモードでブートします。  
この場合、ファイルシステム上に動作は存在しません。

例:

```
init 0
ok boot net -s
.
.
.
#
```

- 5 手順 3 で特定したルート (/) ファイルシステム、`/usr` ファイルシステム、または `/var` ファイルシステムが含まれるデバイスを検査します。  
検査または修復するファイルシステムのハードウェアが変更されている場合には、そのデバイス名も変更されている可能性があります。 `fsck -n` メッセージの「Last Mounted on ...」を利用して、そのファイルシステムのデバイスが予期したデバイスであることを確認してください。

たとえば、検査するルート (/) ファイルシステムが `/dev/dsk/c0t0d0s0` の場合は、次のように出力されます。

```
fsck -n /dev/dsk/c0t0d0s0
** /dev/dsk/c0t0d0s0 (NO WRITE)
** Last Mounted on /
.
.
```



```

.
fsck /dev/rdsk/c0t0d0s0
** /dev/rdsk/c0t0d0s0
** Last Mounted on /
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
.
.
.

```

- 6 報告された **fsck** エラーをすべて修正します。

1 つまたは複数の UFS ファイルシステムを対話式で検査しながら、エラーメッセージのプロンプトに応答する方法については、『[Solaris のシステム管理 \(上級編\)](#)』の第 20 章「[UFS ファイルシステムの不整合解決 \(手順\)](#)」を参照してください。

- 7 **fsck** コマンドを実行してもすべての問題を修復できない場合は、[363 ページ](#)の「[fsck コマンドで修復できない UFS ファイルシステムの修正](#)」を参照してください。
- 8 修復したファイルシステムをマウントして、**lost+found** ディレクトリにファイルが存在するかどうかを確認します。

**fsck** コマンドによって **lost+found** ディレクトリに入れられた各ファイルの名前は、その i ノード番号を使用して変更されます。可能であれば、ファイル名を変更し、ファイルが含まれるべきディレクトリに移動してください。**grep** コマンドを使用して各ファイル内の語句を探したり、**file** コマンドを使用してファイルタイプを識別してください。

どうしても特定できないファイルまたはディレクトリについては、**lost+found** ディレクトリからそれらを削除して、不要な領域を解放してください。

- 9 システムをマルチユーザーモードに戻します。

```
init 6
```

- 10 ルート (/) ファイルシステムがミラー化されているシステムの場合のみ: ルート (/) ミラーを再接続します。

## ▼ ルート (/)、/usr、または /var 以外の UFS ファイルシステムを検査する方法

Solaris 10 6/06 リリースの **fsck** に関する最新情報については、[342 ページ](#)の「[UFS ファイルシステムユーティリティ \(fsck、mkfs、および newfs\) の拡張機能](#)」を参照してください。次のメッセージが表示される場合は、**fsck** を再実行する必要はありません。

```
***** FILE SYSTEM WAS MODIFIED *****
```



ただし、このメッセージのあとに `fsck` を再実行してもファイルシステムは損傷しません。このメッセージは、`fsck` の修正処置に関する情報を示すだけのものです。

この手順では、検査するファイルシステムがマウント解除されていることを前提としています。

不正なスーパーブロックの復元については、[365 ページの「不正な UFS スーパーブロックを復元する方法 \(Solaris 10 6/06 リリース\)」](#) または [368 ページの「不正なスーパーブロックを復元する方法 \(Solaris 8、9、および 10 リリース\)」](#) を参照してください。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 ローカルファイルシステムをマウント解除し、ファイルシステム上でほかの動作が存在しないようにします。  
`fsck` コマンドの引数として、マウントポイントディレクトリや `/dev/dsk/ device-name` を指定します。整合性が失われている場合には、そのことを示すメッセージが表示されます。  
 例:  

```
umount /export/home
fsck /dev/rdsk/c0t0d0s7
** /dev/dsk/c0t0d0s7
** Last Mounted on /export/home
.
.
.
```
- 3 報告された `fsck` エラーをすべて修正します。  
 1 つまたは複数の UFS ファイルシステムを対話式で検査しながら、エラーメッセージのプロンプトに応答する方法については、『[Solaris のシステム管理 \(上級編\)](#)』の第 20 章「[UFS ファイルシステムの不整合解決 \(手順\)](#)」を参照してください。
- 4 `fsck` コマンドを実行してもすべての問題を修復できない場合は、[363 ページの「fsck コマンドで修復できない UFS ファイルシステムの修正」](#) を参照してください。
- 5 修復したファイルシステムをマウントして、`lost+found` ディレクトリにファイルが存在するかどうかを確認します。  
`fsck` コマンドによって `lost+found` ディレクトリに入れられた各ファイルの名前は、その `i` ノード番号を使用して変更されます。
- 6 `lost+found` ディレクトリに保存されているファイルの名前を変更して移動します。  
 可能であれば、ファイル名を変更し、ファイルが含まれるべきディレクトリに移動してください。`grep` コマンドを使用して各ファイル内の語句を探したり、`file` コマンドを使用してファイルタイプを識別してください。

どうしても特定できないファイルまたはディレクトリについては、`lost+found` ディレクトリからそれらを削除して、不要な領域を解放してください。

### 例 17-1 ルート (/) 以外、または /usr 以外の UFS ファイルシステムの対話型検査

次の例は、`/dev/rdisk/c0t0d0s6` ファイルシステムを検査し、不正なブロック数を訂正する方法を示しています。この例では、ファイルシステムがマウント解除されていることを前提としています。

```
fsck /dev/rdisk/c0t0d0s6
** Phase 1 - Check Block and Sizes
INCORRECT BLOCK COUNT I=2529 (6 should be 2)
CORRECT? y

** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Cylinder Groups
929 files, 8928 used, 2851 free (75 frags, 347 blocks, 0.6%
fragmentation)

***** FILE SYSTEM WAS MODIFIED *****
#
```

## UFS ファイルシステムの修復

`fsck -o p` コマンド (`p` は `preen` (修復) を表す) は、UFS ファイルシステムを検査し、通常は予期しないシステムのシャットダウンによって発生する問題を自動的に修正します。オペレータの介入が必要な問題が発見されると、このコマンドは即座に終了します。このコマンドによって、ファイルシステムを並列に検査することも可能です。

システムがクリーンな状態でシャットダウンしなかった後のファイルシステムの修復にも、`fsck -o p` コマンドを実行できます。このモードでは、`fsck` コマンドはクリーンフラグを調べずに完全検査を実行します。これらの処理は、`fsck` コマンドを対話式で実行した場合の処理のサブセットです。

## ▼ UFS ファイルシステムを修復する方法

この手順では、ファイルシステムがマウント解除されているか、非アクティブであることを前提としています。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。

- 2 UFS ファイルシステムをマウント解除します。

```
umount /mount-point
```

- 3 修復オプションを指定して UFS ファイルシステムを検査します。

```
fsck -o p /dev/rdisk/device-name
```

fsck コマンドの引数として */mount-point* または */dev/rdisk/device-name* を使用すると、個々のファイルシステムを修復できます。

#### 例 17-2 UFS ファイルシステムを修復する

次の例は、*/export/home* ファイルシステムの修復方法を示します。

```
fsck -o p /export/home
```

## fsck コマンドで修復できない UFS ファイルシステムの修正

fsck コマンドを何度か実行しているとき、問題の修正を進めていく過程で、以前の実行で検出されていた問題が再度発生することがあります。場合によっては、問題が報告されなくなるまで、繰り返し fsck を実行する必要があります。これにより、すべてのエラーが検出され、修復されます。

fsck コマンドで表示される情報に注目してください。問題を解決する上で参考になります。たとえば、メッセージは損傷したディレクトリを指している場合があります。そのディレクトリを削除すると、fsck コマンドが問題なく実行されるようになる場合もあります。

それでも fsck コマンドでファイルシステムを修復できない場合は、ff、clri、および ncheck コマンドを使用し、間違いを特定して修正します。これらのコマンドの使用方法については、次の情報を参照してください。

- fsdb(1M)
- ff(1M)
- clri(1M)
- ncheck(1M)

最終的には、ファイルシステムを作成し直し、その内容をバックアップメディアから復元せざるをえない場合があります。

ファイルシステム全体の復元については、[404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する \(タスクマップ\)」](#)を参照してください。

ファイルシステムを完全に修復できないが、読み取り専用としてマウントできる場合は、`cp`、`tar`、または `cpio` コマンドを使用して、データのすべてまたは一部をファイルシステムから取り出してください。

問題の原因がハードウェア上のディスクエラーであれば、ファイルシステムを作成し直して復元する前に、ディスクをフォーマットし直して再びパーティション分割を行わなければならない場合があります。ディスクデバイスを交換する前に、デバイスのケーブルおよびコネクタが正常に機能するかどうかを検査してください。一般に、ハードウェアエラーが発生すると、さまざまなコマンドで同じエラーが繰り返し表示されます。`format` コマンドはディスク上の不良ブロックを使用しないようにします。ただし、ディスクの損傷が致命的な場合、フォーマットし直したあとも問題が解決されないことがあります。`format` コマンドの使用方法については、[format\(1M\)](#) のマニュアルページを参照してください。新しいディスクを取り付ける方法については、[第 10 章「SPARC: ディスクの設定 \(手順\)」](#) または [第 11 章「x86: ディスクの設定 \(手順\)」](#) を参照してください。

## 不正な UFS スーパーブロックの復元

ファイルシステムのスーパーブロック内のデータが破壊された場合は、復元しなければなりません。スーパーブロックが不正なときには、`fsck` コマンドからメッセージが表示されます。さいわい、スーパーブロックのコピーがファイルシステム内に格納されています。

`fsck -o b` コマンドを使用すると、スーパーブロックをいずれかのコピーと置き換えることができます。または `fsck` のバックアップスーパーブロックの自動検索機能を使用します。この機能の詳細は、[343 ページの「バックアップスーパーブロックの自動検索」](#) を参照してください。

スーパーブロックの詳細は、[372 ページの「スーパーブロック」](#) を参照してください。

ルート (`/`) ファイルシステム内のスーパーブロックが損傷し、修復できない場合は、次のどちらかの操作を実行します。

- システムをインストールし直します。
- ネットワークまたはローカルの Solaris DVD からブートし、次の手順を実行します。それらの手順で失敗する場合は、`newfs` コマンドを使ってルート (`/`) ファイルシステムを作成し直し、バックアップコピーから復元します。

## ▼ 不正な UFS スーパーブロックを復元する方法 (Solaris 10 6/06 リリース)

ファイルシステムに不正な UFS スーパーブロックがある場合は、次のメッセージに示すように `fsck` によって自動的に代替のスーパーブロックが計算されます。

```
BAD SUPERBLOCK AT ...
```

```
LOOK FOR ALTERNATE SUPERBLOCKS WITH MKFS?
LOOK FOR ALTERNATE SUPERBLOCKS WITH NEWFS?
```



注意 - 損傷したスーパーブロックが含まれるファイルシステムが、`ntrack` や `nsect` などの `newfs` または `mkfs` のカスタマイズされたパラメータを使用して作成されている場合、修復処理のために `fsck` で自動的に計算されたスーパーブロックがファイルシステムを損傷させる可能性があります。

カスタマイズされたパラメータを使用して作成されたファイルシステムで、ファイルシステムに不正なスーパーブロックが含まれている場合、`fsck` は、`fsck` セッションを取り消すための次のようなプロンプトを表示します。

```
CANCEL FILESYSTEM CHECK?
```

ファイルシステムがカスタマイズされたパラメータを使用して作成されている場合またはこのシステム上での `fsck` の実行に関する他の心配がある場合は、`fsck` セッションの取り消しが適切な応答です。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 不正なスーパーブロックがあると思われるファイルシステムを検査します。  

```
fsck /dev/rdisk/c0t1d0s0
```

```
** /dev/rdisk/c0t1d0s0
```

```
BAD SUPERBLOCK at ...
```
- 3 ファイルシステムの作成方法を特定し、次のいずれかを選択します。
  - ファイルシステムが **newfs** コマンドを使用して作成された。
    - `fsck` がすべてのスーパーブロックが破損していることを報告し、汎用のスーパーブロックを使用する必要があります。下の例で説明するように、`fsck` プロンプトに応答します。



注意-カスタマイズされたパラメータを使用してファイルシステムが作成されている場合はこのオプションを使用しないでください。このオプションは最後の手段として使用してください。バックアップコピーからファイルシステムを復元する準備をします。

```
fsck /dev/dsk/c1t2d0s0
** /dev/rdsk/c1t2d0s0
BAD SUPERBLOCK AT BLOCK 16: BLOCK SIZE LARGER THAN MAXIMUM SUPPORTED

LOOK FOR ALTERNATE SUPERBLOCKS WITH MKFS? no

LOOK FOR ALTERNATE SUPERBLOCKS WITH NEWFS? yes

SEARCH FOR ALTERNATE SUPERBLOCKS FAILED.

USE GENERIC SUPERBLOCK FROM MKFS? no

USE GENERIC SUPERBLOCK FROM NEWFS? yes

CALCULATED GENERIC SUPERBLOCK WITH NEWFS
If filesystem was created with manually-specified geometry, using
auto-discovered superblock may result in irrecoverable damage to
filesystem and user data.

CANCEL FILESYSTEM CHECK? no

** Last Mounted on
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3a - Check Connectivity
** Phase 3b - Verify Shadows/ACLs
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cylinder Groups
CORRECT GLOBAL SUMMARY
SALVAGE? y

UPDATE STANDARD SUPERBLOCK? y

81 files, 3609 used, 244678 free (6 frags, 30584 blocks, 0.0% fragmentation)
```

```
***** FILE SYSTEM WAS MODIFIED *****
```

- fsck が次のようなメッセージを表示し、代替のスーパーブロックが見つかったことを報告します。

```
FOUND ALTERNATE SUPERBLOCK 32 WITH NEWFS
```

この fsck のシナリオでは、[343 ページ](#)の「バックアップスーパーブロックの自動検索」に示すようにプロンプトに従います。

- ファイルシステムが **mkfs** コマンドを使用して作成された。

- `fsck` がすべてのスーパーブロックが破損していることを報告し、汎用のスーパーブロックを使用する必要があります。下の例で説明するように、`fsck` プロンプトに応答します。



注意-カスタマイズされたパラメータを使用してファイルシステムが作成されている場合はこのオプションを使用しないでください。このオプションは最後の手段として使用してください。バックアップコピーからファイルシステムを復元する準備をします。

```
fsck /dev/dsk/c1t2d0s0
** /dev/rdsk/c1t2d0s0
BAD SUPERBLOCK AT BLOCK 16: BLOCK SIZE LARGER THAN MAXIMUM SUPPORTED

LOOK FOR ALTERNATE SUPERBLOCKS WITH MKFS? yes

LOOK FOR ALTERNATE SUPERBLOCKS WITH NEWFS? no

SEARCH FOR ALTERNATE SUPERBLOCKS FAILED.

USE GENERIC SUPERBLOCK FROM MKFS? yes

CALCULATED GENERIC SUPERBLOCK WITH MKFS
If filesystem was created with manually-specified geometry, using
auto-discovered superblock may result in irrecoverable damage to
filesystem and user data.

CANCEL FILESYSTEM CHECK? no

** Last Mounted on
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3a - Check Connectivity
** Phase 3b - Verify Shadows/ACLs
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cylinder Groups
CORRECT GLOBAL SUMMARY
SALVAGE? y

UPDATE STANDARD SUPERBLOCK? y

81 files, 3609 used, 243605 free (117 frags, 30436 blocks, 0.0% fragmentation)
```

- `fsck` が次のようなメッセージを表示し、代替のスーパーブロックが見つかったことを報告します。

```
FOUND ALTERNATE SUPERBLOCK 32 WITH MKFS
```

この `fsck` のシナリオでは、[343 ページの「バックアップスーパーブロックの自動検索」](#) に示すようにプロンプトに従います。

- 4 プロンプトに応答し、スーパーブロックを復元します。

次のメッセージが表示される場合は、fsck を再実行する必要はありません。

```
***** FILE SYSTEM WAS MODIFIED *****
```

ただし、このメッセージのあとに fsck を再実行してもファイルシステムは損傷しません。このメッセージは、fsck の修正処置に関する情報を示すだけのものです。

## ▼ 不正なスーパーブロックを復元する方法 (Solaris 8、9、および 10 リリース)

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 不正なスーパーブロックがルート (/)、/usr、または /var ファイルシステム内にあるかどうかを調べ、次のどちらかの操作を実行します。

- ルート (/) ファイルシステム、/usr ファイルシステム、または /var ファイルシステムに不正なスーパーブロックが存在する場合、ネットワークまたはローカルに接続された Solaris DVD からブートします。

ローカル接続された DVD からブートする場合は、次のコマンドを使用します。

```
ok boot cdrom -s
```

ブートサーバーまたはインストールサーバーがすでに設定済みのネットワークからブートする場合は、次のコマンドを使用します。

```
ok boot net -s
```

システムを停止する必要がある場合は、『Oracle Solaris の管理: 基本管理』の第 12 章「Oracle Solaris システムのブート (タスク)」または『Oracle Solaris の管理: 基本管理』の「GRUB を使用して x86 システムをブートする (タスクマップ)」を参照してください。

- ルート (/)、/usr、または /var のどのファイルシステムにも不正なスーパーブロックが存在しない場合は、損傷を受けたファイルシステム以外のディレクトリに移動し、マウントを解除します。

```
umount /mount-point
```



注意 - 次の手順では、必ず newfs -N オプションを使用してください。-N オプションを指定しない場合は、そのファイルシステムのデータがすべて破壊され、空のファイルシステムに置き換わります。

- 3 newfs -N コマンドを使用して、スーパーブロックの値を表示します。

```
newfs -N /dev/rdisk/device-name
```



このコマンドの出力には、`newfs` コマンドによってファイルシステムが作成されたときに、スーパーブロックのコピーとして使用されることになったブロック番号が表示されます。カスタマイズされたファイルシステムを作成する方法については、[375 ページの「UFS ファイルシステムパラメータのカスタマイズ」](#)を参照してください。

#### 4 fsck コマンドを使用して、代替スーパーブロックを指定します。

```
fsck -F ufs -o b=block-number /dev/rdsk/device-name
```

`fsck` コマンドは、指定された代替スーパーブロックを使用して、プライマリスーパーブロックを復元します。代替ブロックとしては、常に 32 を使用できます。または、`newfs -N` コマンドの実行結果として出力される任意の代替ブロックを使用します。

### 例 17-3 不正なスーパーブロックの復元 (Solaris 8、9、および 10 リリース)

次の例は、スーパーブロックのコピー 5264 を復元する方法を示しています。

```
newfs -N /dev/rdsk/c0t3d0s7
/dev/rdsk/c0t3d0s7: 163944 sectors in 506 cylinders of 9 tracks, 36 sectors
83.9MB in 32 cyl groups (16 c/g, 2.65MB/g, 1216 i/g)
super-block backups (for fsck -b #) at:
 32, 5264, 10496, 15728, 20960, 26192, 31424, 36656, 41888,
 47120, 52352, 57584, 62816, 68048, 73280, 78512, 82976, 88208,
 93440, 98672, 103904, 109136, 114368, 119600, 124832, 130064, 135296,
 140528, 145760, 150992, 156224, 161456,
fsck -F ufs -o b=5264 /dev/rdsk/c0t3d0s7
Alternate superblock location: 5264.
** /dev/rdsk/c0t3d0s7
** Last Mounted on
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
36 files, 867 used, 75712 free (16 frags, 9462 blocks, 0.0% fragmentation)

***** FILE SYSTEM WAS MODIFIED *****
#
```

## fsck コマンドの構文とオプション

`fsck` コマンドは、ファイルシステム内の不整合を検査して修復します。オプションを指定しないで `fsck` コマンドを実行した場合は、修復が行われる前に確認を求めるプロンプトが表示されます。このコマンドには、次の 4 つのオプションがあります。

コマンドとオプション	説明
<code>fsck -m</code>	ファイルシステムがマウント可能かどうかの検査を行います
<code>fsck -y</code>	すべての修復に対して <code>yes</code> の応答が指定されたものとして処理を行います
<code>fsck -n</code>	すべての修復に対して <code>no</code> の応答が指定されたものとして処理を行います
<code>fsck -o p</code>	確認を求めるプロンプトを表示することなく、ファイルシステムを修復し、想定される (軽微な) 不整合箇所をすべて修正しますが、重大な問題が見つかったと終了します

## UFS ファイルシステム (参照情報)

---

この章の内容は次のとおりです。

- 371 ページの「UFS ファイルシステムのシリンダグループの構造」
- 375 ページの「UFS ファイルシステムパラメータのカスタマイズ」

### UFS ファイルシステムのシリンダグループの構造

UFS ファイルシステムを作成すると、ディスクスライスがシリンダグループに分割されます。「シリンダグループ」は、1つ以上の連続したディスクシリンダで構成されます。シリンダグループはさらにアドレス指定可能なブロックに分割され、このブロックによって、シリンダグループ内のファイルの構造が編成され、制御されます。各種のブロックは、ファイルシステム内で特定の機能を持っています。UFS ファイルシステムには、次の4種類のブロックがあります。

ブロックの種類	格納されている情報の種類
ブートブロック	システムブート時に使用される情報
スーパーブロック	ファイルシステムに関する大部分の情報
i ノード	ファイルに関する名前以外のすべての情報
記憶域またはデータブロック	各ファイルのデータ

次のセクションでは、これらのブロックの編成と機能について説明します。

## ブートブロック

ブートブロックには、システムのブート時に使用されるオブジェクトが格納されます。ファイルシステムがブートに使用されなければ、ブートブロックは空白のままです。ブートブロックは最初のシリンダグループ(シリンダグループ0)にのみ置かれ、スライス内の最初の8K バイトです。

## スーパーブロック

スーパーブロックには、次のようなファイルシステムに関する大部分の情報が格納されます。

- ファイルシステムのサイズとステータス
- ラベル(ファイルシステム名とボリューム名を含む)
- ファイルシステムの論理ブロックのサイズ
- 最終更新日時
- シリンダグループのサイズ
- シリンダグループ内のデータブロック数
- サマリーデータブロック
- ファイルシステムの状態
- 最後のマウントポイントのパス名

スーパーブロックには重要なデータが入っているので、ファイルシステムの作成時には複数のスーパーブロックが作成されます。

サマリー情報ブロックは、スーパーブロック内に保管されます。サマリー情報ブロックは複製されませんが、通常はシリンダグループ0内の主スーパーブロックと一緒にグループに分類されます。サマリーブロックには、ファイルシステムが使用されるときの発生する変更が記録されます。さらに、サマリーブロックには、ファイルシステム内のiノード数、ディレクトリ数、フラグメント数、および記憶ブロック数が表示されます。

## iノード

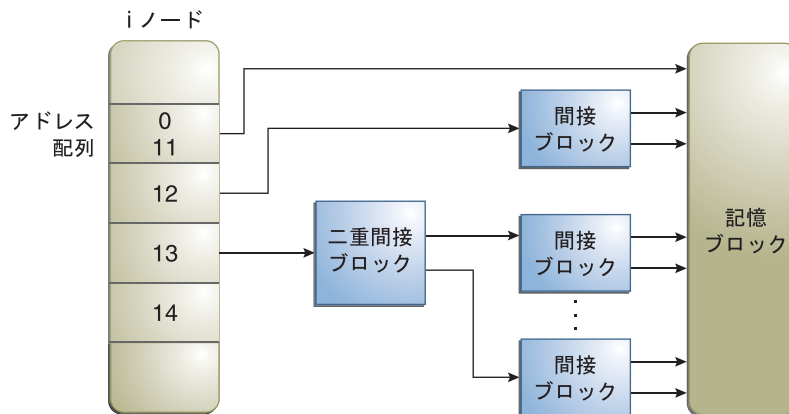
iノードには、ファイルに関して名前以外のすべての情報が入っており、ディレクトリ内に保管されます。iノードは128バイトです。iノード情報はシリンダ情報ブロック内に保管され、次の情報が入っています。

- ファイルのタイプ
  - 通常ファイル
  - ディレクトリ
  - ブロック型特殊ファイル
  - キャラクタ型特殊ファイル

- FIFO (名前付きパイプとも呼ぶ)
- シンボリックリンク
- ソケット
- その他のiノード - 属性ディレクトリとシャドウファイル (ACL 用)
- ファイルのモード (読み取り権 - 書き込み権 - 実行権のセット)
- ファイルへのハードリンク数
- ファイルの所有者のユーザー ID
- ファイルが属するグループ ID
- ファイル内のバイト数
- 15 個のディスクブロックアドレスの配列
- ファイルの最終アクセス日時
- ファイルの最終変更日時
- iノードの変更日時

