



Solstice DiskSuite 4.2.1 ユーザーズ ガイド

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303
U.S.A.

Part Number 806-3749-10
2000年3月

Copyright 2000 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303-4900 U.S.A. All rights reserved.

本製品およびそれに関連する文書は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社の書面による事前の許可なく、本製品および関連する文書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびに他の国における登録商標です。フォント技術を含む第三者のソフトウェアは、著作権により保護されており、提供者からライセンスを受けているものです。

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

本製品に含まれる HG 明朝 L と HG ゴシック B は、株式会社リコーがリコービイマジクス株式会社からライセンス供与されたタイプフェイスマスタをもとに作成されたものです。平成明朝体 W3 は、株式会社リコーが財団法人日本規格協会 文字フォント開発・普及センターからライセンス供与されたタイプフェイスマスタをもとに作成されたものです。また、HG 明朝 L と HG ゴシック B の補助漢字部分は、平成明朝体 W3 の補助漢字を使用しています。なお、フォントとして無断複製することは禁止されています。

Sun, Sun Microsystems, docs.sun.com, AnswerBook, AnswerBook2Open Windows, Solstice, Solstice AdminSuite, Solstice Backup, SPARCstorage, SunNet Manager, Online:DiskSuite, AutoClient, NFS, Solstice DiskSuite は、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。

サンロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャに基づくものです。

OPENLOOK、OpenBoot、JLE は、サン・マイクロシステムズ株式会社の登録商標です。

Wnn は、京都大学、株式会社アステック、オムロン株式会社で共同開発されたソフトウェアです。

Wnn6 は、オムロン株式会社で開発されたソフトウェアです。(Copyright OMRON Co., Ltd. 1999 All Rights Reserved.)

「ATOK」は、株式会社ジャストシステムの登録商標です。

「ATOK8」は株式会社ジャストシステムの著作物であり、「ATOK8」にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。

「ATOK Server/ATOK12」は、株式会社ジャストシステムの著作物であり、「ATOK Server/ATOK12」にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

本製品に含まれる郵便番号辞書 (7 桁/5 桁) は郵政省が公開したデータを元に制作された物です (一部データの加工を行なっています)。

本製品に含まれるフェイスマーク辞書は、株式会社ビレッジセンターの許諾のもと、同社が発行する『インターネット・パソコン通信フェイスマークガイド '98』に添付のものを使用しています。© 1997 ビレッジセンター

Unicode は、Unicode, Inc. の商標です。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザインタフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

DtComboBox ウィジェットと DtSpinBox ウィジェットのプログラムおよびドキュメントは、Interleaf, Inc. から提供されたものです。(© 1993 Interleaf, Inc.)

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

原典: *Solstice DiskSuite 4.2.1 User's Guide*

Part No: 806-3205-10

Revision A



目次

- はじめに 19
- Solstice DiskSuite 4.2.1 情報の検索 25**
- 1. 概要 39
 - はじめに 39
 - 構成の計画 40
 - ▼ 動作環境に合わせた DiskSuite の使用法 40
 - 初期状態データベースの複製の作成 42
 - 状態データベースの複製を作成するための予備情報 42
 - ▼ 初期状態データベースの複製を作成する方法について 43
 - 状態データベースの複製の前提条件 44
 - ▼ 初期状態データベースの複製を最初から作成する方法 (DiskSuite ツール) 45
 - ▼ 初期状態データベースの複製を最初から作成する方法 (コマンド行) 48
 - ▼ 既存の未使用スライスに初期状態データベースの複製を作成する方法 (DiskSuite ツール) 50
 - ▼ swap パーティションから領域を割り当て、状態データベースの複製を作成する方法 50
 - 次の作業 53
- 2. **DiskSuite オブジェクトの作成 55**
 - DiskSuite オブジェクト作成の概要 57

DiskSuite オブジェクトを作成するための前提条件	57
状態データベースの追加複製の作成	58
状態データベースの追加複製を作成するための予備情報	58
▼ 状態データベースの追加複製を作成する方法 (DiskSuite ツール)	59
▼ 状態データベースの追加複製を作成する方法 (コマンド行)	60
ストライプと連結の作成	61
ストライプと連結を作成するための予備情報	61
▼ ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)	63
▼ ストライプ方式メタデバイスの作成方法 (コマンド行)	66
▼ 連結の作成方法 (DiskSuite ツール)	67
▼ 連結の作成方法 (コマンド行)	69
ミラーの作成	70
ミラーを作成するための予備情報	70
▼ 未使用スライスからミラーを作成する方法 (DiskSuite ツール)	71
▼ 未使用スライスからミラーを作成する方法 (コマンド行)	73
▼ マウント解除可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)	74
▼ マウント解除可能なファイルシステムからミラーを作成する方法 (コマンド行)	78
▼ マウント解除不可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)	80
▼ マウント解除不可能なファイルシステムからミラーを作成する方法 (コマンド行)	83
▼ SPARC: ルート (/) からミラーを作成する方法 (コマンド行)	86
▼ x86: ルート (/) からミラーを作成する方法 (コマンド行)	88
RAID5 メタデバイスの作成	93
RAID5 メタデバイスを作成するための予備情報	93
▼ RAID5 メタデバイスの作成方法 (DiskSuite ツール)	94
▼ RAID5 メタデバイスの作成方法 (コマンド行)	97

トランスメタデバイスの作成 97

トランスメタデバイスを作成するための予備情報 98

- ▼ マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール) 99
- ▼ マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (コマンド行) 101
- ▼ マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール) 103
- ▼ マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (コマンド行) 105
- ▼ ミラーを使用してトランスメタデバイスを作成する方法 (DiskSuite ツール) 106
- ▼ ミラーを使用してトランスメタデバイスを作成する方法 (コマンド行) 106

ホットスペア集合の作成 107

ホットスペア集合を作成するための予備情報 108

- ▼ ホットスペア集合の作成方法 (DiskSuite ツール) 109
- ▼ ホットスペア集合の作成方法 (コマンド行) 110
- ▼ ホットスペア集合を関連付ける方法 (DiskSuite ツール) 111
- ▼ ホットスペア集合を関連付ける方法 (コマンド行) 112
- ▼ ホットスペア集合にホットスペアスライスを追加する方法 (DiskSuite ツール) 114
- ▼ ホットスペア集合にホットスペアスライスを追加する方法 (コマンド行) 115
- ▼ 関連付けられたホットスペア集合の変更方法 (DiskSuite ツール) 116
- ▼ 関連付けられたホットスペア集合の変更方法 (コマンド行) 117

ディスクセットの作成 118

ディスクセットを作成するための予備情報 118

- ▼ ディスクセットの作成方法 (コマンド行) 120
- ▼ ディスクセットにドライブを追加する方法 (コマンド行) 122

ディスクセット内に DiskSuite オブジェクトを作成 124

- ▼ ディスクセット内に DiskSuite オブジェクトを作成する方法 (DiskSuite ツール) 124

- ▼ ディスクセット内に DiskSuite オブジェクトを作成する方法 (コマンド行) 124
 - メタデバイス上にファイルシステムを作成 125
 - メタデバイス上にファイルシステムを作成するための予備情報 125
- ▼ メタデバイス上にファイルシステムを作成する方法 (ファイルシステムマネージャ) 125
- ▼ メタデバイス上にファイルシステムを作成する方法 (コマンド行) 126
- 3. **DiskSuite** オブジェクトの保守 129
 - DiskSuite オブジェクトの保守の概要 131
 - DiskSuite オブジェクトを保守するための前提条件 131
 - DiskSuite オブジェクトの状態チェック 132
 - DiskSuite ツールによる状態のチェック 132
 - コマンド行による状態のチェック 133
 - ▼ 状態データベースの複製の状態のチェック方法 (DiskSuite ツール) 134
 - ▼ 状態データベースの複製の状態のチェック方法 (コマンド行) 135
 - ▼ メタデバイスとホットスペア集合の状態のチェック方法 (DiskSuite ツール) 136
 - ▼ メタデバイスとホットスペア集合の状態のチェック方法 (コマンド行) 144
 - ▼ ディスクセットの状態のチェック方法 (コマンド行) 153
 - オブジェクトの交換と有効化 155
 - 状態データベースの複製を有効にするための予備情報 155
 - ▼ 状態データベースの複製を有効にする方法 (DiskSuite ツール) 156
 - ストライプや連結を再作成するための予備情報 156
 - ▼ スライス障害の後でストライプや連結を再作成する方法 (DiskSuite ツール) 157
 - ▼ スライス障害の後でストライプや連結を再作成する方法 (コマンド行) 158
 - ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要 160
 - ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報 163
 - ▼ サブミラー内のスライスを有効にする方法 (DiskSuite ツール) 164
 - ▼ サブミラー内のスライスを有効にする方法 (コマンド行) 165

- ▼ サブミラー内のスライスを交換する方法 (DiskSuite ツール) 166
- ▼ サブミラー内のスライスを交換する方法 (コマンド行) 167
- ▼ サブミラーの交換方法 (DiskSuite ツール) 168
- ▼ サブミラーの交換方法 (コマンド行) 169
- ▼ RAID5 メタデバイス内のスライスを有効にする方法 (DiskSuite ツール) 170
- ▼ RAID5 メタデバイス内のスライスを有効にする方法 (コマンド行) 171
- ▼ RAID5 スライスの交換方法 (DiskSuite ツール) 171
- ▼ RAID5 スライスの交換方法 (コマンド行) 172
 - ホットスペア集合を交換するための予備情報 174
- ▼ ホットスペア集合内のホットスペアを交換する方法 (DiskSuite ツール) 174
- ▼ ホットスペア集合内のホットスペアの交換方法 (コマンド行) 175
- ▼ ホットスペアを有効にする方法 (DiskSuite ツール) 176
- ▼ ホットスペアを有効にする方法 (コマンド行) 178
- トランスメタデバイスの障害の修復 178
 - ファイルシステムのパニック 178
 - トランスメタデバイスのエラー 179
- ▼ ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行) 179
- ▼ ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行) 179
- スライスとメタデバイスの拡張 183
 - スライスとメタデバイスを拡張するための予備情報 183
- ▼ 既存のデータを含むスライスの拡張方法 (DiskSuite ツール) 184
- ▼ 既存のデータを含むスライスの拡張方法 (コマンド行) 186
- ▼ 既存の連結方式の拡張方法 (DiskSuite ツール) 187
- ▼ 既存のストライプの拡張方法 (コマンド行) 189
- ▼ ミラーの拡張方法 (DiskSuite ツール) 191
- ▼ ミラーの拡張方法 (コマンド行) 192
- ▼ RAID5 メタデバイスの拡張方法 (DiskSuite ツール) 193

- ▼ RAID5 メタデバイスの拡張方法 (コマンド行) 194
- ▼ トランスメタデバイスの拡張方法 (DiskSuite ツール) 195
- ▼ トランスメタデバイスの拡張方法 (コマンド行) 196
- ファイルシステムの拡張 198
 - ファイルシステムを拡張するための予備情報 198
- ▼ ファイルシステムの拡張方法 (コマンド行) 198
- メタデバイスのリネーム 199
 - メタデバイスをリネームするための予備情報 200
- ▼ メタデバイスのリネーム方法 (DiskSuite ツール) 200
- ▼ メタデバイスのリネーム方法 (コマンド行) 201
- ミラーの操作 202
 - ミラーの予備情報 202
- ▼ ファイルシステムのミラー化を解除する方法 (DiskSuite ツール) 203
- ▼ ファイルシステムのミラー化を解除する方法 (コマンド行) 204
- ▼ マウント解除不可能なファイルシステムをミラー化解除する方法 (コマンド行) 206
- ▼ サブミラーの接続方法 (DiskSuite ツール) 208
- ▼ サブミラーの接続方法 (コマンド行) 209
- ▼ サブミラーを切断する方法 (DiskSuite ツール) 210
- ▼ サブミラーを切断する方法 (コマンド行) 211
- ▼ サブミラーのオフライン / オンライン設定 (DiskSuite ツール) 211
- ▼ サブミラーのオフライン / オンライン設定方法 (コマンド行) 213
- ディスクセットの操作 214
 - ディスクセットを操作するための予備情報 214
- ▼ ディスクセットの予約方法 (コマンド行) 214
- ▼ ディスクセットを解放する方法 (コマンド行) 216
- ▼ ディスクセットに新たなドライブを追加する方法 (コマンド行) 217
- ▼ ディスクセットに別のホストを追加する方法 (コマンド行) 219

- 4. **DiskSuite** オブジェクトの変更 221
 - DiskSuite オブジェクトの変更の概要 222
 - DiskSuite オブジェクトを変更するための前提条件 222
 - DiskSuite 構成の操作 222
 - DiskSuite 構成のための予備情報 223
 - ▼ DiskSuite 構成をディスクに保存する方法 (DiskSuite ツール) 223
 - ▼ ディスクから DiskSuite 構成を復元する方法 (DiskSuite ツール) 224
 - 状態データベースの複製の変更 226
 - 状態データベースの複製を変更するための予備情報 226
 - ▼ 状態データベースの複製の変更方法 (DiskSuite ツール) 226
 - ミラーオプションの変更 228
 - ミラーオプションを変更するための予備情報 228
 - ▼ ミラーのオプションの変更方法 (DiskSuite ツール) 229
 - ▼ ミラーのオプションの変更方法 (コマンド行) 231
 - ロギングデバイスをファイルシステム間で共有 232
 - ロギングデバイスを共有するための予備情報 232
 - ▼ ロギングデバイスをファイルシステム間で共有する方法 (DiskSuite ツール) 233
 - ▼ ロギングデバイスをファイルシステム間で共有する方法 (コマンド行) 235
- 5. **DiskSuite** オブジェクトの除去 237
 - DiskSuite オブジェクトの削除の概要 238
 - DiskSuite オブジェクトを除去するための前提条件 238
 - 状態データベースの複製を除去 239
 - 状態データベースの複製を除去するための予備情報 239
 - ▼ 状態データベースの複製の除去方法 (DiskSuite ツール) 239
 - ▼ 状態データベースの複製を除去する方法 (コマンド行) 240
 - ストライプと連結の除去 241
 - ストライプと連結を除去するための予備情報 241

- ▼ ストライプ、連結、またはストライプ方式の連結の除去方法 (DiskSuite ツール) 241
- ▼ ストライプ、連結、またはストライプ方式の連結の除去方法 (コマンド行) 242
- ミラーの除去 243
 - ミラーを除去するための予備情報 243
- ▼ ミラーとサブミラーを除去する方法 (DiskSuite ツール) 244
- ▼ ミラーとサブミラーの除去方法 (コマンド行) 245
- RAID5 メタデバイスの除去 246
 - RAID5 メタデバイスを除去するための予備情報 247
- ▼ RAID5 メタデバイスの除去方法 (DiskSuite ツール) 247
- ▼ RAID5 メタデバイスの除去方法 (コマンド行) 248
- トランスメタデバイスの除去 248
 - トランスメタデバイスを除去するための予備情報 248
- ▼ トランスメタデバイスの除去方法 (DiskSuite ツール) 249
- ▼ トランスメタデバイスの除去方法 (コマンド行) 250
- ▼ マウント解除不可能なファイルシステムからトランスメタデバイスを除去する方法 (DiskSuite ツール) 251
- ▼ マウント解除不可能なファイルシステムからトランスメタデバイスを除去する方法 (コマンド行) 253
- ホットスペアとホットスペア集合の除去 254
 - ホットスペアとホットスペア集合を除去するための予備情報 254
- ▼ ホットスペアをホットスペア集合から除去する方法 (DiskSuite ツール) 255
- ▼ ホットスペアをホットスペア集合から除去する方法 (コマンド行) 255
- ▼ ホットスペア集合の除去方法 (DiskSuite ツール) 256
- ▼ ホットスペア集合の除去方法 (コマンド行) 257
- ディスクセットの除去 258
 - ホストとディスクをディスクセットから除去するための予備情報 258
- ▼ ディスクセットからホストを除去する方法 (コマンド行) 259
- ▼ ディスクセットからドライブを除去する方法 (コマンド行) 260

- ▼ ディスクセットの除去方法 (コマンド行) 261
- 6. システムの管理 263
 - システム管理の概要 264
 - システムを管理するための前提条件 265
 - SPARCstorage Array のグラフィック表示を操作 265
 - SPARCstorage Array をグラフィカルに操作するための予備情報 266
 - ▼ 「ディスク表示」ウィンドウでオブジェクトを選択する方法 (DiskSuite ツール) 268
 - ▼ SPARCstorage Array ディスクの状態のチェック方法 (DiskSuite ツール) 269
 - ▼ SPARCstorage Array コントローラのファンとバッテリーの状態をチェックする方法 (DiskSuite ツール) 270
 - ▼ SPARCstorage Array コントローラの World Wide Name を表示する方法 (DiskSuite ツール) 271
 - SPARCstorage Array の管理 271
 - NVRAM を有効または無効にするための予備情報 272
 - ▼ コントローラ、トレイ、ディスク上で NV RAM を有効にする方法 (DiskSuite ツール) 273
 - ▼ コントローラ、トレイ、ディスクに対する同期書き込み用の NV RAM を有効にする方法 (DiskSuite ツール) 274
 - ▼ コントローラ、トレイ、ディスク上で NV RAM を無効にする方法 (DiskSuite ツール) 275
 - NVRAM データをパージおよびフラッシュするための予備情報 275
 - ▼ 未処理の書き込みを NV RAM からフラッシュする方法 (DiskSuite ツール) 276
 - ▼ 高速書き込みデータを NV RAM からパージする方法 (DiskSuite ツール) 276
 - ディスクを予約および解放するための予備情報 277
 - ▼ ディスクをホスト専用予約する方法 (DiskSuite ツール) 277
 - ▼ ホストによって予約されたディスクを解放する方法 (DiskSuite ツール) 278
 - ディスクを停止および起動するための予備情報 278
 - ▼ ディスクの停止方法 (DiskSuite ツール) 279
 - ▼ ディスクの起動方法 (DiskSuite ツール) 280

- パフォーマンスの監視とグラフ化 281
 - パフォーマンスの監視と解析 281
 - パフォーマンスの監視とグラフ化のための予備情報 282
 - ▼ デバイス統計情報の表示方法 (DiskSuite ツール) 282
 - ▼ デバイス統計情報をグラフ化する方法 (DiskSuite ツール) 283
 - ▼ 「統計情報グラフ」ウィンドウにデバイスを追加する方法 (DiskSuite ツール) 284
 - ▼ 「統計情報グラフ」ウィンドウからデバイスを除去する方法 (DiskSuite ツール) 284
- SunNet Manager と DiskSuite の統合 285
 - ▼ SunNet Manager を有効にして DiskSuite ツールを起動する方法 (SunNet Manager) 285
 - ▼ SunNet Manager から DiskSuite ツールを起動する方法 (SunNet Manager) 286
- SNMP 警告と DiskSuite の統合 286
 - ▼ DiskSuite の SNMP サポートの構成方法 (コマンド行) 287
- ストレージマネージャと DiskSuite の統合 289
 - ▼ DiskSuite を有効にしてストレージマネージャを起動する方法 (コマンド行) 289
 - ▼ ファイルシステムマネージャとディスクマネージャの起動方法 (DiskSuite ツール) 290
- 7. システムのトラブルシューティング 291
 - システムのトラブルシューティングの概要 292
 - システムのトラブルシューティングのための前提条件 292
 - トラブルシューティングを行う時の注意 292
 - DiskSuite 設定の回復 293
 - ▼ md.cf ファイルを使用して DiskSuite 設定を回復する方法 293
 - DiskSuite のデフォルトの変更 294
 - メタデバイスの予備情報 294
 - ▼ デフォルトのメタデバイス数を増やす方法 (コマンド行) 295
 - ディスクセットのための予備情報 296

- ▼ デフォルトディスクセットの数を増やす方法 (コマンド行) 296
 - 状態データベースの複製のための予備情報 297
- ▼ 大きな状態データベースの複製を追加する方法 (コマンド行) 297
 - エラーのチェック 298
- ▼ メタデバイス内のスライスエラーのチェックを自動化する方法 (コマンド行) 298
 - ブート障害 300
 - ブート障害のための予備情報 301
- ▼ 不適切な /etc/vfstab エントリからの回復方法 (コマンド行) 301
- ▼ 状態データベースの複製の不足からの回復方法 (コマンド行) 304
- ▼ ブートデバイス障害からの回復方法 (コマンド行) 307
- ▼ 代替ブートデバイスへのパスを記録する方法 (コマンド行) 312
- ▼ SPARC: 代替デバイスからのブート方法 (コマンド行) 313
- ▼ x86: 代替デバイスからのブート方法 (コマンド行) 314
- SCSI ディスクの交換 315
 - ▼ 障害の発生した SCSI ディスクの交換方法 (コマンド行) 315
- SPARCstorage Array の操作 319
 - インストール 319
 - デバイスの命名規則 319
 - SPARCstorage Array コンポーネントを交換するための予備情報 320
 - ▼ ミラー内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール) 321
 - ▼ RAID5 メタデバイス内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール) 327
 - ▼ SPARCstorage Array トレイの除去方法 (コマンド行) 327
 - ▼ SPARCstorage Array トレイの交換方法 328
 - ▼ SPARCstorage Array の電源断からの回復方法 (コマンド行) 329
 - ▼ ホスト間で SPARCstorage Array ディスクを移動する方法 (コマンド行) 331
 - SPARCstorage Array をシステムディスクとして使用 333

SPARCstorage Array をブート可能にする 333

▼ SPARCstorage Array ディスクをブートプロセスの初期段階で使用可能にする方法 333

8. ヒントとテクニック 335

状態データベースの複製とトランスメタデバイス 336

DiskSuite と Prestoserve 336

Prestoserve と互換性のある DiskSuite オブジェクト 336

Prestoserve と互換性のない DiskSuite オブジェクト 337

Prestoserve をミラーと使用することが好ましくない理由 337

Prestoserve をトランスメタデバイスと使用することが好ましくない理由 337

▼ Prestoserve と DiskSuite の設定方法 (コマンド行) 337

DiskSuite 構成のガイドライン 339

一般的なガイドライン 339

状態データベースの複製のガイドライン 340

ストライプ化のガイドライン 340

連結のガイドライン 341

連結方式ストライプのガイドライン 341

ミラーのガイドライン 341

RAID5 メタデバイスのガイドライン 343

UFS ロギングのガイドライン 344

ホットスベアのガイドライン 344

ファイルシステムのガイドライン 345

ラベル付きパーティション 345

機密保護対策 345

互換性 345

ディスクドライブの操作 346

▼ fmthard(1M) の使用方法 346

トランスメタデバイス (UFS ロギング) とディスクの制限 347

DiskSuite ツールの使用法	347
制約	347
メタデバイスエディタの使用法	348
スライス表示、ディスク表示、フィルタの使用法	348
▼ スライスサイズの選別方法 (DiskSuite ツール)	348
▼ 交換用スライスの選別方法 (DiskSuite ツール)	349
DiskSuite ツールの色とフォントの変更	351
▼ DiskSuite ツールのデフォルトカラーとフォントの変更方法	356
メタデバイスの命名規則	357
メタデバイス名の切り替え	358
メタデバイス名の切り替えを使用するための前提条件	359
名前切り替えを使用したメタデバイスの作成	360
▼ 既存の連結方式からミラーを作成する方法 (コマンド行)	360
▼ 既存のメタデバイスからトランスメタデバイスを作成する方法 (コマンド行)	361
名前切り替えを使用したメタデバイスの除去	362
▼ ファイルシステムをミラー化解除し、マウントデバイスを保持する方法 (コマンド行)	362
▼ トランスメタデバイスを除去し、マウントデバイスを保持する方法 (コマンド行)	363
ストライプの操作	365
▼ ストライプを別のコントローラに移動する方法 (コマンド行)	365
ミラーの操作	367
高度なミラー手法	367
▼ ミラー内のストライプの飛び越し値を変更する方法 (DiskSuite ツール)	367
▼ ミラーを使用してオンラインバックアップを行う方法 (コマンド行)	368
シングルユーザーモードへのブートがミラーに与える影響	371
ホットスベア	372
ディスクセットの操作	372

- ▼ ディスクセット用のディスクドライブデバイス名を設定する方法 (コマンド行) 372
- ▼ ディスクセット内で状態データベースの複製のサイズを変更する方法 (コマンド行) 374
- A. ストレージマネージャの使用法 377
 - 「コンテキストの読み込み」プロパティブック 378
 - ▼ 最初のコンテキストの読み込み 380
 - ▼ 別のコンテキストの読み込み 381
 - ファイルシステムマネージャの概要 382
 - ファイルシステムマネージャのメインウィンドウ 383
 - ファイルシステムマネージャのプロパティブック 385
 - ファイルシステム、マウント先、ディレクトリの管理 387
 - ▼ UFS ファイルシステムの作成 388
 - ▼ マウント先の作成 389
 - ▼ マウント先またはディレクトリのプロパティの修正 391
 - ▼ ファイルシステムのマウントとマウント解除 392
 - ▼ ディレクトリの共有と共有解除 393
 - ▼ スタティックなクライアントファイルシステムの表示 395
 - ▼ アクティブなサーバーファイルシステムの表示 396
 - ▼ スタティックなサーバーファイルシステムの表示 397
 - ▼ /etc/vfstab ファイルからのマウント先の削除 398
 - ディスクマネージャの概要 399
 - ディスクマネージャのメインウィンドウ 400
 - 複数のディスクの選択 402
 - ディスクマネージャ・プロパティブック 402
 - ディスクマネージャによるディスクの管理 404
 - ▼ 表示フィルタの指定 405
 - ▼ ボリュームラベルの指定 407
 - ▼ fdisk パーティションの変更 407

▼ スライス配置の変更 409

▼ ディスクの複写 410

索引 411

はじめに

Solstice™ DiskSuite™ 4.2.1 は、データとディスクドライブを管理するためのソフトウェア製品です。

Solstice DiskSuite 4.2.1 は、Solaris™ 8 が動作するすべての SPARC™ システムと、Solaris 8 が動作するすべての x86 システムで実行できます。

DiskSuite のディスクセット機能は、SPARC 版の Solaris でのみサポートされており、x86 システムではサポートされていません。

対象読者

本書は、システム管理者およびディスク記憶装置の管理担当者を対象としています。

内容の紹介

本書は、製品の機能別ではなく、類似した作業別に構成されています。

xxv ページの「Solstice DiskSuite 4.2.1 情報の検索」を利用すれば、一般に実行される作業を簡単にアクセスし、すぐに DiskSuite を使用できます。

本書の残りの部分は次のように構成されています。

第 1 章

新たに DiskSuite を構成するための準備とその実行方法について説明します。

第 2 章

DiskSuite オブジェクトの作成方法について説明します。

第 3 章

状態チェックやエラーの発生したスライスの交換など、DiskSuite オブジェクトの保守関連作業について説明します。

第 4 章

DiskSuite オブジェクトのパラメータの変更方法について説明します。

第 5 章

システムから DiskSuite オブジェクトを削除する方法について説明します。

第 6 章

DiskSuite ツールによる SPARCstorage Array の管理方法、パフォーマンス統計の取得方法、DiskSuite と他の Solstice 製品との統合方法について説明します。

第 7 章

DiskSuite 管理者が遭遇する可能性のある、さまざまなトラブルとその解決方法について説明します。

第 8 章

DiskSuite の可能性を最大限に引き出すための参考情報を提供します。

付録 A

ファイルシステムマネージャやディスクマネージャのグラフィカルツールを使用して、ファイルシステムの作成やディスクのパーティション分割などの作業を実行する方法について説明します。

関連マニュアル

DiskSuite やディスクの保守と構成に関連するマニュアルには、次のものがあります。

- 『Solstice AdminSuite 2.3 管理者ガイド』
- 『Solstice DiskSuite 4.2.1 リファレンス』
- 『Solaris のシステム管理 (第 1 巻)』

- 『Solaris のシステム管理 (第 2 巻)』
- 『SPARCstorage Array User's Guide』 および 『SPARCstorage Array Configuration Guide』

表記上の規則

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

表 P-1 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 system%
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。	system% su password:
AaBbCc123	変数を示します。実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、rm <i>filename</i> と入力します。
『 』	参照する書名を示します。	『コードマネージャ・ユーザーズガイド』を参照してください。
「 」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第 5 章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	sun% grep \^#define \ XV_VERSION_STRING'