15 個のディスクブロックアドレスの配列 (0 - 14) は、ファイルの内容が格納されているデータブロックを指します。最初の 12 個は直接アドレスです。つまり、ファイルの内容のうち最初の 12 個の論理記憶ブロックを直接指します。ファイルが論理ブロック 12 個分より大きい場合は、13 番目のアドレスは間接ブロックを指します。間接ブロックには、ファイルの内容ではなく直接ブロックのアドレスが入っています。14 番目のアドレスは、二重間接ブロックを指します。二重間接ブロックには、間接ブロックのアドレスが入っています。15 番目のアドレスは三重間接アドレス用です。次の図に、iノードから始まるこのアドレスブロックチェーンを示します。

図 18-1 UFS ファイルシステムのアドレスチェーン



# データブロック

ファイルシステムに割り当てられた残りの領域には、データブロック (記憶ブロックともいう) が入っています。これらのデータブロックのサイズは、ファイルシステムの作成時に決定されます。デフォルトでは、データブロックは、2つのサイズに割り当てられます。8K バイトの論理ブロックサイズと 1K バイトのフラグメントサイズです。

通常ファイルの場合、データブロックにはファイルの内容が入っています。ディレクトリの場合、データブロックにはディレクトリ内のファイルの i ノード番号とファイル名を示すエントリが入っています。

# 空きブロック

現在、i ノード、間接アドレスブロック、または記憶ブロックとして使用されていないブロックには、シリンダグループマップ内で空きを示すマークが付けられます。また、このマップはフラグメントを追跡し、断片化によるディスクパフォーマンスの低下を防止します。

UFS ファイルシステムの内容の構造を理解しやすいように、次の図に、一般的な UFS システム内の一連のシリンダグループを示します。

図 18-2 一般的な UFS ファイルシステム



# UFS ファイルシステムパラメータのカスタマイズ

`newfs` コマンドによって割り当てられるデフォルトのファイルシステムパラメータを変更しようとする前に、各パラメータについて理解しておく必要があります。このセクションでは、次の各パラメータについて説明します。

- 375 ページの「論理ブロックサイズ」
- 376 ページの「フラグメントサイズ」
- 376 ページの「最小空き容量」
- 377 ページの「回転待ち」(廃止)
- 377 ページの「最適化のタイプ」
- 377 ページの「i ノード数(ファイルの数)」

これらのパラメータをカスタマイズするコマンドオプションについては、[newfs\(1M\)](#) と [mkfs\\_ufs\(1M\)](#) のマニュアルページを参照してください。

## 論理ブロックサイズ

論理ブロックサイズは、UNIX カーネルがファイルの読み書きに使用するブロックのサイズです。一般に、論理ブロックサイズは物理ブロックサイズとは異なりま  
す。物理ブロックサイズは、通常は 512K バイトで、ディスクコントローラが読み書きできる最小ブロックのサイズです。

論理ブロックサイズは、デフォルトでシステムのページサイズに設定されます。UFS ファイルシステムの場合、このデフォルト論理ブロックサイズは 8192 バイト (8K バイト) です。UFS ファイルシステムでは、ブロックサイズとして 4096 バイトまたは 8192 バイト (4K または 8K バイト) がサポートされます。論理ブロックの推奨サイズは 8K バイトです。

---

**SPARC のみ** - `sun-4u` プラットフォームでは、8192 バイトのブロックサイズしか指定できません。

---

システムに最善の論理ブロックサイズを選択するには、必要なパフォーマンスと使用可能容量を検討してください。ほとんどの UFS システムでは、8K バイトのファイルシステムが最高のパフォーマンスを発揮し、ディスクパフォーマンスとプライマリメモリーやディスク上の領域の使用量が適切なバランスに保たれます。

原則として、効率を高めるには、ほとんどのファイルがきわめて大きいファイルシステムには大きめの論理ブロックサイズを使用します。ほとんどのファイルがきわめて小さいファイルシステムには、小さめの論理ブロックサイズを使用します。ファイルシステム上で `quot -c filesystem` コマンドを使用すると、ファイルの分散に関する詳細なレポートをブロックサイズ別に表示できます。

ただし、通常は、ファイルシステムの作成時に設定されたページサイズが最適です。

## フラグメントサイズ

ファイルが作成または拡張されると、論理ブロック全体または「フラグメント」と呼ばれる部分のディスク容量が割り当てられます。ファイルのためにディスク容量が必要になると、まずブロック全体が割り当てられ、次に残りの部分にブロックのうち1つまたは複数のフラグメントが割り当てられます。小型ファイルの場合、割り当てはフラグメントから始まります。

ブロック全体ではなく、そのフラグメントを割り当てることができるので、ブロック内の未使用のホールによって生じるディスク容量の「断片化」が減少し、容量の節約になります。

UFS ファイルシステムを作成するときに、「フラグメントサイズ」を定義します。デフォルトのフラグメントサイズは1K バイトです。各ブロックは、1個、2個、4個、または8個のフラグメントに分割できます。この場合、フラグメントサイズは8192 バイトから512 バイト (4K バイトのファイルシステムのみ) までです。実際には、下限はディスクのセクターサイズ、通常は512 バイトに連動します。

マルチテラバイトのファイルシステムの場合、フラグメントサイズはファイルシステムのブロックサイズに等しくなります。

---

注-フラグメントサイズの上限は論理ブロックサイズに等しくなります。この場合、フラグメントは存在しないことになります。容量よりも速度を重視する場合、きわめて大型のファイルがあるファイルシステムには、この構成が最適なことがあります。

---

フラグメントサイズを選択するときには、処理時間と容量の兼ね合いを考慮してください。フラグメントサイズが小さければ容量の節約になりますが、割り当てには時間がかかります。原則として、格納効率を高めるには、ほとんどのファイルが大型のファイルシステムには、大きめのフラグメントサイズを使用します。ほとんどのファイルが小型のファイルシステムには、小さめのフラグメントサイズを使用します。

## 最小空き容量

「最小空き容量」とは、ファイルシステムの作成時に予約分として保持されるディスク容量の割合です。デフォルトの予約分は、 $((64\text{M バイト}/\text{パーティションサイズ}) * 100)$  で算出し、その値はもっとも近い整数に切り捨てられ、ディスク容量の1% から 10% の範囲に制限されます。

ファイルシステム内の空き容量が少なくなるほど、アクセス速度が低下するので、空き容量は重要です。十分な空き容量があれば、UFS ファイルシステムは効率



よく動作します。ファイルシステムがいっぱいになって、使用可能なユーザー領域を使い果たすと、スーパーユーザー以外は予約済みの空き容量にアクセスできなくなります。

`df` などのコマンドは、最小空き容量として割り当て済みの分を差し引いて、ユーザーに使用可能な容量をパーセントで表示します。コマンドでファイルシステム内のディスク容量の 100 パーセント以上が使用中であると表示される場合は、予約分の一部がスーパーユーザーによって使用されています。

ユーザーに割り当てを適用する場合に、各ユーザーが使用可能な容量には予約分の空き容量は含まれません。`tunefs` コマンドを使用すると、既存のファイルシステムの最小空き容量の値を変更できます。

## 回転待ち

このパラメータはもう使用しません。指定した値に関わらず、値は常に 0 に設定されます。

## 最適化のタイプ

「最適化のタイプ」パラメータの設定には、「*space*」と「*time*」があります。

- **space** – 最適化のタイプに `space` を選択すると、断片化を最小限度に抑え、ディスクの使用状況が最適化されるようにディスクブロックが割り当てられます。
- **time** – 最適化のタイプに `time` を選択すると、配置はあまり重視されず、できるだけ高速になるようにディスクブロックが割り当てられます。空き容量が十分ある場合、それほど細かく断片化しなくても、比較的簡単にディスクブロックを割り当てることができます。デフォルトは `time` です。

既存のファイルシステムの場合は、`tunefs` コマンドを使用して最適化タイプのパラメータ値を変更できます。

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

## i ノード数(ファイルの数)

i ノード 1 個あたりのバイト数によって、ファイルシステム内における i ノードの密度が決まります。ファイルシステムの合計サイズをこの値で割ると、作成すべき i ノードの個数が得られます。i ノードが割り当てられたら、ファイルシステムを作成し直さないかぎり、その数は変更できません。

1G バイト未満ファイルシステムが 1G バイト以下の場合、i ノード 1 個あたりのデフォルトのバイト数は 2048 バイト (2K バイト) です。ファイルシステムが 1G バイトを超える場合、次の公式が使用されます。

ファイルシステムのサイズ	iノードごとのバイト数
1G バイト以下	2048
2G バイト未満	4096
3G バイト未満	6144
3G バイト以上、1T バイト以下	8192
1T バイト超または -t オプションを指定して作成した場合	1048576

多数のシンボリックリンクを持つファイルシステムでは、平均ファイルサイズが小さくなることがあります。ファイルシステムに多数の小型ファイルが格納される場合は、このパラメータに小さい値を与えてもかまいません。iノード数が少ないためにiノードが不足するよりも、多すぎる方が好ましいことを留意してください。iノード数が少なすぎると、実際には空のディスクスライス上でも最大ファイル数に達してしまうことがあります。

## UFS ファイルとファイルシステムの最大サイズ

UFS ファイルシステムの最大サイズは、およそ 16T バイトで、使用できる領域は、そこから約 1 パーセントのオーバーヘッドを引いたサイズです。「空白」ファイルの論理サイズは 1T バイトです。ただし、ファイルに格納できる実際のデータ容量は、1T バイトから約 1 パーセントのファイルシステムオーバーヘッドを引いたサイズです。

## UFS サブディレクトリの最大数

UFS ファイルシステム内のディレクトリあたりのサブディレクトリの最大数は、32,767 です。この制限はあらかじめ定義されたものなので、変更できません。

## UFS ファイルシステムのバックアップと復元 (概要/タスク)

---

この章では、`ufsdump` コマンドを使用したファイルシステムのバックアップ手順について説明します。

手順の詳細は、[396 ページの「UFS ファイルとファイルシステムのバックアップ \(タスクマップ\)」](#)を参照してください。

バックアップの実行の概要については、[379 ページの「UFS ファイルシステムのバックアップと復元の概要」](#)を参照してください。

個々のファイルをフロッピーディスクにバックアップする方法については、[第 21 章「ファイルとファイルシステムのコピー \(タスク\)」](#)を参照してください。

`ufsdump` コマンドの追加情報については、[第 23 章「UFS バックアップおよび復元コマンド \(参照情報\)」](#)を参照してください。

## UFS ファイルシステムのバックアップと復元の概要

ファイルシステムの「バックアップ」とは、消失、損傷、または破損に備えて、ファイルシステムをリムーバブルメディア (テープなど) にコピーすることを意味します。ファイルシステムの「復元」とは、現在のバックアップファイルをリムーバブルメディアから作業ディレクトリに適宜コピーすることを意味します。

この章では、UFS ファイルシステムのバックアップおよび復元に使用する `ufsdump` および `ufsrestore` コマンドについて説明します。ほかのコマンドを使用してファイルやファイルシステムをコピーし、ファイルの共有や移動を行うこともできます。次の表に、個々のファイルやファイルシステムをほかのメディアにコピーするためのすべてのコマンドの参照先を示します。

表 19-1 ファイルとファイルシステムのバックアップおよび復元用コマンド

タスク	コマンド	参照先
1 つまたは複数のファイルシステムをローカルまたはリモートのテープデバイスにバックアップします。	ufsdump	396 ページの「UFS ファイルとファイルシステムのバックアップ (タスクマップ)」または第 23 章「UFS バックアップおよび復元コマンド (参照情報)」
ファイルシステムの読み取り専用コピーを作成します。	fssnap	第 20 章「UFS スナップショットの使用 (手順)」
ネットワーク上のシステムのファイルシステム全体をバックアップサーバーからバックアップします。	Solstice Backup ソフトウェア	『Solstice Backup 6.1 Administration Guide』
NIS+ マスターサーバーをバックアップし、復元します。	nisbackup および nisrestore	『Solaris のシステム管理 (ネーミングとディレクトリサービス : NIS+ 編)』
テープまたはフロッピーディスク上でファイルをコピー、表示、検索します。	tar、cpio、または pax コマンド	第 21 章「ファイルとファイルシステムのコピー (タスク)」
マスターディスクをクローンディスクにコピーします。	dd	第 21 章「ファイルとファイルシステムのコピー (タスク)」
ファイルシステム全体または個々のファイルを、リムーバブルメディアから作業ディレクトリに復元します。	ufsrestore	404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する (タスクマップ)」

## ファイルシステムをバックアップする理由

ファイルのバックアップは、もっとも重要なシステム管理作業の 1 つです。次のような問題によるデータの消失に備えて、定期的にバックアップを実行することをお勧めします。

- システムのクラッシュ
- 不注意によるファイルの削除
- ハードウェア障害
- 天災 (火災、台風、地震など)
- システムの再インストールやアップグレード時に発生する問題

## どの **UFS** ファイルシステムをバックアップするか の計画

頻繁に更新されるファイルシステムなど、ユーザーにとって重要なファイルシステムはバックアップしておく必要があります。次の表に、スタンドアロンシステムおよびサーバーについてバックアップすべきファイルシステムに関する一般的なガイドラインを示します。

表 19-2 スタンドアロンシステムにおけるバックアップ対象のファイルシステム

バックアップするファイルシステム	説明	バックアップ間隔
ルート (/) – スライス 0	このファイルシステムにはカーネルが格納されていますが、/var ディレクトリが格納されていることもあります。/var ディレクトリには、一時ファイル、ログインファイル、またはステータスファイルのほか、頻繁に更新されるシステムアカウンティングファイルやメールファイルが含まれる場合もあります。	定期的 (毎週、毎日など)
/usr – スライス 6、/opt	/usr および /opt ファイルシステムには、ソフトウェアと実行可能ファイルが含まれます。/opt ディレクトリは、ルート (/) の一部であるか、独自のファイルシステムです。	随時
/export/home – スライス 7	このファイルシステムには、スタンドアロンシステム上のすべてのユーザーのディレクトリおよびサブディレクトリを格納できます。	ルート (/) や /usr よりも頻繁に、サイトのニーズによっては毎日

表 19-2 スタンドアロンシステムにおけるバックアップ対象のファイルシステム (続き)

バックアップするファイルシステム	説明	バックアップ間隔
/export、/var、またはその他のファイルシステム	/export ファイルシステムには、ディスクレスクライアント用のカーネルおよび実行可能プログラムが含まれます。/var ディレクトリには、一時ファイル、ログインファイル、またはステータスファイルが含まれます。	サイトの必要に応じて

表 19-3 UFS ファイルシステムをサーバー用にバックアップする

バックアップするファイルシステム	説明	バックアップ間隔
ルート (/) - スライス 0	このファイルシステムには、カーネルおよび実行可能プログラムが含まれます。	サイトの必要に応じて毎日 1 度ないし月に 1 度。  ネットワーク上でユーザーとシステムの追加および削除を頻繁に実行する場合、このファイルシステム内の構成ファイルを変更する必要があります。この場合、週に 1 度から月に 1 度の間隔で、ルート (/) ファイルシステムの完全バックアップを実行する必要があります。  サイトでユーザーのメールをメールサーバー上の /var/mail ディレクトリに保管しておき、後でクライアントシステムがマウントする場合は、ルート (/) ディレクトリを毎日バックアップすることをお勧めします。別のファイルシステムの場合は、/var を毎日バックアップすることをお勧めします。
/export - スライス 3	このファイルシステムには、ディスクレスクライアント用のカーネルおよび実行可能プログラムが含まれます。	サイトの必要に応じて毎日 1 度ないし月に 1 度。  このファイルシステム内の情報はスライス 0 にあるサーバーのルートディレクトリと同様なので、ファイルシステムは頻繁に変化することはありません。サイトからメールをクライアントシステムに送信していない場合、このファイルシステムのバックアップは随時に行うだけでかまいません。送信している場合は、/export のバックアップをさらに頻繁に行う必要があります。

表 19-3 UFS ファイルシステムをサーバー用にバックアップする (続き)

バックアップするファイルシステム	説明	バックアップ間隔
/usr – スライス 6、/opt	/usr および /opt ファイルシステムには、ソフトウェアと実行可能ファイルが含まれます。/opt ディレクトリは、ルート (/) の一部であるか、独自のファイルシステムです。	サイトの必要に応じて毎日 1 度ないし月に 1 度。  ソフトウェアの追加や削除が頻繁に行われなければ、これらのファイルシステムは比較的静的です。
/export/home – スライス 7	このファイルシステムには、システム全体のユーザーのホームディレクトリを格納できます。このファイルシステム内のファイルは、頻繁に変更されます。	毎日または毎週。

## バックアップタイプの選択

ufsdump コマンドを使用すると、完全バックアップまたは増分バックアップを実行できます。fssnap コマンドを使用すると、ファイルシステムの一時イメージを作成できます。次の表に、各バックアップタイプの手順の違いを示します。

表 19-4 バックアップタイプの相違点

バックアップのタイプ	結果	長所	短所
完全	ファイルシステムやディレクトリ全体をコピーします	すべてのデータが 1 か所に格納されます	大量のバックアップテープが必要であり、書き込みに時間がかかります。ドライブはテープ上でファイルが入っている位置に順番に移動しなければならないので、個々のファイルの検索時間が長くなります。複数のテープを検索しなければならない場合もあります。
スナップショット	ファイルシステムの一時イメージを作成します	システムはマルチユーザーモードで動作可能です	スナップショットの作成中は、システムのパフォーマンスが低下する可能性があります。

表 19-4 バックアップタイプの相違点 (続き)

バックアップのタイプ	結果	長所	短所
増分	指定されたファイルシステム内で、前回のバックアップ以降に変更されたファイルのみをコピーします	ファイルシステム内の小さな変化を簡単に検索できます	どの増分テープにファイルが入っているかを探すのに時間がかかることがあります。最後の完全バックアップに戻らなければならない場合もあります。

## テープデバイスの選択

次の表に、バックアップ処理中にファイルシステムを格納するための典型的なテープデバイスを示します。記憶容量は、ドライブのタイプとテープに書き込むデータのタイプによって異なります。テープデバイスの詳細は、[第 22 章「テープドライブの管理 \(タスク\)」](#)を参照してください。

表 19-5 ファイルシステムのバックアップに使用する一般的なメディア

バックアップメディア	記憶容量
1/2 インチのリールテープ	140M バイト (6250 bpi)
2.5G バイト、1/4 インチのカートリッジ (QIC) テープ	2.5G バイト
DDS3 4mm カートリッジテープ (DAT)	12 - 24G バイト
14G バイト、8mm カートリッジテープ	14G バイト
DLT 7000 1/2 インチ カートリッジテープ	35 - 70G バイト

# UFS ファイルシステムのバックアップおよび復元の概要 (タスクマップ)

このタスクマップを使用して、ファイルシステムのバックアップおよび復元タスクをすべて確認できます。各タスクは、実行するバックアップタイプの判別などの一連の追加タスクを指し示します。

タスク	説明	参照先
1. バックアップするファイルシステムを識別します。	毎日、毎週、または毎月、バックアップの必要なファイルシステムを識別します。	<a href="#">381 ページの「どの UFS ファイルシステムをバックアップするか計画」</a>



タスク	説明	参照先
2. バックアップタイプを決定します。	サイトに適したファイルシステムのバックアップタイプを決定します。	383 ページの「バックアップタイプの選択」
3. バックアップを作成します。	次のいずれかの方法を使用します。	
	<p>ファイルシステムの完全バックアップおよび増分バックアップを実行する場合は、<code>ufsdump</code> コマンドを使用します。</p> <p>アクティブかつマウント中のファイルシステムのスナップショットを作成する場合は、<code>fssnap</code> コマンドの使用を考慮します。</p> <p>個人のホームディレクトリや、小規模で重要度の低いファイルシステムの完全バックアップを作成する場合は、<code>tar</code>、<code>cpio</code>、または <code>pax</code> コマンドを使用します。</p>	<p>396 ページの「UFS ファイルとファイルシステムのバックアップ (タスクマップ)」</p> <p>第 20 章「UFS スナップショットの使用 (手順)」</p> <p>第 21 章「ファイルとファイルシステムのコピー (タスク)」</p>
4. (オプション) ファイルシステムを復元します。	<p>ファイルまたはファイルシステムのバックアップに使用するコマンドに基づいて、復元方法を選択します。</p> <p><code>ufsdump</code> コマンドを使用して作成したファイルシステムのバックアップを復元します。</p> <p><code>tar</code>、<code>cpio</code>、または <code>pax</code> コマンドを使用して作成したファイルシステムを復元します。</p>	<p>404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する (タスクマップ)」</p> <p>第 21 章「ファイルとファイルシステムのコピー (タスク)」</p>
5. (オプション) ルート (/) または /usr ファイルシステムを復元します。	ルート (/) または /usr ファイルシステムの復元は、それほど重要でないファイルシステムの復元よりも複雑です。これらのファイルシステムの復元中に、ローカルの Solaris DVD またはネットワークからブートする必要があります。	416 ページの「UFS ルート (/) および /usr ファイルシステムを復元する方法」

# バックアップスケジュールに関するその他の注意事項

「バックアップスケジュール」とは、`ufsdump` コマンドを実行するように設定するスケジュールです。このセクションでは、バックアップスケジュールの作成時に考慮すべき点について説明します。また、バックアップスケジュールの例もいくつか紹介します。

バックアップスケジュールは、次の点を考慮に入れて作成します。

- バックアップに使用するテープの数を最小限に抑える
- バックアップの実行に使用できる時間
- 損傷したファイルシステムの完全復元に使用できる時間
- 不注意に削除した個々のファイルの検索に使用できる時間

## バックアップ頻度

バックアップに費やす時間とメディアの数を最小限度に抑える必要がない場合は、完全バックアップを毎日実行してもかまいません。しかし、多くのサイトの場合、このようなバックアップ方法は現実的ではないので、ほとんどの場合は増分バックアップが使用されます。その場合は、サイトが過去4週間分のバックアップからファイルを十分復元できるようにしてください。そのためには、少なくとも1週分ごとに1組ずつ、合計4組のテープが必要です。各組のテープを毎月使い回すことになります。また、少なくとも1年分の月別のバックアップを保存する必要があります。そして、数年分の年度別バックアップを保管しておく必要があります。

## バックアップ間隔の用語と定義

次の表に、バックアップ間隔の用語および定義を示します。

用語	定義
スナップショット	ファイルシステムの一時イメージを作成します。
完全バックアップ	ファイルシステムまたはディレクトリの完全なコピーを作成します。

用語	定義
増分バックアップ	<p>指定したファイルシステムのファイルのうち、前回のバックアップ以降に変更されたファイルだけをコピーします。増分バックアップには、次の種類があります。</p> <ul style="list-style-type: none"><li>■ 日単位累積 - 月曜日に1日分のファイル変更をコピーします。次に、月曜日のバックアップを火曜日、水曜日のように、各曜日のファイル変更で上書きしていきます。</li><li>■ 日単位増分 - 1日分のファイル変更をコピーします。つまり、月曜日、火曜日のように、各曜日の変更の差分テープを記録していきます。</li><li>■ 週単位累積 - その週に変更されたファイルをコピーします。前の週のファイル変更も残しておきます。</li><li>■ 週単位増分 - 前の週のバックアップ以降に変更されたファイルをコピーします。</li></ul>

## バックアップスケジュールを設定する際のガイドライン

次の表に、バックアップスケジュールのガイドラインを示します。バックアップスケジュールに関するその他の注意事項については、[386 ページの「バックアップスケジュールに関するその他の注意事項」](#)を参照してください。

表 19-6 バックアップスケジュールのガイドライン

必要なファイル復元	バックアップ間隔	コメント
各ファイルの別バージョン (ワード処理に使用するファイルシステムなど) を復元する必要があります	作業日ごとに日単位増分バックアップを実行します。  日単位増分バックアップには同じテープを再利用しません。	このスケジュールでは、その日に変更があったすべてのファイルが保存されるだけでなく、下のレベルの最後のバックアップ以降に変更があったファイルがディスク上に残ります。ただし、このスケジュールの場合は、毎日異なるテープを使用する必要があります。このように運用しないと、必要なバージョンのファイルを復元できなくなることがあるためです。  たとえば、火曜日に変更があったファイルが木曜日にも変更されると、金曜日の下のレベルのバックアップでは、火曜日の夜ではなく木曜日の夜に変更されたように見えます。ユーザーが火曜日のバージョンを必要とする場合には、火曜日 (または水曜日) のバックアップテープを保存しておかなければ、それを復元できません。また、火曜日と水曜日には存在したファイルが木曜日に削除されても、金曜日の下のレベルのバックアップには表示されません。
ファイルシステム全体を短時間で復元する必要があります	下位のレベルのバックアップを頻繁に実行します。	-
同じサーバー上で多数のファイルシステムのバックアップを作成します	ファイルシステムごとにスケジュールをずらすことを検討します。	この方法では、すべてのレベル 0 のバックアップを同じ日に実行しないことになります。
使用するテープ数を最小限に抑えます	1 週間に実行する増分バックアップのレベルを上げます。  週末に実行するバックアップのレベルを上げます。日単位と週単位の増分バックアップを同じテープに入れます。	毎日の変更のみが各日単位テープに保存されます。  変更は (月単位ではなく) 週単位でしか週単位テープに保存されません。

表 19-6 バックアップスケジュールのガイドライン (続き)

必要なファイル復元	バックアップ間隔	コメント
	日単位と週単位の増分バックアップを同じテープに入れます。	この操作には、 <code>ufsdump</code> の 'no rewind' オプションを使用します。たとえば、 <code>/dev/rmt/0n</code> のように指定します。

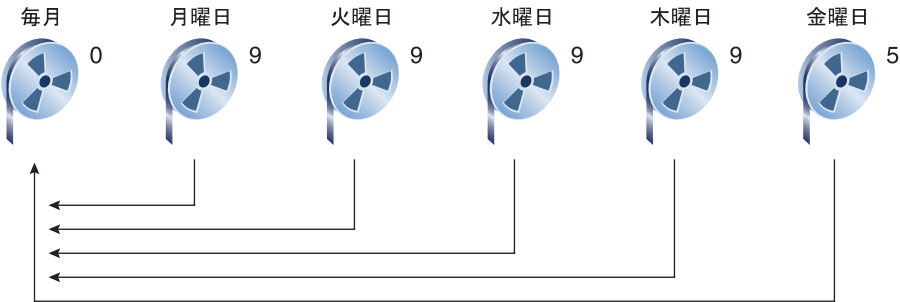
## ダンプレベルを使用して増分バックアップを作成する

`ufsdump` コマンドで指定するダンプレベル (0-9) により、バックアップするファイルを決定できます。ダンプレベル 0 を指定すると、完全バックアップが作成されます。増分バックアップのスケジュール設定にレベル 1 から 9 までの番号が使用されますが、意味は定義されていません。レベル 1 から 9 は、累積バックアップまたは個別バックアップのスケジュール設定に使用する番号の範囲にすぎません。レベル 1 から 9 までは意味するのは、大小による相互関係だけです。下位ダンプレベル番号を使用した場合は、常に完全バックアップまたは累積バックアップが再開されます。次の例で、1 から 9 までのレベルを使用する増分ダンプの手順を示します。

### 例—日単位累積バックアップのダンプレベル

日単位の累積増分バックアップを実行するのが、もっとも一般に使用されるバックアップスケジュールで、ほとんどの場合に推奨できます。次の例で、月曜日から木曜日までレベル 9 のダンプを使用し、金曜日にはレベル 5 のダンプを使用してプロセスを再開するスケジュールを示します。

図 19-1 増分バックアップ: 日単位累積

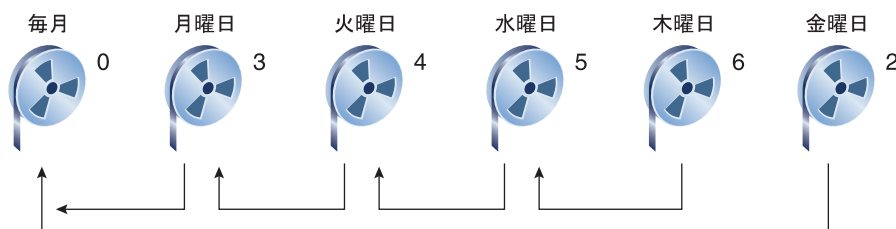


前の例では、1 から 9 までの範囲内で他の番号を使用しても同じ結果が得られます。ポイントは、月曜日から木曜日まで同じ番号を使用し、金曜日にはそれより「小さい」番号を使用することです。たとえば、レベル 4、4、4、4、2 や 7、7、7、7、5 を指定してもかまいません。

## 例—日単位増分バックアップのダンプレベル

次の例で、1 日分の作業内容のみを別々のテープ上で保存するスケジュールを示します。このようなバックアップは、日単位増分バックアップと呼ばれます。この場合、月曜日から木曜日までは連続するダンプレベル番号(3、4、5、6)を使用し、金曜日にはそれより小さい番号(2)を使用します。金曜日はレベル番号が小さいので、プロセスが再開されます。

図 19-2 増分バックアップ: 日単位増分



前の例では、連番 6、7、8、9 の次に 2 を使用したり、5、6、7、8 の次に 3 を使用したりすることもできます。番号自体は特定の意味を持たないことに注意してください。この例が示しているように、連続した番号を並べることによって、それらの番号に意味が生まれます。

## バックアップスケジュールの例

バックアップスケジュールの例をいくつか紹介します。どのスケジュールも、完全バックアップ(ダンプレベル 0)から始めることと、`-u` オプションを使用して `/etc/dumpdates` ファイルに各バックアップを記録することを前提としています。

### 例—日単位累積、週単位累積バックアップスケジュール

表 19-7 に、よく使用される増分バックアップスケジュールを示します。これは、ほとんどの場合に推奨できるスケジュールです。このスケジュールで実行される処理は次のとおりです。

- 前週の終わりの下位レベルのバックアップ以降に変更があったすべてのファイルが毎日保存されます。
- 月-金のレベル 9 のバックアップの場合は、直前のレベル 0 またはレベル 5 のバックアップがもっとも近い下位バックアップレベルになります。したがって、各週のテープには、前週の終わり(第 1 週の場合は初期レベル 0 のバックアップ)以降に変更があったすべてのファイルが累積されます。

- 毎週金曜日のレベル5のバックアップの場合、もっとも近い下位レベルのバックアップは、月初めに実行されたレベル0のバックアップです。したがって、毎週金曜日のテープには、月初めからその時点までに変更があったすべてのファイルが入っています。

表 19-7 日単位累積/週単位累積バックアップスケジュール

	開始日	月	火	水	木	金
月の1日目	0					
第1週		9	9	9	9	5
第2週		9	9	9	9	5
第3週		9	9	9	9	5
第4週		9	9	9	9	5

次の表に、日単位累積、週単位累積スケジュールを使用して、テープの内容が2週間でどのように変化するかを示します。各文字はそれぞれ異なるファイルを示します。

表 19-8 日単位累積、週単位累積スケジュールのテープの内容

	月	火	水	木	金
第1週	a b	a b c	a b c d	a b c d e	a b c d e f
第2週	g	g h	g h i	g h i j	a b c d e f g h i j k

## 日単位累積、週単位累積スケジュールのテープ要件

このスケジュールでは、日単位テープを再利用する場合には、6本のテープが必要になります。ただし、曜日ごとに4本の日単位テープを別々に使用する場合は、次に示すように9本のテープが必要になります。

- レベル0のバックアップ用に1本
- 金曜日用に4本
- 日単位テープ用に1本または4本

ファイルシステム全体を復元する必要がある場合は、次のテープが必要になります。

- レベル0テープ
- 前週の金曜日用に1本
- 前週の金曜日以降の最新の日単位テープ1本

### 例 一日単位累積、週単位増分バックアップスケジュール

次の表に、各曜日のテープに月曜日 (第 1 週の場合は初期レベル 0 のバックアップ) 以降に変更があったすべてのファイルが累積されるスケジュールを示します。また、各金曜日のテープには、その週に変更されたすべてのファイルが入ります。

表 19-9 日単位累積、週単位増分バックアップスケジュール

	開始日	月	火	水	木	金
月の 1 日目	0					
第 1 週		9	9	9	9	3
第 2 週		9	9	9	9	4
第 3 週		9	9	9	9	5
第 4 週		9	9	9	9	6

次の表に、日単位累積、週単位増分バックアップスケジュールの場合、テープの内容が 2 週間でどのように変化するかを示します。各文字はそれぞれ異なるファイルを示します。

表 19-10 日単位累積、週単位増分バックアップスケジュールのテープの内容

	月	火	水	木	金
第 1 週	ab	abc	abcd	abcde	abcdef
第 2 週	g	gh	ghi	ghij	ghijk

### 日単位累積、週単位増分バックアップスケジュールのテープ要件

このスケジュールでは、日単位テープを再利用する場合には、6 本のテープが必要になります。ただし、曜日ごとに 4 本の日単位テープを別々に使用する場合は、次に示すように 9 本のテープが必要になります。

- レベル 0 のバックアップ用に 1 本
- 金曜日用に 4 本
- 日単位テープ用に 1 本または 4 本

ファイルシステム全体を復元する必要がある場合は、次のテープが必要になります。

- レベル 0 テープ
- すべての金曜日のテープ
- 前週の金曜日以降の最新の日単位テープ 1 本



## 例 一日単位増分、週単位累積バックアップスケジュール

次の表に、各曜日のテープには前日以降に変更があったファイルのみが入るスケジュールを示します。また、各金曜日のテープには、月初めの初期レベル0バックアップ以降に変更されたすべてのファイルが入ります。

表 19-11 日単位増分、週単位累積バックアップスケジュール

	開始日	月	火	水	木	金
月の1日目	0					
第1週		3	4	5	6	2
第2週		3	4	5	6	2
第3週		3	4	5	6	2
第4週		3	4	5	6	2

次の表に、日単位増分、週単位累積スケジュールの場合、テープの内容が2週間でどのように変化するかを示します。各文字はそれぞれ異なるファイルを示します。

表 19-12 日単位増分、週単位累積バックアップスケジュールのテープの内容

	月	火	水	木	金
第1週	ab	cd	efg	hi	abcdefghi
第2週	jkl	m	no	pq	abcdefghijklm nopqrs

## 日単位増分、週単位累積スケジュールのテープ要件

このスケジュールでは、少なくとも9本のテープが必要になります。これは日単位テープを毎週再利用することが前提となっていますが、これはお勧めできません。週単位テープを1か月保存する場合は、21本のテープが必要になります。可能であれば、こちらの方法をお勧めします。内訳は、レベル0に1本、金曜日用に4本、日単位テープ4本または16本です。

- レベル0のバックアップ用に1本
- 金曜日のバックアップ用に4本
- 日単位のテープを4本または16本

ファイルシステム全体を復元する必要があるれば、次のテープが必要になります。

- レベル0テープ
- 前週の金曜日用に1本
- 前週の金曜日以降のその週のすべての日単位テープ

## 例—サーバーの月単位バックアップスケジュール

次の表に、ユーザーがプログラム開発やドキュメント作成のようなファイル集約型の作業を実行する小型ネットワーク上の、使用頻度の高いファイルサーバーのバックアップ方針の例を示します。この例は、バックアップ期間が日曜日に始まり、1 週 7 日間を 4 週間行うものと想定しています。

表 19-13 例—サーバーの月単位バックアップスケジュール

ディレクトリ	日付	ダンプレベル	テープ名
ルート (/)	第 1 日曜日	0	<i>n</i> 本のテープ
/usr	第 1 日曜日	0	<i>n</i> 本のテープ
/export	第 1 日曜日	0	<i>n</i> 本のテープ
/export/home	第 1 日曜日	0	<i>n</i> 本のテープ
	第 1 月曜日	9	A
	第 1 火曜日	9	B
	第 1 水曜日	5	C
	第 1 木曜日	9	D
	第 1 金曜日	9	E
	第 1 土曜日	5	F
ルート (/)	第 2 日曜日	0	<i>n</i> 本のテープ
/usr	第 2 日曜日	0	<i>n</i> 本のテープ
/export	第 2 日曜日	0	<i>n</i> 本のテープ
/export/home	第 2 日曜日	0	<i>n</i> 本のテープ
	第 2 月曜日	9	G
	第 2 火曜日	9	H
	第 2 水曜日	5	I
	第 2 木曜日	9	J
	第 2 金曜日	9	K
	第 2 土曜日	5	L
ルート (/)	第 3 日曜日	0	<i>n</i> 本のテープ
/usr	第 3 日曜日	0	<i>n</i> 本のテープ
/export	第 3 日曜日	0	<i>n</i> 本のテープ

表 19-13 例—サーバーの月単位バックアップスケジュール (続き)

ディレクトリ	日付	ダンプレベル	テープ名
/export/home	第 3 日曜日	0	$n$ 本のテープ
	第 3 月曜日	9	M
	第 3 火曜日	9	N
	第 3 水曜日	5	O
	第 3 木曜日	9	P
	第 3 金曜日	9	Q
	第 3 土曜日	5	R
ルート (/)	第 4 日曜日	0	$n$ 本のテープ
/usr	第 4 日曜日	0	$n$ 本のテープ
/export	第 4 日曜日	0	$n$ 本のテープ
/export/home	第 4 日曜日	0	$n$ 本のテープ
	第 4 月曜日	9	S
	第 4 火曜日	9	T
	第 4 水曜日	5	U
	第 4 木曜日	9	V
	第 4 金曜日	9	W
	第 4 土曜日	5	X

このスケジュールでは、 $4n$  本のテープ (ルート (/)、/usr、/export、および /export/home ファイルシステムの 4 回の完全バックアップに必要な本数) を使用します。また、/export/home ファイルシステムの増分バックアップ用に 24 本のテープを使用します。このスケジュールは、増分バックアップごとに 1 本ずつテープを使用し、それを 1 カ月は保存することを前提としています。

このスケジュールの機能を、次に示します。

1. 日曜日ごとに、ルート (/)、/usr、/export、および /export/home ファイルシステムの完全バックアップ (レベル 0) を実行します。レベル 0 のテープを少なくとも 3 カ月は保存します。
2. 月の第 1 月曜日に、テープ A を使用して /export/home ファイルシステムのレベル 9 のバックアップを実行します。ufsdump コマンドは、下のレベルのバックアップ以降に変更があったすべてのファイルをコピーします。この場合、直前の下位レベルのバックアップは、日曜日に実行したレベル 0 のバックアップになります。

- 3. 月の第1火曜日に、テープBを使用して /export/home ファイルシステムのレベル9のバックアップを実行します。この場合も、`ufsdump` コマンドは、直前の下のレベルのバックアップ(日曜日のレベル0のバックアップ)以降に変更があったすべてのファイルをコピーします。
  - 4. 月の第1水曜日に、テープCを使用して /export/home ファイルシステムのレベル5のバックアップを実行します。`ufsdump` コマンドは、日曜日以降に変更があったすべてのファイルをコピーします。
  - 5. テープDとE上で、/export/home ファイルシステムの木曜日と金曜日のレベル9バックアップを実行します。`ufsdump` コマンドは、直前の低レベルバックアップ(水曜日のレベル5バックアップ)以降に変更があったすべてのファイルをコピーします。
  - 6. 月の第1土曜日に、テープFを使用して /export/home のレベル5のバックアップを実行します。`ufsdump` コマンドは、下のレベルのバックアップ(この場合は日曜日に実行したレベル0のバックアップ)以降に変更があったすべてのファイルをコピーします。テープを再利用する場合は、テープAからFまでを次の4週間の第1月曜日までは保存しておきます。
  - 7. 次の3週間は、テープGからLまでと、日曜日のレベル0のバックアップ用に4n本のテープを使用して、手順1から6までを繰り返します。
  - 8. 4週ごとに、レベル0のバックアップ用に新しいテープ1組と、増分バックアップ用のテープAからXまでを再利用して、手順1から7までを繰り返します。レベル0のテープは、3カ月後に再利用できるようになります。
- このスケジュールでは、各ファイルを1カ月間で段階別に保存できます。多数のテープが必要ですが、テープのライブラリを確実に用意できます。テープの本数を減らすには、テープAからFまでを毎週再利用します。

## UFS ファイルとファイルシステムのバックアップ(タスクマップ)

タスク	説明	参照先
1. ファイルシステムのバックアップの準備をします。	バックアップで使用するファイルシステム、バックアップタイプ、テープデバイスを確認します。	<a href="#">397 ページの「ファイルシステムバックアップの実行準備」</a>
2. ファイルシステムのバックアップに必要なテープの数を決定します。	ファイルシステムの完全バックアップに必要なテープの数を決定します。	<a href="#">398 ページの「完全バックアップに必要なテープ数を決定する」</a>

タスク	説明	参照先
3. ファイルシステムのバックアップを実行します。	<p>ファイルシステムの完全バックアップを実行して、全ファイルの基準コピーを取得します。</p> <p>日単位で変更されたファイルのコピーを保持することが、サイトで重要かどうかに基づいて、ファイルシステムの増分バックアップを実行します。</p>	399 ページの「UFS ファイルシステムのバックアップをテープに作成する方法」

# ファイルシステムバックアップの実行準備

ファイルシステムのバックアップの準備は、計画 (379 ページの「UFS ファイルシステムのバックアップと復元の概要」を参照) から始め、さらに、次を選択します。

- バックアップを作成するファイルシステム
- 実行するバックアップのタイプ (完全または増分)
- バックアップのスケジュール
- テープドライブ

詳細は、379 ページの「UFS ファイルシステムのバックアップと復元の概要」を参照してください。

このセクションでは、ファイルシステムのバックアップを作成する前に実行する必要がある、次の2つのタスクについて説明します。

- バックアップを作成するファイルシステムの名前を検索する
- 完全バックアップの作成に必要なテープの本数を決定する

## ▼ UFS ファイルシステム名を検索する方法

- 1 `/etc/vfstab` ファイルの内容を表示します。  
\$ `more /etc/vfstab`
- 2 `mount point` 列に表示されるファイルシステム名を調べます。
- 3 ファイルシステムのバックアップを作成する際、`mount point` 列に表示されたディレクトリ名を使用します。

### 例 19-1 ファイルシステム名を検索する

この例では、バックアップ対象のファイルシステムはルート (`/`)、`/usr`、および `/export/home` です。

```
more /etc/vfstab
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
fd - /dev/fd fd - no -
/proc - /proc proc - no
/dev/dsk/c0t0d0s1 - - swap - no -
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
/dev/dsk/c0t0d0s6 /dev/rdisk/c0t0d0s6 /usr ufs 1 no -
/dev/dsk/c0t0d0s7 /dev/rdisk/c0t0d0s7 /export/home ufs 2 yes -
/devices - /devices devfs - no -
sharefs - /etc/dfs/sharetab sharefs - no -
ctfs - /system/contract ctfs - no -
objfs - /system/object objfs - no
swap - /tmp tmpfs - yes
```

▼ 完全バックアップに必要なテープ数を決定する

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 バックアップのサイズをバイト単位で予測します。

# **ufsdump** [0]S *file-system*

ファイルシステムの1回目のバックアップ時には、s オプションを使用して、バックアップに必要な予想バイト数を表示します。

ファイルシステムの2回目以降のバックアップ時には、0s オプションを使用して、バックアップに必要な予想バイト数を表示します。

- 3 予測サイズをテープの容量で割り、必要なテープの数を決定します。  
テープの容量の一覧は、[表 19-5](#) を参照してください。

例 19-2 テープの本数を決定する

次の例では、150M バイトのテープにファイルシステムが入ります。

```
ufsdump S /export/home
178176
```

UFS ファイルシステムのバックアップ

- バックアップを実行する際の一般的なガイドラインは次のとおりです。
- シングルユーザーモードを使用するか、ファイルシステムをマウント解除します (ファイルシステムのスナップショットを作成している場合を除く)。UFS スナップショットの詳細は、[第 20 章「UFS スナップショットの使用\(手順\)」](#) を参照してください。

- ディレクトリレベルの処理 (ファイルの作成、削除、名前変更など) とファイルレベルの処理が同時に行われているときにファイルシステムのバックアップを実行すると、バックアップに組み込まれないデータがあるので注意してください。
- 単一のシステムで `ufsdump` コマンドを実行し、リモートシェルまたはリモートログインを通じてネットワーク上のシステムグループのバックアップをリモート実行できます。また、出力をテープデバイスがあるシステムに転送できます。通常、テープデバイスは `ufsdump` コマンドを実行するシステム上にありますが、ほかの場所にあってもかまいません。

ファイルのバックアップをリモートデバイスに作成するには、`ufsdump` コマンドの出力を `dd` コマンドにパイプする方法もあります。`dd` コマンドの使用方法については、[第 21 章「ファイルとファイルシステムのコピー \(タスク\)」](#)を参照してください。

- ネットワーク上でリモートバックアップを実行する場合、テープデバイスを持つシステムの `/rhosts` ファイル中には、デバイスを使用する各クライアントのエントリが入っている必要があります。また、バックアップを実行する各システムの `/rhosts` ファイルには、バックアップを開始するシステムのエントリが入っている必要があります。

## ▼ UFS ファイルシステムのバックアップをテープに作成する方法

`ufsdump` 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作成する一般的な手順を示します。この例では、オプションと引数の使用方法を示しています。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 システムをレベル 5 (シングルユーザーモード) に移行します。

例:

```
shutdown -g30 -y
```

- 3 (オプション) ファイルシステムの整合性を検査します。

例:

```
fsck -m /dev/rdisk/c0t0d0s7
```

`fsck -m` コマンドを実行すると、ファイルシステムの整合性が検査されます。たとえば、電源障害が発生すると、ファイルが不整合になることがあります。`fsck` コマンドの詳細は、[第 17 章「UFS ファイルシステムの整合性検査 \(手順\)」](#)を参照してください。

- 4 ファイルシステムをリモートテープドライブにバックアップする場合は、次の手順を実行します。
  - a. テープドライブが接続されているシステム(テープサーバー)の `./rhosts` ファイルに、次のエントリを追加します。  

```
host root
```

  
`host` エントリには、`ufsdump` コマンドを実行してバックアップを行うシステムの名前を指定します。
  - b. テープサーバー上で、上記の `./rhosts` ファイルに追加したホストに、ネームサービス経由でアクセスできることを確認します。

- 5 テープドライブのデバイス名を確認します。  
デフォルトのテープドライブは、`/dev/rmt/0` です。

- 6 書き込み可能なテープをテープドライブに挿入します。

- 7 システムのバックアップを実行します。

```
ufsdump options arguments filenames
```

ファイルシステムやディレクトリ、またはファイルシステム内のファイルをバックアップできます。ファイルを個別にバックアップする方法については、[tar\(1\)](#) または [cpio\(1\)](#) のマニュアルページを参照してください。

次の例では、もっとも一般的な `ufsdump` オプションおよび引数の使用方法を示します。

- [例 19-3](#)
- [例 19-4](#)
- [例 19-5](#)
- [例 19-6](#)

その他の `ufsdump` オプションおよび引数については、[第 23 章「UFS バックアップおよび復元コマンド \(参照情報\)」](#)を参照してください。

- 8 プロンプトが表示されたら、テープを取り出して、次のテープを挿入します。
- 9 各テープにボリューム番号、ダンプレベル、日付、システム名、ディスクスライス、ファイルシステム名を記入したラベルを貼ります。
- 10 **Control+D** キーを押してシステムをレベル 3 の動作に戻します。
- 11 バックアップが正常に実行されたことを確認します。

```
ufsrestore tf device-name
```



**例 19-3 UFS ルート (/) ファイルシステムの完全バックアップの実行**

次の例では、ルート (/) ファイルシステムの完全バックアップを実行する方法を示します。この例では、バックアップの実行前にシステムをシングルユーザーモードにしています。次の `ufsdump` オプションが含まれます。

- `0` は、レベル `0` のダンプ (完全バックアップ) であることを示します。
- `u` は、このバックアップの実行日に `/etc/dumpdates` ファイルが更新されることを示します。
- `c` は、カートリッジテープデバイスを示します。
- `f /dev/rmt/0` は、テープデバイスを示します。
- `/` はバックアップするファイルシステムを示します。

例:

```
init 0
ok boot -s
.
.
.
ufsdump 0ucf /dev/rmt/0 /
DUMP: Date of this level 0 dump: Mon Jul 12 10:00:50 2010
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c0t0d0s0 (starbug:/) to /dev/rmt/0.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 63 Kilobyte records
DUMP: Estimated 1843276 blocks (900.04MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 1843252 blocks (900.03MB) on 1 volume at 6602 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 0 dump on Mon Jul 12 10:22:50 MDT 2010
ufsrestore tf /dev/rmt/0
2 .
3 ./lost+found
4 ./usr
5 ./var
6 ./export
7 ./bin
8 ./dev
.
.
.
(Press Control-D to bring system to run level 3)
```

**例 19-4 UFS ルート (/) ファイルシステムの増分バックアップの実行**

次の例は、ルート (/) ファイルシステムのシングルユーザーモードでの増分バックアップの方法を示します。次の `ufsdump` オプションが含まれます。

- `9` は、レベル `9` のダンプ (増分バックアップ) であることを示します。

- `u` は、このバックアップの実行日に `/etc/dumpdates` ファイルが更新されることを示します。
- `c` は、カートリッジテープデバイスを示します。
- `f /dev/rmt/0` は、テープデバイスを示します。
- `/` はバックアップするファイルシステムを示します。

```
init 0
ok boot -s
ufsdump 9ucf /dev/rmt/0 /
DUMP: Date of this level 9 dump: Mon Jul 12 10:52:50 2010
DUMP: Date of last level 0 dump: Mon Jul 12 10:50:50 2010
DUMP: Dumping /dev/rdisk/c0t0d0s0 (starbug:/) to /dev/rmt/0.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Mapping (Pass II) [directories]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 63 Kilobyte records
DUMP: Estimated 422 blocks (211KB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 376 blocks (188KB) on 1 volume at 1843 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 9 dump on Mon Jul 12 10:55:50 2010
ufsrestore tf /dev/rmt/0
2 .
9 ./etc
304 ./etc/dumpdates
1117 ./devices
1118 ./devices/pseudo
3381 ./devices/pseudo/pts@0:1
.
.
.

init 0
ok boot -s
ufsdump 9ucf /dev/rmt/0 /
DUMP: Date of this level 9 dump: Mon Jul 12 10:33:14 2010
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c2t1d0s0 (v880-brm-03:/root) to /dev/rmt/0.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 63 Kilobyte records
DUMP: Estimated 120694 blocks (58.93MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 120580 blocks (58.88MB) on 1 volume at 9026 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 9 dump on Mon Jul 12 10:33:14 2010
ufsrestore tf /dev/rmt/0
2 .
9 ./etc
304 ./etc/dumpdates
1117 ./devices
1118 ./devices/pseudo
```

```

3381 ./devices/pseudo/pts@0:1
.
.
.