ただし AnswerBook2™ では、ユーザーが入力する文字と画面上のコンピュータ出力は区別して表示されません。

コード例は次のように表示されます。

- C シェルプロンプト

```
system% command y|n [filename]
```

- Bourne シェルおよび Korn シェルのプロンプト

```
system$ command y|n [filename]
```

- スーパーユーザーのプロンプト

```
system# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します (例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ (-) は 2 つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

一般規則

- このマニュアルでは、英語環境での画面イメージを使っています。このため、実際に日本語環境で表示される画面イメージとこのマニュアルで使っている画面イメージが異なる場合があります。本文中で画面イメージを説明する場合には、日本語のメニュー、ボタン名などの項目名と英語の項目名が、適宜併記されています。
- このマニュアルでは、「IA」という用語は、Intel 32 ビットのプロセッサアーキテクチャを意味します。これには、Pentium、Pentium Pro、Pentium II、Pentium

II Xeon、Celeron、Pentium III、Pentium III Xeon の各プロセッサ、および AMD、Cyrix が提供する互換マイクロプロセッサチップが含まれます。

Solstice DiskSuite 4.2.1 情報の検索

『Solstice DiskSuite 4.2.1 ユーザーズガイド』では、Solstice DiskSuite 4.2.1 を使用してシステムを設定および保守する方法について説明します。

DiskSuite 製品をすぐにご使用になりたい場合は、この章の情報をご利用ください。この章は「ロードマップ」として構成されており、記憶容量の設定など、特定の DiskSuite 作業に関する情報を見つけるためのガイドとして役立ちます。この章では、DiskSuite の使用に必要なすべての作業について言及するのではなく、次に示す DiskSuite の概念に関連した作業と、その実行方法を記述した参照先をまとめています。

- 記憶容量
- 可用性
- 入出力パフォーマンス
- 管理
- トラブルシューティング

この章では、機能別に構成された作業の詳細な一覧も提供します。32ページの「DiskSuite 作業手順のまとめ」も参照してください。



注意 - DiskSuite を正しく使用しないと、データが破壊されることがあります。DiskSuite は、ディスクとディスク上のデータを管理するための強力な方法を提供します。DiskSuite をご使用になる前に、最低限の安全対策として、DiskSuite の動作について確認しておくことが必要です。

DiskSuite の概要

表 P-1 DiskSuite ロードマップ — 記憶容量

作業	説明	参照先
記憶容量の設定	スライスにまたがる記憶容量を作成するには、ストライプ方式メタデバイスまたは RAID5 メタデバイスを作成します。これによって、そのメタデバイスは、raw デバイスにアクセスするデータベースなどのアプリケーションやファイルシステムに使用できます。	63ページの「ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)」 94ページの「RAID5 メタデバイスの作成方法 (DiskSuite ツール)」
既存ファイルシステムの拡張	既存ファイルシステムの容量を増大させるには、連結を作成してから新たなスライスを追加します。	184ページの「既存のデータを含むスライスの拡張方法 (DiskSuite ツール)」
既存の連結方式の拡張	既存の連結方式を拡張するには、DiskSuite ツールを使用してストライプ方式の連結を作成します。	187ページの「既存の連結方式の拡張方法 (DiskSuite ツール)」
RAID5 メタデバイスの拡張	RAID5 メタデバイスの容量を拡張する必要がある場合、これに新たなスライスを連結できます。	193ページの「RAID5 メタデバイスの拡張方法 (DiskSuite ツール)」
UFS のサイズ増大	growfs(1M) コマンドを使用すれば、データへのアクセスを中断することなく、UFS をマウントしたままでサイズを拡張できます。	198ページの「ファイルシステムの拡張方法 (コマンド行)」

表 P-1 DiskSuite ロードマップ — 記憶容量 続く

作業	説明	参照先
ファイルシステムの作成	ストライプ、連結、ミラー、RAID5 メタデバイス、またはトランスメタデバイス上にファイルシステムを作成できます。	125ページの「メタデバイス上にファイルシステムを作成する方法 (ファイルシステムマネージャ)」

表 P-2 DiskSuite ロードマップ — 可用性

作業	説明	参照先
データの可用性を最大にする	データの可用性を最大限に高めたい場合、DiskSuite のミラー化機能を使用して、データのコピーを複数保持します。データ準備用の未使用スライスからミラーを作成したり、ルート (/) や /usr など既存のファイルシステムをミラー化することができます。	<p>71ページの「未使用スライスからミラーを作成する方法 (DiskSuite ツール)」</p> <p>74ページの「マウント解除可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)」</p> <p>80ページの「マウント解除不可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)」</p>
最小限のハードウェアコストでデータの可用性を増大させる	最小限のハードウェアでデータの可用性を増大させるには、DiskSuite の RAID5 メタデバイス機能を使用します。	94ページの「RAID5 メタデバイスの作成方法 (DiskSuite ツール)」
既存のミラーや RAID5 メタデバイスのデータの可用性を増大させる	ミラーや RAID5 メタデバイスのデータの可用性を増大させるには、ホットスペア集合を作成して、これにミラーのサブミラーや RAID5 メタデバイスを関連付けます。	109ページの「ホットスペア集合の作成方法 (DiskSuite ツール)」

表 P-2 DiskSuite ロードマップ — 可用性 続く

作業	説明	参照先
リブート後にファイルシステムの可用性を増大させる	リブート後にファイルシステム全体の可用性を増大させるには、システムに UFS ロギング (トランスメタデバイス) を追加します。ファイルシステムをロギングすれば、システムのリブート時の fsck(1M) の実行時間が減少します。	99ページの「マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」 103ページの「マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」

表 P-3 DiskSuite ロードマップ — 入出力パフォーマンス

作業	説明	参照先
ミラーパフォーマンスの改善	ミラーを作成する前に、状態データベースの複製を追加すると、ミラーのパフォーマンス改善に役立ちます。	59ページの「状態データベースの追加複製を作成する方法 (DiskSuite ツール)」
ミラーの読み書きオプションのチューニング	ミラーの読み書きオプションを指定すれば、特定の構成に対するパフォーマンスを改善することができます。	229ページの「ミラーのオプションの変更方法 (DiskSuite ツール)」
デバイスパフォーマンスの最適化	ストライプを作成すると、そのストライプを構成するデバイスのパフォーマンスが最適化されます。ストライプの飛び越し値は、ランダムアクセスや順次アクセスに対して最適化できます。	63ページの「ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)」
既存ストライプの範囲内でデバイスパフォーマンスを維持	ストライプ方式の連結では、領域を使い果たした連結やストライプを拡張します。ストライプの連結は、スライスの連結よりもパフォーマンスが優れています。	186ページの「既存のデータを含むスライスの拡張方法 (コマンド行)」

表 P-3 DiskSuite ロードマップ — 入出力パフォーマンス 続く

作業	説明	参照先
システムパフォーマンスの改善	UFS ログイン (トランスメタデバイス) を実施すれば、同期ディスク書き込みの数を減らすことによって、パフォーマンスの改善に役立ちます。	99ページの「マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」 103ページの「マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」

表 P-4 DiskSuite ロードマップ — 管理

作業	説明	参照先
大規模な構成における管理の簡易化	DiskSuite ツールのグラフィカルインタフェースを使用すれば、多数のディスクをすばやく簡単に操作できます。このインタフェースではドラッグ&ドロップ操作をサポートし、システムの物理表示と論理表示を提供します。	『Solstice DiskSuite 4.2.1 リファレンス』の第4章「DiskSuite ツール」
スライスやファイルシステムをグラフィカルに管理	DiskSuite には、Solstice ストレージマネージャのグラフィカルユーザーインタフェースが組み込まれています。これを使用してディスクとファイルシステムを管理し、ディスクのパーティション分割や UFS ファイルシステムの構築などの作業を実行します。	付録 A
SPARCstorage Array の管理	DiskSuite ツールのグラフィカルインタフェースを使用すれば、ディスクトレイの起動と停止や NVRAM の操作など、SPARCstorage Array 上で多数の保守作業を実施することができます。	第6章

表 P-4 DiskSuite ロードマップ — 管理 続く

作業	説明	参照先
メタデバイスの再構成	metarename(1M) コマンドを使用すれば、メタデバイスを簡単に管理できます。	201ページの「メタデバイスのリネーム方法 (コマンド行)」
Solstice DiskSuite 4.2.1 の最適化	DiskSuite のパフォーマンスは、適切な構成が行われているかどうか依存します。一度構成を行ったら、モニタリングとチューニングが必要です。	283ページの「デバイス統計情報をグラフ化する方法 (DiskSuite ツール)」
将来的な拡張計画	ファイルシステムは領域を使い果たす傾向があるため、ファイルシステムを連結することによって、将来の拡張計画をたてることができます。	184ページの「既存のデータを含むスライスの拡張方法 (DiskSuite ツール)」
DiskSuite モニタリングの自動化	DiskSuite の SNMP 機能を使用して、SunNet Manager と統合的に警報管理を行います。	285ページの「SunNet Manager を有効にして DiskSuite ツールを起動する方法 (SunNet Manager)」 287ページの「DiskSuite の SNMP サポートの構成方法 (コマンド行)」

表 P-5 DiskSuite ロードマップ — トラブルシューティング

作業	説明	参照先
<p>障害の発生したスライスの交換</p>	<p>メタデバイス内で障害の発生したスライスの交換が必要になることもあります。ストライプと連結の場合、新しいスライスを使用し、メタデバイスを削除してから再作成し、バックアップからデータを復元しなければなりません。ミラーや RAID5 メタデバイス内のスライスは、データを失うことなく、交換および再同期することができます。</p>	<p>157ページの「スライス障害の後でストライプや連結を再作成する方法 (DiskSuite ツール)」</p> <p>164ページの「サブミラー内のスライスを有効にする方法 (DiskSuite ツール)」</p> <p>170ページの「RAID5 メタデバイス内のスライスを有効にする方法 (DiskSuite ツール)」</p>
<p>ブート障害からの回復</p>	<p>システムをブートするとき、ハードウェア障害やオペレータのミスによって特別な障害が発生することがあります。</p>	<p>301ページの「不適切な /etc/vfstab エントリからの回復方法 (コマンド行)」</p> <p>304ページの「状態データベースの複製の不足からの回復方法 (コマンド行)」</p> <p>307ページの「ブートデバイス障害からの回復方法 (コマンド行)」</p>
<p>SSA ディスクの障害時の作業</p>	<p>多くの場合、SPARCstorage Array では DiskSuite を透過的に使用できます。ディスク障害の解決といった一部の作業では、手順が若干異なります。</p>	<p>321ページの「ミラー内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール)」</p>

表 P-5 DiskSuite ロードマップ — トラブルシューティング 続く

作業	説明	参照先
トランスメタデバイスの障害時の作業	トランスメタデバイスの障害は、マスターデバイスやロギングデバイスで発生することがあり、その原因としては、エラーデータやデバイスの障害があります。同一のロギングデバイスを共有するすべてのトランスメタデバイスを使用可能な状態に戻すには、これらを修復する必要があります。	179ページの「ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行)」 179ページの「ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行)」

DiskSuite 作業手順のまとめ

この節の情報は DiskSuite の機能別に構成されており、すべての DiskSuite 作業手順に対するクイックリファレンスとして使用できます。作業手順ごとに、コマンド行の同等機能を記載しています。

表 P-6 DiskSuite の機能と作業手順

機能	DiskSuite ツールのメニューまたは作業手順	コマンド
連結		
状態のチェック	オブジェクト -> 情報	metastat(1M)
作成	編集 -> 作成 -> 連結 (Concat/Stripe)	metainit(1M)
拡張	未使用スライスをオブジェクトにドラッグ	metattach(1M)
スライスに障害が発生した後の再作成	メタデバイスを削除してから再作成	metaclear(1M)、metainit(1M)
削除	編集 -> 削除	metaclear(1M)

表 P-6 DiskSuite の機能と作業手順 続く

機能	DiskSuite ツールのメニューまたは 作業手順	コマンド
構成		
メタデバイスのリネーム	「情報」ウィンドウを使用	metarename(1M)
未確定構成の復元	ファイル -> ファイルから復元	
すべての未確定操作の取り消し	編集 -> すべての変更を取消し	
未確定操作の取り消し	編集 -> 最後の変更を取消し	
未確定構成の保存	ファイル -> ファイルに保存	
DiskSuite ツール		
オブジェクトの拡大表示解除	オブジェクト -> 拡大表示解除	
オブジェクトの確定	オブジェクト -> 確定	metainit(1M)
コンフィグレーションログの表示	ブラウズ -> コンフィグレーションロ グ	
オブジェクトの削除	編集 -> 削除	metaclear(1M)
ディスクの表示	ブラウズ -> ディスク表示	
オブジェクト全体の表示	オブジェクト -> 拡大表示	
オブジェクトの複製	編集 -> 複製	
オブジェクトの評価	オブジェクト -> 評価	
終了	ファイル -> 終了	
メタデバイスの検索	ブラウズ -> 検索	
メタデバイスの表示	ブラウズ -> メタデバイス	
オブジェクト情報	オブジェクト -> 情報	metastat(1M)

表 P-6 DiskSuite の機能と作業手順 続く

機能	DiskSuite ツールのメニューまたは作業手順	コマンド
障害の表示	ブラウズ -> 障害リスト	
オブジェクトをキャンバスから移動	オブジェクト -> キャンバスから移動	
キャンバス上のオブジェクトの再編成	編集 -> キャンバスを整理	
スライスの表示	ブラウズ -> スライス	prtvtoc(1M)、 format(1M)
ホットスペア		
ホットスペア集合にスライスを追加	ホットスペア集合オブジェクトに使用可能なスライスをドラッグ	metahs(1M)
ホットスペア集合の関連付け	サブミラーまたは RAID5 オブジェクトにホットスペア集合オブジェクトをドラッグ	metaparam(1M)
関連付けられたホットスペア集合の変更	サブミラーまたは RAID5 オブジェクトに交換用のホットスペア集合オブジェクトをドラッグ	metaparam(1M)
状態のチェック	オブジェクト -> 情報	metahs(1M)
ホットスペア集合の作成	編集 -> 作成 -> スペア集合	metainit(1M)
ホットスペアの有効化	「ホットスペア集合情報」ウィンドウを使用	metahs(1M)
ホットスペアの削除	「ホットスペア集合情報」ウィンドウを使用	metahs(1M)
ホットスペア集合の削除	編集 -> 削除	metaparam(1M)、 metahs(1M)
コンポーネントの交換	ホットスペア集合オブジェクトに交換用のスライスをドラッグ	metahs(1M)
ホットスペア集合の表示	ブラウズ -> ホットスペア集合	metastat(1M)

表 P-6 DiskSuite の機能と作業手順 続く

機能	DiskSuite ツールのメニューまたは作業手順	コマンド
ミラー		
サブミラーの接続	ミラーオブジェクトにサブミラーをドラッグ	metattach(1M)
オプションの変更	「ミラー情報」ウィンドウを使用	metaparam(1M)
状態のチェック	オブジェクト -> 情報	metastat(1M)
作成	編集 -> 作成 -> ミラー	metainit(1M)
サブミラーの切断	「ミラー情報」ウィンドウを使用	metadetach(1M)
拡張	サブミラーに未使用スライスをドラッグ	metattach(1M)、 growfs(1M)
サブミラーのオフライン / オンライン設定	「ミラー情報」ウィンドウを使用	metaonline(1M)、 metaoffline(1M)
障害の発生したコンポーネントの交換	エラーの発生したスライスに交換用スライスをドラッグ	metareplace(1M)
削除	編集 -> 削除	metadetach(1M)、 metaclear(1M)
ファイルシステムのミラー化解除		
パフォーマンス監視		
デバイス統計情報の表示	オブジェクト -> 統計情報	iostat(1M)
グラフの表示	ブラウザ -> 統計情報グラフ	iostat(1M)
RAID5 メタデバイス		
状態のチェック	オブジェクト -> 情報	metastat(1M)
作成	編集 ->作成 -> RAID	metainit(1M)

表 P-6 DiskSuite の機能と作業手順 続く

機能	DiskSuite ツールのメニューまたは作業手順	コマンド
拡張	RAID5 オブジェクトに未使用スライスをドラッグ	metattach(1M)
障害の発生したコンポーネントの交換	エラーの発生したスライスに交換用のスライスをドラッグ	metareplace(1M)
削除	編集 -> 削除	metaclear(1M)
SPARCstorage Array		
ファンとバッテリーの状態チェック	「ディスク表示」ウィンドウを表示し、オブジェクト -> 情報	ssaadm(1M)
コントローラ情報の表示	「ディスク表示」ウィンドウを表示し、オブジェクト -> 情報	ssaadm(1M)
NVRAM の無効化	「ディスク表示」ウィンドウを表示し、オブジェクト -> 高速書き込み -> 無効	ssaadm(1M)
NVRAM の有効化	「ディスク表示」ウィンドウを表示し、オブジェクト -> 高速書き込み -> 有効	ssaadm(1M)
NVRAM の有効化 (同期書き込み)	「ディスク表示」ウィンドウを表示し、オブジェクト -> 高速書き込み -> 同期	ssaadm(1M)
NVRAM から未処理の書き込みをフラッシュ	「ディスク表示」ウィンドウを表示し、オブジェクト -> NVRAM の同期処理	ssaadm(1M)
NVRAM から高速書き込みデータをパージ	「ディスク表示」ウィンドウでオブジェクトを選択し、そのポップアップメニューから「NVRAM をパージ」を選択	ssaadm(1M)
ディスクの予約	「ディスク表示」ウィンドウを表示し、オブジェクト -> ディスクの予約	ssaadm(1M)
ディスクの解放	「ディスク表示」ウィンドウを表示し、オブジェクト -> ディスクの解放	ssaadm(1M)

表 P-6 DiskSuite の機能と作業手順 続く

機能	DiskSuite ツールのメニューまたは作業手順	コマンド
ディスク / トレイ / コントローラの起動	「ディスク表示」ウィンドウを表示し、オブジェクト -> ディスクの利用開始	ssaadm(1M)
ディスク / トレイ / コントローラの停止	「ディスク表示」ウィンドウを表示し、オブジェクト -> ディスクの利用停止	ssaadm(1M)
状態データベースの複製		
さらに追加	初期化されたメタデバイスの状態データベースオブジェクトにスライスをドラッグ	metadb(1M)
接続	「メタデバイスの状態データベース情報」ウィンドウを使用	metadb(1M)
状態のチェック	オブジェクト -> 情報	metadb(1M)
作成	初期化されていないメタデバイスの状態データベースオブジェクトにスライスをドラッグ	metadb(1M)
削除	「メタデバイスの状態データベース情報」ウィンドウを表示し、削除を選択	metadb(1M)
復元	「メタデバイスの状態データベース情報」ウィンドウを表示し、復元を選択	
表示と変更	「メタデバイスの状態データベース情報」ウィンドウを使用	
ストライプ		
状態のチェック	オブジェクト -> 情報	metastat(1M)
作成	編集 -> 作成 -> 連結 (Concat/Stripe)	metainit(1M)

表 P-6 DiskSuite の機能と作業手順 続く

機能	DiskSuite ツールのメニューまたは作業手順	コマンド
拡張	オブジェクトに未使用スライスをドラッグ	metattach(1M)
スライスに障害が発生した後の再作成	メタデバイスを削除し、再作成	metaclear(1M)、metainit(1M)
削除	編集 -> 削除	metaclear(1M)
トランスメタデバイス		
ロギングデバイスの接続	トランスデバイスのログ矩形にスライスまたはメタデバイスをドラッグ	metattach(1M)
状態のチェック	オブジェクト -> 情報	metastat(1M)
作成	編集 -> 作成 -> トランス	metainit(1M)
ロギングデバイスの切断	トランスメタデバイスからロギングデバイスをドラッグ	metadetach(1M)
削除	編集 -> 削除	metaclear(1M)
トランスメタデバイス間でのログ共有	編集 -> 作成 -> トランス ログの場合、すでに他のトランスメタデバイスによって使用されているものと同じスライスを使用。	metainit(1M)

概要

この章では、DiskSuite ソフトウェアの紹介を行い、新しい構成の立ち上げと実行に必要な手順について説明します。この章の作業を次に示します。

- 40ページの「構成の計画」
- 42ページの「初期状態データベースの複製の作成」

既存の DiskSuite を単にアップグレードするだけの場合、この章を読む必要はありません。代わりに、Solstice DiskSuite 4.2.1 へのアップグレードについては、『Solstice DiskSuite 4.2.1 ご使用にあたって』を参照してください。

はじめに

Solstice DiskSuite 4.2.1 を使用すれば、多数のディスクとそのディスク上のデータを管理することができます。

DiskSuite にはいろいろな利用法がありますが、その操作のほとんどは、次のテーマに関連しています。

- 記憶容量の増大
- データの可用性の向上

DiskSuite では、仮想ディスクを使用することによって、物理ディスクとそれに対応するデータを管理します。DiskSuite の場合、仮想ディスクはメタデバイスと呼ばれます。アプリケーションから見れば、メタデバイスと物理ディスクは機能的に同一です。DiskSuite では、メタデバイスに送られた入出力要求を、その配下にあるメンバーディスクへの入出力要求に変換します。

DiskSuite のメタデバイスは、スライス (ディスクパーティション) から構築されます。メタデバイスを構築する簡単な方法は、DiskSuite に付属しているグラフィカルユーザーインターフェースである、DiskSuite ツールを使用する方法です。DiskSuite ツールでは、使用できるすべてのスライスを表示できます。スライスをメタデバイスにすばやく割り当てるには、スライスをメタデバイスオブジェクトにドラッグしてください。

たとえば、記憶容量を増大させたい場合、DiskSuite を使用すれば、システムは多数の小さなスライスから構成される集合を 1 つの物理ディスクであるとみなすことができます。これらのスライスからメタデバイスを作成した後すぐに、これをあたかも任意の「実」ディスクのように使用することができます。

DiskSuite では、ミラーと RAID5 メタデバイスを使用することによって、データの可用性を高めることもできます。ミラーと RAID5 メタデバイスは、ディスクに障害が発生してもそこに格納されているデータが破壊されないよう、データを複写します。

メタデバイスの詳細、および DiskSuite のすべてのツールの詳細については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

構成の計画

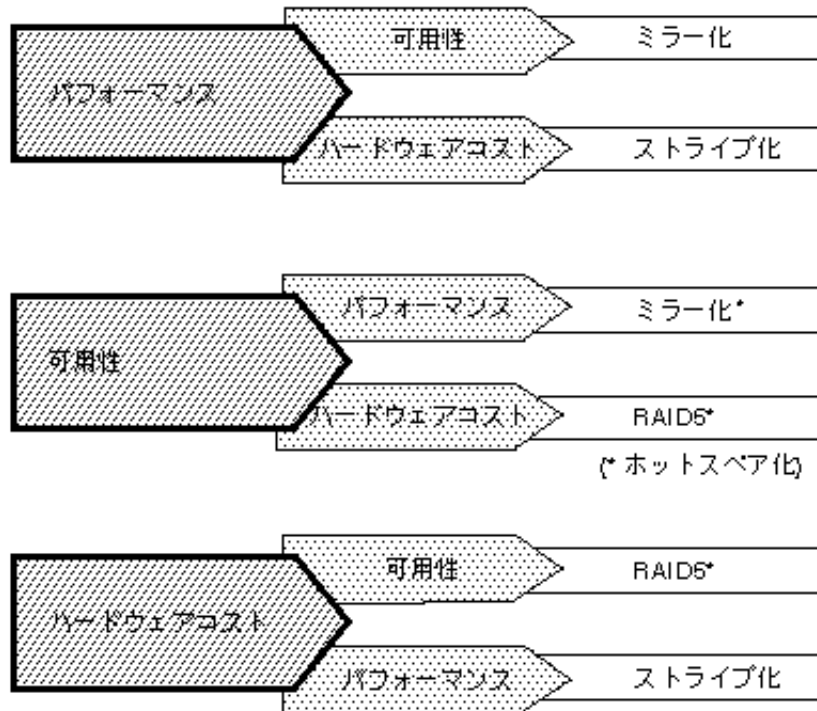
DiskSuite の機能とその機能をうまく活用する方法については、まず『*Solstice DiskSuite 4.2.1* リファレンス』をお読みください。

『*Solstice DiskSuite 4.2.1* リファレンス』の巻末にある用語集では、DiskSuite で使用している用語を説明しています。

▼ 動作環境に合わせた DiskSuite の使用法

動作環境に合わせた DiskSuite の使用法を決定するには、次の表と図を参考にしてください。

手順1	手順2	手順3
自分にとって最も重要な要因を選択する。	次に重要な要因を選択する。	ニーズに合う最適な DiskSuite 機能



例：ハードウェアコストの削減が最重要課題であり、その次が可用性である場合、RAID5 メタデバイスを使用すればその要求に最も適合します。

注 - DiskSuite の計画方法の詳細については、『Solstice DiskSuite 4.2.1 リファレンス』を参照してください。

初期状態データベースの複製の作成

この節では、システム構成にもとづいて初期状態データベースを作成する方法について説明します。

状態データベースの複製を作成するための予備情報

- 状態データベースの複製には、DiskSuite の構成と状態情報が格納されます。DiskSuite を使用するには、その前に状態データベースの複製を作成しなければなりません。
- 状態データベースの複製は、少なくとも3つ作成します。状態データベースの複製が任意の時点で半分以上使用できる場合、システムは実行を継続します。使用できる状態データベースの複製の数が半分を割ると、システムはパニック状態になります。使用できる状態データベースの複製の数が全体の半分以上でないとシステムはリブートせず、代わりに管理操作のシングルユーザーモードになります。
- 要求された状態データベースの複製の概算値は、システムのハードドライブの数にもとづいて、表 1-1 の概算値を使用します。必要に応じて調整してください。

表 1-1 要求された状態データベースの複製の概算値

ハードドライブの数	作成する状態データベースの複製の数
1	3つ、すべて1つのスライスで ¹
2～4	各ドライブに2つ
5以上	各ドライブに1つ

1. この構成では、システムのアキレス腱となる部分が生まれるため、好ましくない。

注 - 2 ドライブ構成では、各ドライブに状態データベースの複製を必ず2つ作成してください。その理由は、たとえば一方のスライスには状態データベースの複製を2つ作成し、他方のスライスには状態データベースの複製を1つしか作成しなかったとすると、状態データベースの複製を2つもつスライスに障害が発生した場合、残りのスライスには状態データベースの複製が1つしかないため、DiskSuite は機能できないからです。状態データベースの複製の操作については、『Solstice DiskSuite 4.2.1 リファレンス』を参照してください。

- 状態データベースの複製を作成できるのは、専用のスライス上、あるいはシンプルメタデバイス、RAID5 メタデバイス、トランスメタデバイスの一部として使用されるスライス上です。状態データベースの複製の位置を計画する際のガイドラインについては、『Solstice DiskSuite 4.2.1 リファレンス』を参照してください。
- 状態データベースの複製の最大数は50です。この値は、ディスクセットの一部である複製にも適用されます。

▼ 初期状態データベースの複製を作成する方法について

初期状態データベースの複製を作成するには、次の3つの方法のいずれかを使用します。これらの方法は、システム上のスライスの構成方法によって決まります。システムに使用できるスライスが存在しない場合、または既存スライスのパーティションを再分割できない場合には、DiskSuite ソフトウェアを使用することはできません。

1. 最初から始める：新しいシステムがある場合、状態データベースの複製を作成する最も簡単な方法としては、シンプルメタデバイス (ストライプ / 連結)、RAID5 メタデバイス、またはトランスメタデバイスの一部となるスライス上に、状態データベースの複製を置く方法があります。状態データベースの複製は、ルート (/)、swap、/usr、既存のファイルシステム、およびデータを含んだスライスの一部となることはできません。

新しいシステムの場合は、45ページの「初期状態データベースの複製を最初から作成する方法 (DiskSuite ツール)」を参照してください。

注 - 状態データベースの複製とメタデバイスを同じスライス上にあわせもつ場合、DiskSuite が状態データベースの複製を検出し、それに応じてメタデバイスのサイズと開始アドレスを調節します。この方法のメリットは、スライスのパーティション再分割など余分な操作が不要になり、スライス上の領域を状態データベースの複製のためにだけ確保することによる領域のムダを心配する必要がないことです。

2. 既存の未使用スライスを使用して始める : 既存の構成に未使用の専用スライスがある場合、状態データベースの複製をこれらのスライス上に置きます。

しかし、状態データベースの複製は比較的小さい (517K バイト、または 1034 セクター) ため、小さな状態データベースの複製を保持するために必要以上に大きなスライスを確保することはありません。スライスによっては、現在よりも小さくした方がよいこともあります。

既存の未使用スライスがある場合、50ページの「既存の未使用スライスに初期状態データベースの複製を作成する方法 (DiskSuite ツール)」を参照してください。

3. 未使用スライスのないところから始める : 既存の構成に使用できるスライスがない場合、状態データベースの複製用の領域は、swap など他のスライスから「奪取する」必要があります。この場合、50ページの「swap パーティションから領域を割り当て、状態データベースの複製を作成する方法」を参照してください。



注意 - ファイルシステムやデータベースによって使用されているスライスには、状態データベースの複製を置かないでください。そのスライス上にあるデータがすべて破壊されます。

状態データベースの複製の前提条件

このセクションで説明する作業の前提条件を次に示します。

- 現在のデータをすべてバックアップしている。
- Solstice DiskSuite 4.2.1 ソフトウェアをインストールしている (DiskSuiteのインストールについては、『Solstice DiskSuite 4.2.1 ご使用にあたって』を参照)。
- ルート (スーパーユーザー) 権限を持っている。

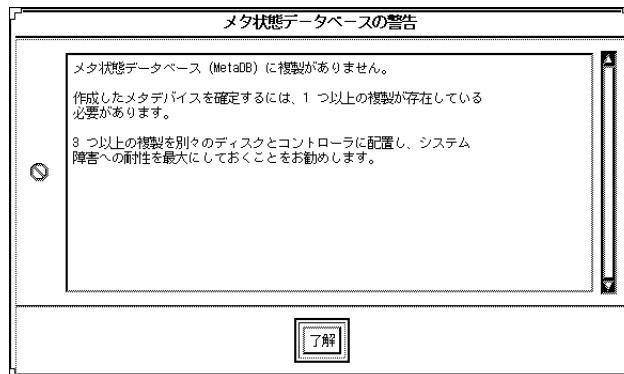
▼ 初期状態データベースの複製を最初から作成する方法 (DiskSuite ツール)

この作業では、DiskSuite ツールの 3 つのウィンドウを使用して構成を識別および表示し、状態データベースの複製をスライス上に置く方法を示します。

1. 前提条件 (44ページの「状態データベースの複製の前提条件」) を満たし、予備情報 (42ページの「状態データベースの複製を作成するための予備情報」) を読んでいることを確認する。
2. 表 1-1の表にもとづいて、必要となる状態データベースの複製の数を決定する。
3. スーパーユーザーになって **DiskSuite** ツールを起動し、構成の識別と表示を行う。

```
# metatool &
```

4. **DiskSuite** を初めて実行すると、次のメッセージが表示される。「了解」をクリックして続行する。



5. オブジェクトリストから、**MetaDB** オブジェクトをキャンバスにドラッグする。

オブジェクトのドラッグには、マウスの ADJUST ボタン (デフォルトでは中央ボタン) を使用します。

注 - この MetaDB オブジェクトは、ご使用の構成に含まれるすべての状態データベースの複製を表わします (含みます)。最初から始めるため、MetaDB オブジェクトには、状態データベースの複製を追加する必要があることを示すために、「重大な障害」ラベルが付いています。

6. 「ディスク表示」をクリックして、「ディスク表示」ウィンドウを表示する。
7. 「ディスク表示」ウィンドウの「表示」メニューから「すべてのコントローラを表示」を選択する。

これにより、記憶領域の構成 (コントローラ、ディスク、スライス) が表示されます。「ディスク表示」ウィンドウの「表示」メニューを使用して、50%、100%、200% の表示を設定できます。デフォルトは 100% です。
8. スライスをクリックして、「スライスブラウザ」ウィンドウを表示する。

このウィンドウは、スライスの状態と現在の使用状況を表示します (スライスの上にカーソルを置くと、「ディスク表示」ウィンドウのメッセージ行 (最下行) にこの情報が表示されます)。
9. 「ディスク表示」ウィンドウに表示されたディスクまたは「スライスブラウザ」ウィンドウから、使用できるスライスをキャンバス内の **MetaDB** オブジェクトにドラッグすることによって、**MetaDB** オブジェクトを構築 (状態データベースの複製を作成) する。

「ディスク表示」ウィンドウを開いたときに MetaDB オブジェクトが選択されている限り、スライスが MetaDB オブジェクトの中に置かれると、「ディスク表示」ウィンドウでは、状態データベースの複製が収められているスライスが着色されます (カラー画面では青色、モノクロ画面では黒色)。この機能は、たとえば、複数のコントローラにまたがって状態の複製を探す場合に役立ちます。

冗長性を確保するためには、コントローラ全体にわたって状態データベースの複製をバランスよく配分することが大切です。多数のディスクを備えた構成の場合、状態データベースの複製を各ディスクに置きます (「ディスク表示」ウィンドウの「フィルタの設定」ボタンをクリックすれば、現在使用されていないスライスを選別できます)。

10. [オプション] 複数の状態データベースの複製を同じスライスに追加する、
MetaDB オブジェクトの「情報」ウィンドウを表示する、「スライス」フィールドにスライス名を入力する、「複製」フィールドに追加する複製の数を入力する、「接続」をクリックしてから「閉じる」をクリックする。
この方法では、スライス上の状態データベースの複製の数を「変更」することはできません。最初に指定できるだけです。

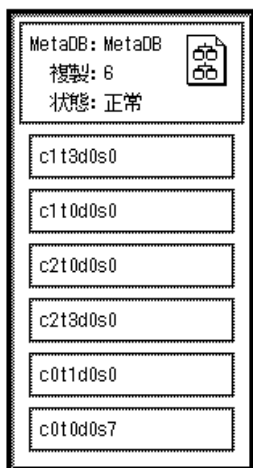
11. **MetaDB** オブジェクトを選択してから、「確定」をクリックする。

12. コンフィグレーションログを表示して、**MetaDB** オブジェクトが確定されたことを確認する。

「ブラウザ」メニューから「コンフィグレーションログ」を選択します。

例 - 6つの状態データベースの複製から成る確定済みの **MetaDB** オブジェクト

この例では、6つのスライスから成る確定済みの **MetaDB** オブジェクトを示します。それぞれのスライスには、1つの状態データベースの複製が含まれます。**MetaDB** オブジェクトの状態は「正常」であり、状態データベースの複製がとりあえず3つのコントローラに分散されていることを示します。



▼ 初期状態データベースの複製を最初から作成する方法 (コマンド行)

この作業は前のものと同等のものです。ここでは、コマンド行ユーティリティを使用して、状態データベースの複製を作成する方法を示します。

44ページの「状態データベースの複製の前提条件」の前提条件と 42ページの「状態データベースの複製を作成するための予備情報」の予備情報を確認した後で、`metadb(1M)` コマンドを使用して、状態データベースの複製を作成します。詳細については、`metadb(1M)` のマニュアルページを参照してください。

注・`metadb(1M)` コマンドを初めて実行すると、システムは、この宿主には状態データベースの複製が存在しないことを示す警告メッセージを表示します。このメッセージは無視してください。このメッセージが表示されるのは、状態データベースの複製を初めて作成するときだけです。

例 - 5つのディスクを持つシステムに初期状態データベースの複製を作成

```
# metadb -a -f c0t1d0s3 c1t1d0s3 c2t1d0s3 c3t1d0s3 c4t1d0s3
# metadb
```

	flags		first blk	block count	
	a	u	16	1034	/dev/dsk/c0t1d0s3
	a	u	16	1034	/dev/dsk/c1t1d0s3
	a	u	16	1034	/dev/dsk/c2t1d0s3
	a	u	16	1034	/dev/dsk/c3t1d0s3
	a	u	16	1034	/dev/dsk/c4t1d0s3

`-a` と `-f` のオプションは、初期状態データベースの複製を作成するために、一緒に使用されます。5つの初期状態データベースの複製が、5つの各スライスに1つずつ作成されます。状態データベースの複製をコントローラ全体に分散させることによって、メタデバイスのパフォーマンスと信頼性を向上させることができます。`metadb` コマンドは、`-a` フラグで示されるように、複製がアクティブであることをチェックします。

例 - 3 つのディスクを持つシステムに初期状態データベースの複製を作成

```
# metadb -a -f -c 2 c0t1d0s3 c1t1d0s3 c2t1d0s3
# metadb
```

	flags		first blk	block count	
a	u		16	1034	/dev/dsk/c0t1d0s3
a	u		1050	1034	/dev/dsk/c0t1d0s3
a	u		16	1034	/dev/dsk/c1t1d0s3
a	u		1050	1034	/dev/dsk/c1t1d0s3
a	u		16	1034	/dev/dsk/c2t1d0s3
a	u		1050	1034	/dev/dsk/c2t1d0s3

-a と -f のオプションは、初期状態データベースの複製を作成するために、一緒に使用されます。-c 2 オプションは、2つの状態データベースの複製を指定された各スライスに置き、合計で6つの複製を作成します。状態データベースの複製をコントローラ全体に分散させることによって、メタデバイスのパフォーマンスと信頼性を向上させることができます。metadb コマンドは、-a フラグで示されるように、複製がアクティブであることをチェックします。

例 - 1 つのディスクだけを持つシステムに初期状態データベースの複製を作成

```
# metadb -a -f -c 3 c0t0d0s3
# metadb
```

	flags		first blk	block count	
a	u		16	1034	/dev/dsk/c0t0d0s3
a	u		1050	1034	/dev/dsk/c0t0d0s3
a	u		2084	1034	/dev/dsk/c0t0d0s3

この例に示すシステムは、1つのディスクだけで構成されています (DiskSuite は、その UFS ロギング機能を実現するためにインストールされています)。-a と -f のオプションは、初期状態データベースの複製を作成するために、一緒に使用されます。-c 3 オプションは、システムの唯一のディスクの同じスライスに、状態データベースの複製を3つ作成します。metadb コマンドは、-a フラグで示されるように、複製がアクティブであることをチェックします。



注意 - この1ディスク構成では、システムのアキレス腱となる部分が生まれるため、他に選択肢のない場合にのみ使用します。

▼ 既存の未使用スライスに初期状態データベースの複製を作成する方法 (DiskSuite ツール)

この作業では、状態データベースの複製を保持するスライスのサイズを変更します。たいいていのスライスでは、状態データベースの複製に必要なとされるサイズよりも、はるかに大きいことが多いようです。大量のディスク領域を浪費しないために、1つのスライスから領域を取り出し、それを未使用スライス名や新しいスライスに割り当てることができます。

1. 前提条件 (44ページの「状態データベースの複製の前提条件」) を満たし、予備情報 (42ページの「状態データベースの複製を作成するための予備情報」) を読んでいることを確認する。また、構成の中に含まれるスライスと現在使用されていないスライスの名前を確認する。
2. 表 1-1の表にもとづいて、必要とする状態データベースの複製の数を決定する。
3. 新しい「小さな」スライスを少なくとも 1 つ作成する。

`format(1M)` コマンド、`fmthard(1M)` コマンド、またはストレージマネージャを使用できます。ディスクとスライスの管理の詳細については、『*Solaris* のシステム管理 (第 1 巻)』を参照してください。

注 - メタデバイスの状態データベース用に作成しているスライスに割り当てられたディスク領域は、他のスライスと共有されていないことを確認してください。

4. リブートする。
5. 45ページの「初期状態データベースの複製を最初から作成する方法 (DiskSuite ツール)」の 45ページの手順 3 ~ 47ページの手順 12 に従う。

▼ swap パーティションから領域を割り当て、状態データベースの複製を作成する方法

この作業は、状態データベースの複製を置くための未使用スライスがない場合に実行します。たとえ未使用スライスがない場合でも、`swap` パーティションの最後尾の領域を未使用スライス名に割り当てることによって、新しいスライス名を作成できます。



注意 - 新しいスライスに対して領域を再割り当てする手法は、`swap` パーティションの場合にのみ実行できます。この操作をファイルシステムで行うと、そのファイルシステムのデータが失われます。

1. スーパーユーザーになってシステムを停止し、シングルユーザーモードでブートする。

```
# halt
...
ok boot -s
...
INIT: SINGLE USER MODE

Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance):<パスワードを入力>
Entering System Maintenance Mode

Mar 12 16:52:38 su: 'su root' succeeded for root on /dev/syscon
Sun Microsystems Inc.   SunOS 5.5           Generic November 1995
```

2. `swap -l` コマンドを使用して、システムのスワップ領域を識別する。
`swapfile` カラムの下には、起動されたスワップデバイスやファイルが一覧表示されます。

```
# swap -l
swapfile          dev  swaplo blocks  free
/dev/dsk/c0t2d0s1 32,17    8 205624 192704
```

3. スワップをオフにし、`swap` に使用されているスライスが存在しないことを確認する。

```
# swap -d /dev/dsk/<スライス名>
# swap -l
```

4. **format(1M)** コマンド、**fmthard(1M)** コマンド、またはストレージマネージャを使用してディスクのパーティション再分割を行なってから、システムをリブートする。

swap スライスからシリンダを削除し、現在使用されていない新しいスライス名にそれらのシリンダを追加します。

5. システムに再びログインし、**prtvtoc(1M)** コマンドを使用することによって、この新しいスライスが存在することを確認する。

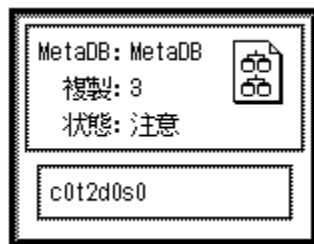
```
# prtvtoc /dev/rdisk/<スライス名>
```

6. 45ページの「初期状態データベースの複製を最初から作成する方法 (DiskSuite ツール)」の45ページの手順3～47ページの手順12に従う。

「メタデバイス状態データベース情報」ウィンドウを使用して、**swap** のパーティション再分割によって作成したスライスに、状態データベースの複製を3つ作成します。

例 - 3つの状態データベースの複製から成る、確定済み MetaDB オブジェクト

この例では、同じスライス上に状態データベースの複製を3つ備えた、確定済み MetaDB オブジェクトを示します。このオブジェクトは、1つのスライスしか含まないにもかかわらず、3つの状態データベースの複製を示していることに注目してください。これらの複製は、少なくとも3つのコントローラに分散されていないため、MetaDB オブジェクトの状態は「注意」です。



次の作業

以上により、DiskSuite を使用して構成を管理する準備ができました。第 2 章の作業を使用して、メタデバイスやホットスペア集合などの DiskSuite オブジェクトを作成します。

DiskSuite オブジェクトの作成

この章では、DiskSuite グラフィカルユーザーインターフェースとコマンド行ユーティリティの両方を使用して、DiskSuite オブジェクトを作成する方法について説明します。

DiskSuite ツールを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 59ページの「状態データベースの追加複製を作成する方法 (DiskSuite ツール)」
- 63ページの「ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)」
- 67ページの「連結の作成方法 (DiskSuite ツール)」
- 71ページの「未使用スライスからミラーを作成する方法 (DiskSuite ツール)」
- 74ページの「マウント解除可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)」
- 80ページの「マウント解除不可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)」
- 94ページの「RAID5 メタデバイスの作成方法 (DiskSuite ツール)」
- 99ページの「マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」
- 103ページの「マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」
- 106ページの「ミラーを使用してトランスメタデバイスを作成する方法 (DiskSuite ツール)」
- 109ページの「ホットスペア集合の作成方法 (DiskSuite ツール)」

- 111ページの「ホットスペア集合を関連付ける方法 (DiskSuite ツール)」
- 114ページの「ホットスペア集合にホットスペアスライスを追加する方法 (DiskSuite ツール)」
- 116ページの「関連付けられたホットスペア集合の変更方法 (DiskSuite ツール)」
- 124ページの「ディスクセット内に DiskSuite オブジェクトを作成する方法 (DiskSuite ツール)」
- 125ページの「メタデバイス上にファイルシステムを作成する方法 (ファイルシステムマネージャ)」

コマンド行インタフェースを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 60ページの「状態データベースの追加複製を作成する方法 (コマンド行)」
- 66ページの「ストライプ方式メタデバイスの作成方法 (コマンド行)」
- 69ページの「連結の作成方法 (コマンド行)」
- 73ページの「未使用スライスからミラーを作成する方法 (コマンド行)」
- 78ページの「マウント解除可能なファイルシステムからミラーを作成する方法 (コマンド行)」
- 83ページの「マウント解除不可能なファイルシステムからミラーを作成する方法 (コマンド行)」
- 86ページの「SPARC: ルート (/) からミラーを作成する方法 (コマンド行)」
- 88ページの「x86: ルート (/) からミラーを作成する方法 (コマンド行)」
- 97ページの「RAID5 メタデバイスの作成方法 (コマンド行)」
- 101ページの「マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (コマンド行)」
- 105ページの「マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (コマンド行)」
- 106ページの「ミラーを使用してトランスメタデバイスを作成する方法 (コマンド行)」
- 110ページの「ホットスペア集合の作成方法 (コマンド行)」
- 115ページの「ホットスペア集合にホットスペアスライスを追加する方法 (コマンド行)」
- 117ページの「関連付けられたホットスペア集合の変更方法 (コマンド行)」

- 120ページの「ディスクセットの作成方法 (コマンド行)」
- 122ページの「ディスクセットにドライブを追加する方法 (コマンド行)」
- 124ページの「ディスクセット内に DiskSuite オブジェクトを作成する方法 (コマンド行)」
- 126ページの「メタデバイス上にファイルシステムを作成する方法 (コマンド行)」

DiskSuite オブジェクト作成の概要

DiskSuite オブジェクトを作成する場合、物理スライスを論理 DiskSuite 名に割り当てます。この章に含まれる手順を使用して作成できる DiskSuite オブジェクトには、次のものが含まれます。

- 状態データベースの複製の追加
- メタデバイス (ストライプ、連結、ミラー、RAID5 メタデバイス、トランスメタデバイス)
- ホットスペア集合
- ディスクセット

DiskSuite の概要については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

注 - メタデバイスに名前を付けるためのヒントについては、357ページの「メタデバイスの命名規則」を参照してください。

DiskSuite オブジェクトを作成するための前提条件

この章に含まれる手順を実行するための前提条件を次に示します。

- 初期状態データベースの複製を作成してある。
まだ作成していない場合は、第1章を参照して作成してください。
- DiskSuite で使用できるスライスを識別する。
必要に応じて `format(1M)`、`fmthard(1M)`、またはディスクマネージャを使用して、既存ディスクのパーティションを再分割してください。
- ルート (スーパーユーザー) 権限を持っている。

- 現在のデータをすべてバックアップしている。
- グラフィカルユーザーインターフェースを使用して作成する場合は、DiskSuite ツールを起動する。

「ローカル」メタデバイス (ディスクセット構成に含まれないメタデバイス) を操作するには、次のように入力する。

```
# metatool &
```

ディスクセットに含まれるメタデバイスを操作するには、自分がディスクセットの所有者であることを確認し、次のように入力する。

```
# metatool -s <ディスクセット名> &
```

状態データベースの追加複製の作成

この節では、実行中のシステムに状態データベースの追加複製を作成する方法について説明します。

状態データベースの追加複製を作成するための予備情報

- 状態データベースの複製を作成できるのは、専用のスライス上、あるいはシンプルメタデバイス、ミラー、RAID5 メタデバイス、またはトランスメタデバイスの一部として使用されるスライス上です。状態データベースの複製の位置を計画する際のガイドラインについては、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。
- 状態データベースの複製は、いつでもシステムに追加できます。状態データベースの複製を追加すると、DiskSuite の可用性を確保するのに役立ちます。
- ミラーを作成する前に状態データベースの複製を追加すると、ミラーのパフォーマンスを向上させることができます。一般に、システムにミラーを1つ追加するごとに複製を1つ追加します。
- 状態データベースの複製の最大数は50です。この値は、ディスクセットの一部となる複製にも適用されます。

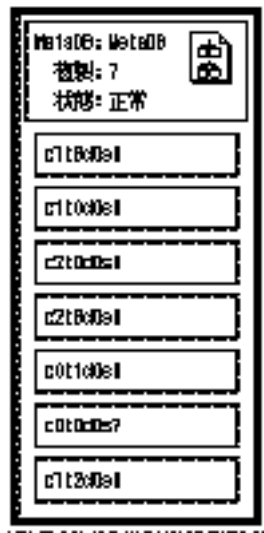
▼ 状態データベースの追加複製を作成する方法 (DiskSuite ツール)

初期状態データベースの複製が作成された後で、必要に応じて、状態データベースの複製を追加作成できます。

1. **前提条件** (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、**予備情報** (58ページの「状態データベースの追加複製を作成するための予備情報」) を読んでいることを確認する。
2. オブジェクトリストの **MetaDB** オブジェクトをダブルクリックする。
キャンバスに **MetaDB** オブジェクトが表示されます。
3. スライスをクリックして、「スライスブラウザ」ウィンドウを開く。
4. 状態データベースの複製の追加用スライスを選択する。**MetaDB** オブジェクトの先頭の矩形にスライスをドラッグする。
複数のスライスを選択するには、Control キーを押しながらクリックします。
5. **[オプション]** 同じスライスに状態データベースの複製を複数追加する、**MetaDB** オブジェクトの「情報」ウィンドウを表示する、「スライス」フィールドにスライス名を入力する、「複製」フィールドに追加する複製の数を入力する、「接続」をクリックしてから、「閉じる」をクリックする。
6. この方法では、スライス上の状態データベースの複製の数を「変更」することはできません。最初に指定できるだけです。**MetaDB** オブジェクトが選択されていることを確認し、「確定」をクリックする。
7. コンフィグレーションログを表示して、**MetaDB** オブジェクトが確定されたことを確認する。

例 - 確定済みの**MetaDB** オブジェクト

この例では、**MetaDB** オブジェクトに追加された、新しい状態データベースの複製を示します。



状態データベースの複製を含む追加のスライスである c1t2d0s0 は、確定済みの MetaDB オブジェクトのスライスリストの下に表示されます。

▼ 状態データベースの追加複製を作成する方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 58ページの「状態データベースの追加複製を作成するための予備情報」の予備情報をチェックした後で、metadb(1M) コマンドを使用して、状態データベースの複製を作成します。詳細については、metadb(1M) のマニュアルページを参照してください。

例 - 状態データベースの複製を追加

```
# metadb -a c0t2d0s0
# metadb
      flags          first blk      block count
...
      a              u              16          1034          /dev/dsk/c0t2d0s0
```

-a オプションは、状態データベースの複製をシステムに追加します。metadb コマンドは、-a フラグで示されるように、複製がアクティブであることをチェックします。

例 - 同一のスライスに2つの状態データベースの複製を追加