```

### 例 19-5 UFS ホームディレクトリの完全バックアップ

次の例では、`/export/home/kryten` ホームディレクトリの完全バックアップを実行する方法を示します。次の `ufsdump` オプションが含まれます。

- `0` は、レベル `0` のダンプ (完全バックアップ) であることを示します。
- `u` は、このバックアップの実行日に `/etc/dumpdates` ファイルが更新されることを示します。
- `c` は、カートリッジテープデバイスを示します。
- `f /dev/rmt/0` は、テープデバイスを示します。
- `/export/home/kryten` は、バックアップするディレクトリを示します。

```

ufsdump 0ucf /dev/rmt/0 /export/home/kryten
DUMP: Date of this level 0 dump: Wed Jul 10:35:18 2010
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c0t0d0s7 (starbug:/export/home) to /dev/rmt/0.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 63 Kilobyte records
DUMP: Estimated 105158 blocks (51.35MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 105082 blocks (51.31MB) on 1 volume at 5025 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 9 dump on Mon Jul 12 10:38:14 2010
ufsrestore tf /dev/rmt/0
232 ./kryten
233 ./kryten/filea
234 ./kryten/fileb
235 ./kryten/filec
236 ./kryten/letters
237 ./kryten/letters/letter1
238 ./kryten/letters/letter2
239 ./kryten/letters/letter3
240 ./kryten/reports
241 ./kryten/reports/reportA
242 ./kryten/reports/reportB
243 ./kryten/reports/reportC

```

### 例 19-6 リモートシステムへの完全バックアップの実行 (Solaris 10 データを Solaris 10 システムへ)

次の例は、Solaris 10 システム (mars) 上のローカルの `/export/home` ファイルシステムをリモートの Solaris 10 システム (earth) へ、シングルユーザーモードで完全バックアップする方法を示します。リモートアクセスが有効になるようにシステムを設定する必要があります。次の `ufsdump` オプションが含まれます。

- 0 は、レベル 0 のダンプ (完全バックアップ) であることを示します。
- u は、このバックアップの実行日に /etc/dumpdates ファイルが更新されることを示します。
- c は、カートリッジテープデバイスを示します。
- f earth:/dev/rmt/0 は、リモートシステム名およびテープデバイスを示します。
- /export/home は、バックアップするファイルシステムを示します。

```
ufsdump 0ucf earth:/dev/rmt/0 /export/home
DUMP: Date of this level 0 dump: Mon Jul 12 10:36:30 MDT 2010
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c0t0d0s7 (mars:/export/home) to earth:/dev/rmt/0.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 63 Kilobyte records
DUMP: Estimated 105172 blocks (51.35MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 105082 blocks (51.31MB) on 1 volume at 4425 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 0 dump on Mon Jul 12 10:39:30 MDT 2010
ufsrestore tf earth:/dev/rmt/0
2 .
3 ./lost+found
4 ./kryten
5 ./kryten/filea
6 ./kryten/fileb
7 ./kryten/filec
8 ./kryten/letters
9 ./kryten/letters/letter1
10 ./kryten/letters/letter2
11 ./kryten/letters/letter3
12 ./kryten/reports
.
.
.
```

# UFS ファイルおよびファイルシステムのバックアップを復元する (タスクマップ)

次のタスクマップに、ファイルおよびファイルシステムの復元手順を示します。

タスク	説明	参照先
ファイルとファイルシステムを復元するための準備を行います。	復元するファイルシステムやファイル、テープデバイス、およびそれらの復元方法を決定します。	405 ページの「UFS ファイルとファイルシステムを復元するための準備」

タスク	説明	参照先
使用するテープを決定します。	バックアップテープを調べ、復元するファイルまたはファイルシステムを含む最新のバックアップの日付を確認します。	<a href="#">407 ページの「使用するテープを決定する方法」</a>
ファイルを復元します。	<p>次のいずれかの復元方法を選択します。</p> <p>ファイルの対話的な復元 – メディアの内容を参照してファイルおよびディレクトリ単位で選択できるため、正確なファイル名を把握していない場合、この方法を使用します。</p> <p>ファイルの非対話的な復元 – 復元するファイル名をすでに把握している場合には、この方法を使用します。</p> <p>ファイルシステムの復元 – 復元時の手順の一部として新規ディスクドライブを使用できる場合は、この方法を使用します。</p>	<p><a href="#">408 ページの「対話式で UFS ファイルを復元する方法」</a></p> <p><a href="#">410 ページの「特定の UFS ファイルを復元する方法」</a></p> <p><a href="#">412 ページの「UFS ファイルシステム全体を復元する方法」</a></p>
ルート (/) または /usr ファイルシステムを復元します。	ルート (/) または /usr ファイルシステムを復元する場合には、ローカルの Solaris DVD またはネットワークからシステムをブートする必要があります。	<a href="#">416 ページの「UFS ルート (/) および /usr ファイルシステムを復元する方法」</a>

## UFS ファイルとファイルシステムを復元するための準備

ufsrestore コマンドは、ufsdump コマンドを使用して作成されたバックアップから、ディスクの現在の作業ディレクトリにファイルをコピーします。ufsrestore コマンドを使用すると、レベル 0 のダンプとそれ以降の増分ダンプからファイルシステム階層全体を読み込み直すことができます。また、このコマンドを使用して、任意のバックアップテープから個々のファイルを復元することもできます。ufsrestore コマンドをスーパーユーザーとして実行した場合には、ファイルの所有者、最新の変更時間、モード (ファイルのアクセス権) は元のまま、ファイルが復元されます。

ファイルまたはファイルシステムの復元を開始する前に、次の点を確認してください。

- 復元に必要なテープ (またはフロッピーディスク)
- ファイルシステム全体を復元する raw デバイス名
- 使用するテープデバイスのタイプ
- テープデバイスのデバイス名 (ローカルまたはリモート)

## UFS ファイルシステム名の確認

バックアップテープに適切な名前が付いている場合は、テープラベルに入っているファイルシステム名 (`/dev/rdisk/device-name`) が使えるはずです。詳細については、[397 ページの「UFS ファイルシステム名を検索する方法」](#)を参照してください。

## 必要なテープデバイスのタイプの決定

ファイルを復元するには、バックアップメディアと互換性のあるテープデバイスを使用する必要があります。バックアップメディアの形式によって、ファイルの復元にどんなドライブを使用しなければならないかが決まります。たとえば、使用するバックアップメディアが 8mm テープの場合、ファイルの復元には 8mm テープデバイスを使わなければなりません。

## テープデバイス名の決定

テープデバイス名 (`/dev/rmt/n`) をバックアップテープラベル情報の一部として指定している可能性があります。同じドライブを使ってバックアップテープを復元しようとする場合には、ラベル内にあるデバイス名を使うことができます。メディアデバイスおよびデバイス名の詳細については、[第 22 章「テープドライブの管理\(タスク\)」](#)を参照してください。

## UFS ファイルとファイルシステムの復元

バックアップを実行すると、ファイルやディレクトリは、それらが含まれるファイルシステムからの相対的な位置に保存されます。ファイルとディレクトリを復元するときは、`ufsrestore` コマンドが現在の作業ディレクトリにファイル階層を作成し直します。

たとえば、`/export/doc/books` ディレクトリ (`/export` はファイルシステム) からバックアップされたファイルは、`/export` からの相対的な位置に保存されます。つまり、`books` ディレクトリ内の `book` ファイルは、テープ上で `./doc/books/book1.` として保存されます。あとで、`./doc/books/book1` ファイルを `/var/tmp` ディレクトリに復元する場合、そのファイルは `/var/tmp/doc/books/book1` に復元されます。

個別のファイルやディレクトリを復元するときには、`/var/tmp` などの一時的な場所に復元する必要があります。ファイルを確認したら、それを適当な位置に移動させてもかまいません。ただし、個別のファイルやディレクトリはそれぞれ元の位置に復元できます。その場合には、新しいファイルをバックアップテープからの古いバージョンで上書きしないかどうか確かめてください。

ほかのユーザーとの重複を防ぐために、`/var/tmp/restore` ファイルなどのサブディレクトリを作成し、そのディレクトリに移動して、ファイルを復元することをお勧めします。

階層を復元する場合、ファイルを配置するファイルシステム内の一時ディレクトリにファイルを復元する必要があります。復元後に、`mv` コマンドを使用して階層全体を適切な場所に移動します。

---

注—一時的な場合でも、`/tmp` ディレクトリにファイルを復元してはいけません。`/tmp` ディレクトリは通常、TMPFS ファイルシステムとしてマウントされます。TMPFS は ACL などの UFS ファイルシステム属性をサポートしていません。

---

## ▼ 使用するテープを決定する方法

- 1 復元するファイルを最後に変更したのはいつごろだったかをユーザーに尋ねます。
- 2 バックアップ計画を参照し、該当するファイルまたはファイルシステムを含む最後のバックアップの日付を調べます。  
ファイルの最新バージョンを検索するには、特にユーザーの要求がない限り、増分バックアップファイルを最高のダンプレベルから最低のダンプレベルへ、最新の日付からもっとも古い日付へ逆方向に調べます。

- 3 オンラインアーカイブファイルを保持している場合、使用に適したメディアを判別します。

```
ufsrestore ta archive-name ./path/filename ./path/filename
```

`t`                    テープ上の各ファイルを表示します。

`a`                    内容一覧をテープからではなく、オンラインアーカイブファイルから読み取ります。

`archive-name`       オンラインアーカイブファイル名を指定します。

`./path/filename`    オンラインアーカイブ上で検索するファイル名を指定します。コマンドが成功した場合は、`ufsrestore` は `i` ノード番号とファイル名を出力します。成功しなかった場合、`ufsrestore` はエラーメッセージを出力します。

詳細は、[ufsrestore\(1M\)](#) を参照してください。

- 4 復元するファイルを含むメディアをドライブに挿入し、適切なメディアであることを確認します。

```
ufsrestore tf /dev/rmt/n ./path/filename ./path/filename
```

各ファイル名に必ず完全パス名を使用してください。ファイルがバックアップに入っていれば、その名前とiノード番号が表示されます。ファイルがバックアップに入っていない場合、ファイルがそのボリュームに入っていないことを示すメッセージが表示されます。

- 5 同じテープに複数のバックアップファイルが入っている場合は、使用するバックアップファイルの位置までテープを移動します。

```
ufsrestore xfs /dev/rmt/n tape-number
```

### 例 19-7 使用するテープを決定する

次の例は、オンラインアーカイブに /etc/passwd ファイルが存在するかどうかを確認する方法を示します。

```
ufsrestore ta /var/tmp/root.archive ./etc/passwd
```

次の例は、バックアップテープに /etc/passwd ファイルが存在するかどうかを確認する方法を示します。

```
ufsrestore tf /dev/rmt/0 ./etc/passwd
```

## ▼ 対話式で **UFS** ファイルを復元する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 (オプション) 安全性のため、書き込み保護を設定します。
- 3 ボリューム 1 のテープをテープドライブに挿入します。
- 4 ファイルを一時的に復元するためのディレクトリに移動します。

```
cd /var/tmp
```

- 5 対話式で復元を開始します。

```
ufsrestore if /dev/rmt/n
```

情報を伝えるためのメッセージと `ufsrestore>` プロンプトが表示されます。

- 6 復元するファイルのリストを作成します。

- a. ディレクトリの内容を表示します。

```
ufsrestore> ls [directory-name]
```



- b. ディレクトリを変更します。

```
ufsrestore> cd directory-name
```

- c. 復元するファイルとディレクトリのリストを作成します。

```
ufsrestore> add filenames
```

- d. (省略可能) 必要に応じ、復元するファイルのリストからディレクトリまたはファイルを削除します。

```
ufsrestore> delete filename
```

- 7 (省略可能) 復元処理中にファイル名を表示します。

```
ufsrestore> verbose
```

- 8 ファイルを復元します。

```
ufsrestore> extract
```

どのボリューム番号を使用するかを指定するプロンプトが表示されます。

- 9 ボリューム番号を入力して、**Return** キーを押します。ボリュームが1つしかない場合には、**1**を入力して**Return** キーを押します。

```
Specify next volume #: 1
```

リスト内のファイルとディレクトリが抽出され、現在の作業ディレクトリに復元されます。

- 10 現在の作業ディレクトリのモードを変更しない場合は、**set owner/mode** プロンプトが表示されたときに**n**を入力します。

```
set owner/mode for '.?' [yn] n
```

ufsrestore コマンドによる最後の整理が完了すると、ufsrestore のプロンプトが表示されます。

- 11 **ufsrestore** プログラムを終了します。

```
ufsrestore> quit
```

シェルプロンプトが表示されます。

- 12 復元されたファイルを確認します。

- a. 復元されたファイルとディレクトリを表示します。

```
ls -l
```

ファイルとディレクトリのリストが表示されます。

- b. リストを検査して、リストに指定したすべてのファイルとディレクトリが復元されていることを確認します。

### 13 ファイルを適切なディレクトリに移動します。

#### 例 19-8 対話式で UFS ファイルを復元する

次の例では、バックアップテープから `/etc/passwd` ファイルと `/etc/shadow` ファイルを抽出する方法について説明します。

```
cd /var/tmp
ufsrestore if /dev/rmt/0
ufsrestore> ls
.:
.:
.sunw/ export/ net/ /sbin/ usr/
Sources/ etools/ opt/ /scde/ var/
b/ home/ pttools/ set/ vol/
bin kernel/ pkg/ share/
dev/ lib/ platform/ shared/
devices/ lost+found/ proc/ src/
etc/ mnt/ rtools/ tmp/
ufsrestore> cd etc
ufsrestore> add passwd shadow
ufsrestore> verbose
verbose mode on
ufsrestore> extract
Extract requested files
You have not read any volumes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 1
extract file ./etc/shadow
extract file ./etc/passwd
Add links
Set directory mode, owner, and times.
set owner/mode for './?' [yn] n
ufsrestore> quit
cd etc
mv passwd /etc
mv shadow /etc
ls -l /etc
```

## ▼ 特定の UFS ファイルを復元する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 (オプション) 安全性のため、書き込み保護を設定します。
- 3 ボリューム 1 のテープをテープドライブに挿入します。
- 4 ファイルを一時的に復元するためのディレクトリに移動します。

```
cd /var/tmp
```

## 5 ファイルを復元します。

```
ufsrestore xvf /dev/rmt/n filename
```

x *filename* 引数内に指定されたファイルまたはディレクトリをコピーするように ufsrestore コマンドに指定します。

v 復元処理中にファイル名を表示します。

f /dev/rmt/n テープデバイス名を識別します。

*filename* 1つまたは複数のファイル名またはディレクトリ名を指定します。複数の場合は、空白で区切って指定します。次に例を示します。 ./export/home/user1/mail ./export/home/user2/mail .

## 6 ファイルが入っているボリューム番号を入力します。Return を押します。

```
Specify next volume #: 1
```

ファイルは現在の作業ディレクトリに復元されます。

## 7 現在のディレクトリのモードを変更しない場合は、set owner/mode プロンプトが表示されたときに n と入力して Return キーを押します。

```
set owner/mode for './'? [yn] n
```

## 8 復元されたファイルを確認します。

## a. 復元されたファイルとディレクトリを表示します。

```
ls -l
```

ファイルとディレクトリのリストが表示されます。

## b. リストを検査して、リストに指定したすべてのファイルとディレクトリが復元されていることを確認します。

## 9 ファイルを適切なディレクトリに移動します。

## 例 19-9 特定のファイルを復元する

次の例では、passwd ファイルと shadow ファイルを /var/tmp ディレクトリに非対話式で復元する方法を説明します。

```
cd /var/tmp
ufsrestore xvf /dev/rmt/0 ./etc/passwd ./etc/shadow
Verify volume and initialize maps
Media block size is 126
Dump date: Wed Jul 14 13:36:17 2010
Dumped from: the epoch
Level 0 dump of / on starbug:/dev/dsk/c0t0d0s0
Label: none
```

```
Extract directories from tape
Initialize symbol table.
Extract requested files
You have not read any volumes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 1
extract file ./etc/passwd
extract file ./etc/shadow
Add links
Set directory mode, owner, and times.
Specify next volume #:1
extract file ./etc/passwd
extract file ./etc/shadow
Add links
Set directory mode, owner, and times.
set owner/mode for './?' [yn] n
cd etc
mv passwd /etc
mv shadow /etc
ls -l /etc
```

#### 例 19-10 リモートのテープデバイスからファイルを復元する

ufsrestore コマンドを使用するときにテープデバイス名の前に *remote-host:* を追加することにより、ファイルをリモートドライブから復元できます。

次の例では、システム *venus* 上のリモートテープドライブ */dev/rmt/0* を使用してファイルを復元します。

```
ufsrestore xf venus:/dev/rmt/0 ./etc/hosts
```

## ▼ UFS ファイルシステム全体を復元する方法

ときには、全面的に復元しなければならないほどファイルシステムが破壊される場合があります。一般的な例として、ディスク障害が発生した場合には、ファイルシステムを全面的に復元する必要があります。この場合、ハードウェアを交換してからソフトウェアを復元しなければならないこともあります。ディスクの交換方法については、[217 ページの「SPARC: UFS ファイルシステム用のディスクの設定 \(タスクマップ\)」](#)または [237 ページの「x86: UFS ファイルシステム用のディスクの設定 \(タスクマップ\)」](#)を参照してください。

*/export/home* などのファイルシステムを全面的に復元するには、時間がかかります。ファイルシステムを一貫性のある方法でバックアップしていれば、最後の増分バックアップ時の状態に復元できます。

---

注- ルート (/) または /usr ファイルシステムの復元には、この手順は使用できません。これらのファイルシステムを復元する方法については、[416 ページの「UFS ルート \(/\) および /usr ファイルシステムを復元する方法」](#)を参照してください。

---

- 1 スーパーユーザーになるか、同等の役割を引き受けます。

- 2 必要に応じ、ファイルシステムをマウント解除します。

```
umount /dev/rdisk/device-name
```

または

```
umount /file-system
```

- 3 ファイルシステムを新規作成します。

```
newfs /dev/rdisk/device-name
```

raw デバイス上に新しいファイルシステムを構築するかどうかを確認するプロンプトが表示されます。意図しないファイルシステムを間違って損失してしまわないように、*device-name* が正しいことを確認します。

詳細は、[newfs\(1M\)](#)を参照してください。

- 4 新しいファイルシステムを作成しなければならないかどうかを確認します。

```
newfs: construct a new file system /dev/rdisk/cwtxdysz:(y/n)? y
```

新しいファイルシステムが作成されます。

- 5 新しいファイルシステムを一時的なマウントポイントにマウントします。

```
mount /dev/dsk/device-name /mnt
```

- 6 マウントポイントのディレクトリに移動します。

```
cd /mnt
```

- 7 (オプション) 安全性のため、書き込み保護を設定します。

- 8 レベル 0 テープの第 1 ボリュームをテープドライブに挿入します。

- 9 ファイルを復元します。

```
ufsrestore rvf /dev/rmt/n
```

ダンプレベル 0 のバックアップが復元されます。バックアップの実行に複数のテープが必要な場合は、番号の順にテープをロードするようにプロンプトが表示されます。

- 10 テープを外し、次のレベルのテープをドライブにロードします。  
テープの復元は必ずダンプレベル 0 から始め、一番高いダンプレベルまで続けてください。
- 11 ダンプレベルごとに、一番低いレベルから一番高いレベルまで、[手順 8](#) から [手順 10](#) までの操作を繰り返します。
- 12 ファイルシステムが復元されたことを確認します。  
`# ls`
- 13 **restoresymtable** ファイルを削除します。  
`# rm restoresymtable`  
復元のチェックポイントのために `ufsrestore` コマンドが作成し、使用した `restoresymtable` ファイルを削除します。
- 14 別のディレクトリに移動します。  
`# cd /`
- 15 新しく復元されたファイルシステムをマウント解除します。  
`# umount /mnt`
- 16 最後のテープを取り出し、書き込み保護されていない新しいテープをテープドライブに挿入します。
- 17 新しく復元されたファイルシステムのレベル 0 のバックアップを作成します。  
`# ufsdump 0ucf /dev/rmt/n /dev/rdisk/device-name`  
レベル 0 のバックアップが実行されます。`ufsrestore` コマンドはファイルの位置を移動し、i ノード割り当てを変更するので、新しく作成されたファイルシステムの完全バックアップは、必ずすぐに作成するようにしてください。
- 18 復元されたファイルシステムをマウントします。  
`# mount /dev/dsk/device-name mount-point`  
復元されたファイルシステムがマウントされ、使用できるようになります。
- 19 復元およびマウントされたファイルシステムが使用できることを確認します。  
`# ls mount-point`

#### 例 19-11 UFS ファイルシステム全体を復元する

次の例では、`/export/home` ファイルシステムを復元する方法を示します。

```

newfs /dev/rdisk/c0t0d0s7
newfs: /dev/rdisk/c0t0d0s7 last mounted as /export/home
newfs: construct a new file system /dev/rdisk/c0t0d0s7: (y/n)? y
/dev/rdisk/c0t0d0s7: 143358064 sectors in 23334 cylinders of 48 tracks, 128 sectors
69999.1MB in 1459 cyl groups (16 c/g, 48.00MB/g, 5824 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 98464, 196896, 295328, 393760, 492192, 590624, 689056, 787488, 885920,
Initializing cylinder groups:
.....
super-block backups for last 10 cylinder groups at:
 142447776, 142546208, 142644640, 142743072, 142841504, 142939936, 143038368,
 143136800, 143235232, 143333664
mount /dev/dsk/c0t0d0s7 /mnt
cd /mnt
ufsrestore rvf /dev/rmt/0
Verify volume and initialize maps
Media block size is 126
Dump date: Wed Jul 14 12:45:33 2010
Dumped from: the epoch
Level 0 dump of /export/home on starbug:/dev/dsk/c0t0d0s7
Label: none
Begin level 0 restore
Initialize symbol table.
Extract directories from tape
Calculate extraction list.
Warning: ./lost+found: File exists
Make node ./rimmer
Make node ./rimmer/wdir
Make node ./lister
Make node ./pmorph
Make node ./inquisitor
Make node ./kryten
Make node ./kryten/letters
Make node ./kryten/reports
Extract new leaves.
Check pointing the restore
extract file ./rimmer/words
extract file ./rimmer/words1
extract file ./rimmer/words2
extract file ./rimmer/words3
extract file ./rimmer/wdir/words
extract file ./rimmer/wdir/words1
extract file ./rimmer/wdir/words2
extract file ./rimmer/wdir/words3
.
.
.
Add links
Set directory mode, owner, and times.
Check the symbol table.
Check pointing the restore
rm restoresymtable
cd /
umount /mnt
ufsdump 0ucf /dev/rmt/0 /export/home
.
.
.