```
# metadb -a -c 2 c0t2d0s0
# metadb
      flags          first blk      block count
...
  a      u           16             1034      /dev/dsk/c0t2d0s0
  a      u          1050            1034      /dev/dsk/c0t2d0s0
```

-a オプションは、状態データベースの複製をシステムに追加します。-c 2 オプションは、2つの複製を指定されたスライスに格納します。metadb コマンドは、-a フラグで示されるように、複製がアクティブであることをチェックします。

ストライプと連結の作成

この節では、ストライプと連結の作成方法について説明します。既存のストライプに連結する(スライスを追加する)には、187ページの「既存の連結方式の拡張方法(DiskSuite ツール)」を参照してください。

ストライプと連結を作成するための予備情報

- DiskSuite の用語では、ストライプと連結は、スライスから構成されるという点において、「シンプルメタデバイス」です。ストライプと連結のいずれを使用しても、ディスクの記憶容量を拡張できます。両方とも、直接使用したり、ミラーやトランスメタデバイス用の構築ブロックとして使用することが可能です。
- ストライプと連結を作成するためには、その前にシステムには状態データベースの複製が少なくとも3つあることが必要です(第1章を参照)。
- 次に示すものを除く任意のファイルシステムに対して、いろいろな様式のストライプと連結(複数のスライスから構成されるストライプや連結)を使用できます。
 - ルート (/)
 - swap (複数の単一スライス swap メタデバイスを作成できる)
 - /usr
 - /var
 - /opt

- Solaris のインストール中やアップグレード中にアクセスされる、その他のファイルシステム
- 同じ物理ディスク上にあるスライスは、ストライプ化しないでください。ストライプ化しても同時アクセスができないので、パフォーマンスが低下します。
- 可能ならば、ディスクの幾何学的配置が同じディスクからメタデバイスを作成してください。その歴史的な理由としては、UFS はディスクの幾何学的配置にもとづいてディスクブロックを使用することが挙げられます。今日では、この問題はパフォーマンスの向上に関連します。幾何学的配置の異なるディスクで構成されたストライプは、その最も遅いディスクと同じ速さになります。
- 可能な限り、連結やストライプのスライスを、さまざまなコントローラとバスに分散させてください。それぞれが異なるコントローラに置かれているストライプを使用すれば、同時に実行できる読み書きの数が増大します。
- 既存のファイルシステムやデータからは、ストライプ方式メタデバイスを作成しないでください。データが破壊されてしまいます。その代わりに、連結を使用してください (既存のデータからストライプ方式メタデバイスを作成することもできますが、その場合データをダンプしてメタデバイスに復元する必要があります)。
- サイズの異なるスライスを使用して、ストライプ化しないでください。ディスク領域が無駄に消費されてしまいます。必要に応じて、未使用部分を他のスライスに割り当てることができます。未使用ディスク領域を他の使用可能なスライス名に割り当てするには、スライスに対してパーティションの再設定 (切り直し) を行う必要があります (`format(1M)`、`fmthard(1M)`、またはストレージマネージャを使用)。

注 - DiskSuite ツールでは、連結方式オブジェクトを使用して、単純連結方式メタデバイスとストライプ方式メタデバイスの両方を表現しています。したがって、これらを一目で区別するためには、連結方式オブジェクト内の矩形パターンを調査するしかありません。

▼ ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)



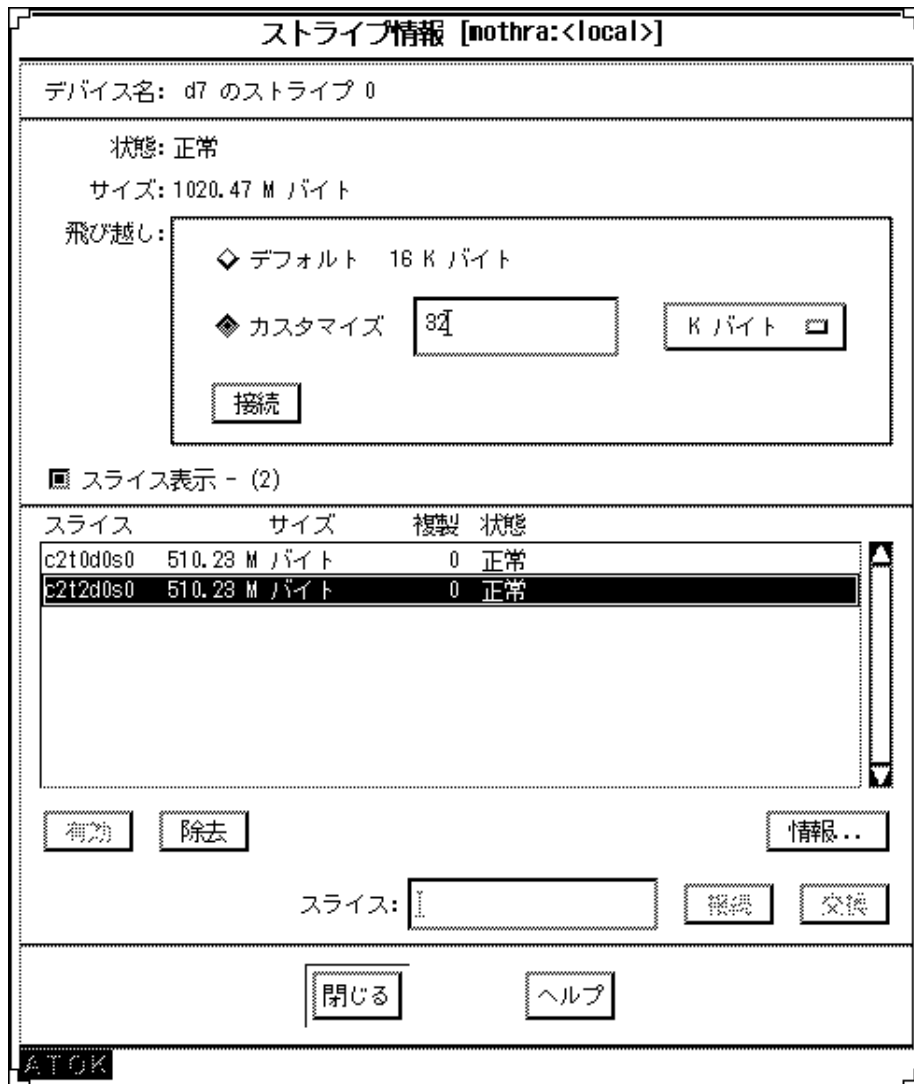
注意 - 既存のファイルシステムやデータからは、ストライプ方式メタデバイスを作成しないでください。これを行なった場合、データが破壊されます。既存のデータからストライプ方式メタデバイスを作成するには、データをダンプしてメタデバイスに復元する必要があります。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (61ページの「ストライプと連結を作成するための予備情報」) を読んでいることを確認する。
2. 連結方式テンプレートをクリックし、未割り当てで未確定の連結方式オブジェクトをキャンバスに表示する。
メタデバイス名は自動的に割り当てられます。
3. [オプション] デフォルトのメタデバイス名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 新しいメタデバイス名を「デバイス名」フィールドに入力し、「接続」ボタンをクリックする。
 - c. 「閉じる」ボタンをクリックする。
4. スライスをクリックして、「スライスブラウザ」ウィンドウを開く。
5. **Control** キーを押しながらクリックして複数のスライスを選択し、連結方式オブジェクトにドラッグする。
6. 「連結方式の指定」ダイアログボックスで「ストライプ」をクリックする。

注 - スライスを1つずつドラッグする場合、「連結方式の指定」ダイアログボックスは表示されないため、連結方式オブジェクトのどこにドロップするかについては注意が必要です。ストライプを作成するには、「dxのストライプ0」というラベルの付いた矩形にスライスをドロップします。

7. [オプション] 飛び越し値をデフォルトの **16K** バイトから変更する。

「dxのストライプ0」が表示されている矩形の内部にカーソルを置いて、このストライプ方式メタデバイス用の「ストライプ情報」ウィンドウを表示します。



- 飛び越し値を変更する。「カスタマイズ」をクリックして、「カスタマイズ」ボタンの横のフィールドに新しい値を入力する。

単位には、K バイト、M バイト、またはセクターを使用できます。単位を設定するには、横のプルダウンメニューから任意の単位を選択します。「接続」をクリックして値を設定し、「閉じる」をクリックする。

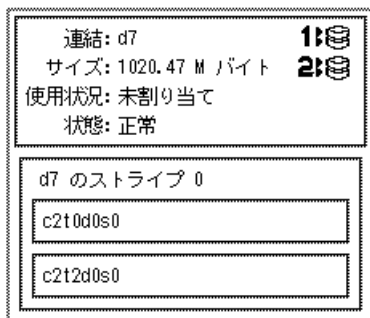
飛び越し値の設定については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

注 - メタデバイスを確定した後では、飛び越し値を変更できません。

- 連結方式オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
- コンフィグレーションログを表示して、ストライプ方式メタデバイスが確定されたことを確認する。

例 - 確定済みのストライプ

この例では、2つのスライスから成る、新しく作成されたストライプを示します。ストライプオブジェクトは、1つの矩形内の互いの先頭にスライスを表示します。68ページの「例 - 確定済みの連結オブジェクト」の連結メタデバイスの表現と比較してください。



次の作業

ファイルシステムに対して新しく作成されたストライプを準備するには、125ページの「メタデバイス上にファイルシステムを作成する方法 (ファイルシステムマネー

ジャ)」を参照してください。データベースなど、raw メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

▼ ストライプ方式メタデバイスの作成方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 61ページの「ストライプと連結を作成するための予備情報」の予備情報をチェックした後で、`metainit(1M)` コマンドを使用してストライプ方式メタデバイスを作成します。詳細は、`metainit(1M)` のマニュアルページを参照してください。



注意 - 既存のファイルシステムやデータからは、ストライプ方式メタデバイスを作成しないでください。これを行なった場合、データが破壊されます。既存のデータからストライプ方式メタデバイスを作成するには、データをダンプ操作 (バックアップの取得) をしたものをメタデバイスに復元する必要があります。

例 - 32K バイトの飛び越しで 2 つのスライスから成るストライプ方式メタデバイスの作成

```
# metainit d10 1 2 c0t1d0s2 c0t2d0s2 -i 32k
d10: Concat/Stripe is setup
```

ストライプ方式メタデバイスである `d10` は、2 つのスライス (数字 2) から成る 1 つのストライプ (数字 1) から構成されます。`-i` オプションは飛び越しに 32K バイトを設定します (この飛び越しは、8K バイト未満であったり 100M バイトを超えてはなりません)。飛び越しが指定されない場合、ストライプ方式メタデバイスはデフォルトの 16K バイトを使用します。システムは、連結方式オブジェクトが設定されたことを確認します。

飛び越し値の設定については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

例 - 3つのスライスから成るストライプ方式メタデバイスの作成

```
# metainit d20 1 3 c0t1d0s2 c0t2d0s2 c0t3d0s2
d20: Concat/Stripe is setup
```

ストライプ方式メタデバイスである d20 は、3つのスライス (数字 3) から成る 1つのストライプ (数字 1) から構成されます。飛び越しが指定されていないため、ストライプ方式メタデバイスはデフォルトの 16K バイトを使用します。システムは、連結方式オブジェクトが設定されたことを確認します。

次の作業

ファイルシステムに対して新しく作成されたストライプ方式メタデバイスを準備するには、126ページの「メタデバイス上にファイルシステムを作成する方法 (コマンド行)」を参照してください。データベースなど、raw メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

▼ 連結の作成方法 (DiskSuite ツール)

この作業は、何のデータも含まないスライスから連結を作成するために使用します。ファイルシステムやデータベースなど、既存のデータを連結するには、184ページの「既存のデータを含むスライスの拡張方法 (DiskSuite ツール)」を参照してください。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (61ページの「ストライプと連結を作成するための予備情報」) を読んでいることを確認する。
2. 連結方式テンプレートをクリックする。
キャンバスには、未割り当てで未確定の連結方式オブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
3. [オプション] デフォルトのメタデバイス名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。
 - c. 「閉じる」をクリックする。

4. スライスをクリックして、「スライスブラウザ」ウィンドウを表示する。
5. **Control** キーを押しながらクリックして連結するスライスを選択し、連結方式オブジェクトにドラッグする。
6. 表示される「連結方式の指定」ダイアログボックスの「連結」をクリックする。
7. 連結方式オブジェクトの先頭の矩形をクリックしてから、「確定」をクリックする。
8. コンフィグレーションログを表示して、連結が確定されたことを確認する。

例 - 確定済みの連結オブジェクト

この例では、2つのスライスから成る、新しく作成された連結を示します。オブジェクトは、各スライスが自分の矩形内に収まるよう、連結に含まれるスライスを表示します。



次の作業

ファイルシステムに対して新しく作成された連結を準備するには、125ページの「メタデバイス上にファイルシステムを作成する方法 (ファイルシステムマネー

ジャ)」を参照してください。データベースなど、raw メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

▼ 連結の作成方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 61ページの「ストライプと連結を作成するための予備情報」の予備情報をチェックした後で、metainit(1M) コマンドを使用して連結を作成します。詳細については、metainit(1M) のマニュアルページを参照してください。

既存のファイルシステムやデータでは、この作業を使用しないでください。ファイルシステムやデータベースなど、既存のデータを連結するには、186ページの「既存のデータを含むスライスの拡張方法 (コマンド行)」を参照してください。

次の例では、既存のデータを含まない連結を作成します。

例 - 2 つのスライスから成る連結の作成

```
# metainit d25 2 1 c0t1d0s2 1 c0t2d0s2
d25: Concat/Stripe is setup
```

この例では、それぞれが 1 つのスライス (各スライスの前にある数字 1) から成る、2 つの「ストライプ」(数字 2) で構成される連結 d25 を作成します。システムは、連結方式オブジェクトが設定されたことを確認します。

例 - 4 つのスライスから成る連結の作成

```
# metainit d40 4 1 c0t1d0s2 1 c0t2d0s2 1 c0t2d0s3 1 c0t2d1s3
d40: Concat/Stripe is setup
```

この例では、それぞれが 1 つのスライス (各スライスの前にある数字 1) から成る、4 つの「ストライプ」(数字 4) で構成される d40 と呼ばれる連結を作成します。システムは、連結方式オブジェクトが設定されたことを確認します。

次の作業

ファイルシステムに対して新しく作成された連結を準備するには、126ページの「メタデバイス上にファイルシステムを作成する方法 (コマンド行)」を参照してくだ

さい。データベースなど、raw メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

ミラーの作成

この節では、ミラーを最初から作成する方法、およびルート (/) を含めて既存のファイルシステムをミラー化する方法について説明します。

注 - 以前には、ファイルシステムのミラーを作成すると、マウント先が変化しました。現在では、特定の場合には、マウント先を変更することなくミラーを作成できます。詳細については、360ページの「既存の連結方式からミラーを作成する方法 (コマンド行)」を参照してください。

ミラーを作成するための予備情報

- ミラーとは、1つ以上のサブミラーで構成されるメタデバイスのことをいいます。サブミラーは、1つ以上のストライプ方式メタデバイスまたは連結方式メタデバイスで構成されます。データをミラー化すれば、データのコピーを複数保持することによって、最大限のデータ可用性を実現できます。
- ミラーを作成するためには、システムに状態データベースの複製が少なくとも3つ含まれていることが必要です。
- ミラーを作成する前に、ミラーを構成するストライプ方式メタデバイスまたは連結方式メタデバイスを作成します。
- ルート (/)、swap、/usr を始めとする任意のファイルシステム、またはデータベースなどのアプリケーションは、ミラーを使用できます。



注意 - 既存のファイルシステムのミラーを作成するとき、最初のサブミラーにデータが含まれていることを確認してください。

- ミラーを作成するときは、最初に1面のミラーを作成し、その後で2番目のサブミラーを接続してください。

これによって再同期操作が正しく起動され、データが破壊されることはありません。

- 既存のファイルシステムをミラー化する場合は、すでにミラーで使用されているスライス以上のサイズをもつ追加スライスを使用してください。

使用できるのは、ミラーを収めるだけの十分な領域をもつ複数のスライスから成る、連結方式メタデバイスまたはストライプ方式メタデバイスです。

- 将来、2面または3面のミラーを作成するときのことを考えて、前もって1面ミラーを作成しておくことができます。
- 3面ミラーまで作成できます。

しかし通常、2面ミラーでも、大部分のアプリケーションに対して十分なデータ冗長性を提供でき、コストも3面より低く抑えられます。3面ミラーを使用すれば、サブミラーをオフラインにしてバックアップを実行するときも、データの冗長性を得るために2面ミラーを維持できます。

- サブミラーを作成するときは、同じサイズのスライスを使用してください。異なるサイズのスライスを使用すると、ミラー内に未使用領域ができます。
- 同じディスクのスライスをサブミラーにもたせることは避けてください。

また、可能ならば、異なるコントローラに接続されたディスクを使用して、システムのアキレス腱となる部分が生まれないようにしてください。最大限の耐障害性とパフォーマンスを実現するには、各サブミラーを異なる物理ディスクに置き、さらに可能ならば、異なるディスク制御装置の配下に置きます。データの可用性をさらに高めるには、ミラーに対してホットスペアを関連付けてください。

- ミラーを作成する前に状態データベースの複製を追加すると、ミラーのパフォーマンスを高めることができます。一般に、システムに追加するミラーごとに複製を1つ追加します。
- 可能ならば、ディスクの幾何学的配置が同じディスクからミラーを作成してください。

その歴史的な理由としては、UFSはディスクの幾何学的配置にもとづいてディスクブロックを使用することが挙げられます。今日では、この問題はパフォーマンスの向上に関連します。幾何学的配置の異なるディスクで構成されたミラーは、その最も遅いディスクと同じ速さになります。

▼ 未使用スライスからミラーを作成する方法 (DiskSuite ツール)

この作業での手順を次に示します。

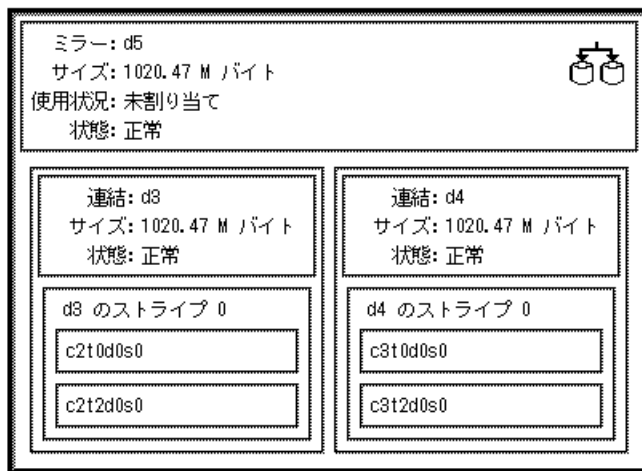
- サブミラーとなる、2つのストライプ方式メタデバイスまたは連結方式メタデバイスの作成。

63ページの「ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)」、または67ページの「連結の作成方法 (DiskSuite ツール)」を参照してください。

- サブミラーの1つから1面のミラーを作成。
 - 2番目のサブミラーから2面のミラーを作成。
1. **前提条件** (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、**予備情報** (70ページの「ミラーを作成するための予備情報」) を読んでいることを確認する。
 2. ミラーテンプレートをクリックする。
キャンバスには、未割り当てで未確定のミラーオブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
 3. **[オプション]** デフォルトのメタデバイス名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。
 - c. 「閉じる」をクリックする。
 4. **1面のミラーを作成するには**、オブジェクトリストから最初の連結方式オブジェクト (サブミラー) をミラーテンプレートにドラッグする。ミラーの先頭の矩形をクリックし、「確定」をクリックする。
 5. **2面のミラーを作成するには**、オブジェクトリストから2番目の連結方式オブジェクト (サブミラー) をミラーにドラッグする。ミラーの先頭の矩形をクリックし、「確定」をクリックする。
これで、2番目のサブミラーの再同期が始まります。ミラーオブジェクトは、再同期の進捗状況を表示します。
 6. **コンフィグレーションログを表示して**、ミラーが確定されたことを確認する。

例 - 確定済みのミラーオブジェクト

この例では、2つのストライプ方式メタデバイス(サブミラー)である d3 と d4 から成る、確定済みのミラーオブジェクト d5 を示します。



次の作業

ファイルシステムに対して新しく作成されたミラーを準備するには、125ページの「メタデバイス上にファイルシステムを作成する方法(ファイルシステムマネージャ)」を参照してください。データベースなど、raw メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

▼ 未使用スライスからミラーを作成する方法(コマンド行)

この作業での手順を次に示します。

- サブミラーとなる、2つのストライプ方式メタデバイスまたは連結方式メタデバイスの作成。
66ページの「ストライプ方式メタデバイスの作成方法(コマンド行)」、または 69ページの「連結の作成方法(コマンド行)」を参照してください。
- `metainit(1M) -m` コマンドを使用して、サブミラーとなるメタデバイスのうちの1つを使用して1面ミラーを作成する。

- `metattach(1M)` コマンドを使用して、2つ目のサブミラーから2面ミラーを作成する。

作業を開始する前に、57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と70ページの「ミラーを作成するための予備情報」の予備情報をチェックします。詳細は、`metainit(1M)` と `metattach(1M)` のマニュアルページを参照してください。

例 - 2面のミラーの作成

```
# metainit d51 1 1 c0t0d0s2
d51: Concat/Stripe is setup
# metainit d52 1 1 c1t0d0s2
d52: Concat/Stripe is setup
# metainit d50 -m d51
d50: Mirror is setup
# metattach d50 d52
d50: Submirror d52 is attached
```

この例では、2面ミラー `d50` を作成します。`metainit(1M)` コマンドは、実際には連結である、2つのサブミラー (`d51` と `d52`) を作成します。`metainit -m` コマンドは、`d51` の連結から1面ミラーを作成します。`metattach(1M)` コマンドは、`d52` を接続することによって、2面ミラーを作成してミラーの再同期を行います (接続されたサブミラー上のデータは、再同期の間に他のサブミラーによって上書きされます)。システムは、オブジェクトが設定されていることを確認します。

次の作業

ファイルシステムに対して新しく作成されたミラーを準備するには、126ページの「メタデバイス上にファイルシステムを作成する方法 (コマンド行)」を参照してください。データベースなど、`raw` メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

▼ マウント解除可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)

マウント解除可能なファイルシステムをミラー化するには、次の作業を使用します。

注 - マウント解除可能なファイルシステムと、マウント解除不可能なファイルシステム (ルート (/)、/usr、/opt、swap など) があります。マウント解除不可能なファイルシステムをミラー化するには、80ページの「マウント解除不可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)」を参照してください。

この作業での手順を次に示します。

- ミラー化される既存のファイルシステムを含んでいるスライスの特定
- ファイルシステムの存在するスライスを連結方式オブジェクト (サブミラー 1 になる) に置く
- 2 番目の連結方式オブジェクト (サブミラー 2 になる) の作成
- ミラーテンプレートに対してサブミラー 1 をドラッグしてミラーを確定する
- ファイルシステムのマウント解除
- ミラーオブジェクトに対してサブミラー 2 をドラッグしてミラーを確定する
- ファイルシステムの再マウント



注意 - 最初に、多面のミラーを作成しないでください。まず 1 面ミラーを確定し、ミラーテンプレートに追加のサブミラーをドラッグし、再びミラーを確定します。こうすることによってミラーの再同期が正しく行われるので、データは破壊されません。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (70ページの「ミラーを作成するための予備情報」) を読んでいることを確認する。
2. スライスをクリックして、「スライスブラウザ」ウィンドウを表示する。
DiskSuite ツールは、マウントされているスライス名の横にファイルシステムの名前を表示します。DiskSuite ツールを起動した後でファイルシステムをマウントした場合、「ファイル」メニューから「構成の再走査」を選択します。
3. 連結方式テンプレートをクリックする。
DiskSuite ツールは、未割り当てで未確定の連結方式オブジェクトをキャンバスに表示し、それにメタデバイス名を与えます。
4. [オプション] デフォルトのメタデバイス名を変更する。

- a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。
 - c. 「閉じる」をクリックする。
5. ファイルシステムのスライスを、「スライスブラウザ」ウィンドウから連結方式オブジェクトにドラッグする。
スライスがドロップされると、スライスがマウントされたことを示す警告ダイアログボックスが表示されます。ここで「継続」をクリックします。
 6. 連結方式オブジェクトが選択されていることを確認し、「確定」をクリックする。
表示される警告ダイアログボックスで、「確定」をクリックします。これによって、最初のサブミラーとして使用されるファイルシステムを含んだメタデバイスが作成されます。

注 - /etc/vfstab ファイルにファイルシステムのエントリが存在し、そのファイルシステムが現在マウントされている場合、DiskSuite ツールはこれを自動的に更新して、連結の名前を使用します。

7. 連結方式テンプレートをクリックする。
DiskSuite ツールは、未割り当てで未確定の連結方式オブジェクトをキャンバスに表示し、それにメタデバイス名を与えます。
8. 【オプション】デフォルトのメタデバイス名を変更する。
75ページの手順4を参照してください。
9. スライスを「スライスブラウザ」ウィンドウから連結方式オブジェクトにドラッグする。
未使用スライス、状態データベースの複製を含むスライス、または既存のファイルシステムのサイズ以上のサイズをもつ複数のスライスを選択します。
10. 連結方式オブジェクトが選択されていることを確認し、「確定」をクリックする。

これによって、2 番目のサブミラーとして使用されるメタデバイスが作成されま
す。

11. ミラーテンプレートをクリックする。

DiskSuite ツールは、未割り当てで未確定のミラーオブジェクトをキャンバスに
表示し、それにメタデバイス名を与えます。

12. [オプション] デフォルトのメタデバイス名を変更する。

75ページの手順 4 を参照してください。

13. 76ページの手順 6 で作成されたファイルシステムを含む連結方式オブジェク
トを、ミラーテンプレートにドラッグする。

警告ダイアログボックスが表示されます。ここで「**継続**」をクリックします。

注 - /etc/vfstab ファイルにファイルシステムのエントリが存在し、その
ファイルシステムが現在マウントされている場合、DiskSuite ツールはこれを自
動的に更新し、ミラーのメタデバイス名を使用します。

14. ミラーオブジェクトの先頭の矩形をクリックし、「**確定**」をクリックする。

これによって 1 面のミラーが作成されます。

15. ファイルシステムをマウント解除する。

たとえば、umount(1M) コマンドやファイルシステムマネージャを使用します。

16. 「ファイル」メニューから「構成の再走査」を選択します。

DiskSuite ツールは、ファイルシステムの現在のマウント状態を更新します。

17. ファイルシステムをミラーに再マウントする。

たとえば、mount(1M) コマンドやファイルシステムマネージャを使用します (現
在、マウント先はスライス名からミラー名に変化しています)。

18. 「ファイル」メニューから「構成の再走査」を選択する。

DiskSuite ツールは、ファイルシステムのマウント先を更新します。ミラーオブ
ジェクトは、ファイルシステムによって使用されていることを表示します。

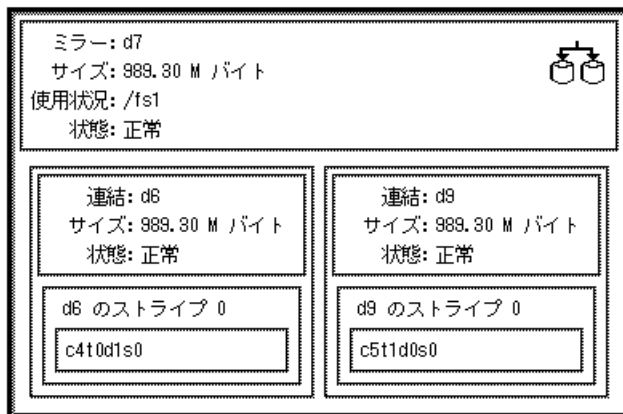
19.2 番目の連結方式オブジェクト (76ページの手順 10 で作成) をミラーオブジェクトにドラッグする。ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。

最初のサブミラーからのデータは、自動的に 2 番目のミラーと再同期されます。

20. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。

例 - マウントされたファイルシステムをもつ確定済みミラーオブジェクト

この例では、サブミラー d6 と d9 から成る、確定済みの 2 面ミラーを示します。d6 と d9 は同様に、それぞれスライス c4t0d1s0 と c5t1d0s0 から成ります。ファイルシステム /fs1 は、ミラー d7 にマウントされています。



▼ マウント解除可能なファイルシステムからミラーを作成する方法 (コマンド行)

この作業は、マウント解除可能な既存のファイルシステムをミラー化するために使用します。

注 - コマンド行を使用して /usr、/opt、または swap をミラー化するには、83 ページの「マウント解除不可能なファイルシステムからミラーを作成する方法 (コマンド行)」を参照してください。コマンド行を使用してルート (/) をミラー化するには、86 ページの「SPARC: ルート (/) からミラーを作成する方法 (コマンド行)」または 88 ページの「x86: ルート (/) からミラーを作成する方法 (コマンド行)」を参照してください。

この作業での手順を次に示します。

- ミラー化される既存のファイルシステムを含むスライスの識別
- `metainit(1M) -f` を使用して、マウントされたファイルシステムのスライスを連結方式 (サブミラー 1) に置く。
- 2 番目の連結方式 (サブミラー 2) の作成
- `metainit(1M) -m` を使用して、サブミラー 1 で 1 面のミラーを作成
- ファイルシステムのマウント解除
- ファイルシステムがミラーを参照するよう、`/etc/vfstab` ファイルを編集
- ファイルシステムの再マウント
- `metattach` を使用してサブミラー 2 を接続

作業を開始する前に、57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 70ページの「ミラーを作成するための予備情報」の予備情報をチェックしてください。詳細は、`metainit(1M)` と `metattach(1M)` のマニュアルページを参照してください。



注意 - 多面のミラーは作成しないでください。まず `metainit(1M)` コマンドで 1 面ミラーを作成してから、`metattach(1M)` コマンドで追加のサブミラーを接続します。`metattach(1M)` コマンドを使用しない場合、再同期の動作が行われず、データが破壊されることがあります。また、ファイルシステムの 2 面ミラーを作成する場合、最初にファイルシステムのマウントを解除し、2 番目のサブミラーを接続する前に `/etc/vfstab` ファイルを編集して、ミラーメタデバイスを参照するようにしてください。

例 - 2 面ミラーの作成

```
# metainit -f d1 1 1 c1t0d0s0
d1: Concat/Stripe is setup
# metainit d2 1 1 c2t0d0s0
d2: Concat/Stripe is setup
# metainit d0 -m d1
d0: Mirror is setup
# umount /master
(ファイルシステムがミラーを参照するように /etc/vfstab ファイルを編集する)
# mount /master
# metattach d0 d2
```

(続く)

```
d0: Submirror d2 is attached
```

-f オプションによって、マウントされたファイルシステム `/master` を `/dev/dsk/c1t0d0s0` にもつ、最初の連結 `d1` が作成されます。2 番目の連結 `d2` は、`/dev/dsk/c2t0d0s0` から作成されます (このスライスのサイズは、`d1` のサイズ以上でなければなりません)。-m オプションを付けた `metainit` コマンドは、`d1` から 1 面ミラー `d0` を作成します。

次に、`/master` がマウント解除され、ミラーを参照するように `/etc/vfstab` ファイル内で変更されます。たとえば、次の行は、

```
/dev/dsk/c1t0d0s0 /dev/rdisk/c1t0d0s0 /master ufs 2 yes -
```

次のように変更してください。

```
/dev/md/dsk/d0 /dev/md/rdsk/d0 /master ufs 2 yes -
```

最後に、`/master` ファイルシステムが再マウントされ、サブミラー `d2` がミラーに接続されるため、ミラーの再同期が行われます (システムは、連結とミラーが設定されたこと、およびサブミラー `d2` が接続されたことを確認します)。

▼ マウント解除不可能なファイルシステムからミラーを作成する方法 (DiskSuite ツール)

この作業は、通常のシステム使用中にはマウント解除が不可能であるルート (`/`)、`/usr`、`/opt`、`swap` などのファイルシステムをミラー化するために使用します。

この作業での手順を次に示します。

- ファイルシステムのスライスを、単一スライスの連結方式オブジェクト (サブミラー 1) に置く
- 2 番目の連結方式オブジェクト (サブミラー 2) の作成
- サブミラー 1 をミラーテンプレートにドラッグして、ミラーを確定
- リポートする
- サブミラー 2 をミラーオブジェクトにドラッグして、ミラーを確定
- ルート (`/`) をミラー化する場合、代替ブートパスを記録

注・ルート (/) をミラー化するとき、一次サブミラーに障害が発生した場合にシステムをリブートするときのために、二次ルートのスライス名を記録しておくことが不可欠です。この情報は、動作できなくなるかもしれないシステム上に記録するのではなく、紙などに書き留めておきます。代替ブートデバイスの記録と代替ブートデバイスからのブート操作については、第7章を参照してください。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (70ページの「ミラーを作成するための予備情報」) を読んでいることを確認する。
2. 連結方式テンプレートをクリックする。
DiskSuite ツールは、未割り当てで未確定の連結方式オブジェクトをキャンバスに表示し、それにメタデバイス名を与えます。
3. [オプション] デフォルトのメタデバイス名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。
 - c. 「閉じる」をクリックする。
4. スライスをクリックし、「スライスブラウザ」ウィンドウを表示する。
5. ミラー化したいファイルシステムを含むスライスを、連結方式オブジェクトの先頭の矩形にドラッグする。
スライスをドロップすると、DiskSuite ツールは警告ダイアログボックスを表示します。そこで「継続」ボタンをクリックします。
6. 連結方式オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
警告ダイアログボックスが表示された場合、「確定」ボタンをクリックします。これによって、最初のサブミラーとして使用されるファイルシステムを含んだ、1方向の連結が作成されます。

注 - /etc/vfstab ファイルにファイルシステムのエントリが存在し、そのファイルシステムが現在マウントされている場合、DiskSuite ツールはこれを自動的に更新し、連結のメタデバイス名を使用します。

7. 連結方式テンプレートをクリックする。

DiskSuite ツールは、未割り当てで未確定の連結方式オブジェクトをキャンバスに表示し、それにメタデバイス名を与えます。

8. [オプション] デフォルトのメタデバイス名を変更する。

81ページの手順3を参照してください。

9. 「スライスブラウザ」ウィンドウからこの連結方式オブジェクトにスライスをドラッグする。

この1方向の連結方式メタデバイスを作成するとき、既存のファイルシステムのサイズ以上のサイズをもつ未使用スライスを使用してください。

10. 連結方式オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。

これによって2番目のサブミラーが作成されます。

11. ミラーテンプレートをクリックする。

DiskSuite ツールは、未割り当てで未確定のミラーオブジェクトをキャンバスに表示し、それにメタデバイス名を与えます。

12. [オプション] デフォルトのメタデバイス名を変更する。

81ページの手順3を参照してください。

13. ファイルシステムを含む連結方式オブジェクトを、ミラーテンプレートにドラッグする。

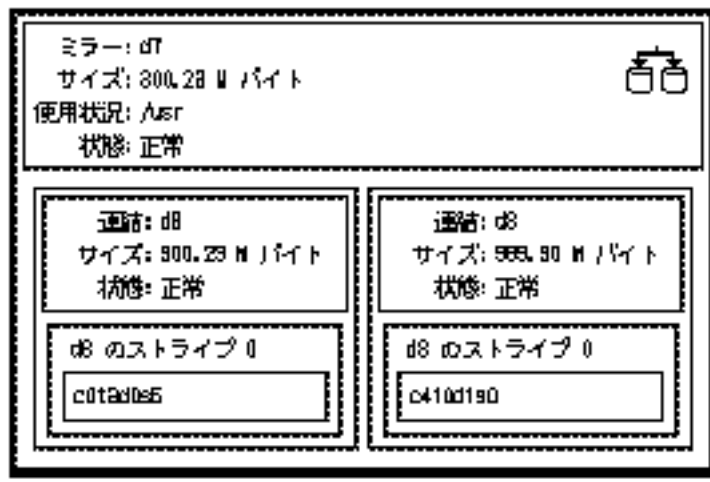
警告ダイアログボックスが表示された場合は、「継続」ボタンをクリックします。

注 - /etc/vfstab ファイルにファイルシステムのエントリが存在し、そのファイルシステムが現在マウントされている場合、DiskSuite ツールはこれを自動的に更新し、ミラーのメタデバイス名を使用します。

14. ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
15. リポートする。
16. **DiskSuite** ツールを再起動する。
17. 82ページの手順 10 で作成した 2 番目の連結方式オブジェクトをミラーオブジェクトにドラッグし、2 面ミラーを作成する。そして「確定」をクリックする。
最初のサブミラーからのデータは、2 番目のミラーに再同期されます。
18. ルート (/) をミラー化する場合は、代替ブートパスを記録する。
詳細については、第 7 章を参照してください。
19. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。

例 - /usr 用の確定済みミラー

この例では、/usr ファイルシステムを含む確定済みのミラーを示します。



▼ マウント解除不可能なファイルシステムからミラーを作成する方法 (コマンド行)

この作業は、通常のシステム使用中にはマウント解除が不可能な、/usr、/opt、swap などのファイルシステムをミラー化するために使用します。

スワップをミラー化したときにクラッシュダンプを保存するには、`dumpadm(1M)` コマンドを使用し、ダンプデバイスをメタデバイスとして構成します。たとえば、スワップデバイスが `/dev/md/dsk/d2` であれば、`dumpadm` を使用してこれをダンプデバイスとして設定します。

注 - コマンド行を使用してルート (/) をミラー化するには、86ページの「SPARC: ルート (/) からミラーを作成する方法 (コマンド行)」、または 88ページの「x86: ルート (/) からミラーを作成する方法 (コマンド行)」を参照してください。

この作業での手順を次に示します。

- `metainit(1M) -f` を使用して、ファイルシステムのスライスを単一スライス (1 面目) の連結方式 (サブミラー 1) に置く。
- 2 面目の連結方式 (サブミラー 2) の作成
- `metainit(1M)` を使用して、サブミラー 1 から 1 面ミラーを作成
- ファイルシステムがミラーを参照するよう、`/etc/vfstab` ファイルを編集
- リポートする
- `metattach(1M)` を使用してサブミラー 2 を接続



注意 - 多面ミラーは作成しないでください。まず `metainit(1M)` コマンドで 1 面ミラーを作成してから、`metattach(1M)` コマンドで追加のサブミラーを接続します。`metattach(1M)` コマンドを使用しない場合、ミラーの再同期が行われず、データが破壊されることがあります。

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 70ページの「ミラーを作成するための予備情報」の予備情報をチェックしてから、`metainit(1M)` コマンドと `metattach(1M)` コマンドを使用して、ミラーを作成します。

例 - /usr からミラーを作成

```
# metainit -f d12 1 1 c0t3d0s6
d12: Concat/Stripe is setup
# metainit d22 1 1 c1t0d0s6
d22: Concat/Stripe is setup
# metainit d2 -m d12
d2: Mirror is setup
```

(続く)

続き

```
(swap がミラーを参照するよう、/etc/vfstab ファイルを編集する)
# reboot
...
# metattach d2 d22
d2: Submirror d22 is attached
```

-f オプションによって、マウントされたファイルシステム /usr を /dev/dsk/c0t3d0s6 にもつ、最初の連結 d12 が作成されます。2 番目の連結 d22 は、/dev/dsk/c1t0d0s6 から作成されます (このスライスのサイズは、d12 のサイズ以上でなければなりません)。-m オプションを付けた metainit コマンドは、/usr を含む連結を使用して 1 面ミラー d2 を作成します。次に、/etc/vfstab ファイルを編集して、/usr のエントリがミラーを参照するように変更しなければなりません。

たとえば、次の行は、

```
/dev/dsk/c0t3d0s6 /dev/rdsk/c0t3d0s6 /usr ufs 1 yes -
```

次のように変更してください。

```
/dev/md/dsk/d2 /dev/md/rdsk/d2 /usr ufs 1 yes -
```

リブートの後、2 番目のサブミラー d22 がミラーに接続され、ミラーの再同期が行われます (システムは、連結とミラーが設定されたこと、およびサブミラー d22 が接続されたことを確認します)。

例 - swap からミラーを作成

```
# metainit -f d11 1 1 c0t0d0s1
d11: Concat/Stripe is setup
# metainit d21 1 1 c1t0d0s1
d21: Concat/Stripe is setup
# metainit d1 -m d11
d1: Mirror is setup
(swap がミラーを参照するよう、/etc/vfstab ファイルを編集する)
# reboot
...
# metattach d1 d21
d1: Submirror d21 is attached
```

-f オプションによって、マウントされたファイルシステム swap を /dev/dsk/c0t0d0s1 にもつ、最初の連結 d11 が作成されます。2 番目の連結 d21 は、/dev/dsk/c1t0d0s1 から作成されます (このスライスのサイズは、d11 のサイズ以上でなければなりません)。-m オプションを付けた metainit コマンドは、swap を含む連結を使用して 1 面ミラー d1 を作成します。次に、/etc/vfstab ファイルに swap 用のエントリがある場合、これを編集してミラーを参照するようにしなければなりません。たとえば、次の行は、

```
/dev/dsk/c0t0d0s1 - - swap - no -
```

次のように変更してください。

```
/dev/md/dsk/d1 - - swap - no -
```

リポートの後、2 番目のサブミラー d21 がミラーに接続され、ミラーの再同期が行われます (システムは、連結とミラーが設定されたこと、およびサブミラー d21 が接続されたことを確認します)。

▼ SPARC: ルート (/) からミラーを作成する方法 (コマンド行)

この作業は、SPARC システム上でルート (/) をミラー化するために使用します。

注 - x86 システム上でコマンド行を使用してルート (/) をミラー化する作業は、SPARC システムの場合とは異なります。x86 システム上でルート (/) をミラー化する方法については、88ページの「x86: ルート (/) からミラーを作成する方法 (コマンド行)」を参照してください。ルート (/) をミラー化するとき、一次サブミラーに障害が発生した場合にシステムをリポートするときのために、二次ルートのスライス名を記録しておくことが不可欠です。この情報は、動作できなくなるかもしれないシステム上に記録するのではなく、紙などに書き留めておきます。代替ブートデバイスの記録と代替ブートデバイスからのブート操作の詳細については、第7章を参照してください。

この作業での手順を次に示します。

- metainit(1M) -f を使用して、ルート (/) スライスを単一スライス (1 面目) の連結 (サブミラー 1) に置く
- 2 面目の連結 (サブミラー 2) を作成する
- metainit(1M) を使用して、サブミラー 1 で 1 面ミラーを作成する
- metaroot(1M) コマンドを実行する

- lockfs(1M) コマンドを実行する
- リブートする
- metattach(1M) を使用して、サブミラー 2 を接続する
- 代替ブートパスを記録する

前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (70ページの「ミラーを作成するための予備情報」) を読んでいることを確認します。詳細は、metainit(1M)、metaroot(1M)、metattach(1M) の各マニュアルページを参照してください。

例 - SPARC: ルート (/) からミラーを作成する方法