```

```
mount /dev/dsk/c0t0d0s7 /export/home
ls /export/home
```

## ▼ UFS ルート (/) および /usr ファイルシステムを復元する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 新しいシステムディスクを、ルート (/) と /usr ファイルシステムが復元されるシステムに追加します。  
システムディスクを追加する方法の詳細については、[219 ページの「SPARC: UFS ルートファイルシステム用のディスクを設定する方法」](#)または [239 ページの「x86: UFS ルートファイルシステム用のディスクを設定する方法」](#)を参照してください。
- 3 新しいファイルシステムを一時的なマウントポイントにマウントします。  

```
mount /dev/dsk/device-name /mnt
```
- 4 /mnt ディレクトリに移動します。  

```
cd /mnt
```
- 5 (オプション) 安全性のため、書き込み保護を設定します。
- 6 テープデバイスのリンクを作成します。  

```
tapes
```
- 7 ルート (/) ファイルシステムを復元します。  

```
ufsrestore rvf /dev/rmt/n
```

ダンプレベル 0 のテープが復元されます。
- 8 テープを外し、次のレベルのテープをドライブにロードします。  
テープの復元は必ずダンプレベル 0 から始め、もっとも低いダンプレベルからもっとも高いダンプレベルまで続けてください。
- 9 必要に応じ、復元を続行します。  

```
ufsrestore rvf /dev/rmt/n
```

次のレベルのテープが復元されます。
- 10 テープを追加するたびに、[手順 8](#) と [手順 9](#) を繰り返します。



- 11 ファイルシステムが復元されたことを確認します。

```
ls
```

- 12 **restoresymtable** ファイルを削除します。

```
rm restoresymtable
```

復元のチェックポイントのために **ufsrestore** コマンドが作成し、使用した **restoresymtable** ファイルを削除します。

- 13 ルート (/) ディレクトリに変更します。

```
cd /
```

- 14 新しく作成されたファイルシステムをマウント解除します。

```
umount /mnt
```

- 15 新しいファイルシステムを検査します。

```
fsck /dev/rdisk/device-name
```

復元されたファイルシステムの整合性が検査されます。

- 16 ルートパーティションにブートブロックを作成します。

SPARC システムの場合:

```
installboot /usr/platform/'uname-i'/lib/fs/ufs/bootblk
/dev/rdisk/device-name
```

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

SPARC システム上での **installboot** コマンドの使用例については、[例 19-12](#) を参照してください。

x86 システムの場合:

```
/sbin/installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/cwtxdysz
```

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

x86 システム上での **installgrub** コマンドの使用例については、[例 19-13](#) を参照してください。

- 17 新しいテープをテープドライブに挿入します。

- 18 新しいファイルシステムのバックアップを作成します。

```
ufsdump 0uf /dev/rmt/n /dev/rdisk/device-name
```

ダンプレベル 0 のバックアップが実行されます。**ufsrestore** コマンドはファイルの位置を移動し、i ノード割り当てを変更するので、新しく作成されたファイルシステムの完全バックアップは、必ずすぐに作成するようにしてください。

19 必要に応じ、**/usr** ファイルシステムについて手順 5 から 16 を繰り返します。

20 システムをリブートします。

```
init 6
```

システムがリブートされます。

#### 例 19-12 SPARC:UFS ルート (/) ファイルシステムを復元する

次の例は、SPARC システム上でルート (/) ファイルシステムを復元する方法を示します。この例では、システムはローカルの Solaris DVD またはネットワークからブートするものとします。

```
mount /dev/dsk/c0t3d0s0 /mnt
cd /mnt
tapes
ufsrestore rvf /dev/rmt/0
ls
rm restoresymtable
cd /
umount /mnt
fsck /dev/rdsk/c0t3d0s0
installboot /usr/platform/sun4u/lib/fs/ufs/bootblk
/dev/rdsk/c0t3d0s0
ufsdump 0uf /dev/rmt/0 /dev/rdsk/c0t3d0s0
init 6
```

#### 例 19-13 x86:UFS ルート (/) ファイルシステムを復元する

次の例は、x86 システム上でルート (/) ファイルシステムを復元する方法を示します。この例では、GRUB フェイルセーフブートセッション、ローカルの Solaris DVD、またはネットワークからシステムがブートされていることを前提としています。

```
mount /dev/dsk/c0t3d0s0 /mnt
cd /mnt
tapes
ufsrestore rvf /dev/rmt/0
ls
rm restoresymtable
cd /
umount /mnt
fsck /dev/rdsk/c0t3d0s0
/sbin/installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c0t3d0s0
stage1 written to partition 0 sector 0 (abs 2016)
stage2 written to to partition 0, 227 sectors starting 50 (abs 2066)
ufsdump 0uf /dev/rmt/0 /dev/rdsk/c0t3d0s0
init 6
```

## UFS スナップショットの使用 (手順)

---

この章では、UFS スナップショットの作成およびバックアップ方法について説明します。

UFS スナップショットの作成に関連した手順については、[419 ページの「UFS スナップショットの使用 \(タスクマップ\)」](#)を参照してください。

バックアップの実行の概要については、[379 ページの「UFS ファイルシステムのバックアップと復元の概要」](#)を参照してください。

### UFS スナップショットの使用 (タスクマップ)

タスク	説明	参照先
1. UFS スナップショットを作成します。	<code>fssnap</code> コマンドを使用して、ファイルシステムの読み取り専用コピーを作成します。	<a href="#">424 ページの「UFS スナップショットを作成する方法」</a>
2. UFS スナップショットの情報を表示します。	UFS スナップショット情報 (raw スナップショットデバイスなど)を確認します。	<a href="#">425 ページの「UFS スナップショットの情報を表示する方法」</a>
3. (オプション) UFS スナップショットを削除します。	バックアップ済みまたは不要になったスナップショットを削除します。	<a href="#">426 ページの「UFS スナップショットを削除する方法」</a>
4. (オプション) UFS スナップショットのバックアップを作成します。	次の方法のいずれかを選択します。	

タスク	説明	参照先
	<p><code>ufsdump</code> コマンドを使用して、UFS スナップショットの完全バックアップを作成します。</p> <p><code>ufsdump</code> コマンドを使用して、UFS スナップショットの増分バックアップを作成します。</p> <p><code>tar</code> コマンドを使用して、UFS スナップショットのバックアップを作成します。</p>	<p>427 ページの「UFS スナップショットの完全バックアップの作成方法 (<code>ufsdump</code>)」</p> <p>428 ページの「UFS スナップショットの増分バックアップの作成方法 (<code>ufsdump</code>)」</p> <p>428 ページの「UFS スナップショットのバックアップ方法 (<code>tar</code>)」</p>
5. (オプション) UFS スナップショットからデータを復元します。	<code>ufsrestore</code> コマンドを使用して、データを復元するのと同じ方法で、UFS スナップショットを復元します。	412 ページの「UFS ファイルシステム全体を復元する方法」

## UFS スナップショットの概要

`fssnap` コマンドを使って、ファイルシステムをマウントした状態で、そのバックアップを作成できます。作成されるのは、ファイルシステムの読み取り専用スナップショットです。「スナップショット」は、バックアップ操作のためのファイルシステムの一時的イメージです。

`fssnap` コマンドを実行すると、1つの仮想デバイスと1つのバックキングストアファイルが作成されます。ユーザーは、既存の Solaris バックアップコマンドを使用して、実際のデバイスのように動作し実際のデバイスのように見える「仮想デバイス」をバックアップできます。「バックキングストアファイル」は、スナップショット作成後に変更されたデータのコピーを含むビットマップファイルです。

バックキングストアファイルを指定する際には、次の注意点に留意してください。

- バックキングストアファイルの格納先は、対象ファイルシステムのデータを格納できるだけの空き容量を備えている必要があります。バックキングストアファイルのサイズは、対象ファイルシステム上における活動量に応じて変わります。
- バックキングストアファイルの格納場所は、スナップショット作成対象のファイルシステム上であってはなりません。
- バックキングストアファイルの格納場所としては、別の UFS ファイルシステムや NFS ファイルシステムなど、任意の種類ファイルシステムを選択可能です。
- 512G バイトを超える UFS ファイルシステムのスナップショットを作成する場合、複数のバックキングストアファイルが作成されます。

- バッキングストアファイルは空白ファイルです。ls コマンドで表示される空白ファイルの論理サイズは、du コマンドで表示されるその割り当て容量と同一ではありません。

512G バイトを超える UFS ファイルシステムのスナップショットを作成する方法の詳細は、[423 ページの「マルチテラバイト UFS のスナップショットの作成」](#)を参照してください。

## なぜ UFS スナップショットを使用するか

UFS スナップショット機能により、ファイルシステムのバックアップがより利用しやすく、簡単になりました。この機能を利用すれば、ファイルシステムをマウントした状態でバックアップを作成できます。また、このときマルチユーザーモードを保持できます。さらに、長期保存が必要な場合は、tar や cpio コマンドを使って、UFS スナップショットをテープにバックアップすることもできます。ufsdump コマンドでバックアップを実行する場合は、シングルユーザーモードにして、バックアップが完了するまでファイルシステムを非アクティブな状態にする必要があります。

fssnap コマンドは、企業レベルでないシステム管理者に、大規模な記憶容量を必要とせず、Sun StorEdge Instant Image のような企業レベルのツールのパワーを提供します。

UFS スナップショットの機能は、Instant Image 製品の機能に似ています。UFS スナップショットは大規模なファイルシステムをコピーできますが、企業レベルのシステムには Instant Image の方が適しています。UFS スナップショットは、小さめのシステムに適しています。Instant Image は、取り込まれるファイルシステム全体のサイズに等しい容量を割り当てます。ただし、UFS スナップショットが作成するバッキングストアファイルは、必要なディスク容量だけを占有します。

次の表は、UFS スナップショットと Instant Image との特徴的な違いを示します。

UFS スナップショット	Sun StorEdge Instant Image
バッキングストアファイルのサイズは、スナップショットがとられた後のデータの変更量によります	バッキングストアファイルのサイズは、コピーされるファイルシステム全体のサイズと同じです
システムのリブート後は保持されません	システムのリブート後も保持されます
UFS ファイルシステムで動作します	ルート (/) または /usr ファイルシステムでは使用できません
Oracle Solaris リリースで使用可能	Sun StorEdge 製品の一部です

## UFS スナップショットのパフォーマンス上の問題

UFS スナップショットの初回作成時に、ファイルシステムのユーザーが短い一時停止に気づく場合があります。一時停止の時間は、取り込まれるファイルシステムのサイズに応じて増加します。スナップショットがアクティブな間、ファイルシステムの書き込み中、パフォーマンスに若干の影響が出る可能性があります。ただし、ファイルシステムの読み込み中には、このような影響はありません。

## UFS スナップショットの作成と削除

`fssnap` コマンドを使用して UFS スナップショットを作成する場合、バックキングストアファイルがどれだけのディスク容量を消費するかを監視してください。バックキングストアファイルは、はじめは容量をまったく使用しませんが、その後特によく使用されているシステムにおいて急速に拡大します。バックキングストアファイルが拡大するのに十分な容量を必ず確保してください。または `-o maxsize=n [k,m,g]` オプション (`n [k,m,g]` はバックキングストアファイル最大限のサイズ) でそのサイズを制限してください。



注意-バックキングストアファイルに容量が不足する場合、スナップショットが削除されてしまうことがあります。バックアップが失敗します。スナップショットのエラーの可能性を調べるため、`/var/adm/messages` ファイルを検査してください。

バックキングストアパスのディレクトリを指定して、バックキングストアファイルは指定のディレクトリに作成することもできます。たとえば、`/var/tmp` を指定した場合、次のバックキングストアファイルが作成されます。

```
/var/tmp/snapshot0
```

`/export/home`、`/usr` などのファイルシステムを個別に作成する代わりに大容量のルート (`/`) ファイルシステムを1つ作成した場合、これらのファイルシステムのスナップショットを個別に作成することはできません。たとえば、このシステムは、`/usr` 用の個別のファイルシステムを持ちません。次の **Mounted on** カラムの下を参照してください。

```
df -k /usr
Filesystem kbytes used avail capacity Mounted on
/dev/dsk/c0t0d0s0 3618177 2190002 1391994 62% /
```

`/usr` ファイルシステムのスナップショットを作成しようとすると、次のようなメッセージが表示されます。

```
fssnap -F ufs -o bs=/snaps/usr.back.file /usr
snapshot error: Invalid backing file path
```

ここで、スナップショットを作成するファイルシステム(この例の場合、`/usr` ファイルシステム)上にバックキングストアファイルを作成することはできません。

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

## マルチテラバイト UFS のスナップショットの作成

マルチテラバイト UFS のスナップショットの作成は、より小さいサイズの UFS ファイルシステムに対するスナップショットの作成とほとんど同じです。唯一の違いは、ファイルシステム領域の 512G バイトごとにバックキングストアファイルが作成されるという点です。

512G バイトを超えるファイルシステムのスナップショットを作成する際には、次の注意点に留意してください。

- 複数のバックキングストアファイルが作成されます。
  - スナップショットの作成時にバックキングストアファイル名を指定した場合、その指定されたファイル名に基づく名前が後続のバックキングストアファイルに繰り返し付けられます。後続のバックキングストアファイルには同じ名前が付けられますが、「.2」、「.3」などの接尾辞がそれぞれ付加されます。
  - バックキングストアの名前は指定せず、その格納先(ディレクトリ)だけを指定した場合、複数のバックキングストアファイル名が、「.2」、「.3」などの接尾辞付きで繰り返し作成されます。
- 複数のバックキングストアファイルが作成された場合でも、`fssnap -i` コマンドの実行時に表示されるのは、最初のバックキングストアファイルの名前だけです。これに対し、バックキングストア長として表示されるのは、スナップショットのすべてのバックキングストアファイルの合計サイズです。

---

注-バックキングストアファイルは空白ファイルです。`ls` コマンドで表示される空白ファイルの論理サイズは、`du` コマンドで表示されるその割り当て容量と同一ではありません。

---

- スナップショットの作成時に `unLink` オプションを指定しなかった場合、スナップショットのバックアップ完了後や単にスナップショットを削除するとき、バックキングストアファイルを手動で削除する必要があります。

512G バイトを超えるファイルシステムのスナップショットの作成例については、[例 20-2](#)を参照してください。

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

## ▼ UFS スナップショットを作成する方法

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 ファイルシステムに、バックイングストアファイル用の十分なディスク容量が存在することを確認してください。

```
df -k
```

- 3 同じ場所に同じ名前の既存のバックイングストアファイルがないことを確認します。

```
ls /backing-store-file
```

- 4 UFS スナップショットを作成します。

```
fssnap -F ufs -o bs=/backing-store-file /file-system
```

---

注-バックイングストアファイルは、UFS スナップショットを使用して取り込むファイルシステムとは異なるファイルシステムに存在する必要があります。

---

- 5 スナップショットが作成されたことを確認します。

```
/usr/lib/fs/ufs/fssnap -i /file-system
```

### 例 20-1 UFS スナップショットを作成する

次の例は、`/usr` ファイルシステムのスナップショットを作成する方法を示します。バックイングストアファイルは `/scratch/usr.back.file` です。仮想デバイスは `/dev/fssnap/1` です。

```
fssnap -F ufs -o bs=/scratch/usr.back.file /usr
/dev/fssnap/1
```

次の例は、バックイングストアファイルを 500M バイトに制限する方法を示します。

```
fssnap -F ufs -o maxsize=500m,bs=/scratch/usr.back.file /usr
/dev/fssnap/1
```

### 例 20-2 マルチテラバイト UFS のスナップショットの作成

次の例は、1.6T バイトの UFS ファイルシステムのスナップショットの作成方法を示しています。

```
fssnap -F ufs -o bs=/var/tmp /datab
/dev/fssnap/2
/usr/lib/fs/ufs/fssnap -i /datab
Snapshot number : 2
Block Device : /dev/fssnap/2
Raw Device : /dev/rfssnap/2
```



```

Mount point : /datab
Device state : idle
Backing store path : /var/tmp/snapshot3
Backing store size : 0 KB
Maximum backing store size : Unlimited
Snapshot create time : Mon Jul 12 10:37:50 2010
Copy-on-write granularity : 32 KB

```

## ▼ UFS スナップショットの情報を表示する方法

`fssnap -i` オプションを使用すると、システムの現在のスナップショットを表示できます。1つのファイルシステムを指定する場合、そのファイルシステムのスナップショットについての詳細な情報が表示されます。特定のファイルシステムを指定しない場合は、現在のUFSスナップショットすべておよび対応する仮想デバイスの情報が表示されます。

---

注-次の例に示すように、拡張スナップショット情報を表示する場合は、UFS ファイルシステム固有の `fssnap` コマンドを使用してください。

---

- 1 スーパーユーザーになるか、同等の役割を引き受けます。

- 2 現在のすべてのスナップショットをリスト表示します。

例:

```

/usr/lib/fs/ufs/fssnap -i
Snapshot number : 0
Block Device : /dev/fssnap/0
Raw Device : /dev/rfssnap/0
Mount point : /export/home
Device state : idle
Backing store path : /var/tmp/snapshot0
Backing store size : 0 KB
Maximum backing store size : Unlimited
Snapshot create time : Mon Jul 12 10:37:50 2010
Copy-on-write granularity : 32 KB

```

- 3 特定のスナップショットについての詳細な情報を表示します。

例:

```

/usr/lib/fs/ufs/fssnap -i /export
Snapshot number : 1
Block Device : /dev/fssnap/1
Raw Device : /dev/rfssnap/1
Mount point : /export
Device state : idle
Backing store path : /var/tmp/snapshot1
Backing store size : 0 KB
Maximum backing store size : Unlimited
Snapshot create time : Mon Jul 12 10:37:50 2010
Copy-on-write granularity : 32 KB

```

## UFS スナップショットの削除

UFS スナップショットを作成する際、バックキングストアファイルがリンクされないように指定できます。リンクされていないバックキングストアファイルは、スナップショットを削除した後で削除されます。UFS スナップショットを作成する際に `-o unlink` オプションを指定しない場合は、あとでバックキングストアファイルを手動で削除する必要があります。

バックキングストアファイルは、バックキングストアファイルの削除に `-o unlink` オプションを使用した場合はスナップショットが削除されるまで、またはバックキングストアファイルを手動で削除するまで、ディスク容量を使用します。

### ▼ UFS スナップショットを削除する方法

スナップショットは、システムをリブートするか、あるいは `fssnap -d` コマンドを使用して削除できます。このコマンドを使用するときには、UFS スナップショットを含むファイルシステムのパスを指定する必要があります。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 削除するスナップショットを特定します。  

```
/usr/lib/fs/ufs/fssnap -i
```
- 3 スナップショットを削除します。  

```
fssnap -d /file-system
Deleted snapshot 1.
```
- 4 スナップショットの作成時に `-o unlink` オプションを使用しなかった場合は、そのバックキングストアファイルを手動で削除してください。  

```
rm /file-system/backing-store-file
```

#### 例 20-3 UFS スナップショットの削除

次の例では、`-o unlink` オプションを使用しなかった場合にスナップショットを削除する方法を示します。

```
fssnap -i
0 /export/home
1 /export
fssnap -d /export
Deleted snapshot 1.
rm /var/tmp/snapshot1
```

# UFS スナップショットのバックアップ

UFS スナップショットの完全バックアップまたは増分バックアップを作成できます。UFS スナップショットのバックアップ作成に、標準の Solaris バックアップコマンドを使用できます。

UFS スナップショットを含む仮想デバイスは、標準の読み取り専用デバイスとして動作します。つまり、仮想デバイスを、ファイルシステムのデバイスをバックアップするかのようにバックアップできます。

`ufsdump` コマンドを使用して UFS スナップショットをバックアップする場合、バックアップ時にスナップショットの名前を指定できます。詳細は、次の手順を参照してください。

## ▼ UFS スナップショットの完全バックアップの作成方法 (`ufsdump`)

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 バックアップする UFS スナップショットを確認します。

```
/usr/lib/fs/ufs/fssnap -i /file-system
```

例:

```
/usr/lib/fs/ufs/fssnap -i /usr
Snapshot number : 1
Block Device : /dev/fssnap/1
Raw Device : /dev/rfssnap/1
Mount point : /usr
Device state : idle
Backing store path : /var/tmp/snapshot2
Backing store size : 544 KB
Maximum backing store size : Unlimited
Snapshot create time : Mon Jul 12 10:37:50 2010
Copy-on-write granularity : 32 KB
```

- 3 UFS スナップショットをバックアップします。

```
ufsdump 0ucf /dev/rmt/0 /snapshot-name
```

例:

```
ufsdump 0ucf /dev/rmt/0 /dev/rfssnap/1
```

- 4 スナップショットのバックアップが作成されたことを確認します。

例:

```
ufsrestore tf /dev/rmt/0
```

## ▼ UFS スナップショットの増分バックアップの作成方法 (ufsdump)

UFS スナップショットの増分バックアップでは、最後のスナップショット以降に変更のあったファイルだけがバックアップされます。ufsdump コマンドと **N** オプションを組み合わせて使用します。このオプションは、増分ダンプをトラックするために /etc/dumpdates ファイルに挿入されるファイルシステムのデバイス名を指定します。

次の例では、ufsdump コマンド内で fssnap コマンドを組み込んでファイルシステムの増分バックアップを作成しています。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 UFS スナップショットの増分バックアップを作成します。

例:

```
ufsdump lufN /dev/rmt/0 /dev/rdsk/c0t1d0s0 'fssnap -F ufs -o raw,bs=
/export/scratch,unlink /dev/rdsk/c0t1d0s0'
```

上記の例では、ブロックデバイスではなく raw デバイスの名前を表示するために -o raw オプションが使用されています。このオプションの使用により、raw デバイスを必要とするコマンド (ufsdump コマンドなど) に fssnap コマンドを組み込むことが簡単になります。

- 3 スナップショットのバックアップが作成されたことを確認します。

```
ufsrestore ta /dev/rmt/0
```

## ▼ UFS スナップショットのバックアップ方法 (tar)

tar コマンドを使用してスナップショットをバックアップする場合、バックアップを行う前にスナップショットをマウントします。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
- 2 スナップショット用のマウントポイントを作成します。

例:

```
mkdir /backups/home.bkup
```

- 3 スナップショットをマウントします。

```
mount -F ufs -o ro /dev/fssnap/1 /backups/home.bkup
```

- 4 マウントスナップショットのディレクトリに移動します。

```
cd /backups/home.bkup
```

- 5 **tar** コマンドを使用して、スナップショットをバックアップします。

```
tar cvf /dev/rmt/0 .
```

## UFS スナップショットのバックアップからのデータの復元

仮想デバイスから作成されたバックアップは、基本的には、スナップショットがとられた時点でのオリジナルのファイルシステムの状態を表しています。ファイルシステムをバックアップから復元する場合は、オリジナルのファイルシステムから直接そのバックアップを作成したかのように復元します。そのような復元では、**ufsrestore** コマンドを使用します。**ufsrestore** コマンドを使用したファイルまたはファイルシステムの復元については、[404 ページの「UFS ファイルおよびファイルシステムのバックアップを復元する \(タスクマップ\)」](#)を参照してください。



## ファイルとファイルシステムのコピー (タスク)

---

この章では、各種のバックアップコマンドを使用して、ファイルとファイルシステムをディスク、テープ、フロッピーディスクにコピーする方法について説明します。

この章で説明する手順は次のとおりです。

- 433 ページの「ファイルシステム間でディレクトリをコピーする方法 (cpio)」
- 435 ページの「ファイルをテープにコピーする方法 (tar)」
- 436 ページの「テープ上のファイルのリストを表示する方法 (tar)」
- 436 ページの「テープからファイルを取り出す方法 (tar)」
- 438 ページの「pax コマンドを使用してファイルをテープにコピーする」
- 439 ページの「ディレクトリ内のすべてのファイルをテープにコピーする方法 (cpio)」
- 440 ページの「テープ上のファイルのリストを表示する方法 (cpio)」
- 441 ページの「テープからすべてのファイルを取り出す方法 (cpio)」
- 442 ページの「テープから特定のファイルを取り出す方法 (cpio)」
- 443 ページの「ファイルをリモートテープデバイスにコピーする方法 (tar と dd)」
- 444 ページの「ファイルをリモートテープデバイスから抽出する方法」

## ファイルシステムをコピーするためのコマンド

個々のファイル、ファイルシステムの一部、またはファイルシステム全体をコピーまたは移動する場合は、この章で説明する手順を使用できます。

次の表に、Oracle Solaris リリースで使用可能な各種のバックアップおよび復元コマンドを示します。エンタープライズ環境では、エンタープライズレベルのバックアップ製品の使用を検討してください。エンタープライズレベルのバックアップ製品に関する情報は、Oracle Technical Network から入手できます。

次の表では、これらのコマンドの一部の長所と短所を説明します。

表 21-1 tar、pax、cpio コマンドの長所と短所

コマンド	機能	長所	短所
tar	ファイルやディレクトリサブツリーを 1 本のテープにコピーする場合に使用します。	<ul style="list-style-type: none"> <li>■ ほとんどの UNIX オペレーティングシステムで利用できます</li> <li>■ パブリックドメインバージョンもすぐに利用できます</li> </ul>	<ul style="list-style-type: none"> <li>■ ファイルシステムの境界を認識しません</li> <li>■ 完全パス名は最大 255 文字です</li> <li>■ 複数のテープボリュームを作成する場合は使用できません</li> </ul>
pax	複数のテープボリュームを必要とするファイル、特殊ファイル、またはファイルシステムをコピーする場合に使用します。または、POSIX 準拠システムとの間でファイルをコピーする場合に使用します。	<ul style="list-style-type: none"> <li>■ POSIX 準拠システムに対する、tar コマンドや cpio コマンドよりも高い互換性</li> <li>■ 複数のベンダーサポート</li> </ul>	tar コマンドの短所を参照してください。ただし、pax は、複数のテープボリュームを作成できます。
cpio	複数のテープボリュームを必要とするファイル、特殊ファイル、またはファイルシステムをコピーする場合に使用します。または、Oracle Solaris の最新リリースを実行しているシステムから古い Solaris リリースを実行しているシステムへファイルをコピーする場合に使用します。	<ul style="list-style-type: none"> <li>■ tar コマンドよりも効率的に、データをテープに書き込みます</li> <li>■ 復元時、テープ中の不良箇所をスキップします</li> <li>■ 異なるシステムタイプ間の互換性のために、異なるヘッダー形式 (tar、ustar、crc、odc、bar など) でファイルを書き込むオプションを提供します</li> <li>■ 複数のテープボリュームを作成します</li> </ul>	コマンド構文が tar コマンドや pax コマンドよりも難解です。