```
# metainit -f d11 1 1 c0t3d0s0
d11: Concat/Stripe is setup
# metainit d12 1 1 c1t3d0s0
d12: Concat/Stripe is setup
# metainit d10 -m d11
d10: Mirror is setup
# metaroot d10
# lockfs -fa
# reboot
...
# metattach d10 d12
d10: Submirror d12 is attached
# ls -l /dev/rdisk/c1t3d0s0
lrwxrwxrwx  1 root      root           88 Feb  8 15:51 /dev/rdisk/c1t3d0s0 ->
../../../../devices/iommu@f,e0000000/vme@f,df010000/SUNW,pn@4d,1080000/ipi3sc@0,0/
i
d@3,0:a,raw
```

-f オプションによって、マウントされたファイルシステムのルート (/) を /dev/dsk/c0t3d0s0 にもつ、最初の連結 d11 が作成されます。2 番目の連結 d12 は、/dev/dsk/c1t3d0s0 から作成されます (このスライスのサイズは、d11 のサイズ以上でなければなりません)。-m オプションを付けた metainit コマンドは、ルート (/) を含む連結を使用して 1 面ミラー d10 を作成します。次に、メタデバイス上のルートファイルシステム (/) でシステムをブートできるように、metaroot コマンドは /etc/vfstab と /etc/system のファイルを編集します (リブートする前に、lockfs -fa を実行することをお勧めします)。リブートの後、サブミラー d12 がミラーに接続され、ミラーの再同期が行われます (システムは、連結とミラーが設定されていること、およびサブミラー d12 が接続されていることを確認します)。システムを代替ルートデバイスからブートする必要がある場合、ルート raw デバイス上で ls -l コマンドを実行して、代替ルートデバイスへのパスを調べます。

▼ x86: ルート (/) からミラーを作成する方法 (コマンド行)

この作業は、x86 システム上でルート (/) をミラー化するために使用します。

注 - ルート (/) をミラー化するとき、一次サブミラーに障害が発生した場合にシステムをリブートするときのために、二次ルートのスライス名を記録しておくことが不可欠です。この情報は、動作できなくなるかもしれないシステム上に記録するのではなく、紙などに書き留めておきます。代替ブートデバイスの記録と代替ブートデバイスからのブート操作の詳細については、第7章を参照してください。

この作業での手順を次に示します。

- ルート (/) ミラー用に十分な大きさをもつ Solaris パーティションを作成する
- 代替ブートディスクにブート情報をインストールする
- `metainit(1M) -f` を使用して、ルート (/) スライスを単一スライス (1 面目) の連結 (サブミラー 1) に置く
- 2 面目の連結 (サブミラー 2) を作成する
- `metainit(1M)` を使用して、サブミラー 1 で 1 面ミラーを作成する
- `metaroot(1M)` コマンドを実行する
- `lockfs(1M)` コマンドを実行する
- リブートする
- `metattach(1M)` を使用してサブミラー 2 を接続する
- 代替ブートパスを記録する

注 - IDE ドライブ上にルート (/) があるときはミラー化することはできません。

次の作業では、代替ディスクは `c0t1d0` であるとしています。

Solaris パーティションの作成

次の手順は、x86 システム上に Solaris パーティションを作成するために使用します。

1. `fdisk(1M)` コマンドを使用して、ディスクパーティションを作成する。

```
# fdisk /dev/rdisk/c0t1d0p0
```


- a. **fdisk(1M)** を初めて実行する場合は、次のように表示されます。

```
The recommended default partitioning for your disk is:

a 100% ``SOLARIS System'' partition

To select this, please type ``y''. To partition your disk
differently, type ``n'' and the ``fdisk'' program will let you
select other partitions.
```

- b. 以前 **fdisk(1M)** を実行したことがある場合は、次のようなメニューが表示されます。

```
Total disk size is 1855 cylinders
Cylinder size is 1110 (512 byte) blocks

Partition  Status  Type      Start  End  Length  %
=====  =====  =====  =====  ===  =====  ==
      1      Active  SOLARIS      1  1854   1854   100

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Change Active (Boot from) partition
3. Delete a partition
4. Exit (Update disk configuration and exit)
5. Cancel (Exit without updating disk configuration)
Enter Selection:
```

2. メニュー項目を選択して、ルート (/) ミラー用に十分な大きさの **Solaris** パーティションがあることを確認する。
- この Solaris パーティションは、ルート (/) スライスを保持するために必要なサイズより 5 シリンダ分大きくします。
- Solaris パーティションがアクティブであることを確認します。そうでない場合、ここからブートすることはできません。
3. **Solaris** パーティション上で **format(1M)** コマンドを実行し、ルート (/) ミラー用のスライスを作成する。

```

# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
    0. c0t0d0 <DEFAULT cyl 2676 alt 2 hd 9 sec 85>
       /eisa/ncrs@8000,0/cmdk@0,0
    1. c1t1d0 <DEFAULT cyl 1865 alt 2 hd 7 sec 80>
       /eisa/eha@1000,0/cmdk@1,0
    2. c1t2d0 <DEFAULT cyl 1461 alt 2 hd 9 sec 64>
       /eisa/eha@1000,0/cmdk@2,0
    3. c1t3d0 <DEFAULT cyl 1461 alt 2 hd 9 sec 64>
       /eisa/eha@1000,0/cmdk@3,0
    4. c1t4d0 <DEFAULT cyl 1865 alt 2 hd 7 sec 80>
       /eisa/eha@1000,0/cmdk@4,0
Specify disk (enter its number): 1

selecting c1t1d0: ABCDEFG
[disk formatted]

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
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
quit

```

4. partition を選択して、パーティションを定義する。

```

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

```

(続く)

続き

```
        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> 0
Part  Tag      Flag  Cylinders      Size  Blocks
   0  unassigned  wm      0              0      (0/0/0)

Enter partition id tag [unassigned]: root
Enter partition permission flags [wm]: wm
Enter new starting cyl [0]: 4
Enter partition size [0b, 0c, 0.00mb]: 400mb
partition> label
Ready to label disk, continue? y
partition>
```

5. quit コマンドを 2 回入力して、partition メニューと format(1M) プログラムを終了する。

```
partition> quit

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
  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
  quit
format> quit
```

Solaris のルート (/) パーティションに関する次の重要な情報に注意してください。

- I.D. タグは root でなければならない

- サイズは元のルートパーティションのサイズ以上でなければならない
- シリンダ 0 ~ 2 を使用してはいけない

ブート情報のインストール

代替ブートディスクにブート情報をインストールするには、installboot コマンドを使用します。

```
# installboot /usr/lib/fs/ufs/pboot \
/usr/lib/fs/ufs/bootblk \
/dev/rdisk/c0t1d0s0
```

例 - x86: ルート (/) からミラーを作成する方法

この例では、上の作業を使用して Solaris パーティションが作成され、ブート情報がインストールされているものと想定します。

```
# metainit -f d10 1 1 c0t0d0s0
d10: Concat/Stripe is setup
# metainit d20 1 1 c1t0d0s0
d20: Concat/Stripe is setup
# metainit d0 -m d10
d10: Mirror is setup
# metaroot d0
# lockfs -fa
# reboot
...
# metattach d0 d20
d0: Submirror d20 is attached
# ls -l /dev/rdsk/c1t0d0s0
lrwxrwxrwx 1 root root 55 Mar 5 12:54 /dev/rdsk/c1t0d0s0 -> ../
devices/eisa/eha@1000,0/cmdk@1,0:a
```

-f オプションによって、マウントされたファイルシステムルート (/) を /dev/dsk/c0t0d0s0 にもつ、最初の連結 d10 が作成されます。2 番目の連結 d20 は、/dev/dsk/c1t0d0s0 から作成されます (このスライスのサイズは、d10 のサイズ以上でなければなりません)。-m オプションを付けた metainit コマンドは、ルート (/) を含む連結を使用して 1 面ミラー d0 を作成します。次に、メタデバイス上のルートファイルシステム (/) でシステムをブートできるよう、metaroot コマンドは /etc/vfstab と /etc/system のファイルを編集します。リブートの後、サブミラー d20 がミラーに接続され、ミラーの再同期が行われます (システムは、連結とミラーが設定されていること、およびサブミラー d20 が接続されている

ことを確認します)。システムを代替ルートデバイスからブートする必要がある場合、ルート raw デバイス上で `ls -l` コマンドを実行して、代替ルートデバイスへのパスを調べます。

RAID5 メタデバイスの作成

この節では、RAID5 メタデバイスの作成方法について説明します。

RAID5 メタデバイスを作成するための予備情報

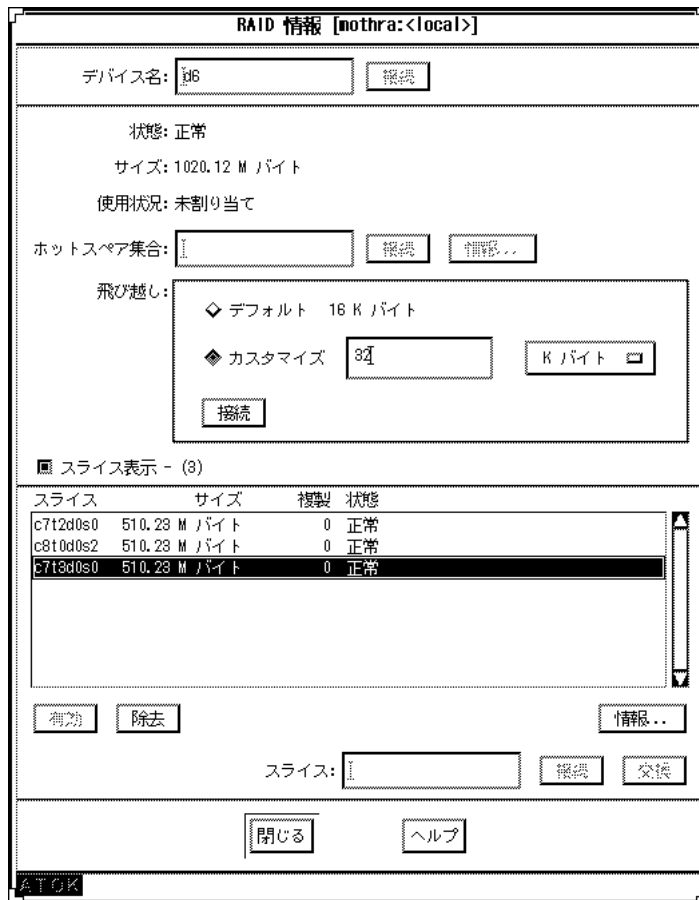
- RAID5 メタデバイスは、メタデバイス内の 1 スライス分に相当する記憶容量を使用して、残りの RAID5 メタデバイスのスライスに格納されたユーザーのデータに関する冗長情報を格納します。この冗長情報は、メタデバイス内のすべてのスライスに分散されます。ミラーと同様、RAID5 メタデバイスはデータの可用性を高めながら、ハードウェアのコストを最小限に抑えることができます。
- RAID5 メタデバイスを作成するには、あらかじめシステムに状態データベースの複製が少なくとも 3 つ含まれている必要があります。
- RAID5 メタデバイスは、単一スライスの障害だけに対処できます。
- RAID5 メタデバイスを作成するときは、20 パーセントルールに従ってください。パリティの計算手順は大変複雑なものであるため、約 20 パーセントを超える書き込みのメタデバイスは、おそらく RAID5 メタデバイスには (速度の点から) 向きません。データの冗長性が必要とされる場合は、ミラー化を考えてください。
- 多数のスライスからなる RAID5 メタデバイスには不利な点があります。RAID5 メタデバイスに含まれるスライスが増えるほど、1 つのスライスに障害が発生した場合に、読み書き操作に長時間を要することになります。
- RAID5 メタデバイスは、少なくとも 3 つのスライスから構成されなければなりません。
- RAID5 メタデバイスは、メタデバイスに追加のスライスを連結することによって拡張できます。新しいスライスにはパリティ保護が行われますが、パリティ情報は格納されません。拡張した RAID5 メタデバイスも、単一スライスの障害に対応しています。
- 飛び越し値は、RAID5 のパフォーマンスに対する鍵となります。この値はメタデバイスの作成時に設定可能であり、作成後は変更できません。デフォルトの飛び

越し値は 16K バイトです。これはほとんどのアプリケーションにとって妥当な値です。

- 同じサイズのディスクスライスを使用してください。異なるサイズのスライスから RAID5 メタデバイスを作成すると、メタデバイス内に未使用ディスク領域が生じます。
- 既存のファイルシステムを含むスライスからは、RAID5 メタデバイスを作成しないでください。これを行なった場合、RAID5 の初期化プロセスの間にデータが消去されます。
- RAID5 メタデバイスをストライプ化、連結、ミラー化することはできません。

▼ RAID5 メタデバイスの作成方法 (DiskSuite ツール)

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (93ページの「RAID5 メタデバイスを作成するための予備情報」) を読んでいることを確認する。
2. **RAID5** テンプレートをクリックする。
キャンバスには、未割り当てで未確定の RAID5 オブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
3. [オプション] デフォルトのメタデバイス名を変更する。
4. [オプション] 飛び越し値をデフォルトの **16K** バイトから変更する。
オブジェクトのポップアップメニューから「情報」を選択する。
RAID5 メタデバイス用の「RAID 情報」ウィンドウが表示されます。



- 飛び越し値を変更する。「カスタマイズ」をクリックし、その横にあるフィールドに新しい値を入力する。

単位には、K バイト、M バイト、またはセクターを指定できます。この横のプルダウンメニューから、任意の単位を選択してください。「接続」をクリックして値を設定し、「閉じる」をクリックします。

飛び越し値の設定については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

- スライスをクリックして「スライスブラウザ」ウィンドウを開く。
- スライスを選択し、**RAID5** オブジェクトの先頭にドラッグする。

複数のスライスを選択するには、Control キーを押しながらクリックします。少なくとも 3 つのスライスが必要です。

8. **RAID5** オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
DiskSuite は、RAID5 メタデバイスの初期化を開始します。
9. コンフィグレーションログを表示して、**RAID5** メタデバイスが確定されたことを確認する。

注 - 初期化が終了したら、RAID5 メタデバイスを使用できます。

例

この例では、RAID5 メタデバイスを示します。このメタデバイスは 3 つの 510.23 M バイトスライスから構成されますが、表示されたサイズは 1020.12 M バイトであることに注目してください (1 スライス分の記憶容量がパリティに使用されます)。



次の作業

新しく作成された RAID5 メタデバイスをファイルシステム用に準備するには、125 ページの「メタデバイス上にファイルシステムを作成する方法 (ファイルシステムマネージャ)」を参照してください。データベースなど、raw メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

ホットスペア集合と RAID5 メタデバイスを関連付けるには、111ページの「ホットスペア集合を関連付ける方法 (DiskSuite ツール)」を参照してください。

▼ RAID5 メタデバイスの作成方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 93ページの「RAID5 メタデバイスを作成するための予備情報」の予備情報をチェックしてから、`metainit(1M)` コマンドを使用して、RAID5 メタデバイスを作成します。詳細については、`metainit(1M)` のマニュアルページを参照してください。

例 - 3つのスライスから構成される RAID5 メタデバイスの作成

```
# metainit d45 -r c2t3d0s2 c3t0d0s2 c4t0d0s2
d45: RAID is setup
```

RAID5 メタデバイスである `d45` は、3つのスライスから `-r` オプションで作成されます。飛び越しが指定されていないため、`d45` はデフォルトの 16K バイトを使用します。システムは、RAID5 メタデバイスが設定されたことを確認し、メタデバイスの初期化を開始します。

注 - 初期化が終了したら、RAID5 メタデバイスを使用できます。

次の作業

新しく作成された RAID5 メタデバイスをファイルシステム用に準備するには、126ページの「メタデバイス上にファイルシステムを作成する方法 (コマンド行)」を参照してください。データベースなど、`raw` メタデバイスを使用するアプリケーションには、メタデバイスを認識する独自の方法が必要です。

ホットスペア集合と RAID5 メタデバイスを関連付けるには、112ページの「ホットスペア集合を関連付ける方法 (コマンド行)」を参照してください。

トランスメタデバイスの作成

この節では、トランスメタデバイスの作成方法について説明します (UFS ロギング)。

トランスメタデバイスを作成するための予備情報

- トランスメタデバイスを使用すると、UFS をロギングすることができます。UFS ロギングとは、UNIX ファイルシステムに更新を適用する前に UFS の更新内容をログに記録する手法のことです。トランスメタデバイスは、システムリブート時に fsck(1M) の実行時間の減少に寄与するので、リブート後のファイルシステム全体の可用性が高くなります。
- トランスメタデバイスを作成するには、その前にシステムには状態データベースの複製が少なくとも 3 つなければなりません。
- 通常、トランスメタデバイスには、マスターデバイスとロギングデバイスという 2 つのデバイスがあります。マスターデバイスには、記録されるファイルシステムが収められています。ロギングデバイスにはログが収められており、複数のファイルシステムで共有できます。ログはレコードが連続したもので、それぞれのレコードがファイルシステムの変更を表わします。マスターデバイスとロギングデバイスには、いずれもスライスまたはメタデバイスを指定することができます。
- ログはファイルシステム間で共有できます。ただし、使用頻度の高いファイルシステムにはそのファイルシステム専用のロギングデバイスが必要です。
- 小さなファイルシステムで、読み取り操作が中心である場合は、ログを取る必要はないでしょう。
- ルート (/) を除く任意の UFS で (トランスメタデバイスを) 利用できます。
- ロギングデバイスに使用できるスライスがない場合でも、ロギングデバイスなしでトランスメタデバイスを設定できます。エクスポートされたファイルシステムでロギングを有効にしたいとしても、この時点ではロギングデバイスに使用できるスライスがない場合、この方法が便利です。
- トランスメタデバイスを作成する前に、マスターデバイスおよびロギングデバイスとして使用されるスライスやメタデバイスを識別します。
- 使用頻度の高いディスクには、ログを置かないでください。
- RAID5 メタデバイスをロギングデバイスとして使用しないでください。代わりに、ミラーを使用してデータに冗長性をもたせてください。
- ログ (ロギングデバイス) は、すでに状態データベースの複製が収められているスライスに置くことができます。
- 少なくとも 1 M バイトのログ領域を確保してください。ディスクの追加時に、100 M バイトのファイルシステムデータあたり 1 M バイトのログ領域を (最大ロ

グサイズ 64 M バイトまで) 追加して使用するよう計画を立ててください。ログ領域では、64 M バイトを超える部分は使用されません。

- 同じトランスメタデバイスを構成しているマスターデバイスとロギングデバイスは、別のドライブ(さらに、できる限り別のコントローラ)に配置します。



注意 - ロギングデバイスのミラー化を強くお勧めします。デバイスエラーによってロギングデバイス内のデータが失われると、ファイルシステムが不安定な状態に置かれることがあります。この場合、`fsck(1M)` ではユーザーの介入なしに修正することはできません。マスターデバイスにミラーを使用することによってデータの冗長性を保証することもできます。

▼ マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)

作業を開始する前に、ファイルシステムが含まれるスライスやメタデバイスを識別します。マスターデバイスを作成するときには、この作業が必要です。また、ロギングデバイスをミラー化するかどうかも決定します。ミラー化する場合は、106ページの「ミラーを使用してトランスメタデバイスを作成する方法 (DiskSuite ツール)」を参照してください。ロギングデバイスをミラー化する場合、マスターデバイスもミラーとすることをお勧めします。

マスターデバイスにまだファイルシステムがない場合は、後からトランスメタデバイスにファイルシステムを作成できます。

通常のシステム操作中にマウント解除不可能な `/usr` などのファイルシステムのトランスメタデバイスを作成するには、103ページの「マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」を参照してください。

1. **前提条件** (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、**予備情報** (98ページの「トランスメタデバイスを作成するための予備情報」) を読んでいることを確認する。
2. トランスメタデバイステンプレートをクリックする。
キャンバスには、未割り当てで未確定のトランスメタデバイスオブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
3. **[オプション]** デフォルトのメタデバイス名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。

b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。

c. 「閉じる」をクリックする。

4. マスターデバイスが収められるスライスまたはメタデバイスを、トランスメタデバイステンプレートのマスター矩形にドラッグする。

「スライスブラウザ」ウィンドウからスライスをドラッグできる場合、またはオブジェクトリストからメタデバイスをドラッグできる場合には、警告ダイアログボックスが表示されます。ここで「継続」をクリックします。

注 - /etc/vfstab ファイルにファイルシステムのエントリが存在し、そのファイルシステムが現在マウントされている場合、DiskSuite ツールはこれを自動的に更新し、トランスメタデバイスの名前を使用します。

5. ロギングデバイスが収められるスライスまたはメタデバイスを選択し、それをトランスメタデバイステンプレートのログ矩形にドラッグする。

ロギングデバイスがミラー化されていない場合、警告ダイアログボックスが表示されます。

6. トランスメタデバイスオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。

7. トランスメタデバイス上のファイルシステムをマウント解除してから、再マウントする。

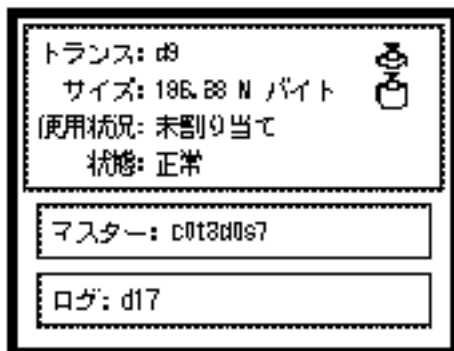
システムを再マウントすると、ファイルシステムのロギングが有効になる。それ以降のリブートでは、ファイルシステムをチェックする代わりに、fsck(1M) が次のメッセージを表示します。

```
/dev/md/rdisk/dx: is logging.
```

8. コンフィグレーションログを表示して、トランスメタデバイスが確定されたことを確認する。

例 - 確定済みのトランスメタデバイスオブジェクト

この例では、マスターデバイス用のスライス c0t3d0s7 とロギングデバイス用のミラー d17 から成る、確定済みのトランスメタデバイス d9 を示します。トランスオブジェクトの内部でマスターデバイスとロギングデバイスがどう表示されるかに注目してください。



▼ マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 98ページの「トランスメタデバイスを作成するための予備情報」の予備情報をチェックしてから、metainit(1M) コマンドでトランスメタデバイスを作成します。詳細については、metainit(1M) のマニュアルページを参照してください。

例 - マスターデバイス用のスライスでトランスメタデバイスを作成

```
# umount /home1
# metainit d63 -t c0t2d0s2 c2t2d0s1
d63: Trans is setup
(ファイルシステムがトランスメタデバイスを参照するよう、/etc/vfstab ファイルを編集する)
the trans metadvice)
# mount /home1
```

スライス /dev/dsk/c0t2d0s2 には、/home1 にマウントされたファイルシステムが収められています。ロギングデバイスを取めるスライスは、/dev/dsk/c2t2d0s1

です。最初に、ファイルシステムがマウント解除されます。-t オプション付きの metainit コマンドは、トランスメタデバイス d63 を作成します。

次に、/etc/vfstab ファイルを編集し、トランスメタデバイスを参照するよう、ファイルシステムのエントリを変更しなければなりません。たとえば、次の行は、

```
/dev/dsk/c0t2d0s2 /dev/rdisk/c0t2d0s2 /home1 ufs 2 yes -
```

次のように変更してください。

```
/dev/md/dsk/d63 /dev/md/rdsk/d63 /home1 ufs 2 yes -
```

ファイルシステムが再マウントされると、ファイルシステムのロギングが有効になります。

それ以降、リブート時に fsck(1M) がファイルシステムをチェックする代わりにトランスメタデバイス用のロギングメッセージを表示します。

```
# reboot
...
/dev/md/rdsk/d63: is logging
```

例 - マスターデバイス用のストライプでトランスメタデバイスを作成

```
# umount /home2
# metainit d40 -t d2 c1t2d0s0
d40: Trans is setup
(ファイルシステムがトランスメタデバイスを参照するよう、/etc/vfstab ファイルを編集する)
# mount /home2
```

d2 には、/home2 にマウントされたファイルシステムが収められています。ロギングデバイスを取めるスライスは、/dev/dsk/c1t2d0s0 です。最初に、ファイルシステムがマウント解除されます。-t オプション付きの metainit コマンドが、トランスメタデバイス d40 を作成します。

次に、/etc/vfstab ファイルを編集し、トランスメタデバイスを参照するよう、ファイルシステムのエントリを変更しなければなりません。たとえば、次の行は、

```
/dev/md/dsk/d2 /dev/md/rdsk/d2 /home2 ufs 2 yes -
```

次のように変更してください。

```
/dev/md/dsk/d40 /dev/md/rdsk/d40 /home2 ufs 2 yes -
```

ファイルシステムが再マウントされると、ファイルシステムのロギングが有効になります。

それ以降のリブートでは、ファイルシステムをチェックする代わりに、`fsck(1M)` はメタデバイス用のロギングメッセージを表示します。

```
# reboot
...
/dev/md/rdisk/d40: is logging
```

▼ マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)

この作業は、通常のシステム操作中にはマウント解除が不可能な `/usr` などのファイルシステムを記録するために使用します。

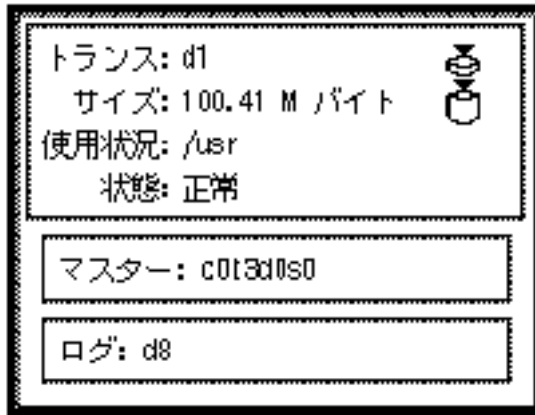
1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (98ページの「トランスメタデバイスを作成するための予備情報」) を読んでいることを確認する。
2. トランスメタデバイステンプレートをクリックする。
キャンバスには、未割り当てで未確定のトランスメタデバイスオブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
3. [オプション] デフォルトのメタデバイス名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。
 - c. 「閉じる」をクリックする。
4. スライスをクリックして「スライスブラウザ」ウィンドウを開く。
5. ログに記録されるファイルシステムが収められているスライスまたはメタデバイスを選択し、それをトランスメタデバイスオブジェクト内のマスター矩形にドラッグする。

これは、ファイルシステムが収められているスライスまたはメタデバイスでなければなりません。

6. マスターとなるデバイスを確認する。
表示されるダイアログボックスの「継続」をクリックします。
7. ログインデバイスが収められるスライスまたはメタデバイスを選択し、それをトランスメタデバイスオブジェクトのログ矩形にドラッグする。
8. ログとなるスライスを確認する。
表示されるダイアログボックスの「継続」をクリックします。
9. トランスメタデバイスオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
表示される確認ダイアログボックスの「確定」をクリックします。
10. コンフィグレーションログを表示して、トランスメタデバイスが確定されたことを確認する。
11. リブートする。
リブートした後、ファイルシステムのログインが有効になります。

例 - /usr 用の確定済みトランスオブジェクト

この例では、/usr ファイルシステムが収められているトランスメタデバイスを示します。



▼ マウント解除不可能なファイルシステム用のトランスメタデバイスの作成方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 98ページの「トランスメタデバイスを作成するための予備情報」の予備情報をチェックしてから、通常システム操作中にはマウント解除不可能な /usr などのファイルシステムのログを取るためにこの作業を使用します。詳細については、metainit(1M) のマニュアルページを参照してください。

例 - /usr 用のトランスメタデバイスの作成

```
# metainit -f d20 -t c0t3d0s6 c1t2d0s1
d20: Trans is setup
(ファイルシステムがトランスメタデバイスを参照するよう、/etc/vfstab ファイルを編集する)
# reboot
```

スライス /dev/dsk/c0t3d0s6 には、/usr ファイルシステムが収められています。ロギングデバイスが収められるスライスは、/dev/dsk/c1t2d0s1 です。/usr はマウント解除できないため、-f オプションを付けて metainit コマンドを実行し、トランスデバイス d20 を作成します。次に、/etc/vfstab ファイル内でファイルシステムをマウントする行を変更して、トランスメタデバイスを参照するようにします。

たとえば、次の行は、

```
/dev/dsk/c0t3d0s6 /dev/rdisk/c0t3d0s6 /usr ufs 1 no -
```

次のように変更してください。

```
/dev/md/dsk/d20 /dev/md/rdsk/d20 /usr ufs 1 no -
```

システムがリブートされると、ファイルシステムのロギングが有効になります。

▼ ミラーを使用してトランスメタデバイスを作成する方法 (DiskSuite ツール)

トランスメタデバイスのデータ可用性を高めるには、マスターデバイスとロギングデバイスにミラーを使用します。ロギングスライスにエラーが発生した場合、ロギングデバイスをミラー化していなければ、重要なデータが失われることとなります。ロギングデバイスをミラー化している場合、マスターデバイスもミラーとすることをお勧めします。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (98ページの「トランスメタデバイスを作成するための予備情報」) を読んでいることを確認する。
2. ミラーの作成に関する作業を参照する (71ページの「未使用スライスからミラーを作成する方法 (DiskSuite ツール)」 ~ 83ページの「マウント解除不可能なファイルシステムからミラーを作成する方法 (コマンド行)」)。
マスターデバイスとロギングデバイスに 1 つずつ、合計 2 つのミラーを作成します。マスターデバイス用のミラーには、ファイルシステムが含まれていなければなりません。
3. 99ページの「マウント解除可能なファイルシステム用のトランスメタデバイスの作成方法 (DiskSuite ツール)」を参照する。
マスターデバイスとロギングデバイスには、手順 2 で作成されたミラーを使用します。

▼ ミラーを使用してトランスメタデバイスを作成する方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 98ページの「トランスメタデバイスを作成するための予備情報」の予備情報をチェックしてから、metainit(1M) を使用してトランスメタデバイスを作成します。

例 - ミラーを使用してトランスメタデバイスを作成する方法

```
# umount /home1
# metainit d64 -t d30 d12
d64: Trans is setup
(ファイルシステムがトランスメタデバイスを参照するよう、/etc/vfstab ファイルを編集する)
# mount /home1
```

ミラー d30 には、/home1 にマウントされたファイルシステムが収められています。ロギングデバイスを収めるミラーは d12 です。最初に、ファイルシステムがマウント解除されます。-t オプションを付けた metainit コマンドは、トランスメタデバイス d64 を作成します。

次に、/etc/vfstab ファイル内でファイルシステムをマウントする行を変更して、トランスメタデバイスを参照するようにします。たとえば、次の行は、

```
/dev/md/dsk/d30 /dev/md/rdisk/d30 /home1 ufs 2 yes -
```

次のように変更してください。

```
/dev/md/dsk/d64 /dev/md/rdisk/d64 /home1 ufs 2 yes -
```

ファイルシステムが再マウントされると、ファイルシステムのロギングが有効になります。

以降のファイルシステムの再マウントやシステムのリブートでは、ファイルシステムをチェックする代わりに、fsck(1M) がメタデバイス用のロギングメッセージを表示します。

```
# reboot
...
/dev/md/rdisk/d64: is logging
```

ホットスペア集合の作成

この節では、ホットスペア集合の作成方法、ホットスペア集合にスライスを追加する方法、ホットスペア集合をメタデバイスに割り当てる方法について説明します。

ホットスペア集合を作成するための予備情報

- ホットスペア集合とは、DiskSuite によって予約されたスライス (ホットスペア) の集まりです。サブミラーや RAID5 メタデバイスのスライスに障害が発生した場合、自動的に交換されます。ホットスペアは、ミラーや RAID5 メタデバイスのデータ可用性を高めます。
- ホットスペアはスライスでなければなりません。つまり、メタデバイスをホットスペアとすることはできません。
- ホットスペア集合は、ミラーと RAID5 メタデバイスで利用されます。ミラーの場合、ホットスペア集合はサブミラーに関連付けられます。
- ホットスペア用のスライスは、複数のホットスペア集合に対して割り当てることができます。
- ホットスペア集合は、複数のサブミラーや RAID5 メタデバイスと関連付けることができます。ただし、サブミラーや RAID5 メタデバイスは、1つのホットスペア集合とだけ関連付けることができます。
- 交換の際には、ホットスペア集合で指定された順序に従って、障害の発生したスライスに最初に適合するものが用いられます。
- ホットスペアは、交換されたままの状態を継続して利用することを想定していません。したがって、修理したスライスや新しいスライスで置き換える必要があります。
- ホットスペアには、状態データベースの複製を置くことができません。
- ホットスペア集合に追加されたスライスは、別のコントローラに接続するのが理想的です。こうすることによって、コントローラのエラーや障害時にもデータの可用性が保証されます。
- 誤ったサイズのホットスペアをサブミラーや RAID5 メタデバイスと関連付けしないでください。
- 1面ミラーのサブミラーには、ホットスペア集合を割り当てないでください。
- ホットスペア集合を構成しているホットスペアのすべてが使用中の状態となることのないようにしてください。こうなった場合、ホットスペア集合に新たなホットスペアを追加するか、ホットスペア交換されたスライスを修復します。
- ホットスペア集合はいつでも、割り当てたり、割り当て解除したり、再割り当てすることができます。ただし、ホットスペア集合内のスライスが、それに関連付けられたメタデバイス内でエラーの発生したスライスを交換するために使用されている場合を除きます。

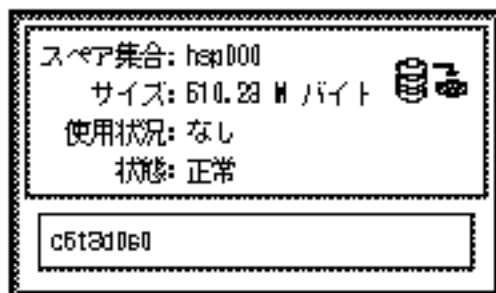
▼ ホットスペア集合の作成方法 (DiskSuite ツール)

空のホットスペア集合を作成し、必要に応じて、後でホットスペアを追加することができます。その場合は、次の作業の手順4 と手順5 を飛ばします。

1. **前提条件** (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、**予備情報** (108ページの「ホットスペア集合を作成するための予備情報」) を読んでいることを確認する。
2. ホットスペア集合テンプレートをクリックする。
キャンバスには、未割り当てで未確定のホットスペア集合オブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
3. **[オプション]** デフォルトのホットスペア集合名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。
 - c. 「閉じる」をクリックする。
4. スライスをクリックし、「スライスブラウザ」ウィンドウを開く。
5. スライスを選択して、ホットスペア集合オブジェクトにドラッグする。
複数のスライスを選択するには、Control キーを押しながらクリックします。
6. ホットスペア集合オブジェクトの先頭の矩形をクリックしてから、「確定」をクリックする。
7. コンフィグレーションログを表示して、ホットスペア集合が確定されたことを確認する。

例- 確定済みのホットスペア集合オブジェクト

この例では、スライス c5t3d0s0 から成る、確定済みのホットスペア集合オブジェクト hsp000 を示します。



次の作業

ホットスペアをホットスペア集合に追加するには、114ページの「ホットスペア集合にホットスペアスライスを追加する方法 (DiskSuite ツール)」を参照してください。ホットスペア集合を作成した後で、これとサブミラーまたは RAID5 メタデバイスを関連付ける必要があります。111ページの「ホットスペア集合を関連付ける方法 (DiskSuite ツール)」を参照してください。

▼ ホットスペア集合の作成方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 108ページの「ホットスペア集合を作成するための予備情報」の予備情報をチェックしてから、`metainit(1M)` コマンドを使用してホットスペア集合を作成します。詳細については、`metainit(1M)` のマニュアルページを参照してください。

例 - ホットスペア集合の作成

```
# metainit hsp001 c2t2d0s2 c3t2d0s2  
hsp001: Hotspare pool is setup
```

ホットスペア集合 `hsp001` には、ホットスペアとして 2 つのディスクが含まれています。システムは、ホットスペア集合が設定されたことを確認します。

次の作業

ホットスペア集合にさらにホットスペアを追加するには、115ページの「ホットスペア集合にホットスペアスライスを追加する方法 (コマンド行)」を参照してくだ

さい。ホットスペア集合を作成した後で、サブミラーまたは RAID5 メタデバイスと関連付ける必要があります。112ページの「ホットスペア集合を関連付ける方法 (コマンド行)」を参照してください。


▼ ホットスペア集合を関連付ける方法 (DiskSuite ツール)


サブミラーや RAID5 メタデバイスと関連付けるためには、ホットスペア集合が存在しなければなりません。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (108ページの「ホットスペア集合を作成するための予備情報」) を読んでいることを確認する。
2. オブジェクトリスト内のミラーまたは **RAID5** メタデバイスをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. 「ブラウズ」メニューから「ホットスペア集合」を選択し、リストからホットスペア集合オブジェクトを選択する。
ホットスペア集合には、ミラー内のサブミラーのサイズまたは RAID5 メタデバイス内のスライスのサイズ以上のサイズをもつスライスが必要です。ホットスペア集合では、メタデバイス内のスライスとは異なるコントローラ上にホットスペアを置くのが理想です。
4. メタデバイスオブジェクトの先頭にホットスペア集合オブジェクトをドラッグする。
5. メタデバイスオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
6. コンフィグレーションログを表示して、ホットスペア集合が関連付けられたことを確認する。

例 - ホットスペア集合の関連付け

この例では、2つのサブミラーと関連付けられたホットスペア集合を示します。

スペア集合: hsp001 サイズ: 989.30 M バイト 使用状況: 共有しているデバイス数 2 状態: 正常	
c6t3d3s0	

ミラー: d7 サイズ: 989.30 M バイト 使用状況: /fs1 状態: 正常		
連結: d6 サイズ: 989.30 M バイト 状態: 正常	連結: d8 サイズ: 989.30 M バイト 状態: 正常	
d6 のストライプ 0	d8 のストライプ 0	
c5t5d2s0	c7t5d4s0	
スペア集合: hsp001	スペア集合: hsp001	

▼ ホットスペア集合を関連付ける方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と108ページの「ホットスペア集合を作成するための予備情報」の予備情報をチェックしてから、metaparam(1M) コマンドを使用してホットスペア集合を関連付けます。詳細については、metaparam(1M) のマニュアルページを参照してください。

例 - ホットスペア集合とサブミラーの関連付け

```
# metaparam -h hsp100 d10
# metaparam -h hsp100 d11
# metastat d0
d0: Mirror
   Submirror 0: d10
     State: Okay
   Submirror 1: d11
     State: Okay
...

d10: Submirror of d0
     State: Okay
     Hot spare pool: hsp100
...

d11: Submirror of d0
     State: Okay
     Hot spare pool: hsp100
...
```

-h オプションは、ホットスペア集合 hsp100 と、ミラー d0 の 2 つのサブミラー d10 と d11 を関連付けます。metastat コマンドは、ホットスペア集合がサブミラーと関連付けられていることを示しています。

例- ホットスペア集合と RAID5 メタデバイスの関連付け

```
# metaparam -h hsp001 d10
# metastat d10
d10: RAID
     State: Okay
     Hot spare pool: hsp001
...
```

-h オプションは、hsp001 という名前のホットスペア集合と、d10 という名前の RAID5 メタデバイスを関連付けます。metastat コマンドは、ホットスペア集合が RAID5 メタデバイスと関連付けられていることを示しています。

▼ ホットスペア集合にホットスペアスライスを追加する方法 (DiskSuite ツール)

1 つ以上のホットスペア集合にスライスを追加することができます。ホットスペアが追加されても、ホットスペアの現在の順序は維持されます。新しいホットスペアは、指定されたホットスペア集合のホットスペアリストの最後に追加されます。

1. **前提条件** (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、**予備情報** (108ページの「ホットスペア集合を作成するための予備情報」) を読んでいることを確認する。
2. オブジェクトリスト内の既存のホットスペア集合オブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. スライスをクリックし、「スライスブラウザ」ウィンドウを開く。
4. スライスを選択してから、ホットスペア集合オブジェクトにドラッグする。
ホットスペア集合にすでにスライスが含まれている場合、ホットスペア集合内のスライス以上のサイズのスライスを探します。
5. ホットスペア集合オブジェクトの先頭の矩形をクリックしてから、「確定」をクリックする。
6. コンフィグレーションログを表示して、ホットスペア集合が確定されたことを確認する。

例 - 2つのスライスをもつホットスペア集合

この例では、最初は 1 つのスライス `c6t3d3s0` で構成されていたホットスペア集合を示します。スライス `c4t5d1s0` が追加され、ホットスペア集合が確定されました。



▼ ホットスペア集合にホットスペアスライスを追加する方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 108ページの「ホットスペア集合を作成するための予備情報」の予備情報をチェックしてから、`metahs(1M)` コマンドを使用してホットスペア集合にスライスを追加します。詳細については、`metahs(1M)` のマニュアルページを参照してください。

注 - 追加されたホットスペアは、ホットスペア集合内にすでに存在するホットスペアの後に続きます。

例 - 1 つのホットスペア集合にホットスペアスライスを追加

```
# metahs -a hsp001 /dev/dsk/c3t0d0s2
hsp001: Hotspare is added
```

`-a` オプションは、ホットスペア集合 `hsp001` にスライス `/dev/dsk/c3t0d0s2` を追加します。システムは、ホットスペア集合にスライスが追加されたことを確認します。

例 - すべてのホットスペア集合にホットスペアスライスを追加

```
# metahs -a -a11 /dev/dsk/c3t0d0s2
hsp001: Hotspare is added
hsp002: Hotspare is added
hsp003: Hotspare is added
```

-a と -a11 のオプションは、システム上で構成されているすべてのホットスペア集合に、スライス /dev/dsk/c3t0d0s2 を追加します。システムは、すべてのホットスペア集合にスライスが追加されたことを確認します。

▼ 関連付けられたホットスペア集合の変更方法 (DiskSuite ツール)

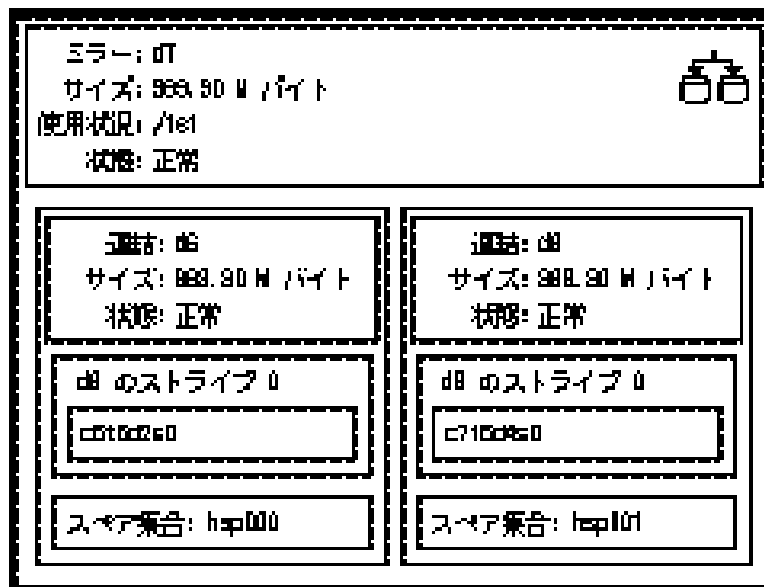
ホットスペア集合の関連付けは、そのスライスのいずれもホットスペアとして現在使用されていない限り、いつでも変更できます。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (108ページの「ホットスペア集合を作成するための予備情報」) を読んでいることを確認する。
2. オブジェクトリストにおいて、ホットスペア集合の関連付けを変更するミラーまたは **RAID5** オブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. オブジェクトリストにおいて、交換に使用するホットスペア集合オブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
4. 交換されるホットスペア集合が収められているサブミラーまたは **RAID5** オブジェクトの矩形に、ホットスペア集合オブジェクトをドラッグする。
5. オブジェクトの先頭の矩形内部をクリックしてから、「確定」をクリックする。
6. コンフィグレーションログを表示して、ホットスペア集合が確定されたことを確認する。

注 - 誤ってデータが生成されるのを回避するため、メタデバイス内に「最後にエラーが発生した状態」のスライスが含まれる場合には、DiskSuite はメタデバイスのホットスペアリングを許可しません。詳細については、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」を参照してください。

例 - ホットスペア集合の関連付けがあるミラーオブジェクト

この例では、各サブミラーと関連付けられたホットスペア集合 hsp000 で始まるミラーを示します。現在サブミラー d8 は、hsp001 と関連付けられています。



▼ 関連付けられたホットスペア集合の変更方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 108ページの「ホットスペア集合を作成するための予備情報」の予備情報をチェックしてから、metaparam(1M) コマンドを使用して、メタデバイスの関連付けられたホットスペア集合を変更します。詳細については、metaparam(1M) のマニュアルページを参照してください。

例 - ホットスペア集合の関連付けを変更

```
# metastat d4
d4: RAID
    State: Okay
    Hot spare pool: hsp001
...
# metaparam -h hsp002 d4
# metastat d4
d4: RAID
    State: Okay
    Hot spare pool: hsp002
...
```

この例では、現在、ホットスペア集合 hsp001 は、d4 という名前の RAID5 メタデバイスと関連付けられています。ホットスペア集合の関連付けは、hsp002 に変更されます。metastat コマンドは、その前後におけるホットスペア集合の関連付けを示しています。

ディスクセットの作成

この節では、ディスクセットの作成方法、およびそれにホストとディスクドライブを実装する方法について説明します。

注 - 現在、ディスクセットは、SPARCstorage Array ディスクでのみサポートされません。

ディスクセットを作成するための予備情報

- ディスクセットとは、1つまたは2つのホストで (同時にではなく) 排他的に共有が可能な DiskSuite オブジェクトが取められている共有ディスクドライブの場合です。ディスクセットによって、ホストの障害回復シナリオが実行可能となります。
- ディスクセットに接続される各ホストには、DiskSuite をインストールしなければなりません。
- メタデバイス状態データベースは、共有ディスクセットごとに1つ、および「ローカル」ディスクセットに1つ存在します。ディスクセットを作成するに

は、各ホストにローカルなメタデバイス状態データベースが設定されていなければなりません。

- ローカルなメタデバイス管理とは異なり、ディスクセット上で状態データベースの複製を手作業で作成したり削除したりする必要はありません。DiskSuite は、ディスクセット内のすべてのドライブを通じて、適切な数の複製を均一に配置しようとしています。
- ディスクセット内の各ホストには、共有ディスクセットとは別にローカルディスクセットが必要です。ホスト用のローカルディスクセットは、共有ディスクセットに含まれないすべてのドライブから構成されます。
- ディスクセット内のホスト間で共有しようとするすべてのディスクは、各ホストに接続し、各ホスト上で同じ名前になっていなければなりません。ディスクセット構成で使用するためのハードウェアの設定は、少々やっかいです。ディスクドライブは対称でなければなりません。つまり、共有ドライブのデバイス番号は同じでなければなりません。このことは、デバイスの名前 / 番号 (コントローラ / ターゲット / ドライブ) が同じであることを暗示します。372ページの「ディスクセット用のディスクドライブデバイス名を設定する方法(コマンド行)」を参照してください。
- 1つのホストをもつディスクセットを作成し、後から2番目のホストを追加することができます。
- 使用中のドライブをディスクセットに追加することはできません。ドライブを追加する前に、そのドライブがファイルシステム、データベース、またはその他のアプリケーションで現在使用されていないことを確認してください。
- ディスクセットに対して、保存する必要のあるデータをもつドライブを追加しないでください。これをディスクセットに追加すると、ディスクのパーティション再分割が行われ、すべてのデータが破壊されます。
- ドライブがディスクセットとして受け付けられると、ディスクセット用のメタデバイス状態データベースの複製をドライブに置けるように、DiskSuiteがこれらのパーティションを再分割します。スライス7が正しく設定されていない場合にだけ、ドライブのディスクセットへの追加時に、ドライブのパーティション再分割が行われます。各ドライブのスライス7に、DiskSuiteで使用するための小さな領域が確保されます。各ドライブの残りの領域は、スライス0になります。ディスク上のデータは、パーティション再分割によって失われます。ディスクセットにドライブを追加した後、必要に応じてパーティションを再分割できます。ただし、スライス7は例外であり、いかなる方法でも変更できません。スライス7がシリンダ0から始まり、しかも状態データベースの複製を収めるだけの大きさがある場合、ディスクはパーティションに再分割されません。

- ディスクセットにドライブが追加されると、DiskSuite は、残りのドライブ全体にわたって、状態データベースの複製のバランスを再調整します。必要ならば、後から `metadb(1M)` コマンドを使用して、複製の配置を変更できます。
- ディスクセットを作成するには、`root` がグループ 14 のメンバーであるか、または `./rhosts` ファイルに (各ホスト上の) 他のホスト名のエントリが含まれている必要があります。

注 - ディスクセットの作成と構成には、DiskSuite のコマンド行インタフェースを使用する必要があります。ディスクセットを作成したら、DiskSuite ツールまたはコマンド行ユーティリティを使用して、ディスクセット内でホットスペア集合、メタデバイス、状態データベースの複製を管理できます。

▼ ディスクセットの作成方法 (コマンド行)

ディスクセットを作成する手順を次に示します。

- ホストをディスクセットに追加 (これでディスクセットを作成)
- ドライブをディスクセットに追加

ディスクセットを作成する前に、次の作業を行います。

- ディスクセット内の各ホストでディスクドライブのデバイス名が同じになるよう、ディスクドライブのデバイス名を設定する。
- 共有されるドライブを両方のホストに接続する。
- 各ホスト上で、ローカルメタデバイスの状態データベースの複製を構成する。

注 - ディスクセット内の共有ドライブに同じデバイス名を設定する方法については、372ページの「ディスクセット用のディスクドライブデバイス名を設定する方法 (コマンド行)」を参照してください。

1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (118ページの「ディスクセットを作成するための予備情報」) を読んでいることを確認する。
2. デバイスにアクセスできるホストを定義することによって、ディスクセットを作成する。

```
# metaset -s <セット名> -a -h <ホスト>...
```


このコマンドでは、

<code>-s <セット名></code>	<code>metaset(1M)</code> が作用するディスクセットの名前を指定します。
<code>-a</code>	指定されたディスクセットにホストを追加します。 <code>DiskSuite</code> では、ディスクセットあたり最大 2 つのホストをサポートします。
<code>-h <ホスト>...</code>	ディスクセットに追加される 1 つ以上のホストを指定します。最初のホストを追加すると、セットが作成されます。2 番目のホストは後で追加できますが、指定されたホスト上でセット内のすべてのドライブが見つからない場合、これは受け付けられません。<ホスト>には、 <code>/etc/nodename</code> にあるものと同じ名前を指定します。

3. `metaset(1M)` コマンドにより、新しいディスクセットの状態をチェックする。

```
# metaset
```

例 - 2 つのディスクセットを作成

```
red# metaset -s relo-red -a -h red blue
red# metaset -s relo-blue -a -h red blue
red# metaset
Set name = relo-red, Set number = 1

Host                Owner
  red
  blue

Set name = relo-blue, Set number = 2

Host                Owner
  red
  blue
```

この例では、ホスト `red` から、`relo-red` と `relo-blue` の 2 つの共有ディスクセットを作成します。ホスト名は `red` と `blue` であり、それぞれに専用のローカルディスクセットを保有しています。`metaset` コマンドで状態が表示されます。この時点では、どちらのセットにも所有者はありません。セットにディスクを追加するホストは、デフォルトで所有者となります。

▼ ディスクセットにドライブを追加する方法 (コマンド行)

ドライブがディスクセットに受け付けられるためには、次の条件が満足されなければなりません。

- メタデバイス内で使用されていたり、状態データベースの複製を含んでいてはならない。
 - 現在マウントされていたり、スワップされていたり、アプリケーションでの使用のために開かれていてはならない。
1. 前提条件 (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、予備情報 (118ページの「ディスクセットを作成するための予備情報」) を読んでいることを確認する。
 2. ドライブをディスクセットに追加する。

```
# metaset -s <セット名> -a <ドライブ名>...
```

このコマンドでは、

-s <セット名>	metaset(1M) が作用するディスクセットの名前を指定します。
-a	指定されたディスクセットにドライブを追加します。
<ドライブ名>...	ディスクセットに追加するドライブを指定します。ドライブ名の形式は cxtidx です。名前の最後に、「sx」スライス識別子は存在しません。これらは、ディスクセット内のすべてのホストで同じある必要があります。

ディスクセットにドライブを追加する最初のホストが、ディスクセットの暗黙の所有者となります。



注意 - データのあるディスクを追加しないでください。これをディスクセットに追加すると、ディスクのパーティションが再分割され、すべてのデータが破壊されます。

3. `metaset(1M)` コマンドを使用して、ディスクセットとドライブの状態を確認する。

```
# metaset
```

例 - ディスクセットにドライブを追加

```
red# metaset -s relo-red -a c1t2d0 c1t3d0 c2t2d0 c2t3d0 c2t4d0 c2t5d0
red# metaset
Set name = relo-red, Set number = 1

Host                Owner
red                 Yes
blue

Drive              Dbase
c1t2d0             Yes
c1t3d0             Yes
c2t2d0             Yes
c2t3d0             Yes
c2t4d0             Yes
c2t5d0             Yes

Set name = relo-blue, Set number = 2

Host                Owner
red
blue
```

この例では、ホスト名は `red` と `blue` であり、それぞれ専用のローカルディスクセットを保有します。共有された2つのディスクセットは、`relo-red` と `relo-blue` です。セット `relo-red` 内のディスクは、通常はホスト `red` によってアクセスされますが、`red` に障害が発生した場合、ホスト `blue` からアクセスできます。この時点では、ディスクセット `relo-blue` には、まだディスクが追加されていません。

ディスクセット内に **DiskSuite** オブジェクトを作成

ディスクセットを作成したら、ディスクセットに追加したドライブを使用して、メタデバイスとホットスペア集合を作成できます。DiskSuite ツールまたはコマンド行ユーティリティを使用できます。

▼ ディスクセット内に **DiskSuite** オブジェクトを作成する方法 (DiskSuite ツール)

DiskSuite ツールを使用してディスクセット内に DiskSuite オブジェクトを作成するには、自分がディスクセットの所有者であることを確認し、次のコマンド行を入力します。

```
# metatool -s <ディスクセット名> &
```

この章に記載の DiskSuite ツールにおける作業を使用して、ミラーやトランスメタデバイスなどの DiskSuite オブジェクトを作成します。

▼ ディスクセット内に **DiskSuite** オブジェクトを作成する方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件をチェックしてから、metainit(1M) コマンドを使用して、ディスクセット内に DiskSuite オブジェクトを作成します。詳細は、metainit(1M) のマニュアルページを参照してください。

例 - ディスクセット内にミラーを作成

```
# metainit -s relo-red d51 1 1 /dev/dsk/c0t0d0s2
relo-red/d51: Concat/Stripe is setup
# metainit -s relo-red d52 1 1 /dev/dsk/c1t0d0s2
relo-red/d52: Concat/Stripe is setup
# metainit -s relo-red d50 -m d51
relo-red/d50: mirror is setup
# metattach -s relo-red d50 d52
```

(続く)

```
relo-red/d50: Submirror d52 is attached
```

この例では、ディスクセット `relo-red` にミラー `d50` を作成します。

メタデバイス上にファイルシステムを作成

この節では、メタデバイス上に新しいファイルシステムを作成する方法について説明します。

メタデバイス上にファイルシステムを作成するための予備情報

- 新しく作成されたメタデバイス上にファイルシステムを作成するには、`newfs(1M)` コマンドまたはファイルシステムマネージャを使用します。
- ストライプ、連結、ミラー、RAID5、またはトランスメタデバイス上にファイルシステムを作成できます。

▼ メタデバイス上にファイルシステムを作成する方法 (ファイルシステムマネージャ)

次の作業では、操作するホストに `Solstice` ストレージマネージャがインストールしており、ストレージマネージャアプリケーションが起動されているものと想定します。詳細は、付録 A を参照してください。

1. **前提条件** (57ページの「DiskSuite オブジェクトを作成するための前提条件」) を満たし、**予備情報** (125ページの「メタデバイス上にファイルシステムを作成するための予備情報」) を読んでいることを確認する。
2. ファイルシステムマネージャを起動する。
DiskSuite ツールを正しく構成していれば、DiskSuite ツールの「ツール」メニューからファイルシステムマネージャを起動できます。

それ以外の場合は、まず、Solstice 起動ツールからストレージマネージャを起動します。

注 - ストレージマネージャを操作できるよう DiskSuite を設定するには、289 ページの「DiskSuite を有効にしてストレージマネージャを起動する方法 (コマンド行)」を参照してください。

3. ファイルシステムマネージャのメインウィンドウで、「動作」メニューから「作成」->「ファイルシステム」を選択し、「新ファイルシステム」プロパティブックを表示する。
4. ウィンドウのプロパティビューア部分のデバイスをクリックする。
5. **DiskSuite** ツールの「メタデバイスエディタ」ウィンドウからメタデバイスを選択し、「デバイス名」テキストフィールドにドラッグする。
6. 「適用」をクリックする。
ファイルシステムマネージャは、メタデバイス上に新しいファイルシステムを作成します。

▼ メタデバイス上にファイルシステムを作成する方法 (コマンド行)

57ページの「DiskSuite オブジェクトを作成するための前提条件」の前提条件と 125ページの「メタデバイス上にファイルシステムを作成するための予備情報」の予備情報をチェックしてから、`newfs(1M)` コマンドを使用して、メタデバイス上に新しいファイルシステムを作成します。詳細については、`newfs(1M)` のマニュアルページを参照してください。

例 - 連結上にファイルシステムを作成

```
# newfs /dev/md/rdisk/d3
newfs: construct a new file system /dev/md/rdisk/d3: (y/n)? y
/dev/md/rdisk/
d3:          917280 sectors in 1638 cylinders of 7 tracks, 80 sectors
```

(続く)

```

          447.9MB in 103 cyl groups (16 c/g, 4.38MB/g, 2112 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 9072, 18112, 27152, 36192, 45232, 54272, 63312, 71712, 80752, 89792,
98832, 107872, 116912, 125952, 134992, 143392, 152432, 161472, 170512,
179552, 188592, 197632, 206672, 215072, 224112, 233152, 242192, 251232,
260272, 269312, 278352, 286752, 295792, 304832, 313872, 322912, 331952,
340992, 350032, 358432, 367472, 376512, 385552, 394592, 403632, 412672,
421712, 430112, 439152, 448192, 457232, 466272, 475312, 484352, 493392,
501792, 510832, 519872, 528912, 537952, 546992, 556032, 565072, 573472,
582512, 591552, 600592, 609632, 618672, 627712, 636752, 645152, 654192,
663232, 672272, 681312, 690352, 699392, 708432, 716832, 725872, 734912,
743952, 752992, 762032, 771072, 780112, 788512, 797552, 806592, 815632,
824672, 833712, 842752, 851792, 860192, 869232, 878272, 887312, 896352,
905392, 914432,
# fsck /dev/md/rdsk/d3
** /dev/md/rdsk/d3
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
2 files, 9 used, 942654 free (14 frags, 117830 blocks, 0.0% fragmentation)

```

この例では、d3 という名前の連結上にファイルシステムを作成します。fsck(1M) コマンドで、新しいファイルシステムを確認します。

例 - ディスクセット内部で連結上にファイルシステムを作成

```

# newfs /dev/md/relo-red/rdsk/d33
newfs: construct a new file system /dev/md/relo-red/rdsk/d33: (y/n)? y
/dev/md/relo-red/rdsk/
d33:          917280 sectors in 1638 cylinders of 7 tracks,
80 sectors
          447.9MB in 103 cyl groups (16 c/g, 4.38MB/g, 2112 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 9072, 18112, 27152, 36192, 45232, 54272, 63312, 71712, 80752, 89792,
98832, 107872, 116912, 125952, 134992, 143392, 152432, 161472, 170512,
179552, 188592, 197632, 206672, 215072, 224112, 233152, 242192, 251232,
260272, 269312, 278352, 286752, 295792, 304832, 313872, 322912, 331952,
340992, 350032, 358432, 367472, 376512, 385552, 394592, 403632, 412672,
421712, 430112, 439152, 448192, 457232, 466272, 475312, 484352, 493392,
501792, 510832, 519872, 528912, 537952, 546992, 556032, 565072, 573472,
582512, 591552, 600592, 609632, 618672, 627712, 636752, 645152, 654192,
663232, 672272, 681312, 690352, 699392, 708432, 716832, 725872, 734912,
743952, 752992, 762032, 771072, 780112, 788512, 797552, 806592, 815632,
824672, 833712, 842752, 851792, 860192, 869232, 878272, 887312, 896352,

```

(続く)

続き

```
905392, 914432,
```

この例では、ディスクセット `relo-red` 内の連結名 `d33` の上にファイルシステムを作成します。

注 - ディスクセット中のメタデバイス上に存在するファイルシステムは、`/etc/vfstab` ファイルを介してブート時に自動的にマウントできません。必要なディスクセット RPC デーモン (`rpc.metad` と `rpc.metamhd`) の起動が、ブートプロセスの後の方で行われるためです。またリブート中は、ディスクセットの所有権が失われます。

次の作業

ファイルシステムを使用可能にしたい タイミング	操作
----------------------------	----

即時	<code>mount(1M)</code> コマンドでファイルシステムをマウントし、 <code>/dev/md/dsk/d30</code> のように、マウントデバイスとしてメタデバイス名を指定します。
システムのブート時に自動的に	メタデバイスブロック名と raw デバイス名を使用して、 <code>/etc/vfstab</code> ファイルに含まれるファイルシステムのエントリを作成または変更します。
共有 (エクスポートされた) 資源として	<code>/etc/dfs/dfstab</code> ファイル内にファイルシステムエントリを作成します。

DiskSuite オブジェクトの保守

この章では、DiskSuite ツールのグラフィカルユーザーインターフェースとコマンド行ユーティリティを使用して、DiskSuite オブジェクトを保守する方法について説明します。

DiskSuite ツールを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 134ページの「状態データベースの複製の状態のチェック方法 (DiskSuite ツール)」
- 136ページの「メタデバイスとホットスペア集合の状態のチェック方法 (DiskSuite ツール)」
- 156ページの「状態データベースの複製を有効にする方法 (DiskSuite ツール)」
- 157ページの「スライス障害の後でストライプや連結を再作成する方法 (DiskSuite ツール)」
- 164ページの「サブミラー内のスライスを有効にする方法 (DiskSuite ツール)」
- 166ページの「サブミラー内のスライスを交換する方法 (DiskSuite ツール)」
- 168ページの「サブミラーの交換方法 (DiskSuite ツール)」
- 170ページの「RAID5 メタデバイス内のスライスを有効にする方法 (DiskSuite ツール)」
- 171ページの「RAID5 スライスの交換方法 (DiskSuite ツール)」
- 174ページの「ホットスペア集合内のホットスペアを交換する方法 (DiskSuite ツール)」
- 176ページの「ホットスペアを有効にする方法 (DiskSuite ツール)」
- 184ページの「既存のデータを含むスライスの拡張方法 (DiskSuite ツール)」

- 187ページの「既存の連結方式の拡張方法 (DiskSuite ツール)」
- 191ページの「ミラーの拡張方法 (DiskSuite ツール)」
- 193ページの「RAID5 メタデバイスの拡張方法 (DiskSuite ツール)」
- 195ページの「トランスメタデバイスの拡張方法 (DiskSuite ツール)」
- 200ページの「メタデバイスのリネーム方法 (DiskSuite ツール)」
- 203ページの「ファイルシステムのミラー化を解除する方法 (DiskSuite ツール)」
- 208ページの「サブミラーの接続方法 (DiskSuite ツール)」
- 210ページの「サブミラーを切断する方法 (DiskSuite ツール)」
- 211ページの「サブミラーのオフライン / オンライン設定 (DiskSuite ツール)」

コマンド行インタフェースを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 135ページの「状態データベースの複製の状態のチェック方法 (コマンド行)」
- 144ページの「メタデバイスとホットスペア集合の状態のチェック方法 (コマンド行)」
- 153ページの「ディスクセットの状態のチェック方法 (コマンド行)」
- 158ページの「スライス障害の後でストライプや連結を再作成する方法 (コマンド行)」
- 165ページの「サブミラー内のスライスを有効にする方法 (コマンド行)」
- 167ページの「サブミラー内のスライスを交換する方法 (コマンド行)」
- 169ページの「サブミラーの交換方法 (コマンド行)」
- 171ページの「RAID5 メタデバイス内のスライスを有効にする方法 (コマンド行)」
- 175ページの「ホットスペア集合内のホットスペアの交換方法 (コマンド行)」
- 178ページの「ホットスペアを有効にする方法 (コマンド行)」
- 179ページの「ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行)」
- 179ページの「ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行)」
- 186ページの「既存のデータを含むスライスの拡張方法 (コマンド行)」
- 189ページの「既存のストライプの拡張方法 (コマンド行)」
- 192ページの「ミラーの拡張方法 (コマンド行)」

- 194ページの「RAID5 メタデバイスの拡張方法 (コマンド行)」
- 196ページの「トランスメタデバイスの拡張方法 (コマンド行)」
- 198ページの「ファイルシステムの拡張方法 (コマンド行)」
- 201ページの「メタデバイスのリネーム方法 (コマンド行)」
- 204ページの「ファイルシステムのミラー化を解除する方法 (コマンド行)」
- 209ページの「サブミラーの接続方法 (コマンド行)」
- 211ページの「サブミラーを切断する方法 (コマンド行)」
- 213ページの「サブミラーのオフライン / オンライン設定方法 (コマンド行)」
- 214ページの「ディスクセットの予約方法 (コマンド行)」
- 216ページの「ディスクセットを解放する方法 (コマンド行)」
- 217ページの「ディスクセットに新たなドライブを追加する方法 (コマンド行)」
- 219ページの「ディスクセットに別のホストを追加する方法 (コマンド行)」

DiskSuite オブジェクトの保守の概要

この章では、DiskSuite オブジェクトを作成した後で実行する必要がある、次のような保守作業について説明します。

- オブジェクト状態のチェック
- エラーの発生したオブジェクトの交換や修復
- メタデバイスの拡張とファイルシステムの成長
- オブジェクトのリネーム
- ファイルシステムのミラー化解除

DiskSuite についての概要は、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。メタデバイスやホットスペア集合の作成については、第2章を参照してください。

DiskSuite オブジェクトを保守するための前提条件

この章に含まれる手順の前提条件を次に示します。

- 現在の時点で、すべてのデータをバックアップしている。

- ルート権限がある (スーパーユーザーになれる) ことを確認する。
- グラフィカルユーザーインターフェースを使用する場合は、DiskSuite ツールを起動する。

ローカルなメタデバイス (ディスクセット構成に含まれないメタデバイス) を操作するには、次のように入力します。

```
# metatool &
```

ディスクセット内のメタデバイスを操作するには、自分がディスクセットの所有者であることを確認し、次のように入力します。

```
# metatool -s <ディスクセット名> &
```

DiskSuite オブジェクトの状態チェック

この節では、状態データベースの複製、メタデバイス、ホットスペア、ディスクセットなど、DiskSuite オブジェクトの状態をチェックする作業について説明します。この章で後述する次のような作業を実行する前に、オブジェクトの状態をチェックしてください。

- メタデバイス内でエラーの発生したスライスの交換
- ハードエラーのあるトランスメタデバイスの修復
- メタデバイスの拡張
- メタデバイスのリネーム
- ミラー化されたファイルシステムのミラー化解除

DiskSuite ツールによる状態のチェック

DiskSuite ツールを使用して DiskSuite オブジェクトの状態をチェックするには、次の3とおりの方法があります。

- オブジェクトリストのオブジェクトをダブルクリックして(キャンバスにオブジェクトが表示される)、オブジェクトの状態フィールドをチェックする。

- キャンバス上のオブジェクトをダブルクリックするか、またはオブジェクトのポップアップメニューから「情報」を選択して、オブジェクトの「情報」ウィンドウを表示する。
- 「ブラウズ」メニューから「障害リスト」を選択して、DiskSuite の障害リストを表示する。このリストには、既存の障害(もしあれば)が説明されている。

コマンド行による状態のチェック

metadb(1M) と metastat(1M) という 2 つのコマンドで、DiskSuite オブジェクトの状態をチェックします。

- 状態データベースの複製をチェックするには、metadb(1M) コマンドを使用します。

```
# metadb [-s <セット名>] [-i]
```

このコマンドでは、

- s <セット名> metadb コマンドが作用するディスクセットの名前を指定します。
- i 状態フラグについて説明する凡例を表示します。

- メタデバイスとホットスペア集合をチェックするには、metastat(1M) コマンドを使用します。

```
# metastat [-s <セット名>] [-p] [-t] [<オブジェクト>]
```

このコマンドでは、

- s <セット名> metastat が作用するディスクセットの名前を指定します。
- p md.tab ファイルと同じ形式で状態を表示します。
- t 最後に状態が変化した時間を表示します。
- <オブジェクト> ストライプ、連結、ストライプ方式の連結、ミラー、RAID5 メタデバイス、トランスメタデバイス、またはホットスペア集合の名前。具体的なオブジェクトを省略した場合、すべてのメタデバイスとホットスペア集合の状態が表示されます。

▼ 状態データベースの複製の状態のチェック方法 (DiskSuite ツール)

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たしていることを確認する。
2. オブジェクトの「情報」ウィンドウを表示して、**MetaDB** オブジェクトの状態をチェックする。
その他の方法で状態をチェックするには、132ページの「DiskSuite ツールによる状態のチェック」を参照してください。
3. 表 3-1 の**MetaDB** オブジェクトの状態フィールド、および実行可能な操作の説明を参照する。

表 3-1 MetaDB オブジェクトの状態キーワード

キーワード	意味	操作
正常	MetaDB オブジェクト (状態データベース) にはエラーがなく、正しく機能しています。	ありません。
注意	正常な状態データベースの複製の数が 3 未満であるか、または少なくとも 1 つの複製が壊れています。 この状態は、メタデバイスの状態データベースの複製が 3 つ未満の異なるコントローラ上で作成された場合にも表示されます。	さらに複製を追加する (できれば異なるコントローラに分散) か、または壊れた複製を修復します。 可能ならば別のコントローラを追加し、新しいコントローラに接続されたドライブ上に状態データベースの複製を作成します。 状態データベースの複製を追加するには、59ページの「状態データベースの追加複製を作成する方法 (DiskSuite ツール)」を参照してください。壊れた複製を修復するには、156ページの「状態データベースの複製を有効にする方法 (DiskSuite ツール)」を参照してください。

表 3-1 MetaDB オブジェクトの状態キーワード 続く

キーワード	意味	操作
緊急	正常な状態データベースの複製の数が 2 未満であるか、または 1 つ以上の状態データベースの複製が壊れています。	さらに複製を追加する (できれば異なるコントローラに分散) か、または壊れた複製を修復します。 状態データベースの複製を追加するには、59ページの「状態データベースの追加複製を作成する方法 (DiskSuite ツール)」を参照してください。壊れた複製を修復するには、156ページの「状態データベースの複製を有効にする方法 (DiskSuite ツール)」を参照してください。
重大な障害	正常な状態データベースの複製が存在しません。	状態データベースの複製を最初から少なくとも 3 つ作成してリポートします。さもなければ、システムは正しくブートできません。45ページの「初期状態データベースの複製を最初から作成する方法 (DiskSuite ツール)」を参照してください。

▼ 状態データベースの複製の状態のチェック方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件をチェックしてから、`-i` オプションを付けて `metadb(1M)` コマンドを使用して、状態データベースの複製の状態を表示します。詳細については、`metadb(1M)` のマニュアルページを参照してください。

例 - すべての状態データベースの複製の状態をチェック

```
# metadb -i
      flags      first blk      block count
a      u          16          1034      /dev/dsk/c4t3d0s2
a      u          16          1034      /dev/dsk/c3t3d0s2
```

(続く)

```

    a      u      16      1034      /dev/dsk/c2t3d0s2
o - state database replica active prior to last mddb configuration change
u - state database replica is up to date
l - locator for this state database replica was read successfully
c - state database replica's location was in /etc/opt/SUNWmd/mddb.cf
p - state database replica's location was patched in kernel
m - state database replica is master, this is state database replica
    selected as input
W - state database replica has device write errors
a - state database replica is active, commits are occurring to this
    state database replica
M - state database replica had problem with master blocks
D - state database replica had problem with data blocks
F - state database replica had format problems
S - state database replica is too small to hold current data base
R - state database replica had device read errors

```

デバイス名の前にある文字は、状態を表わします。この例での状態データベースの複製は、a フラグで示されるように、すべて有効です。状態に続いて、すべてのフラグの凡例が表示されます。

大文字は障害状態を示します。小文字は正常状態を示します。

▼ メタデバイスとホットスペア集合の状態のチェック方法 (DiskSuite ツール)

この作業は、メタデバイスとホットスペア集合の状態情報を表示および解釈するために使用します。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満足していることを確認する。
2. オブジェクトの「情報」ウィンドウを表示することによって、メタデバイスやホットスペア集合の状態をチェックする。
その他の状態チェック方法については、132ページの「DiskSuite ツールによる状態のチェック」を参照してください。
3. 表 3-2 のメタデバイスとホットスペア集合によって使用される状態キーワードの説明を参照する。

表 3-2 一般的な状態キーワード

キーワード	意味	使用するもの
正常	メタデバイスまたはホットスペア集合にはエラーがなく、正しく機能しています。	すべての種類のメタデバイスとホットスペア集合
注意	メタデバイスまたはホットスペア集合には障害があるが、すぐにデータが消失する危険はありません。	すべての種類のメタデバイスとホットスペア集合
緊急	メタデバイスにあと 1 つ障害が発生したらデータが消失します。	ミラー / サブミラー、RAID5 メタデバイス、トランスメタデバイス
重大な障害	データが破壊された可能性があります。たとえば、ミラー内のすべてのサブミラーにエラーがあったり、RAID5 メタデバイスの複数のスライスにエラーがあります。メタデバイスの構成が無効である場合、テンプレートオブジェクト (ホットスペア集合テンプレートを除く) も重大な障害状態を示します。	ミラー / サブミラー、RAID5 メタデバイス、トランスメタデバイス、すべてのテンプレートオブジェクト

注 - SPARCstorage Array のファンに障害が発生した場合、その SPARCstorage Array 上のすべてのメタデバイスとスライスに「重大な障害」のマークが付けられます。

4. 次のリストから、**DiskSuite** オブジェクトの具体的な状態と実行可能な操作についての説明を探し、参照する。

- 138ページの「ストライプと連結の状態 (DiskSuite ツール)」
- 138ページの「ミラーとサブミラーの状態 (DiskSuite ツール)」
- 140ページの「RAID5 メタデバイスの状態 (DiskSuite ツール)」
- 141ページの「トランスメタデバイスの状態 (DiskSuite ツール)」
- 143ページの「ホットスペア集合とホットスペアの状態 (DiskSuite ツール)」

ストライプと連結の状態 (DiskSuite ツール)

連結またはストライプがサブミラーとして使用されていない限り、DiskSuite は、エラーの発生した連結やストライプの状態変化を通知しません。スライスエラーやその他のデバイス障害がある場合、DiskSuite は、要求側のアプリケーションにエラーを返し、それを次のようにコンソールに出力します。