次のセクションでは、これらのコマンドの使用例を紹介します。



# cpio を使用してファイルシステム間でディレクトリをコピーする

cpio (コピーインとコピーアウト) コマンドを使用して、個々のファイル、ファイルグループ、またはファイルシステム全体をコピーできます。このセクションでは、cpio コマンドを使ってファイルシステム全体をコピーする方法について説明します。

cpio コマンドは、ファイルのリストを1つの大型出力ファイルにコピーするアーカイブプログラムです。また、復元しやすいように、個々のファイルの間にヘッダーを挿入します。cpio コマンドを使用すると、ファイルシステム全体を別のスライス、別のシステム、またはテープやフロッピーディスクなどのメディアデバイスにコピーできます。

cpio コマンドは、メディアの末尾を認識し、別のボリュームを挿入するように促すプロンプトを表示するので、複数のテープやフロッピーディスクが必要なアーカイブを作成する場合は、ufsdump 以外ではもっとも効率のよいコマンドです。

cpio コマンドの使用時には、しばしば ls や find のコマンドを使用してコピーするファイルを選択し、その出力を cpio コマンドにパイプします。

## ▼ ファイルシステム間でディレクトリをコピーする方法 (cpio)

- 1 目的のディレクトリに移動します。

```
cd filesystem1
```

- 2 find コマンドと cpio コマンドを組み合わせ使用して、filesystem1 から filesystem2 へディレクトリツリーをコピーします。

```
find . -print -depth | cpio -pdm filesystem2
```

.           現在の作業ディレクトリで処理を始めます。

-print      ファイル名を出力します。

-depth      ディレクトリ階層を下降し、すべてのファイル名を出力します。

-p          ファイルのリストを作成します。

-d          必要に応じてディレクトリを作成します。

-m          ディレクトリ上で正しい変更時間を設定します。

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

指定したディレクトリ名からファイルがコピーされます。シンボリックリンクは保持されます。

また、`-u` オプションも指定できます。このオプションは、無条件にコピーを実行します。`u` オプションを指定しない場合、古いファイルが、新しいファイルで置換されません。このオプションは、ディレクトリごとコピーするときに、コピーするファイルの一部がすでにターゲットのディレクトリ中に存在する場合に便利です。

- 3 コピー先ディレクトリの内容を表示して、コピーに成功したかどうかを確認します。

```
cd filesystem2
ls
```

- 4 ソースディレクトリが不要な場合は削除します。

```
rm -rf filesystem1
```

#### 例 21-1 ファイルシステム間でディレクトリをコピーする (cpio)

```
cd /data1
find . -print -depth | cpio -pdm /data2
19013 blocks
cd /data2
ls
rm -rf /data1
```

## ファイルとファイルシステムをテープにコピーする

`tar`、`pax`、および `cpio` コマンドを使用すると、ファイルとファイルシステムをテープにコピーできます。どのコマンドを選択するかは、コピーする目的に応じて異なります。3つのコマンドはすべて `raw` デバイスを使用するので、使用する前にテープ上でファイルシステムをフォーマットまたは作成する必要はありません。

使用するテープドライブとデバイス名は、各システムのハードウェアの構成によって異なります。テープデバイス名の詳細は、[445 ページの「使用するメディアの選択」](#)を参照してください。

## tarを使用してファイルをテープにコピーする

tar コマンドを使用してファイルをテープにコピーする前に、次の情報について知っておかなければなりません。

- tar コマンドに `-c` オプションを指定してファイルをテープにコピーすると、テープに入っているすべての既存のファイルまたはテープの現存の位置以降にある既存のファイルすべてが破壊(上書き)されます。
- ファイルをコピーするとき、ファイル名の一部にファイル名置換ワイルドカード文字(`?`と`*`)を使用できます。たとえば、接尾辞 `.doc` が付いたすべてのドキュメントをコピーするには、ファイル名引数として `*.doc` と入力します。
- tar アーカイブからファイルを抽出するときには、ファイル名置換ワイルドカードは使用できません。

### ▼ ファイルをテープにコピーする方法 (tar)

- 1 コピーするファイルの存在するディレクトリに移動します。
- 2 書き込み可能なテープをテープドライブに挿入します。
- 3 ファイルをテープにコピーします。

```
$ tar cvf /dev/rmt/n filenames
```

`c`                      アーカイブの作成を指定します。

`v`                      各ファイルがアーカイブされるたびに、その名前を表示します。

`f /dev/rmt/n`        アーカイブを指定したデバイスまたはファイルに書き込むように指定します。

`filenames`            コピーするファイルとディレクトリを指定します。ファイルが複数の場合は、各ファイルをスペースで区切ります。

指定した名前のファイルがテープにコピーされ、テープ上の既存のファイルが上書きされます。

- 4 テープをドライブから取り出します。ファイル名をテープのラベルに記入します。
- 5 コピーしたファイルがテープ上に存在することを確認します。

```
$ tar tvf /dev/rmt/n
```

tar テープ上のファイルを表示する方法については、[436 ページの「テープ上のファイルのリストを表示する方法 \(tar\)」](#)を参照してください。

## 例 21-2 ファイルをテープにコピーする (tar)

次の例では、3つのファイルをテープドライブ0のテープにコピーします。

```
$ cd /export/home/kryten
$ ls reports
reportA reportB reportC
$ tar cvf /dev/rmt/0 reports
a reports/ 0 tape blocks
a reports/reportA 59 tape blocks
a reports/reportB 61 tape blocks
a reports/reportC 63 tape blocks
$ tar tvf /dev/rmt/0
```

## ▼ テープ上のファイルのリストを表示する方法 (tar)

- 1 テープをテープドライブに挿入します。
- 2 テープの内容を表示します。

```
$ tar tvf /dev/rmt/n
```

t                      テープ上のファイルのリストが表示されます。

v                      t オプションと一緒に使用すると、テープ上のファイルに関する詳細情報が表示されます。

f /dev/rmt/n          テープデバイスを示します。

## 例 21-3 テープ上のファイルのリストを表示する (tar)

次の例では、ドライブ0のテープに含まれるファイルのリストを表示します。

```
$ tar tvf /dev/rmt/0
drwxr-xr-x 0/0 0 Jul 14 13:50 2010 reports/
-r--r--r-- 0/0 206663 Jul 14 13:50 2010 reports/reportC
-r--r--r-- 0/0 206663 Jul 14 13:50 2010 reports/reportB
-r--r--r-- 0/0 206663 Jul 14 13:50 2010 reports/reportA
```

## ▼ テープからファイルを取り出す方法 (tar)

- 1 ファイルを置きたいディレクトリへ移動します。
- 2 テープをテープドライブに挿入します。

### 3 テープからファイルを取り出します。

```
$ tar xvf /dev/rmt/n [filenames]
```

x 指定したアーカイブファイルからのファイルの抽出を指定します。指定したドライブのテープに含まれるすべてのファイルが現在のディレクトリにコピーされます。

v 各ファイルを取り出すたびに、その名前を表示します。

f /dev/rmt/ n アーカイブを含むテープデバイスを示します。

filenames 取り出すファイルを指定します。ファイルが複数の場合は、各ファイルをスペースで区切ります。

詳細は、[tar\(1\)](#)のマニュアルページを参照してください。

### 4 ファイルがコピーされたことを確認します。

```
$ ls -l
```

## 例 21-4 テープ上のファイルを取り出す (tar)

次の例では、ドライブ 0 のテープからすべてのファイルを取り出す方法を示します。

```
$ cd /var/tmp
$ tar xvf /dev/rmt/0
x reports/, 0 bytes, 0 tape blocks
x reports/reportA, 0 bytes, 0 tape blocks
x reports/reportB, 0 bytes, 0 tape blocks
x reports/reportC, 0 bytes, 0 tape blocks
x reports/reportD, 0 bytes, 0 tape blocks
$ ls -l
```

**注意事項** テープから抽出されるファイル名は、アーカイブに格納されているファイル名と同一でなければなりません。ファイルの名前やパス名が不明な場合は、まずテープ上のファイルのリストを表示します。テープ上のファイルをリスト表示する方法については、[436 ページ](#)の「[テープ上のファイルのリストを表示する方法 \(tar\)](#)」を参照してください。

## pax コマンドを使用してファイルをテープにコピーする

### ▼ ファイルをテープにコピーする方法 (pax)

- 1 コピーするファイルの存在するディレクトリに移動します。
- 2 書き込み可能なテープをテープドライブに挿入します。
- 3 ファイルをテープにコピーします。

```
$ pax -w -f /dev/rmt/n filenames
```

-w                      書き込みモードを有効にします。

-f /dev/rmt/n          テープドライブを識別します。

filenames            コピーするファイルとディレクトリを指定します。ファイルが複数の場合は、各ファイルをスペースで区切ります。

詳細は、[pax\(1\)](#) を参照してください。

- 4 ファイルがテープにコピーされたことを確認します。

```
$ pax -f /dev/rmt/n
```

- 5 テープをドライブから取り出します。ファイル名をテープのラベルに記入します。

#### 例 21-5 ファイルをテープにコピーする (pax)

次の例は、pax コマンドを使用して、現在のディレクトリ内のファイルをすべてコピーする方法を示します。

```
$ pax -w -f /dev/rmt/0 .
$ pax -f /dev/rmt/0
filea fileb filec
```

## cpio コマンドを使用してファイルをテープにコピーする

### ▼ ディレクトリ内のすべてのファイルをテープにコピーする方法(cpio)

- 1 コピーするファイルの存在するディレクトリに移動します。
- 2 書き込み可能なテープをテープドライブに挿入します。
- 3 ファイルをテープにコピーします。

```
$ ls | cpio -oc > /dev/rmt/n
```

ls cpio コマンドにファイル名のリストを渡します。

cpio -oc cpio コマンドがコピーアウトモード (-o) で動作し、ASCII 文字形式 (-c) でヘッダー情報を書き込むように指定します。これらのオプションによりほかのベンダーのシステムとの互換性を保ちます。

> /dev/rmt/n 出力ファイルを指定します。

ディレクトリ内のすべてのファイルは、指定したドライブ内のテープにコピーされ、テープ上の既存のファイルが上書きされます。コピーされた合計ブロック数が表示されます。

- 4 ファイルがテープにコピーされたことを確認します。

```
$ cpio -civt < /dev/rmt/n
```

-c cpio コマンドがファイルを ASCII 文字形式で読み込むように指定します。

-i cpio コマンドがコピーインモードで動作することを指定します。この時点ではファイルをリストするだけです。

-v ls -l コマンドの出力と同様の形式で出力を表示します。

-t 指定したテープドライブ内のテープ上にあるファイルの内容リストを表示します。

< /dev/rmt/n 既存の cpio アーカイブの入力ファイルを指定します。

- 5 テープをドライブから取り出します。ファイル名をテープのラベルに記入します。

**例 21-6** ディレクトリ内のすべてのファイルをテープにコピーする (cpio)

次の例では、/export/home/kryten ディレクトリ内のすべてのファイルをテープドライブ 0 のテープにコピーする方法を示します。

```
$ cd /export/home/kryten
$ ls | cpio -oc > /dev/rmt/0
1280 blocks
$ cpio -civt < /dev/rmt/0
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filea
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, fileb
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filec
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, letters
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, reports
1280 blocks
```

## ▼ テープ上のファイルのリストを表示する方法 (cpio)

注-テープの内容リストを表示するには、cpio コマンドがアーカイブ全体を処理する必要があるため、かなりの時間がかかります。

- 1 テープをテープドライブに挿入します。
- 2 テープ上のファイルのリストを表示します。

```
$ cpio -civt < /dev/rmt/n
```

**例 21-7** テープ上のファイルのリストを表示する (cpio)

次の例では、ドライブ 0 のテープに含まれるファイルのリストを表示します。

```
$ cpio -civt < /dev/rmt/0
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filea
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, fileb
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filec
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, letters
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, reports
1280 blocks
```



## ▼ テープからすべてのファイルを取り出す方法 (cpio)

相対パス名を使用してアーカイブを作成した場合、入力ファイルはそれを取り出すときに現在のディレクトリ内のディレクトリとして作成されます。ただし、絶対パス名を指定してアーカイブを作成した場合は、それと同じ絶対パス名を使用してシステム上でファイルが再作成されます。



注意-絶対パス名を使用すると、自分のシステム上にある元のファイルを上書きすることになるので危険です。

- 1 ファイルを置きたいディレクトリに移動します。

- 2 テープをテープドライブに挿入します。

- 3 テープからすべてのファイルを抽出します。

```
$ cpio -icvd < /dev/rmt/n
```

- i                   標準入力からファイルを取り出します。
- c                   cpio コマンドがファイルを ASCII 文字形式で読み込むように指定します。
- v                   取り出されたファイルを `ls` コマンドの出力と同様の形式で表示します。
- d                   必要に応じてディレクトリを作成します。
- < /dev/rmt/n       出力ファイルを指定します。

- 4 ファイルがコピーされたことを確認します。

```
$ ls -l
```

### 例 21-8 テープからすべてのファイルを取り出す (cpio)

次の例では、ドライブ 0 のテープからすべてのファイルを取り出す方法を示します。

```
$ cd /var/tmp
cpio -icvd < /dev/rmt/0
answers
sc.directives
tests
8 blocks
$ ls -l
```

## ▼ テープから特定のファイルを取り出す方法 (cpio)

- 1 ファイルを置きたいディレクトリに移動します。
- 2 テープをテープドライブに挿入します。
- 3 テープからファイルのサブセットを取り出します。

```
$ cpio -icv "*file" < /dev/rmt/n
```

-i                    標準入力からファイルを取り出します。

-c                    cpio コマンドを使用してヘッダーを ASCII 文字形式で読み込むように指定します。

-v                    取り出されたファイルを `ls` コマンドの出力と同様の形式で表示します。

*"\*file"*            パターンに一致するすべてのファイルを現在のディレクトリにコピーするように指定します。複数のパターンを指定できますが、個々のパターンを二重引用符で囲まなければなりません。

< /dev/rmt/*n*        入力ファイルを指定します。

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

- 4 ファイルがコピーされたことを確認します。

```
$ ls -l
```

### 例 21-9 指定したファイルをテープから取り出す (cpio)

次の例では、末尾に接尾辞 `chapter` が付いているすべてのファイルをドライブ 0 のテープから取り出します。

```
$ cd /home/smith/Book
$ cpio -icv "**chapter" < /dev/rmt/0
Boot.chapter
Directory.chapter
Install.chapter
Intro.chapter
31 blocks
$ ls -l
```

# ファイルをリモートテープデバイスにコピーする

## ▼ ファイルをリモートテープデバイスにコピーする方法 (tar と dd)

- 1 テープドライブにアクセスできるように、リモートシステムで **ssh** を構成します。『[Solaris のシステム管理: セキュリティサービス](#)』の「[Secure Shell の構成 \(タスク\)](#)」を参照してください。

- 2 ファイルを置きたいディレクトリに移動します。

- 3 テープをテープドライブに挿入します。

- 4 ファイルをリモートテープドライブへコピーします。

```
$ tar cvf - filenames | ssh remote-host dd of=/dev/rmt/n obs=block-size
```

**tar cf**                      テープアーカイブを作成し、アーカイブに含まれるファイルをリスト表示し、テープデバイスを指定します。

**v**                              tar ファイルのエントリに関する追加情報を表示します。

**- (ハイフン)**                  可変部としてテープデバイスの代わりに指定します。

**filenames**                  コピーするファイルを指定します。ファイルが複数の場合は、各ファイルをスペースで区切ります。

**ssh | remote-host**          tar コマンドの出力をパイプを通してリモートシステムに渡します。

**dd of= /dev/rmt/n**          出力デバイスを指定します。

**obs=block-size**              ブロック係数を指定します。

- 5 テープをドライブから取り出します。ファイル名をテープのラベルに記入します。

### 例 21-10 ファイルをリモートテープドライブにコピーする (tar と dd)

```
tar cvf - * | ssh mercury dd of=/dev/rmt/0 obs=126b
password:
a answers/ 0 tape blocks
a answers/test129 1 tape blocks
a sc.directives/ 0 tape blocks
a sc.directives/sc.190089 1 tape blocks
a tests/ 0 tape blocks
a tests/test131 1 tape blocks
```

```
6+9 records in
0+1 records out
```

## ▼ ファイルをリモートテープデバイスから抽出する方法

- 1 テープをテープドライブに挿入します。

- 2 一時ディレクトリに移動します。

```
$ cd /var/tmp
```

- 3 リモートテープデバイスからファイルを抽出します。

```
$ ssh remote-host dd if=/dev/rmt/n | tar xvBpf -
```

`ssh remote-host` `dd` コマンドを使用してテープデバイスからファイルを取り出すために起動する Secure Shell です。

`dd if=/dev/rmt/n` 入力デバイスを指定します。

`| tar xvBpf -` `dd` コマンドの出力を `tar` コマンドにパイプして、ファイルを復元します。

- 4 ファイルが抽出されたことを確認します。

```
$ ls -l
```

### 例 21-11 ファイルをリモートのテープドライブから抽出する

```
$ cd /var/tmp
$ ssh mercury dd if=/dev/rmt/0 | tar xvBpf -
password:
x answers/, 0 bytes, 0 tape blocks
x answers/test129, 48 bytes, 1 tape blocks
20+0 records in
20+0 records out
x sc.directives/, 0 bytes, 0 tape blocks
x sc.directives/sc.190089, 77 bytes, 1 tape blocks
x tests/, 0 bytes, 0 tape blocks
x tests/test131, 84 bytes, 1 tape blocks
$ ls -l
```

## テープドライブの管理 (タスク)

---

この章では、Oracle Solaris OS でテープドライブを管理する方法について説明します。

この章の内容は次のとおりです。

- 445 ページの「使用するメディアの選択」
- 446 ページの「バックアップデバイス名」
- 448 ページの「テープドライブのステータスを表示する」
- 449 ページの「磁気テープカートリッジの取り扱い」
- 450 ページの「ドライブの管理とメディア処理のガイドライン」

この章で説明する手順は次のとおりです。

- 448 ページの「テープドライブのステータスを表示する方法」
- 449 ページの「磁気テープカートリッジのたるみを直す」
- 450 ページの「磁気テープカートリッジを巻き戻す」

### 使用するメディアの選択

通常は、次のテープメディアを使用して Oracle Solaris システムのバックアップを作成します。

- 1/2 インチのリールテープ
- 1/4 インチのストリームカートリッジテープ
- 8mm のカートリッジテープ
- 4mm のカートリッジテープ (DAT)

どのメディアを選択するかは、メディアをサポートする機器とファイルの格納に使用するメディア (通常はテープ) の可用性によって決まります。バックアップはローカルシステムから実行しなければなりませんが、ファイルはリモートデバイスに書き込めます。

次の表に、ファイルシステムのバックアップに使用する標準的なテープデバイスを示します。各デバイスの記憶容量は、ドライブのタイプとテープに書き込むデータのタイプによって異なります。

表 22-1   メディアの記憶容量

バックアップメディア	記憶容量
1/2 インチのリールテープ	140M バイト (6250 bpi)
2.5G バイト、1/4 インチのカートリッジ (QIC) テープ	2.5G バイト
DDS3 4mm カートリッジテープ (DAT)	12 - 24G バイト
14G バイト、8mm カートリッジテープ	14G バイト
DLT 7000 1/2 インチ カートリッジテープ	35 - 70G バイト

# バックアップデバイス名

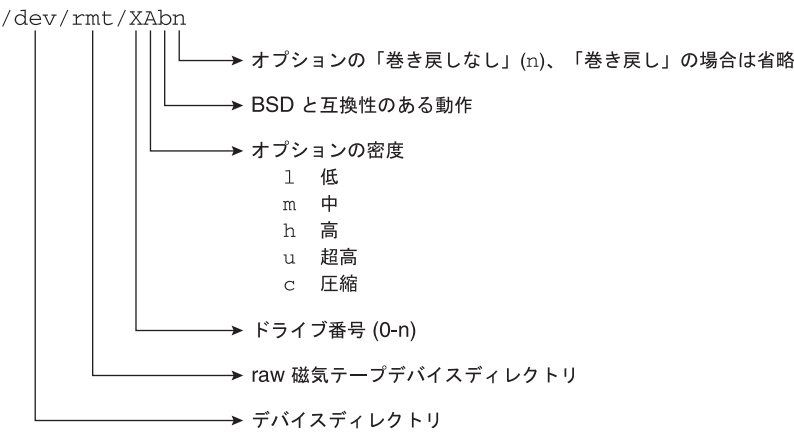
バックアップに使用するテープまたはデバイスに論理デバイス名を与えて指定します。この名前は、「raw」デバイスファイルの格納されたサブディレクトリを指し、ドライブの論理ユニット番号が含まれます。テープドライブの命名規則に従い、物理デバイス名ではなく論理デバイス名を使用します。次の表に、この命名方式を示します。

表 22-2   バックアップデバイスの基本的なデバイス名

デバイスの種類	名前
テープ	/dev/rmt/ <i>n</i>

通常は、次の図に示す方法で、テープデバイスを指定します。

図 22-1 テープドライブデバイス名



密度を指定しないと、テープドライブは通常その「推奨」密度で書き込みます。推奨密度は、一般にそのテープドライブでサポートされる最大密度です。ほとんどの SCSI ドライブはテープ上の密度やフォーマットを自動的に検出し、それに従って読み取りを実行します。ドライブでサポートされる密度を調べるには、`/dev/rmt` サブディレクトリを確認してください。このサブディレクトリには、各テープで異なる出力密度をサポートするためのテープデバイスファイルのセットが含まれます。

SCSI コントローラは、最大7台の SCSI テープドライブを持つこともできます。

## テープドライブの巻き戻しオプションを指定する

通常は、テープドライブを 0 から  $n$  までの論理デバイス番号で指定します。次の表に、「巻き戻し」または「巻き戻しなし」のオプションを付けてテープデバイス名を指定する方法を示します。

表 22-3 テープドライブの「巻き戻し」または「巻き戻しなし」オプション

ドライブおよび巻き戻し	使用するオプション
第 1 のドライブ、巻き戻し	<code>/dev/rmt/0</code>
第 1 のドライブ、巻き戻しなし	<code>/dev/rmt/0n</code>
第 2 のドライブ、巻き戻し	<code>/dev/rmt/1m</code>
第 2 のドライブ、巻き戻しなし	<code>/dev/rmt/1n</code>

# テープドライブに別の密度を指定する

デフォルトでは、テープドライブはその「推奨」密度で書き込みますが、これは一般にそのテープドライブでサポートされる最大密度です。テープデバイスを指定しなければ、コマンドはデバイスでサポートされるデフォルト密度でドライブ番号 0 に書き込みます。

テープを特定の密度しかサポートされないテープドライブが付いたシステムに転送するには、目的の密度で書き込むデバイス名を指定します。次の表に、テープドライブに別の密度を指定する方法を示します。

表 22-4 テープドライブに別の密度を指定する

ドライブ、密度、巻き戻し	使用するオプション
第 1 のドライブ、低密度、巻き戻し	<code>/dev/rmt/0l</code>
第 1 のドライブ、低密度、巻き戻しなし	<code>/dev/rmt/0ln</code>
第 2 のドライブ、中密度、巻き戻し	<code>/dev/rmt/1m</code>
第 2 のドライブ、中密度、巻き戻しなし	<code>/dev/rmt/1mn</code>

密度のオプションについては、[446 ページの「バックアップデバイス名」](#)を参照してください。

# テープドライブのステータスを表示する

`mt` コマンドの `status` オプションを使用すると、テープドライブに関するステータス情報を表示できます。`mt` コマンドは、`/kernel/drv/st.conf` ファイルに記述されているすべてのテープドライブの情報を表示します。

## ▼ テープドライブのステータスを表示する方法

- 1 情報を表示するドライブにテープをロードします。
- 2 テープドライブのステータスを表示します。  
`# mt -f /dev/rmt/n status`
- 3 テープドライブ番号を 0、1、2、3 というように置き換えて手順 1 と 2 を繰り返し、使用可能なすべてのテープドライブに関する情報を表示します。

### 例 22-1 テープドライブのステータスを表示する

次の例は、QIC-150 テープドライブ (`/dev/rmt/0`) のステータスを示します。



```
$ mt -f /dev/rmt/0 status
Archive QIC-150 tape drive:
 sense key(0x0)= No Additional Sense residual= 0 retries= 0
 file no= 0 block no= 0
```

次の例は、Exabyte テープドライブ (/dev/rmt/1) のステータスを示します。

```
$ mt -f /dev/rmt/1 status
Exabyte EXB-8200 8mm tape drive:
sense key(0x0)= NO Additional Sense residual= 0 retries= 0
file no= 0 block no= 0
```

次の方法を使用すると、システムを手早くポーリングしてすべてのテープドライブを検査できます。

```
$ for drive in 0 1 2 3 4 5 6 7
> do
> mt -f /dev/rmt/$drive status
> done
Archive QIC-150 tape drive:
 sense key(0x0)= No Additional Sense residual= 0 retries= 0
 file no= 0 block no= 0
/dev/rmt/1: No such file or directory
/dev/rmt/2: No such file or directory
/dev/rmt/3: No such file or directory
/dev/rmt/4: No such file or directory
/dev/rmt/5: No such file or directory
/dev/rmt/6: No such file or directory
/dev/rmt/7: No such file or directory
$
```

## 磁気テープカートリッジの取り扱い

テープの読み込み中にエラーが発生した場合は、テープのたるみを直し、テープドライブを掃除してからやり直してください。

### 磁気テープカートリッジのたるみを直す

mt コマンドを使用して磁気テープカートリッジのたるみを直します。

例:

```
$ mt -f /dev/rmt/1 retension
$
```

---

注-QIC 以外のテープドライブのたるみは直さないでください。

---

## 磁気テープカートリッジを巻き戻す

磁気テープカートリッジを巻き戻すには、`mt` コマンドを使用します。

例:

```
$ mt -f /dev/rmt/1 rewind
$
```

## ドライブの管理とメディア処理のガイドライン

バックアップテープは読み込めなければ役に立ちません。このため、定期的に掃除および検査を行い、テープドライブが正常に動作するようにしてください。テープドライブのクリーニング手順については、ハードウェアのマニュアルを参照してください。テープハードウェアを検査するには、ファイルをいくつかテープにコピーし、ファイルを読み込んで、オリジナルのファイルとコピーしたファイルを比較します。

ハードウェアは、システムからレポートされないような障害を起こす可能性があるので注意してください。

バックアップ後は、必ずテープにラベルを付けてください。このラベルは変更しないでください。バックアップを実行するたびに、次の情報を記入した別のテープラベルを作成します。

- バックアップ日付
- マシン名およびバックアップを作成したファイルシステム
- バックアップレベル
- テープ番号 (バックアップが複数のボリュームにまたがる場合は  $n$  本のうちの 1 本目)
- サイト特有の情報

テープは、磁気を発生させる機器から離れた埃のない安全な場所に保管してください。たとえば、アーカイブしたテープをリモート地の防火キャビネットに保管します。

各ジョブ (バックアップ) がどのメディア (テープボリューム) に格納されているかということと、各バックアップファイルがどこに保管されているかを記録したログを作成し、管理する必要があります。

## UFS バックアップおよび復元コマンド (参照情報)

---

この章では、`ufsdump` コマンドと `ufsrestore` コマンドの参照情報を示します。

この章で説明する情報は次のとおりです。

- 451 ページの「`ufsdump` コマンドの機能」
- 456 ページの「`ufsdump` コマンドオプションおよび引数の指定」
- 456 ページの「`ufsdump` とセキュリティーに関する注意事項」
- 457 ページの「`ufsrestore` オプションおよび引数の指定」

バックアップの実行の概要については、379 ページの「UFS ファイルシステムのバックアップと復元の概要」を参照してください。

バックアップタスクについては、第 19 章「UFS ファイルシステムのバックアップと復元(概要/タスク)」を参照してください。

### `ufsdump` コマンドの機能

`ufsdump` コマンドは、ファイルシステムのバックアップ作成時に 2 つのパスを作成します。最初のパスでは、このコマンドは `raw` デバイスファイル内でファイルシステムを走査し、メモリー内にディレクトリとファイルのテーブルを作成します。次に、そのテーブルをバックアップメディアに書き込みます。2 つ目のパスでは、`ufsdump` は `i` ノードに番号順にアクセスし、ファイルの内容を読み込んでバックアップメディアに書き込みます。

### デバイス特性の判断

`ufsdump` コマンドに必要なことは、適切なテープブロックの大きさを認識することと、どのようにしてメディアの終わりを検出するかということです。

## メディアの終わりの検出

ufsdump コマンドは、一連の固定長レコードを書き込みます。ufsdump コマンドは、レコードの一部にしか書き込まれていないという通知を受け取ると、メディアの物理的な終わりに達したものと判断します。この方法は、ほとんどのデバイスに有効です。部分的なレコードしか書き込まれなかったことをデバイスが ufsdump に通知できない場合、ufsdump が別のレコードの書き込みを試みると、メディアエラーが発生します。

---

注-DAT デバイスと 8mm テープデバイスでは、メディアの終わりが検出されません。カートリッジテープデバイスと 1/2 インチテープデバイスでは、メディアの終わりは検出されません。

---

ufsdump は、ほとんどのデバイスのメディアの終わりを自動的に検出します。したがって、通常は -c、-d、-s、-t オプションを使用しなくても、複数のボリュームのバックアップを実行できます。

ufsdump コマンドで、デバイスがメディアの終わりを検出する方法を確認できない場合は、メディアの終わりに関するオプションを使用する必要があります。

restore コマンドとの互換性を確保するため、サイズオプションを使用すると、従来どおり、現在のテープやフロッピーディスクの終わりに達する前に、ufsdump を次のテープやフロッピーディスクに強制的に進ませることができます。

## ufsdump コマンドを使用したデータのコピー

ufsdump コマンドは、raw ディスクスライスからデータのみをコピーします。ファイルシステムがまだ有効であれば、メモリーバッファ内のデータがコピーされていない可能性があります。ufsdump コマンドによるバックアップでは、空きブロックはコピーされず、ディスクスライスのイメージも作成されません。シンボリックリンクがほかのスライス上のファイルを指す場合は、リンク自体がコピーされます。

## /etc/dumpdates ファイルの目的

ufsdump コマンドを -u オプション付きで使用すると、/etc/dumpdates というファイルを管理し、更新できます。/etc/dumpdates ファイル内の各行は、次の情報を表しています。

- バックアップが作成されたファイルシステム
- 前回実行されたバックアップのダンプレベル
- バックアップの曜日、日付、および時刻

例:

```
cat /etc/dumpdates
/dev/rdsdsk/c0t0d0s0 0 Wed Jul 7 13:36:17 2010
/dev/rdsdsk/c0t0d0s7 0 Thu Jul 8 12:53:12 2010
/dev/rdsdsk/c0t0d0s7 9 Thu Jul 8 13:41:48 2010
```

増分バックアップの実行時に、ufsdump コマンドは /etc/dumpdates ファイルを検査して、下のダンプレベルの最後のバックアップ日付を調べます。次に、下のレベルのバックアップ以降に更新されたすべてのファイルをメディアにコピーします。バックアップが完了すると、完了したばかりのバックアップを記述する新しい情報行によって、そのレベルの最後のバックアップの情報行が置き換えられます。

/etc/dumpdates ファイルを使用して、バックアップが実行中であるかどうかを検査してください。機器に問題が発生している場合は、この検査が特に重要です。機器の障害が原因でバックアップを完了できないと、そのバックアップは /etc/dumpdates ファイルに記録されません。

ディスク全体を復元する必要がある場合は、/etc/dumpdates ファイル内で最後のバックアップの日付とレベルを検査できるので、ファイルシステム全体を復元するために必要なファイルを判断できます。

---

注 - /etc/dumpdates ファイルは、編集可能なテキストファイルです。ただし、編集するかどうかはユーザーの判断によります。ファイルに変更を加えた結果、アーカイブテープと一致なくなると、必要なテープ (またはファイル) がどれであるかわからなくなることがあります。

---

## バックアップデバイス (*dump-file*) 引数

*dump-file* 引数 (-f オプションで使用) では、バックアップ先を指定します。バックアップ先は、次のいずれかになります。

- ローカルのテープドライブ
- ローカルのフロッピーディスクドライブ
- リモートのテープドライブ
- リモートのフロッピーディスクドライブ
- 標準出力

この引数は、バックアップ先がデフォルトのローカルテープドライブ /dev/rmt/0 でないときに使用します。-f オプションを使用する場合は、*dump-file* 引数の値を指定しなければなりません。

---

注 - *dump-file* 引数では、ローカルディスクまたはリモートディスク上のファイルを指すこともできます。誤用するとファイルシステムが壊れる可能性があります。

---

## ローカルのテープドライブまたはフロッピーディスクドライブ

通常、*dump-file* 引数には、テープデバイスかフロッピーディスク用の raw デバイス ファイルを指定します。ufsdump コマンドは、出力デバイスへの書き込み時にバックアップファイルを1つ作成しますが、このファイルは複数のテープやフロッピーディスクにまたがってもかまいません。

デバイスの省略形を使用して、システム上のテープデバイスかフロッピーディスク デバイスを指定します。最初のデバイスは常に 0 です。たとえば、SCSI テープコントローラが1つと、中密度の形式を使用する QIC-24 テープドライブが1つある場合は、次のデバイス名を使用します。

```
/dev/rmt/0m
```

テープデバイス名を指定するときは、名前の末尾に文字 *n* を付けて、バックアップの完了後にテープドライブを巻き戻さないように指定することもできます。例:

```
/dev/rmt/0mn
```

テープに複数のファイルを格納する場合は、no-rewind オプションを使用します。バックアップ中に領域を使い果たすと、ufsdump コマンドから新しいテープの挿入を促すプロンプトが表示されるまで、テープは巻き戻されません。デバイスの命名規則の詳細は、[446 ページの「バックアップデバイス名」](#)を参照してください。

## リモートのテープドライブまたはフロッピーディスクドライブ

次の形式で、リモートのテープデバイスまたはフロッピーディスクを指定します。*host: device*。ローカルシステム上のスーパーユーザーがリモートシステムへのアクセス権を持っている場合、ufsdump コマンドはリモートデバイスに書き込みます。通常、スーパーユーザーとして ufsdump コマンドを実行するのであれば、ローカルシステム名をリモートシステムの */.rhosts* ファイルに記述しておく必要があります。デバイスを *user@host: device* と指定した場合、ufsdump コマンドは指定されたユーザーでリモートシステム上のデバイスへのアクセスを試みます。この場合、指定されたユーザーの名前が、リモートシステム上の */.rhosts* ファイル中に含まれている必要があります。

## ufsdump コマンドで標準出力を使用する

*dump-file* 引数としてダッシュ (-) を指定すると、ufsdump コマンドは標準出力に書き込みます。

---

注 - *dump-file* 引数として標準出力を指定すると、-v オプション (検査) は機能しません。

---

ufsdump コマンドを使用して標準出力に書き込み、ufsrestore コマンドを使用して標準入力から読み込むと、パイプライン内でファイルシステムをコピーできます。例:

```
ufsdump 0f - /dev/rdisk/c0t0d0s7 | (cd /home; ufsrestore xf -)
```

## バックアップを作成するファイルを指定する

コマンド行の最後の引数として、バックアップするファイル (*filenames*) を必ず指定してください。この引数は、バックアップのコピー元または内容を指定します。

ファイルシステムの場合、次のように raw デバイスファイルを指定します。

```
/dev/rdisk/c0t0d0s7
```

ファイルシステムは、そのエントリが `/etc/vfstab` ファイルに存在すれば、マウントポイントディレクトリ (`/export/home` など) を使用して指定できます。

デバイスの命名規則の詳細は、[446 ページの「バックアップデバイス名」](#)を参照してください。

個々のファイルやディレクトリごとに、1 つまたは複数の名前を空白で区切って入力します。

---

注 - `ufsdump` コマンドを使用して (ファイルシステム全体ではなく) 1 つまたは複数のディレクトリやファイルのバックアップを作成するときには、レベル 0 のバックアップが実行されます。増分バックアップは適用されません。

---

## テープの性質を指定する

テープの性質を指定しなければ、`ufsdump` コマンドはデフォルト設定を使用します。テープカートリッジ (c)、密度 (d)、サイズ (s)、トラック数 (t) を指定できます。オプションの順序とその引数の順番が一致していれば、オプションはいくつでも指定できます。

## ufsdump コマンドの制限

次に、`ufsdump` コマンドでは実行できない操作を示します。

- ファイルシステムのバックアップに必要なテープやフロッピーディスクの数を自動的に計算する。仮実行 (ドライラン) モード (s オプション) を使用すると、実際にファイルシステムをバックアップする前に必要な容量を判定できます。
- アクティブなファイルシステムをバックアップするときの問題を最小限度に抑えるために、組み込みエラー検査機能を提供する。
- サーバーからリモートにマウントされたファイルをバックアップする。サーバー上のファイルのバックアップは、そのサーバー上で実行しなければなりません。ユーザーがサーバー上で所有するファイル上で `ufsdump` コマンドを実行するアクセス権は拒否されます。

## ufsdump コマンドオプションおよび引数の指定

このセクションでは、ufsdump コマンドのオプションと引数の指定方法について説明します。ufsdump コマンドの構文を、次に示します。

```
/usr/sbin/ufsdump options arguments filenames
```

**options**            1 文字のオプション名からなる 1 つの文字列です。

**arguments**        オプションの引数を指定します。複数の文字列も指定できます。オプション文字とそれに関連する引数は、同じ順序で並べる必要があります。

**filenames**        バックアップするファイルを指定します。これらの引数は、空白で区切り、常に最後に指定します。

## ufsdump のデフォルトオプション

オプションを指定せずに ufsdump コマンドを実行する場合は、次の構文を使用します。

```
ufsdump filenames
```

ufsdump コマンドでは、デフォルトで次のオプションと引数を使用されます。

```
ufsdump 9uf /dev/rmt/0 filenames
```

これらのオプションでは、デフォルトのテープドライブ上にその推奨密度でレベル 9 の増分バックアップが作成されます。

ufsdump オプションについては、[ufsdump\(1M\)](#) のマニュアルページを参照してください。

## ufsdump とセキュリティに関する注意事項

セキュリティ保護を適用するには、次の操作を実行する必要があります。

- ufsdump コマンドの実行には、スーパーユーザーのアクセス権を必要とします。
- 集中バックアップを実行する場合は、クライアント上とサーバー上の `/.rhosts` ファイルからスーパーユーザーアクセスのエントリを削除します。

セキュリティに関する一般的な情報は、『[Solaris のシステム管理: セキュリティサービス](#)』を参照してください。



# ufsrestore オプションおよび引数の指定

ufsrestore コマンドの構文を、次に示します。

```
/usr/sbin/ufsrestore options arguments filenames
```

- |                  |                                                                                                                                        |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| <i>options</i>   | 1 文字のオプション名からなる 1 つの文字列です。次のオプションから 1 つだけ選択します。i、r、R、t、または x。ufsrestore オプションについては、 <a href="#">ufsrestore(1M)</a> のマニュアルページを参照してください。 |
| <i>arguments</i> | オプションに対応する引数です。オプション文字とそれに関連する引数は、同じ順序で並べる必要があります。                                                                                     |
| <i>filenames</i> | 復元するファイルを、x または t オプションの引数として指定します。これらの引数は、空白で区切り、常に最後に指定します。                                                                          |



# 索引

---

## 数字・記号

1394 (FireWire) のサポート, 説明, 123  
4.3 Tahoe ファイルシステム, 300  
9660 DVD フォーマット, 34

## A

AutoFS, 310–311

## B

BSD Fat Fast ファイルシステム, 300

## C

CD, 名前, 33  
cdrw コマンド  
    アクセスの制限 (方法), 45  
    説明, 41  
    データ CD、データ DVD、およびオーディオ  
        CD の書き込み (概要), 44  
cfgadm  
    PCI ホットプラグ (概要), 74  
    SCSI ホットプラグ (概要), 74  
cpio コマンド  
    (概要), 433  
    テープからすべてのファイルを抽出 (方  
        法), 441  
    テープ上のファイルのリスト (方法), 440

cpio コマンド (続き)

    ファイルシステム間でのディレクトリのコピー  
        (方法), 433  
CTFS ファイルシステム, 304

## D

datadm コマンド, 163  
/dev/dsk ディレクトリ, 70  
/dev/rdisk ディレクトリ, 70  
devfsadm コマンド, 68  
dfstab ファイル, 共有されたローカルのリムーバ  
    ブルメディアの構成 (方法), 37  
df コマンド  
    -h オプション, 184  
    -t オプション (ブロックの合計数), 185  
    概要, 182  
DOS, ファイルシステム, 301  
driver not attached メッセージ, 59  
dumpadm コマンド, 70  
du コマンド, 189, 190  
DVD, ISO 9660 フォーマット, 34  
DVD-RAM, ファイルシステムの作成 (手順), 30  
DVD-ROM, 302

## E

EFI ラベル  
    VTOC ラベルとの比較, 169  
    (概要), 168  
    制限, 170

## EFI ラベル (続き)

問題のトラブルシューティング, 171  
eject コマンド, リムーバブルメディア (方法), 36  
/etc/dfs/dfstab ファイル, 共有されたりムーバブルメディアの構成 (方法), 37  
/etc/dumpdates ファイル, 452–453  
/export/home ディレクトリ, 306

**F**

FDFS ファイルシステム, 304  
FIFOFS ファイルシステム, 304  
FIFO の i ノード, 352  
find コマンド  
  コアファイル, 194  
  サイズ制限を超えるファイルの検索, 188  
  古い/使用されていないファイル, 192  
  古い/使用されていないファイル, 191  
format ユーティリティー  
  analyze メニュー, 294  
  defect メニュー, 295–296  
  fdisk メニュー, 293  
  partition メニュー, 292, 293  
  Solaris fdisk パーティションの作成 (方法), 254  
  UFS ファイルシステム用のディスクスライスの作成 (手順)  
    x86, 240  
  (概要), 172  
  機能と利点, 172  
  コマンド名の入力 (方法), 296  
  システム上のディスクの確認 (方法), 198  
  システム上のディスクの確認 (例), 199  
  使用上のガイドライン, 174–175  
  使用する場合, 173  
  ディスクがフォーマット済みかどうかの確認 (方法), 201  
  ディスクスライス情報の表示 (例), 204  
  ディスクスライスの作成 (手順)  
    SPARC, 220  
  ディスクラベルの作成  
    例, 207  
  入力, 296, 297  
  破損したディスクラベルの復元 (方法), 212  
  ヘルプ機能の使用, 297

## format ユーティリティー (続き)

  メインメニュー, 290  
free hog スライス, 「提供側のスライス」を参照  
fsck コマンド  
  FSACTIVE 状態フラグ, 349  
  FSBAD 状態フラグ, 349  
  FSCLEAN 状態フラグ, 349  
  FSSTABLE 状態フラグ, 349  
  検査  
    i ノードリストのサイズ, 351  
    空き i ノード数, 352  
    空きブロック数, 352  
    スーパーブロック, 351  
  構文とオプション, 369  
  修復, 362  
  修復する条件, 350  
  状態フラグ, 349  
  対話式で使用, 357  
fssnap コマンド, UFS スナップショットの作成 (方法), 424  
fstypes ファイル, 311  
fuser コマンド  
  リムーバブルメディアが使用中かどうかの調査 (方法), 36  
  リムーバブルメディアにアクセスしているプロセスの強制終了 (方法), 36

**H**

High Sierra ファイルシステム, 301  
/home (自動マウント), 311  
HSFS, 「High Sierra ファイルシステム」を参照

**I**

InfiniBand デバイス  
  HCA に接続されている IB デバイスの構成解除 (方法), 158  
  HCA のカーネル IB クライアントの表示 (方法), 157  
  IB Port、HCA\_SVC、または VPPA デバイスの構成解除 (方法), 155

## InfiniBand デバイス (続き)

IB Port、HCA\_SVC、または VPPA デバイスの構成 (方法), 156

IB 擬似デバイスの構成解除 (方法), 157

IB 擬似デバイスの構成 (方法), 157

IB 通信サービスの表示 (方法), 159

IOC デバイスの構成解除 (方法), 155

IOC デバイスの構成 (方法), 155

IOC の構成の更新 (方法), 161

IP p\_key テーブルの更新 (方法), 159

VPPA 通信サービスの追加 (方法), 160

概要, 149

既存の IB Port、HCA\_SVC、VPPA 通信サービスの削除 (方法), 160

動的再構成 (概要), 152

表示 (方法), 153

installboot コマンド, SPARC, 226

installgrub コマンド, x86, 243

## iSCSI

iSCSI 構成に関する問題のトラブル

シューティング (方法), 281

iSCSI 構成の監視 (方法), 276

iSCSI ターゲット発見の構成 (手順), 264

iSCSI ディスクへのアクセス (方法), 266

iSCSI のイニシエータとターゲットのパラメータの変更 (方法), 279

一般的な iSCSI エラーメッセージ, 284 (概要), 259

静的および動的ターゲット発見, 262

ソフトウェア要件およびハードウェア要件, 261

単方向または双方向 CHAP 認証の構成 (方法), 269

発見された iSCSI ターゲットの削除 (方法), 267

iscsiadm list, iSCSI 構成情報の表示 (例), 276

iscsiadm add コマンド, 静的または動的ターゲットの追加 (例), 265

iscsiadm modify コマンド

CHAP を有効にする (例), 269

静的または動的ターゲットの有効化または無効化 (例), 267

iscsiadm remove コマンド, 静的または動的ターゲットの削除 (例), 267

ISO 9660 ファイルシステム, 301

ISO 標準, 9660 DVD フォーマット, 34

i ノード, 372-373

FIFO, 352

キャラクタ型特殊ファイル, 352

形式とタイプの検査, 352

ごとのバイト数, 377-378

サイズ, 354

シンボリックリンク, 352

通常, 352

ディレクトリ, 352

不正な i ノード番号, 356

ブロック型特殊ファイル, 352  
リンク数, 353

i ノードの形式, 352

i ノードの状態, 353

i ノードのタイプ, 352

i ノードのリンク数, 353

i ノードリストのサイズ, 351

## K

/kernel/drv ディレクトリ, 58

## L

lost+found ディレクトリ, 349

ls コマンド, 185, 186, 192

-l オプション (バイト単位のサイズ), 186

-s オプション (ブロック単位のサイズ), 187

-t オプション (最新ファイル), 191

ディレクトリサイズのチェック, 186

## M

media was found メッセージ, 27

mkisofs コマンド, データ CD または DVD の  
ファイルシステムを作成する (方法), 47

MNTFS ファイルシステム, 307

mnttab ファイル, 308

mt コマンド, 449

**N**

NAMEFS ファイルシステム, 304

NFS

    vfstab エントリ, 324

    サーバーの説明, 309

    説明, 309

nfsd デーモン

    起動, 38

    実行中かどうかの確認, 38

no media was found メッセージ, 28

**O**

OBJFS ファイルシステム, 304

/opt ディレクトリ, 307

**P**

passwd ファイル, テープから復元 (例), 411

PCFS ファイルシステム, 301

PCI デバイス

    PCI アダプタカードの取り付け (方法), 95

    PCI アダプタカードの取り外し (手順), 93

    PCI 構成に関する問題のトラブルシューティング, 97

    PCI スロット構成情報の表示 (方法), 92

PROCFS ファイルシステム, (概要), 303–304

/proc ディレクトリ, 303, 307

prtvtoe コマンド, 71

    例, 211

**Q**

quot コマンド, 190

**R**

raw ディスクデバイスインタフェース, 70

RCM スクリプト

    概要, 99

    コマンド, 100

rm コマンド, 192, 194

Rock Ridge 拡張 (HSFS ファイルシステム), 301

rリムーバブルメディア, フロッピーディスクの  
    ロード (手順), 27

**S**

SATA デバイス

    SATA デバイス情報の表示 (方法), 97

    SATA デバイスの構成手順 (方法), 97

    SATA デバイスの構成 (方法), 98

SCSI テープドライブ, 447

SCSI デバイス

    cfgadm コマンドによる切り離し (方法), 83

    SCSI 構成に関する問題のトラブル

        シューティング, 89

    SCSI コントローラ上の同一 SCSI デバイスの交  
        換 (方法), 86

    SCSI コントローラの構成解除 (方法), 81

    SCSI コントローラの構成 (方法), 82

    SCSI コントローラの接続 (方法), 84

    SCSI デバイスの SCSI バスへの取り付け (方  
        法), 85

    SCSI デバイスの構成 (方法), 83

    SCSI デバイスの取り外し (方法), 88

    失敗した SCSI 構成解除操作の解決 (方法), 90  
        に関する情報の表示 (方法), 81

shareall コマンド, 310

SHAREFS ファイルシステム, 305

share コマンド, 310

    リムーバブルメディアをほかのシステムで使用  
        可能にする (方法), 37

Solaris fdisk パーティション, ガイドライン, 253

space、最適化のタイプ, 377

SPECFS ファイルシステム, 305

svcadm disable, (例), 22

swapadd コマンド, 335

SWAPFS ファイルシステム, 305

swap コマンド, 71

sysdef コマンド, 60

## T

## tar コマンド

dd コマンドによるファイルのリモートテープ  
へのコピー (方法), 443

dd コマンドによるリモートテープからの  
ファイルの取り出し (方法), 444  
(概要), 435

テープからのファイルの取り出し (方法), 437  
テープ上のファイルのリスト (方法), 436

time (最適化のタイプ), 377

TMPFS ファイルシステム, 概要, 303

/tmp ディレクトリ, 303, 307

## U

## uDAPL

DAT 静的レジストリ内のサービスプロバイダ  
の登録解除 (方法), 164

DAT 静的レジストリの更新 (方法), 163

DAT 静的レジストリへのサービスプロバイダ  
の登録 (方法), 163  
(概要), 161

有効にする (方法), 162

UDF ファイルシステム, 301

## ufsdump コマンド

オプションと引数, 456

完全バックアップの例, 401, 403

機能, 451-455

制限, 455

増分バックアップの例, 401

データのコピー方法, 452

メディアの終わりの検出, 452

ufsdump コマンド (概要), 399

ufsrestore コマンド, 457

使用するための準備 (概要), 405

使用するテープの決定 (方法), 407

## UFS スナップショット

完全バックアップの作成 (方法), 427

作成 (方法), 424

説明, 421

UFS ファイルシステム, 300

/etc/vfstab によるマウント, 324

情報の表示, 190

マウント, 324

UFS ファイルシステム用の設定

ディスク (概要)