```
WARNING: md d4: read error on /dev/dsk/clt3d0s6
```

注 - DiskSuite は、SNMP メッセージを受信できる任意のネットワーク管理コンソールに対して、上のメッセージのような SNMP トラップデータ (警告) を送信できます。詳細は、287ページの「DiskSuite の SNMP サポートの構成方法 (コマンド行)」を参照してください。

連結とストライプには複製データが含まれないため、シンプルメタデバイスでのスライスエラーから回復するためには、物理ディスクを交換し、メタデバイスを再作成してデータをバックアップから復元する必要があります。157ページの「スライス障害の後でストライプや連結を再作成する方法 (DiskSuite ツール)」、または 158ページの「スライス障害の後でストライプや連結を再作成する方法 (コマンド行)」を参照してください。

ミラーとサブミラーの状態 (DiskSuite ツール)

ミラーオブジェクトには2つの状態フィールドがあります。つまり、ミラーデバイス自身の状態フィールドと、各サブミラー用の個々の状態フィールドです。ミラー用の状態フィールドは、表 3-3 で示すような状態を提供します。

表 3-3 ミラーの状態キーワード

キーワード	意味
正常	ミラーにはエラーがなく、正しく機能しています。
注意	サブミラーには障害がありますが、すぐにデータが消失する危険はありません。データのコピーがまだ2つある (このミラーは3面ミラーであり、1つのサブミラーだけに障害が発生した) か、またはホットスベアによって交換された状態にあります。

表 3-3 ミラーの状態キーワード 続く

キーワード	意味
緊急	ミラーには正常なサブミラーが 1 つしかなく、データのコピー 1 つだけでサービスを提供しています。この 1 つ残ったミラーに障害が発生したらデータが消失します。
重大な障害	すべてのサブミラーにエラーがあり、データが破壊された可能性があります。

サブミラーの状態フィールドと実行可能な操作を表 3-4 に示します。

表 3-4 サブミラーの状態キーワード

キーワード	意味	操作
正常	サブミラーにはエラーがなく、正しく機能しています。	ありません。
再同期中	サブミラーの再同期処理が活発に行われています。	ありません。エラーの発見、訂正処理を行っていたり、サブミラーがオンライン状態に復帰した直後、または新たにサブミラーが追加された状態です。
コンポーネント再同期	サブミラー内のスライスの再同期処理が活発に行われています。	ありません。ホットスワップスライスまたは他のスライスを利用して、サブミラー内のエラーの発生したスライスを交換しました。
接続中	サブミラーが接続されています。	ありません。
接続済み(再同期中)	接続後、サブミラー全体を再同期しています。	ありません。
オンライン(予定設定済み)	次に「確定」をクリックすると、サブミラーがオンラインとなります。	「確定」をクリックして、サブミラーを有効にします。
オフライン(予定設定済み)	次に「確定」をクリックすると、サブミラーはオフラインとなります。	「確定」をクリックして、サブミラーをオフラインにします。

表 3-4 サブミラーの状態キーワード 続く

キーワード	意味	操作
オフライン済み	サブミラーはオフラインです。	たとえば保守を行なった後など、適当なときに、サブミラーをオンラインに戻します。211ページの「サブミラーのオフライン / オンライン設定 (DiskSuite ツール)」を参照してください。
保守状態	サブミラーにはエラーがあります。	サブミラーを修復します。「エラー」状態にあるサブミラーは任意の順序で修復できます。164ページの「サブミラー内のスライスを有効にする方法 (DiskSuite ツール)」、または166ページの「サブミラー内のスライスを交換する方法 (DiskSuite ツール)」を参照してください。
最後にエラーが発生した状態	サブミラーにエラーがあり、ミラーのデータは破壊された可能性があります。	まず、「保守状態」のサブミラーを修復してから、「最後にエラーが発生した状態」のサブミラーを修復します。164ページの「サブミラー内のスライスを有効にする方法 (DiskSuite ツール)」、または166ページの「サブミラー内のスライスを交換する方法 (DiskSuite ツール)」を参照してください。エラーを修復したら、データの妥当性チェックします。

注 - DiskSuite は、サブミラーではないシンプルメタデバイスに対して、状態情報やホットスペア情報を保持しません。

RAID5 メタデバイスの状態 (DiskSuite ツール)

RAID5 オブジェクトの状態フィールドにあるキーワード、および実行可能な操作について、表 3-5 で説明します。

表 3-5 RAID5 の状態キーワード

キーワード	意味	操作
正常	RAID5 メタデバイスにはエラーがなく、正しく機能しています。	ありません。
接続済み / 初期化 (再同期)	接続が行われた後、または作成された後で、RAID5 メタデバイスが再同期されています。	通常は、ありません。入出力エラーが発生した場合、新しい RAID5 メタデバイスの初期化中に、デバイスは「保守状態」となります。初期化に失敗した場合、メタデバイスは「初期化失敗状態」となり、スライスは「保守状態」となります。この場合、メタデバイスを削除して再作成します。
注意	RAID5 メタデバイスに障害はありますが、すぐにデータが消失する危険はありません。	デバイス状態の監視を続けます。
緊急	RAID5 メタデバイスにはスライスエラーがあり、あと 1 つ障害が発生したらデータが消失します。	エラーの発生したスライスを修復します。170ページの「RAID5 メタデバイス内のスライスを有効にする方法 (DiskSuite ツール)」、または 171ページの「RAID5 スライスの交換方法 (DiskSuite ツール)」を参照してください。
重大な障害	RAID5 メタデバイスには、エラーのあるスライスが複数個存在します。データは破壊された可能性があります。	エラーの発生したスライスを修復するには、170ページの「RAID5 メタデバイス内のスライスを有効にする方法 (DiskSuite ツール)」、または 171ページの「RAID5 スライスの交換方法 (DiskSuite ツール)」を参照してください。バックアップからデータを復元する必要があります。

トランスメタデバイスの状態 (DiskSuite ツール)

トランスメタデバイスオブジェクトの状態フィールドのキーワード、および実行可能な操作について、表 3-6 で説明します。

表 3-6 トランスメタデバイスの状態キーワード

キーワード	意味	操作
正常	デバイスは正しく機能しています。マウントされている場合、ファイルシステムはロギング中なので、ブート時にはチェックされません (つまり、fsck はブート時にファイルシステムをチェックしません)。	ありません。
ログの切断 (進行中)	トランスメタデバイスがマウント解除されたとき、または次のリブート時に、トランスメタデバイスのログが切断されます。	ありません。
ログの切断 (予定設定済み)	次に「確定」をクリックすると、トランスメタデバイスのログが切断されます。	「確定」をクリックしてログを切断します。この切断は、次のリブート時、またはファイルシステムがマウント解除されてから再マウントされたときに行われます。
注意	トランスメタデバイスには障害がありますが、すぐにデータが消失する危険はありません。	トランスメタデバイス状態の監視を続けます。
緊急	トランスメタデバイスに障害があり、あと 1 つ障害が発生したらデータを失います。トランスメタデバイスに RAID5 メタデバイスまたはミラーが含まれる場合にだけ、この状態が発生します。	エラーの発生したミラーまたは RAID5 マスターデバイスを修復します。160 ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」を参照してください。
重大な障害 (ログなし)	トランスメタデバイスにロギングデバイスが接続されていません。	ロギングデバイスを接続します。ロギングデバイスが接続されるまでは、ファイルシステムのロギングを開始できません。

表 3-6 トランスメタデバイスの状態キーワード 続く

キーワード	意味	操作
重大な障害 (ログハードエラー)	デバイスの使用中に、デバイスエラーまたはファイルシステムでパニックが発生しました。デバイスが閉じられるかマウント解除されるまで、読み書きのたびに入出力エラーが返されます。最初に開いたとき、このデバイスはエラー状態に移行します。	トランスメタデバイスを修復します。179ページの「ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行)」、または 179ページの「ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行)」を参照してください。
重大な障害 (エラー)	デバイスは読み書き可能です。ファイルシステムも読み取り専用でマウントできるのに、実際に読み書きしてみるとそのつど入出力エラーが返されています。この状態からは、たとえファイルシステムでパニックが発生したり、デバイスエラーが発生しても、ハードエラー状態には戻りません。	トランスメタデバイスを修復します。179ページの「ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行)」、または 179ページの「ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行)」を参照してください。fsck(1M) または newfs(1M) が正常に終了すると、デバイスは正常状態に移行します。デバイスがハードエラーまたはエラーの状態にある場合、fsck はファイルシステムをブート時に自動的にチェックおよび修復します。newfs は、デバイスにどんなデータがあっても、これを破壊します。

ホットスペア集合とホットスペアの状態 (DiskSuite ツール)

ホットスペア集合オブジェクトの状態フィールドに含まれるキーワードと実行可能な操作を表 3-7 に示します。

表 3-7 ホットスペア集合の状態キーワード

キーワード	意味	操作
正常	ホットスペアは動作中であり、データを受け付ける用意ができていますが、現在のところ、ホットスペアでは読み書きが行われていません。	ありません。
使用中	現在、ホットスペアに読み書きが行われています。	ホットスペアの使用状況を診断してから、ホットスペアが使用されているメタデバイス内のスライスを修復します。
注意	ホットスペアまたはホットスペア集合に障害がありますが、すぐにデータが消失する危険はありません。ホットスペア集合にホットスペアが存在しない場合、またはすべてのホットスペアが使用中であるか壊れているホットスペアがある場合にも、この状態が表示されます。	ホットスペアの使用状況や故障原因を診断します。必要ならば、ホットスペア集合にホットスペアを追加できます。

▼ メタデバイスとホットスペア集合の状態のチェック方法 (コマンド行)

前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たしていることを確認します。metastat(1M) コマンドを使用して、メタデバイスやホットスペア集合の状態を表示します。詳細については、metastat(1M) のマニュアルページを参照してください。

コマンド行の出力と実行可能な操作の説明は、次のリストを使用して、適切なページを参照してください。

- 145ページの「ストライプと連結の状態 (コマンド行)」
- 145ページの「ミラーとサブミラーの状態 (コマンド行)」
- 148ページの「RAID5 メタデバイスの状態 (コマンド行)」
- 150ページの「トランスメタデバイスの状態 (コマンド行)」
- 152ページの「ホットスペア集合とホットスペアの状態 (コマンド行)」

注 - DiskSuite の一般的な状態キーワードの説明については、表 3-2 を参照してください。

ストライプと連結の状態 (コマンド行)

連結またはストライプがサブミラーとして使用されない限り、DiskSuite は連結またはストライプの状態変化を通知しません。詳細については、138ページの「ストライプと連結の状態 (DiskSuite ツール)」を参照してください。

ミラーとサブミラーの状態 (コマンド行)

ミラー上で `metastat(1M)` を実行すると、各サブミラーの状態、パス番号、読み取りオプション、書き込みオプション、ミラー内の総ブロック数サイズが表示されます。ミラーのパス番号、読み取りオプション、書き込みオプションを変更するには、231ページの「ミラーのオプションの変更方法 (コマンド行)」を参照してください。

`metastat` から出力されたミラーの例を次に示します。

```
# metastat
d0: Mirror
  Submirror 0: d1
    State: Okay
  Submirror 1: d2
    State: Okay
  Pass: 1
  Read option: roundrobin (default)
  Write option: parallel (default)
  Size: 5600 blocks

d1: Submirror of d0
  State: Okay
  Size: 5600 blocks
  Stripe 0:
    Device          Start Block  Dbase State      Hot Spare
    c0t2d0s7        0           No   Okay
  ...
```

ミラー内の各サブミラーに対して、`metastat` は、その状態、呼び出し (`Invoke`) 行 (エラーがある場合)、割り当てられたホットスベア集合 (もしあれば)、ブロック単位のサイズ、サブミラー内の各スライスについての情報を表示します。

サブミラーの状態を表 3-8 に示します。

表 3-8 サブミラーの状態 (コマンド行)

状態	意味
Okay	サブミラーにはエラーがなく、正しく機能しています。
Resyncing	サブミラーは、アクティブに再同期されています。エラーが発生して訂正されました。サブミラーがちょうどオンラインに復帰したか、または新しいサブミラーが追加されました。
Needs Maintenance	サブミラー内のスライスに、入出力エラーまたはオープンエラーが発生しました。サブミラー内のこのスライスに対するすべての読み書きは中止されました。

さらに、サブミラー内のストライプごとに、`metastat` は、ストライプ内のスライスのデバイス名 (Device)、スライスが始まる開始ブロック (Start Block)、スライスに状態データベースの複製が含まれるかどうか (Dbase)、スライスの状態 (Status)、障害の発生したスライスのホットスペアに使用されるスライスを示すホットスペア (Hot Spare) を表示します。

ミラーのエラーを解決する場合、おそらくこのスライス状態は最も重要な情報となります。サブミラー状態は、「Okay (正常)」や「Needs Maintenance (要保守)」などの一般的な状態情報を提供するだけです。サブミラーが「Needs Maintenance」状態を通知した場合、スライス状態を参照してください。スライスが「Maintenance (保守状態)」または「Last Erred (最後にエラーが発生した状態)」にある場合、別の障害回復アクションを実行することになります。「Maintenance」のスライスだけが存在する場合、これらのスライスは任意の順序で修復できます。「Maintenance」のスライスと「Last Erred」スライスが存在する場合、最初に「Maintenance」のスライスを修復してから、「Last Erred」スライスを修復しなければなりません。詳細は、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」を参照してください。

サブミラーのスライス状態と実行可能な操作を表 3-9 に示します。

表 3-9 サブミラーのスライス状態 (コマンド行)

状態	意味	操作
Okay	スライスにはエラーがなく、正常に機能しています。	ありません。
Resyncing	サブミラーの再同期処理が活発に行われています。エラーの発見、訂正処理を行っていたり、サブミラーがオンライン状態に復帰した直後、または新たにサブミラーが追加された状態です。	必要ならば、再同期が終了するまで、サブミラーの状態を監視します。
Maintenance	スライスに入出力エラーまたはオープンエラーが発生しました。このスライスとの読み書きはすべて中止されました。	<p>エラーの発生したスライスを有効にするか交換します。165ページの「サブミラー内のスライスを有効にする方法 (コマンド行)」、または 167ページの「サブミラー内のスライスを交換する方法 (コマンド行)」を参照してください。</p> <p>注: <code>metastat(1M)</code> コマンドでは、<code>metareplace(1M)</code> コマンドで実行すべき適切な操作と一緒に、<code>invoke</code> 障害回復メッセージを表示します。<code>metareplace -e</code> コマンドも使用できません。</p>
Last Erred	スライスに入出力エラーまたはオープンエラーが発生しました。しかし、他のスライスにも障害が発生しているため、データは他の場所に複写されません。スライス上では、まだ入出力が実行中です。この入出力がエラーになると、ミラー入出力は異常終了します。	<p>最初に、「保守状態」のスライスを有効にするか交換します。165ページの「サブミラー内のスライスを有効にする方法 (コマンド行)」、または 167ページの「サブミラー内のスライスを交換する方法 (コマンド行)」を参照してください。</p> <p>通常、このエラーによって一部のデータが失われるため、修復してからミラーの妥当性をチェックします。ファイルシステムの場合、<code>fsck(1M)</code> コマンドを使用して「メタデータ」の妥当性をチェックしてから、ユーザーデータをチェックします。アプリケーションやデータベースには、メタデータの妥当性をチェックするための独自の方法が必要です。</p>

RAID5 メタデバイスの状態 (コマンド行)

RAID5 メタデバイス上で `metastat(1M)` コマンドを実行すると、メタデバイスの状態が表示されます。さらに、RAID5 メタデバイス内のスライスごとに、`metastat` は、ストライプ内のスライスのデバイス名 (Device)、スライスが始まる開始ブロック (Start Block)、スライスに状態データベースの複製が含まれるかどうか (Dbase)、スライスの状態 (Status)、障害の発生したスライスのホットスペアに使用されるスライスを示すホットスペア (Hot Spare) を表示します。

`metastat` から出力された RAID5 メタデバイスの例を次に示します。

```
# metastat
d10: RAID
  State: Okay
  Interlace: 32 blocks
  Size: 10080 blocks
  Original device:
    Size: 10496 blocks
    Device          Start Block  Dbase State      Hot Spare
    c0t0d0s1        330         No   Okay
    c1t2d0s1        330         No   Okay
    c2t3d0s1        330         No   Okay
```

RAID5 メタデバイスの状態を表 3-10 に示します。

表 3-10 RAID5 の状態 (コマンド行)

状態	意味
Initializing	スライスは、すべてのディスクブロックをゼロで初期化操作中にあります。この操作は、データとパリティを飛び越しストライピングする RAID5 メタデバイスの特性上、必要なものです。 状態が「Okay (正常)」に変化したら、初期化プロセスが完了しており、デバイスを開くことができます。それまで、アプリケーションはエラーメッセージを受け取ります。
Okay	デバイスの使用準備が整っており、今のところエラーはありません。
Maintenance	読み書き操作中に発生した入出力エラーかオープンエラーが発生したため、1つのスライスがエラーとマークされました。

RAID5 メタデバイスのエラーに対処する場合、おそらくこのスライス状態は最も重要な情報となります。RAID5 状態は、「Okay (正常)」や「Needs Maintenance (要保守)」などの一般的な状態情報を提供するだけです。RAID5 が「Needs

「Maintenance」状態を通知した場合、スライス状態を参照してください。スライスが「Maintenance (保守状態)」または「Last Erred (最後にエラーが発生した状態)」にある場合、別の障害回復操作を実行することになります。「Maintenance」のスライスが1つだけ存在する場合、そのスライスはデータを失うことなく修復できます。「Maintenance」のスライスと「Last Erred」のスライスが存在する場合、データはおそらく破壊されています。最初に「Maintenance」のスライスを修復してから、「Last Erred」のスライスを修復しなければなりません。詳細は、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」を参照してください。

RAID5 メタデバイスのスライス状態と実行可能な操作を、表 3-11 に示します。

表 3-11 RAID5 のスライス状態 (コマンド行)

状態	意味	操作
Initializing	スライスは、すべてのディスクブロックをゼロで初期化操作中にあります。この操作は、データとパリティを飛び越してトライピングする RAID5 メタデバイスの特性上、必要なものです。	通常は、ありません。このプロセス中に入出力エラーが発生した場合、デバイスは「Maintenance (保守状態)」となります。初期化に失敗した場合、メタデバイスは「初期化失敗状態」となり、スライスは「Maintenance」となります。この場合、メタデバイスを削除して再作成します。
Okay	デバイスの使用準備が整っており、今のところエラーはありません。	ありません。必要ならば、スライスの追加や交換が行えます。
Resyncing	スライスの再同期処理が活発に行われています。エラーの発見、訂正処理を行っていたり、スライスがオンライン状態に復帰した直後、または新たにスライスが追加された状態です。	希望により、再同期が完了するまで、RAID5 メタデバイスの状態を監視します。

表 3-11 RAID5 のスライス状態 (コマンド行) 続く

状態	意味	操作
Maintenance	読み書き操作中に発生した入出力エラーやオープンエラーが発生したため、1つのスライスにエラーがマークされました。	エラーの発生したスライスを有効にするか交換します。171ページの「RAID5 メタデバイス内のスライスを有効にする方法 (コマンド行)」、または 172ページの「RAID5 スライスの交換方法 (コマンド行)」を参照してください。 注: <code>metastat(1M)</code> コマンドでは、 <code>metareplace(1M)</code> コマンドで実行すべき適切な操作と一緒に、 <code>invoke</code> 障害回復メッセージを表示します。
Last Erred	複数のスライスにエラーが発生しました。エラーの発生したスライスの状態は、「Maintenance」か「Last Erred」です。この状態では、「Maintenance」にあるスライスには入出力が行われませんが、「Last Erred」とマークされたスライスには入出力が試され、その結果入出力要求の状態は全体的にこの状態 (Last Erred) となります。	エラーの発生したスライスを有効にするか交換します。171ページの「RAID5 メタデバイス内のスライスを有効にする方法 (コマンド行)」、または 172ページの「RAID5 スライスの交換方法 (コマンド行)」を参照してください。 注: <code>metastat(1M)</code> コマンドを起動すると、 <code>metareplace(1M)</code> コマンドに適切なオプション (<code>-f</code> フラグ) を付けて実行するように表示されます。つまり、複数のエラーのスライスから正確でないデータが生成された可能性があることを示します。

トランスメタデバイスの状態 (コマンド行)

トランスメタデバイス上で `metastat(1M)` コマンドを実行すると、メタデバイスの状態が表示されます。

`metastat` から出力されたトランスメタデバイスの例を次に示します。

```
# metastat
d20: Trans
  State: Okay
  Size: 102816 blocks
  Master Device: c0t3d0s4
  Logging Device: c0t2d0s3

      Master Device      Start Block  Dbase
      c0t3d0s4           0            No
```

(続く)

続き

```
c0t2d0s3: Logging device for d0
State: Okay
Size: 5350 blocks

Logging Device      Start Block  Dbase
c0t2d0s3            250          No
```

metastat コマンドは、マスターデバイスとロギングデバイスも表示します。デバイスごとに次の情報が表示されます。スライスまたはメタデバイスのデバイス名 (Device)、デバイスが始まる開始ブロック (Start Block)、デバイスに状態データベースの複製が含まれるかどうか (Dbase)、ロギングデバイス用の状態 (State)。

トランスメタデバイスの状態と実行可能な操作を表 3-12 に示します。

表 3-12 トランスメタデバイスの状態 (コマンド行)

状態	意味	操作
Okay	デバイスは正しく機能しています。マウントされているものの場合、ファイルシステムはロギングされているので、ブート時のチェックはありません。	ありません。
Attaching	トランスが閉じられたりマウント解除されたとき、ロギングデバイスがトランスメタデバイスに接続されます。そのとき、デバイスは「Okay (正常)」状態に移行します。	metattach(1M) のマニュアルページを参照してください。
Detached	トランスメタデバイスにはロギングデバイスがありません。UFS ロギングの利点が無効になります。	fsck(1M) は、デバイスをブート時に自動的にチェックします。metadetach(1M) のマニュアルページを参照してください。
Detaching	トランスが閉じられたりマウント解除されたとき、ロギングデバイスはトランスメタデバイスから切断されます。そのとき、デバイスは「Detached (切断)」状態に移行します。	metadetach(1M) のマニュアルページを参照してください。

表 3-12 トランスメタデバイスの状態 (コマンド行) 続く

状態	意味	操作
Hard Error	デバイスの使用中に、デバイスエラーやファイルシステムのパニックが発生しました。デバイスが閉じられるかマウント解除されるまで、読み書きのたびに入出力エラーが返されます。最初に開いた時点で、デバイスは「Error (エラー)」状態に移行します。	トランスメタデバイスを修復します。179ページの「ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行)」、または 179ページの「ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行)」を参照してください。
Error	デバイスは読み書き可能です。ファイルシステムも読み取り専用でマウントできるのに、実際に読み書きしてみるとそのつど入出力エラーが返されています。この状態からは、たとえファイルシステムでパニックが発生したり、デバイスエラーが発生しても、ハードエラー状態には戻りません。	トランスメタデバイスを修復します。179ページの「ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行)」、または 179ページの「ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行)」を参照してください。fsck(1M) または newfs(1M) が正常に終了すると、デバイスは「Okay」状態に移行します。デバイスが「Hard Error」または「Error」の状態にある場合、fsck はファイルシステムをブート時に自動的にチェックおよび修復します。newfs は、デバイスにどんなデータがあっても、これを破壊します。

ホットスペア集合とホットスペアの状態 (コマンド行)

ホットスペア集合上で `metastat(1M)` コマンドを実行すると、ホットスペア集合とそのホットスペアの状態が表示されます。

`metastat` から出力されたホットスペア集合の例を次に示します。

```
# metastat hsp001
hsp001: 1 hot spare
          c1t3d0s2                Available        16800 blocks
```

ホットスペア集合の状態と実行可能な操作を表 3-13 に示します。

表 3-13 ホットスペア集合の状態 (コマンド行)

状態	意味	操作
Available	ホットスペアは動作中であり、データを受け付ける用意ができていますが、現在のところ読み書きが行われていません。	ありません。
In-use	現在、ホットスペアを使用して読み書きが行われています。	ホットスペアの使用状況を診断してから、ホットスペアが使用されているメタデバイス内のスライスを修復します。
Attention	ホットスペアまたはホットスペア集合に障害がありますが、すぐにデータが消失する危険はありません。ホットスペア集合にホットスペアが存在しない場合、またはすべてのホットスペアが使用中であるか壊れているホットスペアがある場合にも、この状態になります。	ホットスペアの使用状況や故障原因を診断します。必要ならば、ホットスペア集合にホットスペアを追加できます。

▼ ディスクセットの状態のチェック方法 (コマンド行)

前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たしていることを確認します。metaset(1M) コマンドを使用して、ディスクセットの状態を表示します。詳細については、metaset(1M) のマニュアルページを参照してください。

注 - ディスクセットの所有権は、所有者側のホストにだけ表示されます。

例 - 指定されたディスクセットの状態をチェック

```
red# metaset -s relo-red
Set name = relo-red, Set number = 1

Host          Owner
red           Yes
blue

Drive         Dbase
c1t2d0       Yes
```

(続く)

続き

```
c1t3d0      Yes
c2t2d0      Yes
c2t3d0      Yes
c2t4d0      Yes
c2t5d0      Yes
```

-s オプションを付けた `metaset(1M)` コマンドに続けてディスクセットの名前 `relo-red` を指定すると、そのディスクセットの状態情報が表示されます。所有者側のホスト `red` から `metaset` コマンドを発行すると、`red` が実際にディスクセットの所有者であることが確認できます。`metaset` コマンドは、ディスクセット内のドライブも表示します。

例 - すべてのディスクセットの状態をチェック

```
red# metaset
Set name = relo-red, Set number = 1

Host          Owner
red           Yes
blue

Drive         Dbase
c1t2d0        Yes
c1t3d0        Yes
c2t2d0        Yes
c2t3d0        Yes
c2t4d0        Yes
c2t5d0        Yes

Set name = relo-blue, Set number = 2

Host          Owner
red           blue
blue

Drive         Dbase
c3t2d0        Yes
c3t3d0        Yes
c3t4d0        Yes
c3t5d0        Yes

Set name = rimtic, Set number = 3

Host          Owner
red
```

(続く)

blue	
Drive	Dbase
c4t2d0	Yes
c4t3d0	Yes
c4t4d0	Yes
c4t5d0	Yes

metaset コマンドを単独実行すると、すべてのディスクセットの状態が表示されます。この例では、relo-red、relo-blue、および rmtic という名前の 3 つのディスクセットが構成されます。ホスト red は relo-red ディスクセットを所有するため、metaset は red を所有者として表示します。ホスト blue は、残り 2 つのディスクセットである relo-blue と rmtic を所有します。これが明らかになるのは、metaset がホスト blue から実行された場合だけです。

オブジェクトの交換と有効化

この節では、次に示すような DiskSuite オブジェクトに含まれるスライスを交換および有効にするための手順を説明します。

- 状態データベースの複製
- ストライプと連結
- ミラー (サブミラー)
- RAID5 メタデバイス
- ホットスペア

注 - 物理ディスク (SPARCstorage Array 内のディスクも含む) を修復および交換するには、第 7 章を参照してください。

状態データベースの複製を有効にするための予備情報

DiskSuite ツールで状態データベースの複製を有効化 (復元) する場合、2 つの処理が実行されます。DiskSuite ツールは、まず複製を除去 (削除) してから、スライスに

追加して返そうと試みます。スライスにエラーが発生しているなどの障害がある場合、継続して削除が行われます。状態データベースの複製を復元するためには、スライスを修復する必要があります。

▼ 状態データベースの複製を有効にする方法 (DiskSuite ツール)

この作業は、状態データベースによって使用されているスライスをオンラインに復帰させるために実行します。エラーの発生したスライス (ディスク) を物理的に交換してから、この作業を実行してください。

1. **前提条件** (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、**予備情報** (155ページの「状態データベースの複製を有効にするための予備情報」) を読んだことを確認する。
2. オブジェクトリストの **MetaDB** オブジェクトをダブルクリックする。
MetaDB オブジェクトがキャンバスに表示されます。
3. **MetaDB** オブジェクトの「情報」ウィンドウを表示する。
4. スクロールリストからエラーの発生したスライスを選択し、「復元」をクリックする。
選択されたスライスが「正常」状態を表示しない場合にだけ、「復元」ボタンが有効になります。
5. コンフィグレーションログを表示して、復元が行われたことを確認する。

ストライプや連結を再作成するための予備情報

- ストライプや連結には複製されたデータが含まれないため、このようなメタデバイスにスライス障害が発生した場合は、スライスを交換し、ストライプや連結を再作成し、バックアップからデータを復元しなければなりません。
- ストライプを再作成する場合、エラーの発生したスライスと少なくとも同じサイズの交換用スライスを使用します。
- 連結を再作成する場合、障害の発生したスライスと少なくとも同じ容量の交換用スライスを使用します。
- ストライプや連結に含まれる障害の発生したスライスがサブミラーである場合は、164ページの「サブミラー内のスライスを有効にする方法 (DiskSuite ツール)」を参照してください。

ル)」、または 166 ページの「サブミラー内のスライスを交換する方法 (DiskSuite ツール)」を参照してください。

▼ スライス障害の後でストライプや連結を再作成する方法 (DiskSuite ツール)

1. 前提条件 (131 ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (156 ページの「ストライプや連結を再作成するための予備情報」) を読んだことを確認する。
2. メタデバイスへのアクセスを停止する。
たとえば、マウントされたファイルシステムがメタデバイスに含まれる場合、そのファイルシステムをマウント解除します。
3. 可能ならば、ストライプや連結の上で `ufsdump(1M)` コマンドを実行する。

```
# ufsdump [<オプション>...] [<引数>...] <バックアップするファイル>...
```

このコマンドでは、

<オプション>	1 文字のオプション名から成る単一の文字列。
<引数>	オプションの引数であり、複数の文字列が可能です。オプションとそれに伴う引数は、同じ順序にします。
<バックアップするファイル>	バックアップするファイル名。これらのファイル名は、常に最後に記述します。

詳細については、`ufsdump(1M)` のマニュアルページを参照してください。メタデバイスにアクセスできない場合は、最新のバックアップに依存しなければなりません。

4. 削除するストライプや連結をオブジェクトリスト上でダブルクリックする。
メタデバイスオブジェクトがキャンバスに表示されます。
5. オブジェクトのポップアップメニューから「削除」を選択する。
6. 表示されるダイアログボックスで、「削除」をクリックする。
7. メタデバイスを再作成する。

63ページの「ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)」、または 67ページの「連結の作成方法 (DiskSuite ツール)」を参照してください。

注 - メタデバイスがストライプである場合、新しいスライスは、障害の発生したスライスと同じサイズでなければなりません。メタデバイスが連結である場合、新しいスライスは、障害の発生したスライスと少なくとも同じ容量でなければなりません。

8. メタデバイスがファイルシステムに使用された場合、メタデバイス上で新しいファイルシステムを作成する。

125ページの「メタデバイス上にファイルシステムを作成」を参照してください。

9. `ufsrestore(1M)` コマンドでデータを復元する。

```
# ufsrestore [<オプション>...] [<引数>...] [<ファイル名>...]
```

このコマンドでは、

<オプション>	1文字のオプション名から成る単一の文字列。オプション (i、r、R、t、x) の中から 1つだけ選択します。
<引数>	オプション文字列に続けて、オプションに対応する引数を記述します。オプション名とそれに伴う引数は、同じ順序にします。
<ファイル名>	復元されるファイルを、 <code>-x</code> オプションまたは <code>-t</code> オプションに対する引数として、最後に指定します。

詳細は、`ufsrestore(1M)` のマニュアルページを参照してください。

10. メタデバイス上のデータの妥当性をチェックする。

▼ スライス障害の後でストライプや連結を再作成する方法 (コマンド行)

前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) と予備情報 (156ページの「ストライプや連結を再作成するための予備情報」) をチェックしてから、`metaclear(1M)` コマンドと `metainit(1M)` コマンドを使用して、ス

ライスを障害が発生した後のストライプや連結を再作成します。metastat コマンドは、連結やストライプに対するスライスエラーを表示しません。しかし、システムコンソールには次のようなエラーが表示されます。

```
WARNING: md d35: read error on /dev/dsk/c0t0d0s6
```

詳細は、metaclear(1M) と metainit(1M) のマニュアルページを参照してください。

例 - スライス障害の後で連結を再作成

```
# umount /news
# init 0
ok boot -s
...

# ufsdump 0ucf /dev/rmt/0 /news

DUMP: Date of this level 0 dump: Fri Mar 1 15:17:45 1996
...

DUMP: DUMP IS DONE
# metaclear d35
# metainit d35 2 1 c1t0d0s2 1 c1t0d1s2
# newfs /dev/md/rdisk/d35
# mount /dev/md/dsk/d35 /news
# cd /news
# ufsrestore rvf /dev/rmt 0
Verify volume and initialize maps
Media block size is 126
...
Check pointing the restore
# rm restoresymtable
# ls /news
```

d35 にはマウントされたファイルシステム /news が含まれているため、これをマウント解除して、システムをブートしてシングルユーザーモードにします。ufsdump コマンドでデータをテープにダンプし、metaclear コマンドで連結を削除します。metainit コマンドで、障害の発生したスライスと交換する新しいスライスを使用して連結を再作成します。データを ufsrestore コマンドで復元し、さらに妥当性をチェックします (たとえば、ls コマンドを使用)。

注 - メタデバイスがストライプである場合、新しいスライスは、障害の発生したスライスと同じサイズでなければなりません。メタデバイスが連結である場合、新しいスライスには、障害の発生したスライスと少なくとも同じ容量が必要です。

ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要

DiskSuite には、ミラーや RAID5 メタデバイスの内部でスライスの交換と有効化を行う機能があります。

DiskSuite の用語では、スライスの交換とは、サブミラーや RAID5 メタデバイス内の選択されたスライスを、システム上の有効なスライスで置き換えることです。スライスの物理的な交換とは対照的に、この方式を「メタ交換」と呼ぶことができます。スライスを有効化するとは、スライスをアクティブにしたり、スライスをそれぞれで置き換える (つまり、スライス名は同じ) ことです。

使用できる 2 つの方法とその使用タイミングを次に説明します。

注 - ディスクエラーから回復する場合、`/var/adm/messages` を走査して、発生したエラーの種類を調べます。エラーが一時的な性質のものであり、ディスク自身には障害がない場合、エラーの発生したスライスの有効化を試みます。`format(1M)` コマンドを使用しても、ディスクをテストできます。

スライスを有効にする

次のような場合、スライスの有効化が適切です。

- ソフトエラーの可能性が高い場合は、現在エラーの発生しているスライスを有効にすることによって回復を試みます。
- 障害の発生したディスクドライブを新しいディスクドライブと物理的に交換し、元のドライブと同じようにパーティションに再分割し、エラーの発生したスライスをメタデバイス内部で有効にします。

次のような場合、スライスを有効化できます。

1. DiskSuite が物理ドライブにアクセスできないとき。

たとえば、この原因としては、電源断やドライブケーブルが抜けかかっている場合が挙げられます。この場合、DiskSuite はスライスを「保守状態」にします。

ドライブにアクセス可能であることを確認 (電源の復旧、ケーブルの再接続など) してから、メタデバイス内のスライスを有効にします。

2. 物理ドライブに、ディスクに関連しない一時的な障害があると思われるとき。

「保守状態」にあるスライスを、単に有効にするだけで修復できることもあります。これで障害を修復できない場合には、ディスクドライブを物理的に交換してスライスを有効にするか、そのスライスをシステム上の他の使用可能なスライスで「メタ交換」する必要があります。

ドライブを物理的に交換する場合は、古いドライブと同じようにパーティション分割します。なお、ドライブが物理的に交換されて古いドライブと同じようにパーティション分割されたら、エラーの発生したスライスを有効にするための作業は、前述の最初の条件の場合と同じです。

注 - 交換されるドライブ上のホットスペアと状態データベースの複製を必ずチェックしてください。エラーとして示される状態データベースの複製は、ディスクを交換する前に削除し、スライスを有効にする前に追加 (サイズが同じであることを確認) してください。ホットスペアも同様に処置してください。

スライスを他の使用可能なスライスと交換する

既存のスライスを、システム上で使用可能な未使用の別のスライスと置き換えたり交換したりする場合、DiskSuite の「メタ交換」スライス機能を使用します。

次のような場合にこの方法を使用できます。

1. ディスクドライブに障害があり、交換用のドライブはないが、システム上の別の場所に使用可能なスライスがある場合 (交換が絶対的に必要であっても、システムをシャットダウンしたくない場合に、こうすることができます)。
2. ソフトエラーが発生している場合。

DiskSuite は、ミラー / サブミラーや RAID5 メタデバイスが「正常」状態にあるのに物理ディスクよりソフトエラーが通知されることがあります。問題のスライスを他の使用可能なスライスと交換すると、予防的保守を実行することになり、ハードエラーの発生を防げることがあります。

3. パフォーマンスチューニングを行いたい場合。

たとえば、DiskSuite ツールの統計情報表示機能を使用すれば、RAID5 メタデバイスが「OKay」状態であっても、ある特定のスライスに負荷が集中している状態を見ることができます。メタデバイス上の負荷を均一に分散させるため、そのスライスを使用率の少ないディスクのスライスと交換することができます。こ

の種の交換は、メタデバイスへのサービスを中断することなく、オンラインで実行できます。

注 - 必要ならば、DiskSuite ツールを使用して、サブミラー全体を交換できます。そのためには、新しいサブミラー (連結方式オブジェクト) を作成し、交換するサブミラーの先頭にドラッグします。この作業については、168ページの「サブミラーの交換方法 (DiskSuite ツール)」を参照してください。

保守状態と最後にエラーが発生した状態

ミラーや RAID5 メタデバイス内のスライスにエラーが発生した場合、DiskSuite はそのスライスを「Maintenance (保守状態)」にします。「Maintenance」のスライスには、それ以降の読み書きは実行されません。その後、同じメタデバイス内で他のスライスにエラーが発生しても、その取り扱いはメタデバイスの種類に応じて異なります。ミラーは、多数のスライスが「Maintenance」にあっても動作可能であり、まだ読み書きを行うことができます。RAID5 メタデバイスは、定義により、「Maintenance」にあるスライスを1つだけ許容できます。ミラーまたは RAID5 メタデバイスに「Last Erred (最後にエラーが発生した状態)」のスライスがある場合、「Last Erred」とマークされたスライスにはまだ入出力が試みられます。その理由は、「Last Erred」のスライスには、DiskSuite から見て現時点で最新の正しいデータコピーに近いものが含まれているためです。「Last Erred」のスライスをもつメタデバイスは、正常なデバイス (ディスク) のように振る舞い、アプリケーションに入出力エラーを返します。通常、この時点で一部のデータは消失していることがあります。

必ず「Maintenance」のスライスを最初に交換してから、「Last Erred」のスライスを交換します。スライスが交換されて再同期されたら、`metastat(1M)` コマンドを使用してその状態を確認し、データの妥当性をチェックして、その内容が正しいことを確認します。

ミラー: スライスが「Maintenance」にある場合、データは失われていません。スライスは任意の順序で交換したり有効にすることができます。スライスが「Last Erred」にある場合、まず「Maintenance」にある他のすべてのミラー化スライスを交換してから、このスライスを交換できます。通常、「Last Erred」にあるスライスを交換したり有効にするということは、一部のデータが消失した状態であることを意味します。ミラーを修復してから、ミラー上のデータの妥当性をチェックしてください。

RAID5 メタデバイス : RAID5 メタデバイスは、単一のスライス障害に耐性を持ちます。「Maintenance」にあるスライスは、データを失うことなく、安全に交換できます。他のスライスにエラーが発生した場合、そのスライスは「Last Erred」となります。この時点で、RAID5 メタデバイスは読み取り専用デバイスとなります。何らかのエラー回復処置を実行して、RAID5 メタデバイスの状態を非エラー状態とし、データ損失の可能性を低下させる必要があります。RAID5 メタデバイスが「Last Erred」になった場合、データが失われた可能性が高いと言えます。RAID5 メタデバイスを修復したら、そこにあるデータの妥当性をチェックしてください。

ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報

ミラーや RAID5 メタデバイス内のスライスを交換する場合、次のガイドラインに従ってください。

- 必ず「Maintenance (保守状態)」のスライスを先に交換してから、「Last Erred (最後にエラーが発生した状態)」のスライスを交換してください。
- スライスの交換と再同期が終了したら、`metastat(1M)` コマンドを使用してメタデバイスの状態を確認し、データの妥当性をチェックして正しいことを確認してください。通常、「Last Erred」にあるスライスを交換したり有効にするということは、一部のデータが消失した状態であることを意味します。メタデバイスを修復してから、メタデバイス上のデータの妥当性をチェックします。UFSの場合、`fsck(1M)` コマンドを実行してメタデータ (ファイルシステムの構造) の妥当性をチェックしてから、実際のユーザーのデータをチェックします (実際には、ユーザーは自分のファイルを調査する必要があります)。データベースやその他のアプリケーションは、独自の方法でその内部データ構造の妥当性をチェックしなければなりません。
- スライスを交換する場合、状態データベースの複製とホットスペアを必ずチェックしてください。

エラーとして示される状態データベースの複製は、物理ディスクを交換する前に削除し、スライスを有効にする前に追加します。同じことがホットスペアにも適用されます。

- RAID5 メタデバイス : スライスの交換中、データは、現在使用中のホットスペアから復元されるか、使用中のホットスペアがなければ RAID レベル 5 パリティを使用して復元されます。

- ミラー：スライスを交換するとき、DiskSuite は、新しいスライスとミラーの残りの部分との再同期を自動的に開始します。再同期が終了すると、交換されたスライスは読み書きが可能となります。障害の発生したスライスがホットスペアからのデータで交換された場合、ホットスペアは「使用可能」状態となり、他のホットスペア交換にも使用できます。
- 新しいスライスには、古いスライスを置き換えるだけの大きさが必要です。
- 予防対策として、「Last Erred」状態のデバイスを交換する前に、すべてのデータをバックアップしておいてください。

注 - サブミラーや RAID5 メタデバイスは、エラーの発生したスライスの代わりにホットスペアを使用できます。この節で述べた作業を使用して、そのエラーの発生したスライスが有効にされるか交換されると、ホットスペアにはホットスペア集合内で「Available (使用可能)」のマークが付けられ、使用する準備ができます。

▼ サブミラー内のスライスを有効にする方法 (DiskSuite ツール)

この作業は、「エラー状態」にあるサブミラー内のスライスを有効にするために使用します。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、概要 (160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」) と予備情報 (163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」) を読んだことを確認する。
2. オブジェクトリストから、エラーの発生したミラーオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。サブミラーはエラー状態を表示しません。
3. 状態が「重大な障害」となっているサブミラーのストライプ矩形の内部をクリックする。さらにオブジェクトのポップアップメニューを表示し、「情報」を選択する。
「ストライプ情報」ウィンドウが表示されます。
4. 「ストライプ情報」ウィンドウで「重大な障害」状態のスライスを選択し、「有効」をクリックしてから、「閉じる」をクリックする。

スライス状態は、「重大な障害」から「有効」に変化します。サブミラーの状態は、「重大な障害 (エラー状態)」から「重大な障害 (未確定)」に変化します。

5. ミラーオブジェクトの内部をクリックしてから、「確定」をクリックする。
ミラーの再同期が始まります。サブミラーの状態は「コンポーネント再同期」に変化します。
6. 再同期が終了したら、ミラーの状態が「正常」であることを確認する。
7. データの妥当性をチェックする。

注 - スライスを有効化した後でも DiskSuite が「エラー状態」のスライスを通知する場合は、166ページの「サブミラー内のスライスを交換する方法 (DiskSuite ツール)」を参照してください。

▼ サブミラー内のスライスを有効にする方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」の概要、163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」の予備情報をチェックしてから、`metareplace(1M)` コマンドを使用して、サブミラーに含まれるエラーの発生したスライスを有効にします。`metareplace(1M)` では、自動的に再同期を開始して、新しいスライスをミラーの残りの部分と同期させます。

例 - サブミラー内のスライスを有効化

```
# metareplace -e d11 c1t4d0s7
d11: device c1t4d0s7 is enabled
```

ミラー `d11` には、ソフトエラーの発生したスライス `c1t4d0s7` を含むサブミラーがあります。`-e` オプション付きの `metareplace` コマンドを使用すれば、エラーの発生したスライスを有効にできます。

注 - 物理ディスクに欠陥がある場合、167ページの「サブミラー内のスライスを交換する方法 (コマンド行)」の説明に従って、そのディスクをシステム上で使用可能な他のディスク (およびそのスライス) と交換できます。あるいは、ディスクを修復 / 交換し、フォーマットし、この例に示すように `-e` オプション付きの `metareplace` を実行することができます。

▼ サブミラー内のスライスを交換する方法 (DiskSuite ツール)

この作業は、サブミラー内のスライスを新しいスライスと交換するために使用します。DiskSuite によって使用されるディスクのパーティション分割情報は、エラーが発生する前に安全な場所に保存してください。

注 - この作業を行う前に、交換用のスライスが正しくパーティション分割されたことを確認してください。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、概要 (160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」) と予備情報 (163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」) を読んだことを確認する。
2. オブジェクトリストから、エラーの発生したミラーオブジェクトをダブルクリックする。
ミラーがキャンバスに表示されます。サブミラーはエラー状態を表示します。
3. サブミラーオブジェクト (連結方式オブジェクト) の内部からポップアップメニューを表示して「情報」を選択する。
スライスのサイズと状況を調べます。
4. スライスをクリックして、「スライスブラウザ」ウィンドウを開く。
交換の必要なスライスのサイズ以上のサイズをもつ、使用可能なスライスを探します。
5. 交換するスライスが収められているサブミラーオブジェクトのストライプ矩形に、「スライスブラウザ」ウィンドウから交換用のスライスをドラッグする。

6. ミラーオブジェクトの先頭の矩形をクリックしてから「確定」をクリックする。
DiskSuite は、交換されたサブミラーの再同期を始めます。
7. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。
8. データの妥当性をチェックする。

注 - 交換用のスライスをオブジェクトにドラッグするとき、連結方式オブジェクト内の他の矩形ではなく、エラーの発生したスライスのドライブ番号が収められている矩形内のカーソルをポインタで指します。

▼ サブミラー内のスライスを交換する方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」の概要、163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」の予備情報をチェックしてから、`metareplace(1M)` コマンドを使用して、サブミラー内のスライスを交換する。

例 - ミラー内の障害の発生したスライスを交換

```
# metastat d6
d6: Mirror
  Submirror 0: d16
    State: Okay
  Submirror 1: d26
    State: Needs maintenance
...
d26: Submirror of d6
  State: Needs maintenance
  Invoke: metareplace d6 c0t2d0s2 <new device>
...
# metareplace d6 c0t2d0s2 c0t2d2s2
d6: device c0t2d0s2 is replaced with c0t2d2s2
```

`metastat` コマンドは、ミラー `d6` にはサブミラー `d26` があり、スライスが「Needs Maintenance」状態であることを確認します。`metareplace` コマンド

は、metastat 出力の「Invoke」行で指定されたスライスを、システム上で使用可能な他のスライスと交換します。システムは、スライスが交換されたことを確認し、サブミラーの再同期を開始します。

▼ サブミラーの交換方法 (DiskSuite ツール)

サブミラー全体を交換するには、まず、交換対象のサブミラーのサイズ以上のサイズをもつ新しいストライプか連結を構築します。交換用サブミラーに使用されるスライスは、他のサブミラーとは異なるコントローラに置いてください。ストライプや連結の作成方法については、第 2 章を参照してください。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、概要 (160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」) と予備情報 (163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」) を読んだことを確認する。
2. オブジェクトリストから、エラーの発生したミラーオブジェクトをダブルクリックする。
ミラーがキャンバスに表示されます。
3. サブミラーを交換するオブジェクトリストから、連結方式オブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
4. 交換されるサブミラーの先頭に連結方式オブジェクトをドラッグする。
新しい連結方式オブジェクトがエラーの発生したオブジェクトと置き換えられます。
5. ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
新しいサブミラーの再同期が開始されます。
6. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。
7. データの妥当性をチェックする。
8. [オプション] 交換された連結方式オブジェクトを削除する。

▼ サブミラーの交換方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」の概要、163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」の予備情報をチェックしてから、`metadetach(1M)`、`metaclear(1M)`、`metainit(1M)`、`metattach(1M)` の各コマンドを使用して、サブミラー全体を交換します。

例 - ミラー内のサブミラーを交換

```
# metastat d20
d20: Mirror
    Submirror 0: d21
        State: Okay
    Submirror 1: d22
        State: Needs maintenance
...
# metadetach -f d20 d22
d20: submirror d22 is detached
# metaclear d22
d22: Concat/Stripe is cleared
# metainit d22 2 1 c1t0d0s2 1 c1t0d1s2
d22: Concat/Stripe is setup
# metattach d20 d22
d20: components are attached
```

`metastat` コマンドは、2面ミラー `d20` に、「Needs Maintenance」状態のサブミラー `d22` があることを確認します。この場合、サブミラー全体がクリアされてから、再作成されます。`metadetach` コマンドは、`-f` オプション (強制的に切断する) を使用して、エラーの発生したサブミラーをミラーから切断します。`metaclear` コマンドは、サブミラーをクリアします。`metainit` コマンドは、サブミラー `d22` を新しいスライスで再作成します。`metattach` コマンドは、再構築されたサブミラーを接続するため、ミラーの再同期が自動的に開始されます。

注 - ミラーが1面のミラーとなっている間、データの冗長性を実現する機能は一時的に失われます。

▼ RAID5 メタデバイス内のスライスを有効にする方法 (DiskSuite ツール)

この作業は、「保守状態」にある RAID5 メタデバイス内のスライスを有効にするために行います。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、概要 (160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」) と予備情報 (163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」) を読んだことを確認する。
2. オブジェクトリストから、エラーの発生した **RAID5** オブジェクトをダブルクリックする。
RAID5 オブジェクトがキャンバスに表示されます。
3. **RAID5** オブジェクトのポップアップメニューを表示し、「情報」を選択する。
「RAID 情報」ウィンドウが表示されます。
4. 「保守状態」のスライスを選択し、「有効」をクリックしてから、「閉じる」をクリックする。
スライスの状態が「有効」に変化し、RAID5 の状態が「緊急 (未確定)」に変化します。
5. **RAID5** オブジェクトを選択してから「確定」をクリックする。
RAID の状態が「緊急 - 再同期中」に変化し、スライスの状態が「再同期中」に変化します。
6. 再同期が終了したら、**RAID5** オブジェクトの状態が「正常」であることを確認する。
7. データの妥当性をチェックする。

注 - スライスを有効化しても、まだ DiskSuite が「保守状態」のスライスを通知する場合は、171ページの「RAID5 スライスの交換方法 (DiskSuite ツール)」を参照してください。

▼ RAID5 メタデバイス内のスライスを有効にする方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」の概要、163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」の予備情報をチェックしてから、`metareplace(1M)` コマンドを使用して、RAID5 メタデバイス内のエラーの発生したスライスを有効にします。`metareplace(1M)` は、自動的に再同期を開始して、新しいスライスを RAID5 メタデバイス内の残りのスライスと同期させます。

例 - RAID5 メタデバイス内のスライスを有効化

```
# metareplace -e d20 c2t0d0s2
```

RAID5 メタデバイス `d20` には、ソフトウェアのエラーが発生したスライス `c2t0d0s2` があります。`-e` オプション付きの `metareplace` コマンドは、スライスを有効にします。

注 - ディスクドライブに欠陥がある場合、172ページの「RAID5 スライスの交換方法 (コマンド行)」の説明に従って、そのディスクをシステム上で使用可能な他のディスク (およびそのスライス) と交換することができます。あるいは、ディスクを修復もしくは交換し、フォーマットし、`-e` オプション付きの `metareplace` を実行することができます。

▼ RAID5 スライスの交換方法 (DiskSuite ツール)

この作業は、1つのスライスにだけエラーの発生した RAID5 メタデバイス内で、エラーの発生したスライスを交換するために使用します。



注意 - 複数のスライスにエラーが発生している場合、エラーになったスライスを交換すると、正確でないデータが生成される可能性があります。この場合、データの整合性には疑問があります。

1. **前提条件** (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、**概要** (160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」) と**予備情報** (163ページの「ミラーと RAID5 メタデバイス内のスライスを交換および有効にするための予備情報」) を読んだことを確認する。

2. オブジェクトリストから、エラーの発生した **RAID5** メタデバイスをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. スライスをクリックして、「スライスブラウザ」ウィンドウを開く。
4. エラーの発生したスライスと交換するための使用可能なスライスを選択し、**RAID5** メタデバイスオブジェクトの矩形内のスライスにドラッグする。
そのスライスは、少なくともデバイス内で最小のスライスと同じ大きさでなければなりません。
5. **RAID5** メタデバイスオブジェクトの先頭の矩形をクリックして、「確定」をクリックする。
交換中には、メタデバイスと新しいスライスの状態は「再同期中」となります。メタデバイスがこの状態にある間、メタデバイスを使用し続けることができません。
6. 再同期が終了したら、**RAID5** オブジェクトの状態が「正常」であることを確認する。
状態を更新するためには、「ファイル」メニューから「構成の再走査」を選択することが必要な場合もあります。
7. データの妥当性をチェックする。

▼ RAID5 スライスの交換方法 (コマンド行)

この作業は、1つのスライスにだけエラーの発生した RAID5 メタデバイス内で、エラーの発生したスライスを交換するために使用します。



注意 - 複数のスライスにエラーが発生している場合、エラーになったスライスを交換すると、正確でないデータが生成される可能性があります。この場合、データの整合性には疑問があります。

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件、160ページの「ミラーと RAID5 メタデバイスにおけるスライスの交換と有効化の概要」の概要、163ページの「ミラーと RAID5 メタデバイス内のスライスを交換およ

び有効にするための予備情報」の予備情報をチェックしてから、この例に従って RAID5 メタデバイス内のエラーの発生したスライスを交換します。詳細については、metareplace(1M) のマニュアルページを参照してください。

例 - RAID5 スライスの交換

```
# metastat d1
d1: RAID
State: Needs Maintenance
  Invoke: metareplace d1 c0t14d0s6 <new device>
  Interlace: 32 blocks
  Size: 8087040 blocks
Original device:
  Size: 8087520 blocks
Device          Start Block  Dbase State      Hot Spare
c0t9d0s6        330         No    Okay
c0t13d0s6       330         No    Okay
c0t10d0s6       330         No    Okay
c0t11d0s6       330         No    Okay
c0t12d0s6       330         No    Okay
c0t14d0s6       330         No    Maintenance

# metareplace d1 c0t14d0s6 c0t4d0s6
d1: device c0t14d0s6 is replaced with c0t4d0s6
# metatstat d1
d1: RAID
  State: Resyncing
  Resync in progress: 98% done
  Interlace: 32 blocks
  Size: 8087040 blocks
Original device:
  Size: 8087520 blocks
Device          Start Block  Dbase State      Hot Spare
c0t9d0s6        330         No    Okay
c0t13d0s6       330         No    Okay
c0t10d0s6       330         No    Okay
c0t11d0s6       330         No    Okay
c0t12d0s6       330         No    Okay
c0t4d0s6        330         No    Resyncing
```

metastat コマンドは、RAID5 メタデバイス d1 内でエラーの発生したスライスから回復するために実行する操作内容を表示します。使用可能なスライスを探してから、metareplace コマンドを実行します。その際、最初にエラーの発生したスライスを指定し、次に交換用のスライスを指定します (他のスライスが使用できない場合、-e オプション付きの metareplace(1M) コマンドを実行し、エラーの発生したデバイスを再同期することによって、予想されるソフトエラーからの回復を試みます)。複数のエラーが存在する場合、「Maintenance (保守)」状態にあるスライスを最初に交換または有効にする必要があります。そうすれば「Last Erred (最後にエ

ラーが発生した状態)のスライスを修復できます。metareplace の後、metastat が再同期の進捗状況を監視します。交換中には、メタデバイスと新しいスライスの状態は「Resyning (再同期中)」となります。この状態の間は、メタデバイスを使用し続けることができます。

注 - エラーの発生していないデバイス上で metareplace(1M) コマンドを使用して、ディスク (スライス) を変更することができます。この方法は、RAID5 メタデバイスのパフォーマンスをチューニングするときに便利です。

ホットスペア集合を交換するための予備情報

- ホットスペアは、関連付けられていたすべてのホットスペア集合内で交換できます。しかし、「In Use (使用中)」状態のホットスペアは、他のホットスペアと交換できません。
- 交換時に、ホットスペア集合内のホットスペアの順序は変更されません。
- 入出力エラーが発生したら、ホットスペアは「Broken (故障)」状態に置かれます。この状態を修復するには、まず壊れたホットスペアスライスを修復または交換します。次に、DiskSuite ツールを使用するか、-e オプション付きの metahs(1M) コマンドを使用して、スライスを「Available (使用可能)」状態に戻します。

注 - サブミラーや RAID5 メタデバイスは、エラーの発生したスライスの代わりに、ホットスペアを使用していることがあります。エラーの発生したそのスライスが有効にされるか交換されると、ホットスペアにはホットスペア集合内で「Available (使用可能)」のマークが付けられ、使用する準備が整います。

▼ ホットスペア集合内のホットスペアを交換する方法 (DiskSuite ツール)

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (174ページの「ホットスペア集合を交換するための予備情報」) を読んだことを確認する。
2. オブジェクトリストから、ホットスペア集合オブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。

3. スライスをクリックして、「スライスブラウザ」ウィンドウを開く。
4. 交換の必要なスライスと同じサイズのスライスを探す。交換するスライスが収められているホットスペア集合オブジェクトの矩形にスライスをドラッグする。
5. ホットスペア集合オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
6. コンフィグレーションログを表示して、ホットスペア集合が確定されたことを確認する。

▼ ホットスペア集合内のホットスペアの交換方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 174ページの「ホットスペア集合を交換するための予備情報」の予備情報をチェックしてから、`metahs(1M)` を使用してホットスペアを交換します。詳細については、`metahs(1M)` のマニュアルページを参照してください。

例 - 1 つのホットスペア集合内でホットスペアを交換

```
# metastat hsp003
hsp003: 1 hot spare
        c0t2d0s2                Broken        5600 blocks
# metahs -r hsp003 c0t2d0s2 c3t1d0s2
hsp003: Hotspare c0t2d0s2 is replaced with c3t1d0s2
```

`metastat` コマンドは、ホットスペアが使用中でないことを確認します。`metahs -r` コマンドは、ホットスペア `/dev/dsk/c0t2d0s2` を、ホットスペア集合 `hsp003` 内の `/dev/dsk/c3t1d0s2` で置き換えます。

例 - 関連付けられたすべてのホットスペア集合内のホットスペアを交換

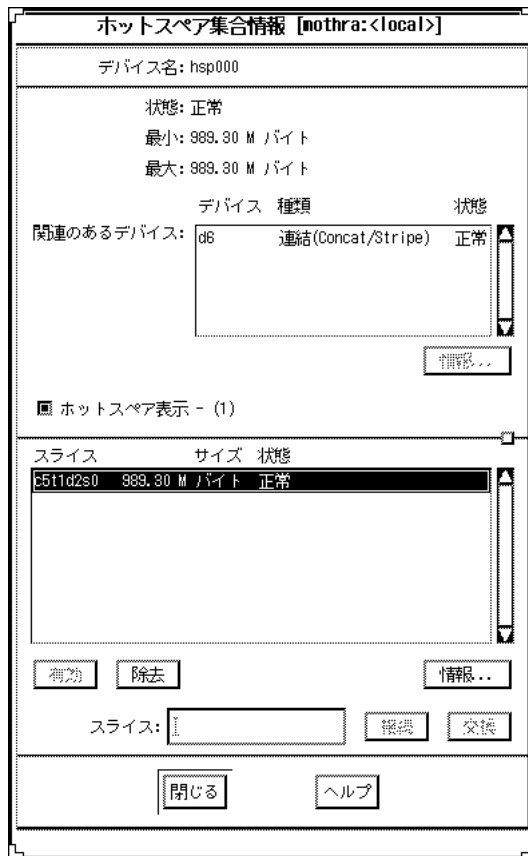
```
# metahs -r all c1t0d0s2 c3t1d0s2
hsp001: Hotspare c1t0d0s2 is replaced with c3t1d0s2
hsp002: Hotspare c1t0d0s2 is replaced with c3t1d0s2
hsp003: Hotspare c1t0d0s2 is replaced with c3t1d0s2
```

キーワード `all` は、ホットスペア `/dev/dsk/c1t0d0s2` を、その関連付けられたすべてのホットスペア集合内の `/dev/dsk/c3t1d0s2` で置き換えます。

▼ ホットスペアを有効にする方法 (DiskSuite ツール)

この作業は、ホットスペアが修復された後で、ホットスペアを有効にする (使用可能にする) ために使用します。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (174ページの「ホットスペア集合を交換するための予備情報」) を読んだことを確認する。
2. オブジェクトリストから、ホットスペア集合オブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. ホットスペア集合オブジェクトのポップアップメニューを表示し、「情報」を選択する。
「ホットスペア情報」ウィンドウが表示されます。



4. 修復されたスライスをスライスリストから選択して、「有効」をクリックする。
5. 「閉じる」をクリックして、「ホットスペア情報」ウィンドウを閉じる。
6. ホットスペア集合オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
7. ホットスペア集合オブジェクトの状態が「正常」に変化していることを確認する。

▼ ホットスペアを有効にする方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 174ページの「ホットスペア集合を交換するための予備情報」の予備情報をチェックしてから、`metahs(1M)` コマンドを使用して、ホットスペアを「Available (使用可能)」状態に戻します。詳細は、`metahs(1M)` のマニュアルページを参照してください。

例 - ホットスペアの有効化

```
# metahs -e c0t0d0s2
```

この例では、ホットスペア `/dev/dsk/c0t0d0s2` は修復された後で使用可能な状態になります。ホットスペア集合を指定する必要はありません。

トランスメタデバイスの障害の修復

トランスメタデバイスは、マスターデバイスとロギングデバイスから成る「階層構造」メタデバイスであり、しかもロギングデバイスは複数のファイルシステムで共有できるため、エラーの発生したトランスメタデバイスの修復には、特別な障害回復作業が必要となります。

デバイスエラーやファイルシステムのパニックが発生した場合、コマンド行ユーティリティを使用して対処することが必要です。

ファイルシステムのパニック

ファイルシステムが、その使用中に内部的な不一致を検出した場合、システムをパニック状態にします。ファイルシステムが UFS ロギング用に設定されている場合、リポート時にファイルシステムのチェックが必要であることをトランスメタデバイスに通知します。トランスメタデバイスは、自分自身を「Hard Error (ハードエラー)」状態に移行します。同じロギングデバイスを共有している他のトランスメタデバイスも、すべて「Hard Error」状態となります。

リポート時に、`fsck` はファイルシステムのチェックと修復を行なって、ファイルシステムの状態を「Okay (正常)」に戻します。この操作は、影響されるロギングデ

バスの /etc/vfstab ファイルに記載されたすべてのトランスメタデバイスに対して行われます。

トランスメタデバイスのエラー

デバイスエラーによってデータが失われることがあります。ロギングデバイスに読み取りエラーが発生すると、重大なデータ損失を引き起こすことがあります。したがって、ロギングデバイスのミラー化を強くお勧めします。

トランスメタデバイスがログに記録されたデータの処理中に、マスターデバイスやロギングデバイスにデバイスエラーが発生した場合、そのデバイスは「Okay」状態から「Hard Error」状態に移行します。デバイスが「Hard Error」状態または「Error (エラー)」状態であれば、デバイスエラーかファイルシステムのパニックが発生しています。

注 - エラーの発生したロギングデバイスを共有するデバイスも、すべて「Error」状態となります。

▼ ファイルシステムのパニックを起こしたトランスメタデバイスの回復方法 (コマンド行)

fsck では修復できないファイルシステムの場合、影響されるロギングデバイスを共有するファイルシステムをもつ各トランスメタデバイス上で fsck を実行します。

例 - トランスメタデバイスの回復

```
# fsck /dev/md/rdisk/<トランスメタデバイス名>
```

影響を受けたすべてのトランスメタデバイスがチェックされ、正しく修復された後に限り、fsck はエラーの発生したトランスメタデバイスの状態を「Okay」にリセットします。

▼ ハードエラーを起こしたトランスメタデバイスの回復方法 (コマンド行)

この作業は、トランスメタデバイスを「Okay (正常)」状態に移行するために使用します。

トランスメタデバイスの状態をチェックするには、144ページの「メタデバイスとホットスベア集合の状態のチェック方法 (コマンド行)」を参照してください。

ログに記録されたデータの処理中に、マスターデバイスまたはログデバイスにエラーが発生した場合、そのデバイスは「Okay」状態から「Hard Error (ハードエラー)」状態に移行します。デバイスが「Hard Error」状態または「Error (エラー)」状態にある場合、デバイスエラーかファイルシステムのパニックが発生しています。この2つの場合からの障害回復の作業は同じです。

注 - ログ (ロギングデバイス) が共有される場合、トランスメタデバイス内のどれかのスライスに障害が発生すると、そのトランスメタデバイスに関連付けられたすべてのスライスやメタデバイスがエラー状態に切り換えられます。

この作業での手順を次に示します。

- 影響を受けたファイルシステムのマウント解除
 - アクセス可能なデータのバックアップ
 - デバイスエラーの修復
 - ファイルシステムの修復 (fsck(1M) または newfs(1M))
1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) と予備情報 (178ページの「トランスメタデバイスの障害の修復」) をチェックしてから、lockfs(1M) コマンドを実行して、どのファイルシステムがロックされているかを調べる。

```
# lockfs
```

影響されるファイルシステムは、ロックタイプが hard となって表示されます。同じロギングデバイスを共有するファイルシステムは、すべてハードロックされます。

2. 影響されるファイルシステムをマウント解除する。

ロックされたファイルシステムは、たとえエラーが発生したときに使用中であっても、マウント解除できます。影響されるプロセスが、ハードロックまたはマウント解除されたファイルシステム上の開かれたファイルやディレクトリにアクセスを試みると、EIO エラーが返されます。

3. [オプション] アクセス可能なデータをすべてバックアップする。

デバイスエラーの修復を試みる前に、できるだけ多くのデータを回収したい場合があります。バックアップ作業に、ファイルシステムをマウントする必要がある場合 (tar や cpio など)、ファイルシステムを読み取り専用でマウントできます。バックアップ作業に、マウントされたファイルシステムが必要ない場合 (dump や volcopy など)、トランスメタデバイスに直接アクセスできます。

4. デバイスエラーを修復する。

この時点では、トランスメタデバイスを読み書きアクセス用にオープンしたりマウントしたりすると、ロギングデバイス上のすべてのアクセス可能データは、適切なマスターデバイスにローリングを開始します。読み書きできないデータは、すべて破棄されます。しかし、トランスメタデバイスを読み取り専用アクセス用にオープンしたりマウントしたりすると、ログは単に再走査されるだけで、マスターデバイスにはロールフォワードされないため、エラーは修復されません。つまり、マスターデバイスとロギングデバイス上のすべてのデータは、最初の読み書きオープンやマウントまでは変化しません。

5. fsck(1M) を実行してファイルシステムを修復するか、データを復元する必要がある場合は newfs(1M) を実行する。

同じロギングデバイスを共有するすべてのトランスメタデバイス上で、fsck を実行します。これらのトランスメタデバイスが fsck によってすべて修復されると、「Okay」状態に戻ります。

newfs(1M) コマンドもファイルシステムを「Okay」状態に戻しますが、ファイルシステム上のデータはすべて破壊されます。一般に、バックアップからファイルシステムの復旧を計画するときは、newfs(1M) が使用されます。

同じロギングデバイスを共有するすべてのトランスメタデバイス上で fsck(1M) コマンドや newfs(1M) コマンドを実行すれば、それらのデバイスは「Okay」状態に戻ります。

6. metastat(1M) コマンドを実行して、影響されるデバイスの状態が「Okay」に戻ったことを確認する。

例 - ロギングデバイスのエラー

```
# metastat d5
d5: Trans
   State: Hard Error
   Size: 10080 blocks
```

(続く)

```
Master Device: d4
Logging Device: c0t0d0s6

d4: Mirror
State: Okay
...
c0t0d0s6: Logging device for d5
State: Hard Error
Size: 5350 blocks
...
# fsck /dev/md/rdisk/d5
** /dev/md/rdisk/d5
** Last Mounted on /fs1
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
WARNING: md: logging device: /dev/dsk/c0t0d0s6 changed state to
Okay
4 files, 11 used, 4452 free (20 frags, 554 blocks, 0.4%
fragmentation)
# metastat d5
d5: Trans
State: Okay
Size: 10080 blocks
Master Device: d4
Logging Device: c0t0d0s6

d4: Mirror
State: Okay
...

c0t0d0s6: Logging device for d5
State: Okay
...
```

この例では、「Hard Error (ハードエラー)」状態のロギングデバイスをもつトランスメタデバイス d5 を修復します。トランスデバイス自身の上で fsck を実行しなければなりません。これによって、トランスメタデバイスの状態は「Okay」に移行します。metastat は、状態が「Okay」であることを確認します。

スライスとメタデバイスの拡張

この節では、スライス (非メタデバイス) やメタデバイスを拡張 (つまり領域を追加) する作業について説明します。たとえば、連結方式メタデバイスがファイルシステムで満杯になった場合、さらにスライスを追加することによって、新しく追加された領域までファイルシステムを拡張することができます。

スライスとメタデバイスを拡張するための予備情報

- メタデバイスは、ファイルシステム、アプリケーション、データベースのいずれで使用していても、スライスを追加することによって拡張できます。これには、ストライプ方式メタデバイス、連結、ストライプ方式の連結、ミラー (サブミラー)、RAID5 デバイス、トランスメタデバイスが含まれます。
- 通常、既存のファイルシステムが使用中であるときに、そのファイルシステムを含むメタデバイスを連結できます。そのファイルシステムが UFS である限り、データへのアクセスを中断することなく、(`growfs(1M)` コマンドによって) より大きな領域を満たすよう拡張できます。
- ファイルシステムをいったん拡張すると、縮小することはできません。これは UFS の制約です。
- raw デバイスを使用するアプリケーションとデータベースには、追加された領域をアプリケーションが認識できるように拡張するための独自の方法が必要となります。DiskSuite はこの機能を提供しません。
- スライスが RAID5 メタデバイスに追加されると、そのデバイスに対する連結となります。新しいスライスにはパリティ情報が含まれません。しかし、新しいスライス上のデータは、デバイスに対して行われる全体的なパリティ計算によって保護されます。
- スライスをいったんメタデバイスに接続すると、除去することはできません。
- ログイングデバイスを拡張するには、新たなスライスを追加します。DiskSuite は、リポート時に追加領域を自動的に認識するため、`growfs(1M)` コマンドを実行する必要はありません。

▼ 既存のデータを含むスライスの拡張方法 (DiskSuite ツール)

この作業は、領域を使い果たした単一スライスから連結を作成するために使用します。この作業は、ファイルシステムやデータベースなどのアプリケーションに使用できます。この作業での手順を次に示します。

- より多くの領域を必要とするスライスを連結に置く
 - メタデバイス上でファイルシステムをマウント解除してから再マウントするか、またはリブートする
 - 連結に他のスライスを追加する
 - ファイルシステムやアプリケーションの拡張
1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (183ページの「スライスとメタデバイスを拡張するための予備情報」) を読んだことを確認する。
 2. 連結方式テンプレートをクリックする。
キャンバスには、未割り当てで未確定の連結方式オブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
 3. [オプション] デフォルトのメタデバイス名を変更する。オブジェクトのポップアップメニューを表示して「情報」を選択する。「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。「閉じる」をクリックする。
 4. スライスをクリックし、「スライスブラウザ」ウィンドウを開く。
 5. データを含むスライスを選択して、連結方式オブジェクトにドラッグする。
スライスがマウントされていることを示すダイアログボックスが表示されたら、「継続」をクリックします。
 6. 連結方式オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
マウントされたファイルシステムの場合: 「確定操作に対する警告」ダイアログボックスが表示されます。ここで「確定」をクリックします (ファイルシステムが `/etc/vfstab` ファイルに記述されている限り、DiskSuite ツールはこの内容を更新してメタデバイス名を使用します。ファイルシステムが手作業でマウント

された場合、ファイルシステムがマウントされるたびにブロックメタデバイス名を使用する必要があります)。

7. [マウントされたファイルシステムの場合] ファイルシステムをマウント解除してから、再マウントする。ファイルシステムが使用中である場合は、リブートする必要がある。
8. **DiskSuite** ツールで、連結オブジェクトと「スライスブラウザ」ウィンドウを開く。別のスライスをオブジェクトにドラッグする。
9. オブジェクトが選択されたことを確認し、「確定」をクリックする。
マウントされたファイルシステムの場合：GrowFS のダイアログボックスが表示されます。「今拡張する」をクリックして、growfs(1M) コマンドの実行を開始します。「ファイルシステムの拡張中」メッセージが表示されます。「ファイルシステム拡張を中断」をクリックした場合、このコマンドは中止されます。そうでない場合、growfs(1M) コマンドが終了すると、「DiskSuite ツール」ウィンドウに戻ります。
raw デバイスを使用するアプリケーション：このようなアプリケーションは、DiskSuite に依存しない独自の方法で拡張された領域を認識する必要があります。
10. コンフィグレーションログを表示して、連結が確定されたことを確認する。

注 - 拡張の際、ファイルシステムはロックされ、書き込みアクセスができなくなります。書き込みアクセスは透過的に中断され、growfs(1M) がファイルシステムをロックを解除した段階で再開されます。読み取りアクセスは影響を受けませんが、ロックが有効な間はアクセス時間は記録されません。

例 - 拡張された連結オブジェクト

この例では、確定された連結を示します。最初のオブジェクトは、1つのスライス c0t1d0s3 から成るファイルシステムで構成されていました。その後、スライス c0t2d0s5 が連結され、growfs(1M) コマンドが自動的に実行されて、領域全体が使用できるようになりました。



▼ 既存のデータを含むスライスの拡張方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 183ページの「スライスとメタデバイスを拡張するための予備情報」の予備情報をチェックしてから、次の例に従って、マウントされたファイルシステムを含むスライスを拡張します。

例 - 連結の作成によるファイルシステムの拡張

```
# umount /docs
# metainit d25 2 1 c0t1d0s2 1 c0t2d0s2
d25: Concat/Stripe is setup
(ファイルシステムがメタデバイス d25 を参照するよう、/etc/vfstab ファイルを編集する)
# mount /docs
```

この例では、`/dev/dsk/c0t1d0s2` (`/docs` にマウントされたファイルシステムを含む) と `/dev/dsk/c0t2d0s2` という 2 つのスライスから、`d25` という連結を作成します。ファイルシステムは、最初にマウント解除しなければなりません。



注意 - metainit(1M) コマンド内の最初のスライスは、ファイルシステムを含むスライスでなければなりません。そうでない場合、データを消去することになります。

次に、`/etc/vfstab` ファイル内のファイルシステムのエントリが、メタデバイスを参照するように変更 (または初めて入力) されます。たとえば、次の行は、

```
/dev/dsk/c0t1d0s2 /dev/rdisk/c0t1d0s2 /docs ufs 2 yes -
```

次のように変更します。

```
/dev/md/dsk/d25 /dev/md/rdsk/d25 /docs ufs 2 yes -
```

最後に、ファイルシステムを再マウントします。

次の作業

UFS の場合、メタデバイス上で `growfs(1M)` コマンドを実行します。198ページの「ファイルシステムの拡張方法 (コマンド行)」を参照してください。

`raw` メタデバイスを使用するデータベースなどのアプリケーションは、指定された方法でメタデバイスの認識や追加領域の拡張を行うことが必要です。

▼ 既存の連結方式の拡張方法 (DiskSuite ツール)

この作業では、既存の連結やストライプに新たなストライプを追加しているものと想定します。障害からの復旧の一部としてストライプ方式の連結を再作成する必要がある場合は、157ページの「スライス障害の後でストライプや連結を再作成する方法 (DiskSuite ツール)」を参照してください。

ストライプ方式の連結を使用すれば、既存のストライプや連結を拡張できます。たとえば、ストライプが領域を使い果たした場合、それをストライプ方式の連結にして、データのバックアップや復元を行うことなく拡張できます。

注 - 複数のスライスを既存の連結方式オブジェクトにドラッグした場合、オプションによって、それらのスライスを連結やストライプにすることができます。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (183ページの「スライスとメタデバイスを拡張するための予備情報」) を読んだことを確認する。
2. オブジェクトリストから連結方式オブジェクトをダブルクリックする。
DiskSuite ツールは、キャンバスに連結方式オブジェクトを表示します。
3. スライスをクリックして、「スライスブラウザ」ウィンドウを開く。
4. 連結したいスライスを別のストライプとして選択する。スライスをメタデバイスオブジェクトの先頭の矩形にドラッグする。

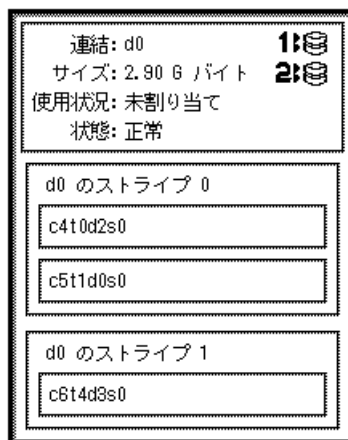
複数のスライスをドラッグした場合、スライスの追加方法 (ストライプ方式として、または単純連結方式として) を選択するよう、ダイアログボックスから要求されます。「ストライプ方式」または「単純連結方式」をクリックします。

新たなストライプ方式メタデバイスや連結方式メタデバイスが、連結方式オブジェクトの末尾に追加されます。

5. 連結方式オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
マウントされたファイルシステムの場合：GrowFS のダイアログボックスが表示されます。「今拡張する」をクリックして `growfs(1M)` コマンドの実行を開始します。「ファイルシステムの拡張中」メッセージが表示されます。「ファイルシステムの拡張を中断」をクリックした場合、このコマンドは中止されます。そうでない場合、`growfs(1M)` コマンドが終了すると、「DiskSuite ツール」ウィンドウに戻ります。
raw デバイスを使用するアプリケーション：このようなアプリケーションは、DiskSuite に依存しない独自の方法で拡張された領域を認識する必要があります。
6. コンフィグレーションログを表示して、連結ストライプが確定されたことを確認する。

例 - 連結ストライプオブジェクト

この例では、他のスライスが追加された 2 つのスライスから成るストライプ方式メタデバイスを示します。連結方式オブジェクトは、ストライプ 0 (元のストライプ) とストライプ 1 (追加されたストライプ) のラベルが付けられた 2 つのストライプ矩形の中にスライスを表示します。



▼ 既存のストライプの拡張方法 (コマンド行)

この作業では、既存のストライプに新たなストライプを追加しているものと想定します。障害からの復旧の一部として、`metainit(1M)` コマンドを使用してストライプ方式の連結を再作成する必要がある場合は、158ページの「スライス障害の後でストライプや連結を再作成する方法 (コマンド行)」を参照してください。

ストライプ方式の連結を使用すれば、既存のストライプを拡張できます。たとえば、ストライプが領域を使い果たした場合、それをストライプ方式の連結にして、データのバックアップや復元を行うことなく拡張できます。

注 - DiskSuite ツールを使用して複数のスライスを既存のストライプ方式メタデバイスにドラッグした場合、オプションによって、それらのスライスを連結やストライプにすることができます。`metattach(1M)` コマンドを使用して複数のスライスを既存のストライプ方式メタデバイスに追加する場合、これらはストライプとして追加しなければなりません。

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 183ページの「スライスとメタデバイスを拡張するための予備情報」の予備情報をチェックしてから、`metattach(1M)` コマンドを使用してストライプ方式の連結を作成します。詳細については、`metattach(1M)` のマニュアルページを参照してください。

例 - 1 つのスライスを接続してストライプ方式の連結を作成

```
# metattach d2 c1t2d0s2
d2: components are attached
```

この例では、既存のストライプ `d2` にスライスを接続します。システムはスライスが連結されたことを確認します。

例 - 既存のメタデバイス内で同じ数のスライスを追加することによって連結ストライプを作成

```
# metattach d25 c1t2d0s2 c1t2d1s2 c1t2d3s2
d25: components are attached
```

この例では、既存の3面のストライプ方式メタデバイス d25 を取り込み、別の3面のストライプを連結します。接続されたスライスには飛び越し値が指定されていないため、これらのスライスは d25 に設定された飛び越し値を継承します。システムは、連結方式オブジェクトが設定されたことを確認します。

注 - アプリケーションの種類にもよりますが、同じ数のスライスを接続することによって、メタデバイスのパフォーマンスが低下しないことがあります。

例 - ストライプ方式の連結を最初から作成

```
# metainit d1 3 2 c0t0d0s2 c1t0d0s2 -i 16k \
  2 c1t2d0s2 c1t2d1s0 -i 32k \
  2 c2t0d0s2 c2t0d1s2
d1: Concat/Stripe is setup
```

通常、このようなメタデバイスを最初から作成することはありません。この例では、d1 が3つのストライプの連結(最初の数字3)であることを示しています。最初のストライプは2つのスライスから構成されます(数字3に続く数字2)。-i 16k は、16K バイトの飛び越しを指定します。2番目のストライプは(2行目の数字2で示されるように)2つのスライスから成り、32K バイトの飛び越しを使用します。最後のストライプは2つのスライスから成ります。3番目のストライプには飛び越しが指定されていないため、その前のストライプから値を継承します(この場合は32K バイト)。

次の作業

UFS の場合、メタデバイス上で growfs(1M) コマンドを実行します。198ページの「ファイルシステムの拡張方法(コマンド行)」を参照してください。

raw メタデバイスを使用するデータベースなどのアプリケーションは、独自の方法でメタデバイスの認識や追加領域の拡張を行う必要があります。

新しく作成したストライプ方式の連結をファイルシステム用に準備するには、126ページの「メタデバイス上にファイルシステムを作成する方法 (コマンド行)」を参照してください。

▼ ミラーの拡張方法 (DiskSuite ツール)

この作業は、ミラーのサブミラーを拡張するために使用します。各サブミラーを拡張する必要があります。サブミラーの使用中に、オフラインにすることなく、サブミラーを拡張できます。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (183ページの「スライスとメタデバイスを拡張するための予備情報」) を読んだことを確認する。
2. オブジェクトリストから既存のミラーオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. スライスをサブミラーに追加するには、スライスをクリックして「スライスブラウザ」ウィンドウを開く。連結方式オブジェクトを追加するには、オブジェクトリストからオブジェクトを選択する。
4. 「スライスブラウザ」ウィンドウからスライスを選択するか、オブジェクトリストから連結方式オブジェクトを選択して、ミラーオブジェクト内の1つのサブミラー矩形の先頭にドラッグする。
適切なサイズのスライスまたは連結方式を選択します。複数のスライスを選択するには、Control キーを押しながらクリックします。
5. スライスや連結方式オブジェクトをミラーオブジェクトにドラッグすると、サブミラーは追加の後で異なるサイズとなるため、警告ダイアログボックスが表示されます。警告ダイアログボックスで「継続」をクリックする。
スライスまたは連結方式オブジェクトは、サブミラー矩形の末尾に追加されません。
6. スライスまたは連結方式オブジェクトを選択し、ミラーオブジェクトの2番目にあるいずれかのサブミラー矩形の先頭にドラッグする。
妥当性検査ダイアログボックスが表示されます。「了解」をクリックします。
3番目のサブミラーがある場合は、この手順を繰り返します。

7. ミラーオブジェクトの先頭をクリックし、「確定」をクリックする。

マウントされたファイルシステムの場合：GrowFS のダイアログボックスが表示されます。「今拡張する」をクリックし、growfs(1M) コマンドの実行を開始します。「ファイルシステムの拡張中」メッセージが表示されます。「ファイルシステムの拡張を中断」をクリックした場合、このコマンドは中止されます。そうでない場合、growfs(1M) コマンドが終了すると、「DiskSuite ツール」ウィンドウに戻ります。

raw デバイスを使用するアプリケーション：このようなアプリケーションは、DiskSuite に依存しない独自の方法で拡張された領域を認識する必要があります。

8. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。

▼ ミラーの拡張方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 183ページの「スライスとメタデバイスを拡張するための予備情報」の予備情報をチェックしたら、metattach(1M) コマンドを使用して各サブミラーにスライスを追加接続します。ミラー内の各サブミラーは、拡張しなければなりません。詳細は、metattach(1M) のマニュアルページを参照してください。

例 - マウントされたファイルシステムを含む 2 面のミラーの拡張