SPARC, 218

x86, 239

UFS ファイルシステム用のディスク

設定 (手順)

SPARC, 220

UFS ルートファイルシステムのブートブロックの  
インストール (手順), SPARC, 226

UNIX ファイルシステム, 300

## USB デバイス

Oracle Solaris USB アーキテクチャー  
(USBA), 117

USB 2.0 のサポート, 112

USB カメラの追加 (方法), 125

USB 大容量ストレージデバイスの追加 (手  
順), 124

USB デバイスサブツリーの論理的な接続解除  
(方法), 147

USB デバイス情報の表示 (方法), 127

USB デバイスの構成 (方法), 145

USB デバイスの接続 (方法), 146

USB デバイスのリセット (方法), 147

USB デバイスの論理的な接続解除 (方法), 146  
オーディオ

(概要), 139

デバイスの所有権, 142

プライマリオーディオデバイスの変更 (方  
法), 142

オーディオデバイスの追加 (方法), 140

オーディオデバイスの問題のトラブル

シューティング, 142

(概要), 116

キーボードとマウスデバイス, 113

ケーブル, 116

合成デバイス, 119

大容量ストレージデバイス上での Solaris

パーティションの作成とスライスの変更 (方  
法), 133

大容量ストレージデバイス上のパーティション  
の変更と PCFS ファイルシステムの作成 (方  
法), 130

大容量ストレージデバイス上のファイルシステ  
ムの作成 (手順), 128

## USB デバイス (続き)

- 大容量ストレージデバイスのトラブル  
シューティングヒント, 137
- 大容量ストレージデバイスのマウント解除 (方法), 135
- 大容量ストレージデバイスのマウント (方法), 135
- 大容量ストレージデバイスのマウントまたはマウント解除 (方法), 135
- デバイスクラス, 120
- デバイスノード, 117
- デバイスの構成解除 (手順), 145
- 電源管理, 115
- ドライバ, 120
- 名前, 119
- バス情報の表示 (方法), 144
- バス電源供給方式のデバイス, 112
- バスの説明, 118
- 複合デバイス, 119
- 物理デバイス階層, 118
- プライマリーオーディオデバイスの識別 (手順), 141
- フロッピーディスクデバイス (概要), 122
- ホイール付きマウスのサポート, 114
- ホストコントローラとルートハブ, 114
- ホットプラグ (概要), 124
- ホットプラグ可能属性, 123
- リムーバブル大容量ストレージ (概要), 121
- 略語, 116

## V

- /var/adm ディレクトリ, サイズの制御, 193
- /var ディレクトリ, 306
- vfstab ファイル, 311, 335
  - LOFS のエントリ, 323
  - エントリの追加 (方法), 323
  - スワップの追加, 335

## Z

- ZFS ファイルシステム, 300

## ZFS ファイルシステム (続き)

- ディスクの設定 (方法)
  - x86, 252
- ZFS ファイルシステム用のディスク  
設定 (方法)
  - SPARC, 234
- ZFS ルートファイルシステムのブートブロックの  
インストール (方法), SPARC, 233
- ZFS ルートファイルシステム用の installboot コ  
マンド, 233
- ZFS ルートファイルシステム用の installgrub コ  
マンド, 251
- ZFS ルートファイルシステム用のディスクの設定  
接続 (方法)
  - SPARC, 228
  - x86, 245

## あ

- 空き i ノード数, 352
- 空きブロック, 374
- 空きブロック数, 352
- 空き容量 (最小), 376-377
- アクセス
  - iSCSI ディスク (方法), 266
  - ディスクデバイス, 69
  - テープデバイス, 72
  - リムーバブルメディア (方法), 35

## い

- 一時ディレクトリ, 191, 194
- 一時ファイルシステム (TMPFS), 概要, 303

## え

- エラーメッセージ, iSCSI, 284

## お

- オプション, ufsdump コマンド, 456



## か

解決, 失敗した SCSI 構成解除操作 (方法), 90  
書き込み, データ CD、データ DVD、および  
オーディオ CD (概要), 44

## 確認

nfsd デーモンが実行中, 38  
システム上のディスク (方法), 198  
ファイルシステム名, 406  
仮想メモリ記憶域、定義, 331  
監視, iSCSI 構成 (方法), 276  
間接ブロック, 355  
完全バックアップ  
  定義, 383  
  例, 401, 403

## き

記憶域 (仮想メモリ)、定義, 331  
記憶容量 (メディア), 384, 446  
擬似ファイルシステム, (概要), 302  
起動, nfsd デーモン, 38  
キャラクタ型特殊ファイルの i ノード, 352  
強制終了  
  ファイルシステムにアクセスしているすべての  
  プロセス (方法), 327  
  リムーバブルメディアにアクセスしているプロ  
  セス (方法), 36  
共有, ファイル, 309–310  
切り離し, SCSI コントローラ (方法), 83  
記録  
  増分バックアップ, 453  
  ダンプ, 452–453

## け

## 決定

テープデバイスのタイプ, 406  
テープデバイス名, 406  
ファイルシステムのタイプ, 311

## 検査

i ノードの形式とタイプ, 352  
i ノードリストの整合性, 351  
空き i ノード数, 352

## 検査 (続き)

空きブロック数, 352  
ファイルシステムサイズ, 351  
ファイルシステムの修復, 357  
リムーバブルメディア上のファイルシステム  
  (方法), 30

## 検索

および古い/使用されていないファイルの削除  
  「削除」を参照  
サイズ制限を超えるファイル, 188  
サイズの大きなファイル, 187

検出、メディアの終わり, ufsdump コマンド, 452

## こ

コアファイル, 検索と削除, 194  
交換, SCSI コントローラ上の同一 SCSI デバイス  
  (方法), 86

## 更新

DAT 静的レジストリ (方法), 163  
IOC の構成 (方法), 161  
IP p\_key テーブル (方法), 159

## 構成

IB Port、HCA\_SVC、または VPPA デバイス (方  
  法), 156  
IB 擬似デバイス (方法), 157  
IOC デバイス (方法), 155  
iSCSI ターゲット発見 (手順), 264  
iSCSI の単方向または双方向 CHAP 認証 (方  
  法), 269  
SATA デバイス (方法), 98  
SCSI コントローラ (方法), 82  
SCSI デバイス (方法), 83  
USB デバイス (方法), 145

## 構成解除

HCA に接続されている IB デバイス (方法), 158  
IB Port、HCA\_SVC、VPPA デバイス (方  
  法), 155  
IB 擬似デバイス (方法), 157  
IOC デバイス (方法), 155  
SATA デバイス (方法), 97  
SCSI コントローラ (方法), 81  
USB デバイス (手順), 145

構造、シリンダグループの, 371–374

構文, fsck コマンド, 369

コピー

cpio コマンドを使って個々のファイルを (概要), 433

cpio コマンドを使ってファイルシステム間の (概要), 433

cpio コマンドを使ってファイルのグループを (概要), 433

さ

最小空き容量, 376–377

サイズ

i ノード, 354

ディレクトリ, 189, 190

ファイル, 185, 186, 189, 190

ファイルシステムの検査, 351

フラグメント, 376

サイズの大きなファイル, 187

最大, 最大サイズを超えるファイルを見つける, 188

削除

UFS スナップショット情報 (例), 426

一時ファイル, 194

既存の IB Port、HCA\_SVC、VPPA 通信サービス (方法), 160

コアファイル, 194

発見された iSCSI ターゲット (方法), 267

古い/使用されていないファイルの検索と削除, 191

作成

DVD-RAM 上のファイルシステム (手順), 30

UFS スナップショット

例, 424

UFS スナップショット情報の完全バックアップ (方法), 427

UFS スナップショット (方法), 424

USB 大容量ストレージデバイス上での Solaris パーティションの作成とスライスの変更 (方法), 133

USB 大容量ストレージデバイス上のファイルシステム (手順), 128

作成 (続き)

ZFS ルートファイルシステム用のディスクスライス

SPARC, 229

ZFS ルートファイルシステム用のディスクスライス (方法)

x86, 246

データ CD または DVD のファイルシステム (方法), 47

リムーバブルメディア上のファイルシステム (方法), 29

ループバックファイルシステム (概要), 314

サポートされていないデバイス, 58

し

識別

デバイス, 60

プライマリ USB オーディオデバイス (手順), 141

自動構成プロセス, 58

自動マウント, /home, 311

修復, リムーバブルメディア上の不良ブロック (方法), 31

重複ブロック, 354

終了, リムーバブルメディアにアクセスしているプロセスの強制終了 (方法), 36

準備

バックアップ (概要), 397–398

ファイルの復元 (概要), 405–406

状態フラグ, fsck, 349

シリンダグループ, 371–374

シンボリックリンク, 352

す

スーパースブロック, 351, 364, 372

スライス (定義), 171

スワップパーティション、定義, 331

スワップファイル

vfstab へ追加, 335

表示, 337–338

## せ

制限, リムーバブルメディアへのアクセス (方法), 45

## 接続

SCSI コントローラ (方法), 84

USB デバイスの論理的な接続 (方法), 146

## 接続解除

USB デバイスサブツリーの論理的な接続解除 (方法), 147

USB デバイスの論理的な接続解除 (方法), 146

## 設定

ZFS ファイルシステム用のディスク (概要)

SPARC, 227

x86, 244-252

## そ

増分バックアップ, 384, 453

例, 401

## た

## ダンプレベル

定義, 389

日単位増分バックアップ, 390

## ち

## 調査

ファイルシステムのタイプ, 311

マウント済みのファイルシステム, 315

## つ

## 追加

USB オーディオデバイス (方法), 140

USB カメラ (方法), 125

USB 大容量ストレージデバイス (手順), 124

VPPA 通信サービス (方法), 160

新しいリムーバブルメディアドライブの追加 (手順), 34

## 追加 (続き)

エントリ/etc/vfstab ファイル (方法), 323

スワップを vfstab へ, 335

通常のノード, 352

## て

停止, ファイルシステムに対するすべてのプロセス (方法), 327

## ディスク

UFS ファイルシステム用の設定 (概要)

SPARC, 218

x86, 239

UFS ファイルシステム用の設定 (例)

SPARC, 223

UFS ルートファイルシステム用のディスクスライスの作成 (例)

SPARC, 222

ZFS ファイルシステム用の設定 (概要)

SPARC, 227

x86, 244-252

ディスクスライスとディスクラベルの作成 (手順)

SPARC, 220

破損したディスクラベルの復元 (概要), 212

破損したディスクラベルの復元 (方法), 212

フォーマット (概要), 175

フォーマット済みかを調べる (方法), 201

フォーマットする場合 (概要), 200

ディスクコントローラ, 71

## ディスクスライス

ZFS ルートファイルシステム用の作成

SPARC, 229

ZFS ルートファイルシステム用 (方法)

x86, 246

情報の表示 (概要), 203

定義, 171

ディスクスライスの指定, 71

ディスクのフォーマット (概要), 175

ディスクベースのファイルシステム, 300

## ディスク容量

サイズ制限を超えるファイルの検索, 188

サイズの大きなファイルを見つける, 187

## ディスク容量 (続き)

## 情報の表示

ディレクトリサイズ, 189, 190

ファイルサイズ, 185, 186, 189

ユーザーの所有するディスク容量, 190

古い/使用されていないファイルの検索と削除, 191

古いファイルまたは使用されていないファイルの検索と削除, 195

## ディスクラベル

作成 (概要), 205

説明, 168

## ディレクトリ

cpio コマンドを使ってファイルシステム間のコピー (概要), 433

i ノード, 352

/proc, 303

/tmp, 303

一時, クリア, 191

一時, クリア, 194

サイズ, 189, 190

情報の表示, 185, 186, 189, 190

未割り当てブロック, 355

ディレクトリデータブロック, 355

データブロック, 356, 374

## テープ

tar コマンドによるファイルの取り出し (方法), 436

記憶容量, 384, 446

サイズ, 384, 446

性質, 455

容量, 455

テープデバイス, 復元するタイプの決定, 406

テープデバイス (命名), 72

## テープドライブ

管理, 450

最大の SCSI テープドライブ台数, 447

巻き戻し, 447-448

テープドライブの管理, 450

デバイス, アクセス, 68

## デバイスドライバ

追加, 67

定義, 57

## デバイス名

テープデバイス名の決定, 406

バックアップ, 446-448

ファイルシステム名の確認, 406

## デフォルト

SunOS ファイルシステム, 306

/tmp 用のファイルシステム (TMPFS), 303

デフォルトの SunOS ファイルシステム, 306

## と

動的再構成, InfiniBand デバイス, 152

動的再構成 (概要), 74

登録, DAT 静的レジストリへのサービスプロバイダの登録 (方法), 163

登録解除, DAT 静的レジストリ内のサービスプロバイダの登録解除 (方法), 164

## トラブルシューティング

EFI ディスクラベル, 171

iSCSI 構成に関する問題 (方法), 281

PCI 構成に関する問題, 97

SCSI 構成に関する問題, 89

USB オーディオデバイスの問題, 142

USB 大容量ストレージデバイス, 137

失敗した SCSI 構成解除操作, 90

## 取り出し

tar コマンドによるテープからのファイルの (方法), 436

リムーバブルメディア (方法), 36

## 取り付け

PCI アダプタカード (方法), 95

SCSI デバイスの SCSI バスへの取り付け (方法), 85

## 取り外し

PCI アダプタカード (手順), 93

SCSI デバイス (方法), 88

## ね

ネットワークベースのファイルシステム, 302

## は

パーティション(スワップ)、定義, 331  
 バイト(iノードごとの数), 377-378  
 バス指向ディスクコントローラ, 71  
 バックアップ  
   完全バックアップおよび増分バックアップの定義, 383  
   準備(概要), 397-398  
   増分バックアップの記録, 453  
   タイプ, 383  
   デバイス名, 446-448  
   ファイルシステムの選択, 381  
   ファイルシステムのバックアップと復元  
     コマンド, 379  
     定義, 379  
   理由, 380  
 バックアップスケジュール  
   ガイドライン, 386, 387  
   サーバー用, 394-396  
   ダンプレベルの使用, 389  
   日単位増分、週単位累積バックアップ, 393  
   日単位累積、週単位増分バックアップ, 392  
   日単位累積、週単位累積バックアップ, 390  
   例, 390, 396  
 バックアップのスケジュール, 386  
 パラメータ(ファイルシステム), 375-378

## ひ

日単位個別バックアップ, 390  
 表示  
   HCAのカーネルIBクライアント(方法), 157  
   IB通信サービス(方法), 159  
   InfiniBandデバイス情報(方法), 153  
   PCIスロット構成情報(方法), 92  
   SATAデバイス(方法), 97  
   SCSIデバイスに関する情報, 81  
   USBデバイス情報(方法), 127  
   USBバス情報(方法), 144  
   システム構成情報, 59, 62  
   スワップ空間, 337-338  
   ディスクスライス情報(概要), 203  
   ディレクトリ情報, 185, 186, 189  
   ファイルシステム情報, 190

## 表示(続き)

ファイル情報  
   duコマンドを使用, 189  
   最新をリスト, 191  
   ファイルサイズ, 185, 187  
 ファイルのサイズ, 186  
 リムーバブルメディアのユーザー(方法), 36

## ふ

ファイル  
   /etc/default/fs, 311  
   /etc/dfs/fstypes, 311  
   /procディレクトリ内, 304  
   tarコマンドによるテープからの取り出し(方法), 436  
   共有, 309-310  
   サイズ, 185, 186, 189, 190  
   サイズ制限を超えるファイルを見つける, 188  
   サイズの表示, 186  
   削除  
     「削除」を参照  
   情報の表示  
     サイズ, 185, 186, 189, 190  
     リスト, 185, 186  
 ファイルシステム  
   4.3 Tahoe, 300  
   BSD Fat Fast, 300  
   CTFS, 304  
   DOS, 301  
   /export/home, 306  
   FDFS, 304  
   FIFOFS, 304  
   High Sierra, 301  
   ISO 9660, 301  
   MNTFS, 307  
   NAMEFS, 304  
   OBJFS, 304  
   /opt, 307  
   PCFS, 301  
   /proc, 307  
   PROCFS、(概要), 303-304  
   SHAREFS, 305  
   SPECFS, 305

## ファイルシステム (続き)

- SWAPFS, 305
- TMPFS, 303
- UFS, 300
- UNIX, 300
- /var, 306
- ZFS, 300
- アクセスしているすべてのプロセスの停止 (方法), 327
- カスタムパラメータ, 375–378
- 擬似, (概要), 302
- 共有, 309–310
- 検査と修復, 357
- サイズの検査, 351
- 作成 (概要)
  - ループバック (LOFS), 314
- 修正, 363
- 修復, 362
- シリンダグループの構造, 371–374
- 損傷, 349
- タイプ, 300
- タイプの調査, 311
- ディスクベース, 300
- デフォルトの SunOS ファイルシステム, 306
- ネットワークベース, 302
- バックアップする理由, 380
- バックアップ対象, 381
- 不整合の原因, 350
- プロセス, (概要), 303–304
- マウントテーブル, 308
- 利用可能にする (概要), 315
- ファイルシステムの修復, 362
- ファイルシステムの損傷, 349
- ファイルシステムの損傷の原因, 349
- ファイルシステムのタイプ, 300
- ファイルシステムの復元
  - 完全 (手段), 413
  - 準備 (概要), 405–406
  - 使用するテープの決定 (方法), 407
  - 全体 (例), 414
  - テープドライブのタイプ, 406
  - ルートと /usr (SPARC) (例), 418
  - ルートと /usr (x86) (例), 418
- ファイルシステムの不整合, 350

- ファイルシステム名, 406
- ファイルシステム用のカスタムパラメータ, 375–378
- ファイルの復元
  - 対話式で復元 (例), 410
  - 非対話式復元 (例), 411
- ブートブロック, 372
- フォーマット, `rmformat` を使用したフォーマット (方法), 28
- 不整合なファイルシステムの修正, 363
- 不正な「.」と「..」エントリ, 356
- 不正な i ノード番号, 356
- 不正なスーパーブロック, 364
- 不正なスーパーブロックの復元, 364
- 不正なブロック番号, 354
- 物理デバイス名
  - 定義, 69
- フラグメントサイズ, 376
- プロセスファイルシステム (PROCFS), 303–304
- ブロック
  - 空き, 374
  - 間接, 355
  - 重複, 354
  - 通常データ, 356
  - ディレクトリデータ, 355
  - データ, 374
  - 特殊ファイル i ノード, 352
  - ブート, 372
  - 不正, 354
  - 論理サイズ, 375
- ブロックディスクデバイスインタフェース
  - いつ使用するか, 70
  - 定義, 70
- フロッピーディスク
  - `rmformat` を使用したフォーマット (方法), 28
  - ファイルシステムの作成 (方法), 29
  - ボリューム管理を使用したロード (手順), 27

## へ

## 変更

- iSCSI のイニシエータとターゲットのパラメータ (方法), 279

## 変更 (続き)

- USB 大容量ストレージデバイス上のパーティションと PCFS ファイルシステムの作成 (方法), 130
- プライマリ USB オーディオデバイス (方法), 142

## ほ

## ホットプラグ

- cfgadm コマンドによる SCSI コントローラの切り離し (方法), 83
- PCI アダプタカードの取り付け (方法), 95
- PCI アダプタカードの取り外し (手順), 93
- PCI デバイス (概要), 91
- SCSI コントローラ上の同一 SCSI デバイスの交換 (方法), 86
- SCSI コントローラの構成 (方法), 82
- SCSI コントローラの接続 (方法), 84
- SCSI デバイスの SCSI バスへの取り付け (方法), 85
- SCSI デバイスの構成解除 (方法), 81
- SCSI デバイスの構成 (方法), 83
- SCSI デバイスの取り外し (方法), 88
- USB デバイスサブツリーの論理的な接続解除 (方法), 147
- USB デバイスの構成解除 (手順), 145
- USB デバイスの構成 (方法), 145
- USB デバイスの論理的な接続解除 (方法), 146
- USB デバイスの論理的な接続 (方法), 146 (概要), 74

## ま

## マウント

- /etc/vfstab によるファイルシステムの, 324
- NFS ファイルシステム, 324
- UFS ファイルシステム, 324
- USB 大容量ストレージデバイス (方法), 135
- ファイルシステムの自動マウント, 310–311
- リムーバブルメディア
  - 自動マウントと手動マウントの比較, 23
  - 自動マウントとの比較, 24

## マウント (続き)

- リモートリムーバブルメディアの手動マウント (例), 39
- マウント解除
  - USB 大容量ストレージデバイス (方法), 135
- マウントテーブル, 308
- マウントポイント定義, 307

## み

- 未割り当てディレクトリブロック, 355
- 未割り当ての i ノード, 353

## む

- 無効にする
  - リムーバブルメディアサービス, 22
  - リムーバブルメディアサービス (方法), 35
  - リムーバブルメディアの書き込み保護 (方法), 32

## め

- メディアの終わりの検出
  - ufsdump コマンド, 452
- メディアの末尾の検出
  - cpio コマンド, 433
- メモリー記憶域 (仮想)、定義, 331

## ゆ

## 有効にする

- uDAPL, 162
  - リムーバブルメディアサービス (方法), 35
  - リムーバブルメディアの書き込み保護 (方法), 32
- ユーザーのディスク容量の所有状況, 190



## リ

## リスト

- ファイルとディレクトリ, 185, 186, 191, 192
- リセット, USB デバイス (方法), 147
- リムーバブルメディア
  - DVD-RAM 上のファイルシステムの作成 (手順), 30
  - rmformat を使用したフロッピーディスクのフォーマット (方法), 28
  - アクセスしているプロセスの強制終了 (方法), 36
  - アクセス (方法), 35
  - 新しいリムーバブルメディアドライブの追加 (手順), 34
  - 書き込み保護を有効にする (方法), 32
  - 管理、利点, 21
  - サービス、無効にする, 22
  - 取り出し (方法), 36
  - 名前, 33
  - ファイルシステムの検査 (方法), 30
  - ファイルシステムの作成 (方法), 29
  - ほかのシステム上のメディアへのアクセス (例), 39
- マウント
  - 手動と自動の比較, 23, 24
  - メディアが使用中かどうかの調査 (方法), 36
  - リムーバブルメディアサービスを無効にするまたは有効にする (方法), 35
  - リムーバブルメディア上の不良ブロックの修復 (方法), 31
  - リムーバブルメディアのマウント (例), 39

## る

- ルート (/) ファイルシステムまたはディレクトリ, 306
- ループバックファイルシステム (LOFS), 作成 (概要), 314

## れ

- レベル 0 バックアップ, 389

## ろ

- ロード, フロッピーディスク (手順), 27
- ログ (ダンプの記録), 452–453
- 論理デバイス名
  - 定義, 69
  - ディスク, 69
  - テープ, 72
  - リムーバブルメディア, 72
- 論理ブロックサイズ, 375

## わ

- 割り当て済みの i ノード, 353