```
# metastat
d8: Mirror
   Submirror 0: d9
       State: Okay
   Submirror 1: d10
       State: Okay
...
# metattach d9 c0t2d0s5
d9: component is attached
# metattach d10 c0t3d0s5
d10: component is attached
```

この例では、2つのディスクドライブをミラーの2つのサブミラーに連結することによって、ミラー化されマウントされたファイルシステムを拡張する方法を示します。ミラーの名前は d8 であり、2つのサブミラー (d9 と d10) が収められています。

次の作業

UFS の場合、ミラーメタデバイス上で `growfs(1M)` コマンドを実行します。198 ページの「ファイルシステムの拡張方法 (コマンド行)」を参照してください。

raw メタデバイスを使用するデータベースなどのアプリケーションは、独自の方法で追加領域の拡張を行うことが必要です。

▼ RAID5 メタデバイスの拡張方法 (DiskSuite ツール)

この作業は、他のスライスを連結することによって既存の RAID5 メタデバイスを拡張するために使用します。一般に、これは領域を使い果たした RAID5 メタデバイスに対する短期的な解決策です。パフォーマンス上の理由により、純粋な RAID5 メタデバイスの使用をお勧めします。

1. **前提条件** (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、**予備情報** (183ページの「スライスとメタデバイスを拡張するための予備情報」) を読んだことを確認する。
2. オブジェクトリストから、既存の **RAID5** メタデバイスオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. スライスをクリックして「スライスブラウザ」ウィンドウを開き、**RAID5** メタデバイスに連結するスライスを選択する。
4. 選択したスライスを **RAID5** オブジェクトの先頭にドラッグする。
スライスは、少なくとも RAID5 メタデバイス内の最小のスライスと同じ大きさでなければなりません。同時に複数のスライスを選択するには、Control キーを押しながらクリックします。
DiskSuite ツールは、追加スライスをオブジェクトの末尾に表示します。
5. **RAID5** オブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
マウントされたファイルシステムの場合：GrowFS のダイアログボックスが表示されます。「今拡張する」をクリックし、`growfs(1M)` コマンドの実行を開始します。「ファイルシステムの拡張中」メッセージが表示される。「ファイルシステム拡張の中断」をクリックした場合、このコマンドは中止されます。そうで

ない場合、growfs(1M) コマンドが終了すると、「DiskSuite ツール」ウィンドウに戻ります。

raw デバイスを使用するアプリケーション：このようなアプリケーションは、DiskSuite に依存しない独自の方法で拡張された領域を認識する必要があります。

6. コンフィグレーションログを表示して、**RAID5** メタデバイスが確定されたことを確認する。

例 - 拡張された **RAID5** メタデバイス

この例では、スライス /dev/dsk/c0t1d0s3 が追加された RAID5 メタデバイス d3 を示します。



▼ RAID5 メタデバイスの拡張方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 183ページの「スライスとメタデバイスを拡張するための予備情報」の予備情報をチェックしてから、metattach(1M) コマンドを使用して、RAID5 メタデバイスにスライスを追加接続します。詳細は、metattach(1M) のマニュアルページを参照してください。

一般に、これは領域を使い果たした RAID5 メタデバイスに対する短期的な解決策です。パフォーマンス上の理由により、純粋な RAID5 メタデバイスの使用をお勧めします。

例 - RAID5 メタデバイスにスライスを追加

```
# metattach d2 c2t1d0s2
d2: column is attached
```

この例では、d2 という名前の既存の RAID5 メタデバイスに対する、スライス /dev/dsk/c2t1d0s2 の追加を示します。

次の作業

UFS の場合、RAID5 メタデバイス上で `growfs(1M)` コマンドを実行します。198 ページの「ファイルシステムの拡張方法 (コマンド行)」を参照してください。

`raw` メタデバイスを使用するデータベースなどのアプリケーションは、独自の方法で追加領域の拡張を行うことが必要です。

▼ トランスメタデバイスの拡張方法 (DiskSuite ツール)

マスターデバイスがメタデバイスである限り、トランスメタデバイス内でマスターデバイスを拡張できます。1つのスライスから成るマスターを拡張するには、トランスを分離 (削除) し、スライスをメタデバイスに置いてから、トランスを再作成します。

この作業は、トランスメタデバイスを構成するマスターデバイスを拡張するために使用します。

注 - マスターデバイスがミラーである場合、各サブミラーを拡張する必要があります。

1. **前提条件** (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、**予備情報** (183ページの「スライスとメタデバイスを拡張するための予備情報」) を読んだことを確認する。
2. オブジェクトリストから既存のトランスメタデバイスオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。

3. マスターデバイスを拡張する。
 - 連結またはストライプ: 187ページの「既存の連結方式の拡張方法 (DiskSuite ツール)」を参照
 - ミラー: 191ページの「ミラーの拡張方法 (DiskSuite ツール)」を参照
 - RAID5 メタデバイス: 193ページの「RAID5 メタデバイスの拡張方法 (DiskSuite ツール)」を参照
4. ログがミラー化されない場合、ダイアログボックスが表示されたら「了解」をクリックする。
5. トランスメタデバイスオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。

マウントされたファイルシステム: GrowFS のダイアログボックスが表示されます。「今拡張する」をクリックし、growfs(1M) コマンドの実行を開始します。「ファイルシステムの拡張中」メッセージが表示されます。「ファイルシステムの拡張を中断」をクリックした場合、このコマンドは中止されます。そうでない場合、growfs(1M) コマンドが終了すると、「DiskSuite ツール」ウィンドウに戻ります。

raw デバイスを使用するアプリケーション: このようなアプリケーションは、DiskSuite に依存しない独自の方法で拡張された領域を認識する必要があります。
6. コンフィグレーションログを表示して、トランスメタデバイスが確定されたことを確認する。

▼ トランスメタデバイスの拡張方法 (コマンド行)

マスターがメタデバイスである限り、トランスデバイス内でマスターデバイスを拡張できます。1つのスライスから成るマスターを拡張するには、トランスを分離 (削除) し、スライスをメタデバイスに置いてから、トランスを再作成します。

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 183ページの「スライスとメタデバイスを拡張するための予備情報」の予備情報をチェックしてから、metattach(1M) コマンドを使用して、トランスメタデバイス内でマスターデバイスにスライスを追加接続します。詳細は、metattach(1M) のマニュアルページを参照してください。

注 - マスターデバイスがミラーである場合、各サブミラーを拡張する必要があります。

例 - トランスメタデバイス内でミラー化されたマスターデバイスを拡張

```
# metastat d10
d10: Trans
    State: Okay
    Size: 102816 blocks
    Master Device: d0
    Logging Device: d1
d0: Mirror
    Submirror 0: d11
        State: Okay
    ...
    Submirror 1: d12
        State: Okay
    ...
# metattach d11 c0t2d0s5
d11: component is attached
# metattach d12 c0t3d0s5
d12: component is attached
```

この例では、2つのサブミラー d11 と d12 を含む 2面のミラー d0 から構成されるマスターデバイスをもつ、トランスデバイス d10 を拡張します。metattach(1M) コマンドは各サブミラー上で実行されます。システムに各スライスが接続されたことを確認します。

次の作業

UFS の場合、(マスターデバイスではなく) トランスメタデバイス上で growfs(1M) コマンドを実行します。198ページの「ファイルシステムの拡張方法 (コマンド行)」を参照してください。

raw メタデバイスを使用するデータベースなどのアプリケーションは、指定された方法で追加領域の拡張を行う必要があります。

ファイルシステムの拡張

この節では、追加スライスによって拡張されたメタデバイスを使用する、UFS の拡張方法について説明します。

ファイルシステムを拡張するための予備情報

- ファイルシステムを含むメタデバイスが拡張 (領域の追加) された後、そのメタデバイスに UFS が含まれる場合は、ファイルシステムを拡張して、追加された領域を認識させる必要があります。
- DiskSuite ツールを使用してメタデバイスに新たな領域を追加した場合は、自動的に `growfs` メッセージを受け取ります。コマンド行インタフェースを使用する場合、`growfs(1M)` コマンドによってファイルシステムを手動で拡張する必要があります。
- `raw` デバイスを使用するデータベースなどのアプリケーションは、独自の方法で追加領域の拡張を行うことが必要となります。DiskSuite はこの機能を提供していません。
- `growfs(1M)` コマンドは、ファイルシステムがたとえマウント中であっても、ファイルシステムを拡張します。

▼ ファイルシステムの拡張方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 198ページの「ファイルシステムを拡張するための予備情報」の予備情報をチェックしてから、`growfs(1M)` コマンドを使用して UFS を拡張します。詳細は、`growfs(1M)` のマニュアルページを参照してください。

例 - 連結上でファイルシステムを拡張

```
# df -k
Filesystem          kbytes   used   avail capacity  Mounted on
...
/dev/md/dsk/d10     69047    65426      0   100%    /home2
...
# growfs -M /home2 /dev/md/rdisk/d10
```

(続く)

```

/dev/md/rdisk/
d10:          295200 sectors in 240 cylinders of 15 tracks, 82 sectors
          144.1MB in 15 cyl groups (16 c/g, 9.61MB/g, 4608 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
  32, 19808, 39584, 59360, 79136, 98912, 118688, 138464, 158240, 178016,
  197792, 217568, 237344, 257120, 276896,
# df -k
Filesystem            kbytes    used  avail capacity  Mounted on
...
/dev/md/dsk/d10        138703    65426  59407    53%    /home2
...

```

マウントされたファイルシステム /home2 を含む連結 d10 に、新しいスライスが追加されました。growfs コマンドは、-M オプションでマウント先を /home2 と指定し、これは raw デバイス /dev/md/rdisk/d10 に拡張されています。growfs(1M) コマンドが終了すると、ファイルシステムはメタデバイス全体をつなぎます。作業の前後で df -k コマンドを使用して、ディスクの合計容量を確認します。

growfs(1M) コマンドは、マウントされたファイルシステムを拡張中に書き込みロックします (lockfs(1M) を参照)。ファイルシステムが書き込みロックされる時間を短縮するには、ファイルシステムを段階的に拡張していきます。たとえば、1G バイトのファイルシステムを 2G バイトに拡張するには、-s オプションを使用して各段階で新しいファイルシステムの合計サイズを指定しながら、ファイルシステムを 16M バイト単位で拡張していきます。

拡張中には、書き込みロックにより、ファイルシステムには書き込みアクセスを実行できません。書き込みアクセスは透過的に中断され、growfs(1M) がファイルシステムのロックを解除した段階で再開されます。読み取りアクセスは影響を受けませんが、ロックが有効な間はアクセス時間は記録されません。

注 - ミラーとトランスメタデバイスの場合、たとえサブミラーやマスターデバイスに領域が追加されても、(サブミラーやマスターデバイスではなく)必ずトップレベルのメタデバイス上で growfs(1M) コマンドを実行してください。

メタデバイスのリネーム

この節では、DiskSuite に備わるメタデバイスのリネーム機能について説明します。

メタデバイスをリネームするための予備情報

- 使用する名前が他のメタデバイスで使用されておらず、メタデバイス自身も使用されていない限り、DiskSuite を使用していつでもメタデバイスをリネームできます。
- メタデバイスをリネームする前に、メタデバイスが現在使用されていないことを確認してください。

ファイルシステムの場合は、マウントされていたり swap として使用されていたりしないことを確認します。raw デバイスを使用するデータベースなどのアプリケーションの場合は、指定された方法でデータへのアクセスを停止する必要があります。

- ログインデバイスとして使用されているメタデバイスを除き、どのメタデバイスでもリネームできます。

ログインデバイスとして使用されるメタデバイスのリネームと同じような結果を得るには、ログインデバイスを切断してから、希望する名前の新しいログインデバイスを再接続します。あるいはログインデバイスを切断し、リネームしてからトランスメタデバイスに再接続します。

- トランスメタデバイスにログインデバイスが接続されている場合、そのトランスメタデバイスはリネームできません。トランスメタデバイスをリネームするには、まずログインデバイスを切断します。
- ホットスペア集合の場合は、いったん作成したらリネームできません。
- ディスクセット内のメタデバイスはリネームできます。メタデバイスを、あるディスクセットから別のディスクセットにリネームすることはできません。

注 -x オプション付きの `metarename` コマンドは、親子関係のあるメタデバイスを切り替えることができます。358ページの「メタデバイス名の切り替え」を参照してください。

▼ メタデバイスのリネーム方法 (DiskSuite ツール)

マウントされていたり開かれているメタデバイスはリネームできません。ログインデバイスが接続されているトランスメタデバイスもリネームできません。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (200ページの「メタデバイスをリネームするための予備情報」) を読んだことを確認する。

2. メタデバイスに対するすべてのアクセスを停止する。
たとえば、メタデバイスにファイルシステムがマウントされている場合は、それをマウント解除します。
3. メタデバイスの「情報」ウィンドウを表示する。
4. 「デバイス名」フィールドに新しいメタデバイス名を入力して、「接続」をクリックする。
「情報」ウィンドウに「デバイス名」フィールドが見当たらない場合、そのメタデバイスはまだ使用中です。メタデバイスへのアクセスを停止したことを確認してください。
5. 「閉じる」をクリックして、「情報」ウィンドウを閉じる。
メタデバイスオブジェクトが新しいメタデバイス名を表示します。
メタデバイスが、`/etc/vfstab` ファイルに記述されているファイルシステムで使用されている場合、DiskSuite ツールは、新しいメタデバイス名を参照するように記述を変更します。
6. メタデバイスへのアクセスを再開する。
たとえば、ファイルシステムをマウントします。
7. コンフィグレーションログを表示して、リネームが確定されたことを確認する。

▼ メタデバイスのリネーム方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 200ページの「メタデバイスをリネームするための予備情報」の予備情報をチェックしてから、`metarename(1M)` コマンドを使用して、メタデバイスをリネームします。詳細は、`metarename(1M)` のマニュアルページを参照してください。

例 - ファイルシステムで使用されるメタデバイスのリネーム

```
# umount /home
# metarename d10 d100
```

(続く)

続き

```
d10: has been renamed to d100  
(ファイルシステムが新しいメタデバイスを参照するよう、/etc/vfstab ファイルを編集する)  
# mount /home
```

メタデバイス d10 は、メタデバイス d100 にリネームされます。d10 にはマウントされたファイルシステムが含まれるため、リネームする前に、ファイルシステムをマウント解除しなければなりません。メタデバイスが、/etc/vfstab ファイルに記述されているファイルシステムで使用される場合、その内容を変更して新しいメタデバイス名を参照しなければなりません。たとえば、次の行は、

```
/dev/md/dsk/d10 /dev/md/rdisk/d10 /docs ufs 2 yes -
```

次のように変更します。

```
/dev/md/dsk/d100 /dev/md/rdisk/d100 /docs ufs 2 yes -
```

最後に、ファイルシステムを再マウントします。

ミラーの操作

この節では、ファイルシステムのミラー化解除、サブミラーの接続と切断し、およびサブミラーのオフライン / オンライン設定など、ミラーに対して行う保守作業について説明します。

ミラーの予備情報

- ミラー化解除 - DiskSuite ツールでは、ルート (/)、/opt、/usr、または swap、あるいはシステムの実行中にマウント解除できないファイルシステムのミラー化解除をサポートしません。これらのファイルシステムに対してはコマンド行を使用してください。
- 接続 - サービスを中断することなく、サブミラーをミラーに接続できます。サブミラーをミラーに接続して、2面と3面のミラーを作成します。

- 切断とオフライン - サブミラーをオフラインにすると、そのミラーでサブミラーの読み書きを禁止できます。その際、ミラーに対するサブミラーの論理的な関連付けは維持できます。サブミラーがオフラインになっている間、DiskSuite がミラーに対するすべての書き込みを記録し、サブミラーがオンラインに復帰したときにサブミラーに書き込みます。最適化された再同期を実行することにより、DiskSuite は、サブミラー全体ではなく、変更されたデータだけを再同期することになります。サブミラーを切断することは、ミラーに対するその論理的な関連付けを切断することになります。一般に、保守を行う場合はサブミラーをオフラインにし、サブミラーを削除する場合にサブミラーを切断します。

▼ ファイルシステムのミラー化を解除する方法 (DiskSuite ツール)

この作業は、システムの実行中にマウント解除できるファイルシステムのミラー化を解除するために使用します。ルート (/)、/opt、/usr、swap、またはシステムの実行中にマウント解除できないその他のファイルシステムをミラー化解除するには、206ページの「マウント解除不可能なファイルシステムをミラー化解除する方法 (コマンド行)」を参照してください。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (202ページの「ミラーの予備情報」) を読んだことを確認する。
2. 少なくとも 1 つのサブミラーの状態が「正常」であることをチェックする。
「正常」状態のサブミラーをもたないミラーは、最初に修復しなければなりません。
3. ファイルシステムをマウント解除する。
4. オブジェクトリストからミラーオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
5. ミラーを削除する。ミラーのプルダウンメニューを表示して「削除」オプションを選択する。表示される「確認」ダイアログボックスで「削除」ボタンをクリックする。
ミラーは、その成分のサブミラーに分割されてキャンバスに表示されます。
6. /etc/vfstab ファイルを編集する。

ファイルシステムのデバイス名を変更する必要があります。データへのアクセスを維持するには、ミラーからいずれかのサブミラーに切り替えます (サブミラーは、ストライプか連結方式メタデバイスです)。データへのアクセスを解消するには、ファイルシステム用のエントリを完全に削除します。

7. [オプション] データへのアクセスを維持するには、ファイルシステムをマウントする。
ファイルシステムが再マウントされると、ストライプまたは連結に置かれます。
8. [オプション] 使用されていないサブミラーを削除するには、サブミラーのプルダウンメニューから「削除」オプションを選択する。
使用する予定のないサブミラーを選択してください。
9. [オプション] サブミラーが 1 つのスライスで構成される場合：ファイルシステムをスライスにマウントできる。ストライプまたは連結を削除する。メタデバイスではなく、物理スライスにファイルシステムをマウントするよう、
/etc/vfstab ファイルを編集する。
10. コンフィグレーションログを表示して、ミラーが削除されたことを確認する。
11. [オプション] 「ファイル」メニューから「構成の再走査」を選択し、ファイルシステムの新しいマウントデバイスを反映させる。

▼ ファイルシステムのミラー化を解除する方法 (コマンド行)

この作業は、システムの実行中にマウント解除できるファイルシステムをミラー化解除するために使用します。ルート (/)、/opt、/usr、swap、またはシステムの実行中にマウント解除できないその他のファイルシステムをミラー化解除するには、206ページの「マウント解除不可能なファイルシステムをミラー化解除する方法 (コマンド行)」のコマンド行作業を使用します。

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 202ページの「ミラーの予備情報」の予備情報をチェックしてから、metadetach(1M) と metaclear(1M) のコマンドを使用して、ファイルシステムをミラー化解除します。詳細については、metadetach(1M) と metaclear(1M) のマニュアルページを参照してください。

ミラーのミラー化解除を行う手順を次に示します。

- `metastat(1M)` コマンドを実行して、少なくとも 1 つのサブミラーが「Okay (正常)」状態であることを確認する
- ファイルシステムをマウント解除する
- ファイルシステム用に継続使用されるサブミラー上で、`metadetach(1M)` コマンドを実行する
- ミラー上で `metaclear(1M) -r` コマンドを実行する
- `/etc/vfstab` ファイルにファイルシステムが記述されている場合は、非ミラーデバイスを使用するように記述を変更する
- ファイルシステムを再マウントする

例 - /var ファイルシステムのミラー化解除

```
# metastat d4
d4: Mirror
   Submirror 0: d2
       State: Okay
   Submirror 1: d3
       State: Okay
...
# umount /var
# metadetach d4 d2
d4: submirror d2 is detached
# metaclear -r d4
d4: Mirror is cleared
d3: Concat/Stripe is cleared
(/etc/vfstab ファイルを編集して、/var を d4 から d2 に変更する)
# mount /var
```

`/var` は、`d4` という名前の 2 面のミラーで構成されます。そのサブミラーは `d2` と `d3` であり、それぞれスライス `/dev/dsk/c0t0d0s0` と `/dev/dsk/c1t0d0s0` から構成されます。`metastat(1M)` コマンドは、少なくとも 1 つのサブミラーが「Okay (正常)」状態であることを確認します (「Okay」状態のサブミラーをもたないミラーは、最初に修復しなければなりません)。ファイルシステムがマウント解除されてから、サブミラー `d2` が切断されます。`metaclear -r` コマンドは、ミラーと他のサブミラー `d3` を削除します。

次に、`/etc/vfstab` ファイル内の `/var` の記述を、サブミラーを参照するように変更します。たとえば、`d4` がミラーで `d2` がサブミラーであった場合、次の行は、

```
/dev/md/dsk/d4 /dev/md/rdisk/d4 /var ufs 2 yes -
```

次のように変更します。

```
/dev/md/dsk/d2 /dev/md/rdisk/d2 /var ufs 2 yes -
```

サブミラーの名前を使用することによって、ファイルシステムをメタデバイスにマウントしたままにできます。最後に、`/var` を再マウントします。

注 - `/etc/vfstab` ファイルで `d4` の代わりに `d2` を使用することによって、ミラーのミラー化解除を行いました。`d2` は 1 つのスライスから成るため、デバイスにメタデバイスをサポートしてほしくない場合、ファイルシステムをスライス名 (`/dev/dsk/c0t0d0s0`) にマウントできます。

▼ マウント解除不可能なファイルシステムをミラー化解除する方法 (コマンド行)

この作業は、ルート (`/`)、`/usr`、`/opt`、`swap` など、通常システム動作中にマウント解除できないファイルシステムをミラー化解除するために使用します。

この作業の手順を次に示します。

- `metastat(1M)` コマンドを実行して、少なくとも 1 つのサブミラーが「Okay (正常)」状態にあることを確認する
- ルート (`/`)、`/usr`、`/opt`、または `swap` を含むミラー上で `metadetach(1M)` コマンドを実行して、1 面のミラーを作成する
- `/usr`、`/opt`、`swap` の場合：`/etc/vfstab` ファイル内のファイルシステムの記述を変更して、非 DiskSuite デバイス (スライス) を使用するよう設定する
- ルート (`/`) の場合：`metaroot(1M)` コマンドを実行する
- システムをリブートする
- `metaclear(1M)` コマンドを実行して、ミラーとサブミラーを削除する

例 - ルート (`/`) のミラー化解除

```
# metadetach d0 d20
d0: submirror d20 is detached
# metaroot /dev/dsk/c0t3d0s0
# reboot
```

(続く)

```

...
# metaclear -r d0
d0: Mirror is cleared
d10: Concat/Stripe is cleared
# metaclear d20
d20: Concat/Stripe is cleared

```

この例では、ルート (/) は d0 という名前の 2 面のミラーです。そのサブミラーは d10 と d20 であり、それぞれスライス /dev/dsk/c0t3d0s0 と /dev/dsk/c1t3d0s0 から構成されています。metastat コマンドは、少なくとも 1 つのサブミラーが「Okay (正常)」状態であることを確認します (「Okay」状態のサブミラーをもたないミラーは、最初に修復する必要があります)。サブミラー d20 が切断され、d0 は 1 面ミラーとなります。そして、システムがブートされるルートスライスを使用して、metaroot コマンドが実行されます。これによって、/etc/system と /etc/vfstab のファイルが編集され、ルート (/) のミラー化を指定する情報が削除されます。リブートの後、metaclear -r コマンドは、ミラーと他のサブミラー d10 を削除します。最後の metaclear コマンドは、サブミラー d20 をクリアします。

例 - swap のミラー化解除

```

# metastat d1
d1: Mirror
  Submirror 0: d11
    State: Okay
  Submirror 1: d21
    State: Okay
...
# metadetach d1 d21
d1: submirror d21 is detached
(/etc/vfstab ファイルを編集し、swap の記述をメタデバイスからスライス名に変更する。)
# reboot
...
# metaclear -r d1
d1: Mirror is cleared
d11: Concat/Stripe is cleared
# metaclear d21
d21: Concat/stripe is cleared

```

この例では、swap は d1 という名前の 2 面のミラーから構成されています。そのサブミラーは d11 と d21 であり、それぞれスライス /dev/dsk/c0t3d0s1 と

/dev/dsk/c1t3d0s1 から構成されています。metastat コマンドは、少なくとも 1 つのサブミラーが「Okay (正常)」状態であることを確認します (「Okay」状態のサブミラーをもたないミラーは、最初に修復する必要があります)。サブミラー d21 が切断され、d1 は 1 面のミラーとなります。次に、swap 用の記述がサブミラー d21 に含まれるスライス参照するよう、/etc/vfstab ファイルを編集する必要があります。たとえば、d1 がミラーであり、d21 がスライス /dev/dsk/c0t3d0s1 を含むサブミラーであった場合、次の行は、

```
/dev/md/dsk/d1 - - swap - no -
```

次のように変更します。

```
/dev/dsk/c0t3d0s1 - - swap - no -
```

リポートした後、metaclear -r コマンドでミラーと他のサブミラー d11 を削除します。最後に metaclear コマンドで、サブミラー d21 を削除します。

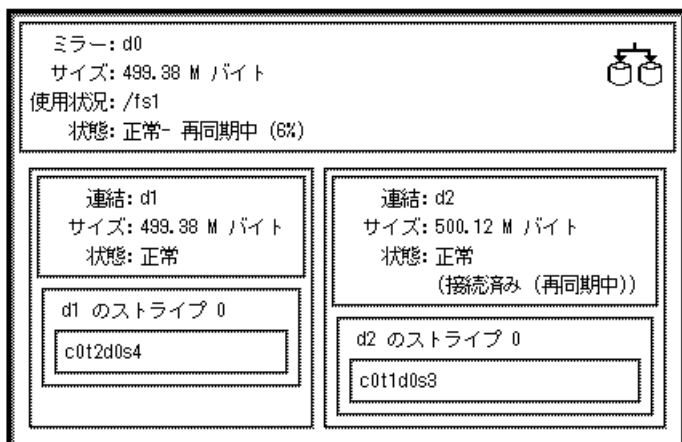
▼ サブミラーの接続方法 (DiskSuite ツール)

作業を始める前に、サブミラーとして使用する連結またはストライプを識別します。そのサイズは、ミラー内の既存のサブミラーのサイズ以上でなければなりません。まだどちらも作成していない場合は、61ページの「ストライプと連結の作成」を参照してください。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (202ページの「ミラーの予備情報」) を読んだことを確認する。
2. オブジェクトリストから既存のミラーオブジェクトをダブルクリックし、その状態が「正常」であることを確認する。
オブジェクトがキャンバスに表示されます。
3. ミラーに接続される連結方式オブジェクトを、ミラーオブジェクトの先頭にドラッグする。
4. ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
新しいサブミラーの再同期が開始されます。
5. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。

例 - サブミラーが接続されたミラーオブジェクト

この例では、サブミラー d2 が接続されたミラー d0 を示します。このミラーは、新しいサブミラー上のデータと自動的に同期します。



▼ サブミラーの接続方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 202ページの「ミラーの予備情報」の予備情報をチェックしてから、`metattach(1M)` コマンドを使用してミラーにサブミラーを接続します。詳細は、`metattach(1M)` のマニュアルページを参照してください。

作業を開始する前に、サブミラーとして使用する連結またはストライプを識別します。そのサイズは、ミラー内の既存のサブミラーのサイズ以上でなければなりません。まだどちらも作成していない場合は、61ページの「ストライプと連結の作成」を参照してください。

例 - サブミラーの接続

```
# metastat d30
d30: mirror
    Submirror 0: d60
    State: Okay
...
# metattach d30 d70
d30: submirror d70 is attached
# metastat d30
d30: mirror
```

(続く)

```
Submirror 0: d60
  State: Okay
Submirror 1: d70
  State: Resyncing
Resync in progress: 41 % done
Pass: 1
Read option: roundrobin (default)
Write option: parallel (default)
Size: 2006130 blocks
...
```

この例では、1面のミラー d30 にサブミラー d70 を接続して、2面のミラーを作成する方法を示します。最初、ミラー d30 はサブミラー d60 から構成されます。d70 は連結方式メタデバイスです。metastat(1M) コマンドによって、ミラーの状態が「Okay (正常)」であることを確認してから、サブミラーを接続します。metattach(1M) コマンドが実行されると、新しいサブミラーが既存のミラーと再同期されます。ミラーにサブミラーを追加接続すると、システムはメッセージを表示します。ミラーが再同期中であることを確認するには、metastat(1M) コマンドを使用します。

▼ サブミラーを切断する方法 (DiskSuite ツール)

配下のディスクを再使用したい場合、サブミラーを切断することもできます。システムからのサービスを中断することなく、サブミラーを切断することができます。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (202ページの「ミラーの予備情報」) を読んだことを確認する。
2. オブジェクトリストから、ミラーオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. 切断するサブミラーの内部をクリックする。
4. サブミラーを、ミラーオブジェクトからキャンバスにドラッグする。
これが2面ミラーである場合は、ミラーの状態が「緊急」に変化します。
5. ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
6. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。

▼ サブミラーを切断する方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 202ページの「ミラーの予備情報」の予備情報をチェックしてから、`metadetach(1M)` コマンドを使用して、サブミラーをミラーから切断します。詳細は、`metadetach(1M)` のマニュアルページを参照してください。

例 - サブミラーの切断

```
# metastat
d5: mirror
    Submirror 0: d50
...
# metadetach d5 d50
d5: submirror d50 is detached
```

この例では、ミラー `d5` にはサブミラー `d50` があり、これは `metadetach(1M)` コマンドで切断されます。`d50` の配下にあるスライスは、どこか他の場所で再使用されます。サブミラーをミラーから切断すると、システムは確認メッセージを表示します。

▼ サブミラーのオフライン / オンライン設定 (DiskSuite ツール)

物理ディスクを修復する場合、サブミラーをオフライン / オンライン設定すると便利です。たとえば、SCSI チェーン内のディスクに障害が発生した場合、そのチェーン内のその他すべてのメタデバイスは、壊れたディスクの交換中にオフラインにすることができます。交換用のディスクを取り付けてから、メタデバイスをオンラインに戻します。

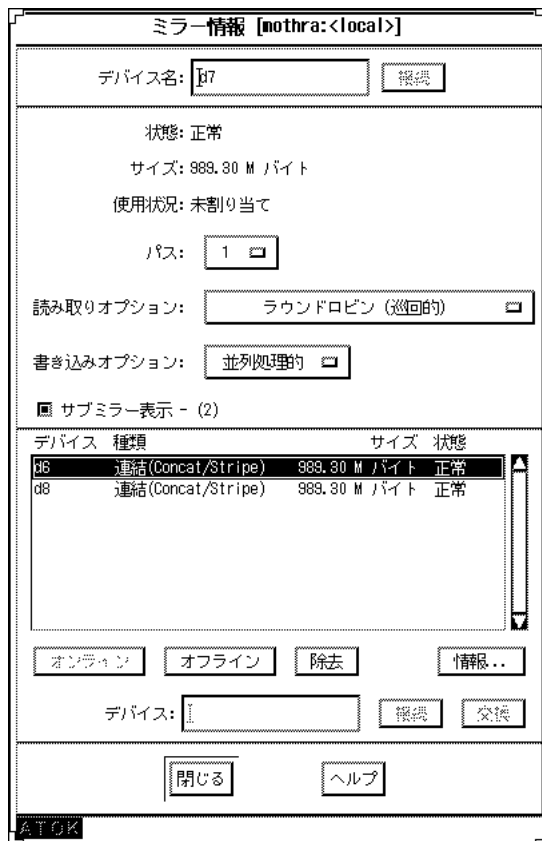
サブミラーをオフラインにすると、DiskSuite は、そのミラーに対するすべての入出力を記録します。サブミラーをオンラインに戻したとき、DiskSuite はデータの最適化された再同期を実行し、サブミラー全体ではなく、変更部分だけを再同期します。

注 - オフラインにされていたサブミラーは、読み取り専用でだけマウントできます。

サブミラーのオフライン / オンライン設定手順は、本質的に同じものです。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (202ページの「ミラーの予備情報」) を読んだことを確認する。

- オブジェクトリストから、既存のミラーオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
- ミラーオブジェクトのポップアップウィンドウを表示し、「情報」を選択する。
「ミラー情報」ウィンドウが表示されます。



- デバイスリストからサブミラーを選択する。
- サブミラーがオフラインであれば、「オンライン」をクリックする。サブミラーがオンラインであれば、「オフライン」をクリックする。

サブミラーの状態は、「オフライン (予定設定済み)」または「オンライン (予定設定済み)」に変化します。

6. 「閉じる」をクリックする。
7. ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
ミラーをオフラインにすると、ミラーの状態は「オフライン」に変化します。ミラーをオンラインにすると、DiskSuite は再同期の操作を開始します。

▼ サブミラーのオフライン / オンライン設定方法 (コマンド行)

131ページの「DiskSuite オブジェクトを保守するための前提条件」の前提条件と 202ページの「ミラーの予備情報」の予備情報をチェックしてから、`metaoffline(1M)` コマンドを使用してサブミラーをオフラインにするか、`metaonline(1M)` コマンドを使用してサブミラーをオンラインにします。詳細については、`metaoffline(1M)` または `metaonline(1M)` のマニュアルページを参照してください。

例 - サブミラーのオフライン設定

```
# metaoffline d10 d11
d10: submirror d11 is offlined
```

ここでは、ミラー d10 からサブミラー d11 をオフラインにします。他のサブミラーからは読み取りを続けることができます。最初の書き込みが行われると、ミラーはすぐに同期を外れます。オフライン設定されたサブミラーがオンラインに復帰すると、この不整合は訂正されます。

例 - サブミラーのオンライン設定

```
# metaonline d10 d11
d10: submirror d11 is onlined
```

用意ができると (たとえば、ディスクを交換した後)、サブミラー d11 はオンラインに復帰します。

metaonline(1M) コマンドを使用できるのは、サブミラーが metaoffline(1M) コマンドによってオフライン設定された場合に限られます。metaonline(1M) コマンドが実行された後、DiskSuite はサブミラーとミラーの再同期を自動的に開始します。

注 - metaoffline(1M) コマンドの機能は、metadetach(1M) によって提供される機能と類似しています。しかし、metaoffline(1M) では、サブミラーとミラー間の論理的な関連付けを切断しません。

ディスクセットの操作

この節では、ディスクセットの予約と解放、ディスクセットに対するホストとディスクの追加など、ディスクセットの保守作業について説明します。

ディスクセットを操作するための予備情報

- ディスクセットの保守を実行するには、ホストがディスクセットの所有者であるか、ディスクセットを予約している必要があります (ホストは、セットに最初のドライブを置くことによって、ディスクセットの暗黙的な所有権を取得します)。
- ディスクセットを操作するには、ルートがグループ 14 のメンバーであるか、./rhosts ファイルに他 (各ホスト) のホスト名のエントリが含まれていることが必要です。

▼ ディスクセットの予約方法 (コマンド行)

ディスクセットは、「安全」または「強制的」に予約できます。ディスクセット内のあるホストがディスクセットを予約すると、ディスクセット内の他のホストは、ディスクセット内のドライブのデータにアクセスできません。

- 安全に - そのディスクセットを予約しているホストが他に存在しない場合にだけ、ディスクセットをユーザーのホスト用に予約します。
- 強制的に - 他のホストがセットを現在予約しているかどうかとは無関係に、ディスクセットを予約します。ディスクセット内のホストがダウンしているか通信し

ていない場合は、この方法を使用します。この時点で他のホストがディスクセットを予約していた場合、予約が失われることによってそのホストはパニック状態となります。

注 - ディスクセット内のホストが通信していることによりかなりの確信がもてる場合、通常は安全な予約を実行するようお勧めします。

前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (214ページの「ディスクセットを操作するための予備情報」) を読みだしたことを確認します。metaset(1M) を使用して、ディスクセットを安全または強制的に予約します。詳細は、metaset(1M) のマニュアルページを参照してください。

注 - 他のホストがディスクセットの所有権をもつ場合、SCSI 予約の競合によって、そのホストはパニック状態になります。

例 - ディスクセットを安全に予約

```
red# metaset
...
Set name = relo-red, Set number = 2

Host                Owner
  red
  blue
...
red# metaset -s relo-red -t
red# metaset
...
Set name = relo-red, Set number = 2

Host                Owner
  red                Yes
  blue
...
```

この例では、ホスト red がホスト blue と通信を行い、ホスト red がセットの予約を試みる前に、ホスト blue がディスクセットの予約を解放したことを確認します。

注 - この例では、ホスト blue がセット relo-red を所有していた場合、上記の出力の「Owner」カラムはまだブランクのままです。metaset(1M) コマンドは、(他のホストではなく) 発行側のホストがディスクセットを所有するかどうかだけを示します。

例 - ディスクセットを強制的に予約

```
# metaset -s relo-red -t -f
```

この例では、ホスト red はホスト blue と通信しません。その代わりに、ディスクセット内のドライブは警告なしに予約されます。ホスト blue がディスクセットを予約している場合、予約が失われるためパニック状態となります。

▼ ディスクセットを解放する方法 (コマンド行)

セット内のドライブの保守を行うとき、ディスクセットを解放すると便利です。ディスクセットが解放されると、ホストからはアクセスできません。ディスクセット内の両方のホストがセットを解放した場合、いずれのホストも、セット内で定義されたメタデバイスやホットスペア集合にアクセスできません。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (214ページの「ディスクセットを操作するための予備情報」) を読んだことを確認する。
2. **metaset(1M)** コマンドを使用して、ディスクセットを解放する。

```
# metaset -s <ディスクセット> -r
```

このコマンドでは、

<code>-s <ディスクセット></code>	metaset が作用するディスクセットの名前を指定します。
<code>-r</code>	ディスクセットの所有権を解放します。セット内でのすべてのディスクの予約は解除されます。セット内部で設定されたメタデバイスは、もうアクセスできません。

3. オプションなしで **metaset(1M)** コマンドを使用することによって、ディスクセットがこのホスト上で解放されたことを確認する。

```
# metaset
```


例 - ディスクセットを解放

```
red# metaset -s relo-red -r
red# metaset -s relo-red

Set name = relo-red, Set number = 1

Host                Owner
red
blue

Drive               Dbase
c1t0d1              Yes
c1t2d0              No
c1t3d0              No
c1t4d1              No
c2t2d0              Yes
c3t0d1              Yes
c3t2d0              No
c3t3d0              No
c3t4d1              No
```

この例では、ディスクセット `relo-red` を解放します。ディスクセットの所有者が存在しないことに注意してください。状態をホスト `red` から見ると、誤解を招くことがあります。ホストが調査できるのは、そのホストがディスクセットを所有するかかどうかだけです。たとえば、ホスト `blue` がディスクセットを予約した場合、ホスト `red` からはそのことがわかりません。この場合、予約を確認できるのはホスト `blue` だけです。

▼ ディスクセットに新たなドライブを追加する方法 (コマンド行)

ディスクセットを定義したら、ディスクセットにドライブを追加できます。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (214ページの「ディスクセットを操作するための予備情報」) を読んだことを確認する。
2. `metaset(1M)` コマンドを使用して、既存のディスクセットにドライブを追加する。

```
# metaset -s <ディスクセット> -a <ドライブ名> ...
```

このコマンドでは、

-s <ディスクセット> metaset が作用するディスクセットの名前を指定します。

-a 指定のディスクセットにドライブを追加します。

<ドライブ名> ... ディスクセットに追加するドライブを指定します。ドライブ名の形式は cxtxdx であり、名前の最後に「sx」スライス識別子を置くことはできません。このドライブ名は、ディスクセット内のすべてのホストに関して、同じメジャー名とマイナー名をもつ必要があります。

ドライブがディスクセットに追加されると、DiskSuite は、残りのドライブ全体を通じてメタデバイスの状態データベースの複製のバランスを再調整します。詳細は、118ページの「ディスクセットの作成」を参照してください。



注意 - データが収められているドライブを追加した場合は、データが失われます。

3. オプションなしの metaset(1M) コマンドを使用して、ホストがディスクセットに追加されたことを確認する。

```
# metaset
```

例 - ディスクセットに新たなドライブを追加

```
red# metaset -s relo-red -a c2t5d0
red# metaset
Set name = relo-red, Set number = 1

Host                Owner
  red                Yes
  blue

Drive               Dbase
  c1t2d0             Yes
  c1t3d0             Yes
  c2t2d0             Yes
  c2t3d0             Yes
  c2t4d0             Yes
```

(続く)

c2t5d0	No
--------	----

この例では、ドライブ c2t5d0 をディスクセット relo-red に追加します。

注 - DiskSuite ツールの実行中にディスクセットに対してドライブの追加または削除を行なった場合、設定が変更されたことを示すダイアログボックスが表示されます。「ファイル」メニューから「構成の再走査」を選択して設定を再ロードするか、または DiskSuite ツールを終了してから再起動します。

▼ ディスクセットに別のホストを追加する方法 (コマンド行)

DiskSuite は、ディスクセットあたり最大 2 つのホストをサポートします。1 つのホストしかもたない既存のディスクセットには、別のホストを追加できます。

1. 前提条件 (131ページの「DiskSuite オブジェクトを保守するための前提条件」) を満たし、予備情報 (214ページの「ディスクセットを操作するための予備情報」) を読んだことを確認する。
2. ホストを追加する。

```
# metaset -s <ディスクセット> -a -h <ホスト> ...
```

このコマンドでは、

-s <ディスクセット> metaset が作用するディスクセットの名前を指定します。

-a 指定のディスクセットにホストを追加します。

-h <ホスト>... ディスクセットに追加される 1 つ以上のホスト名を指定します。最初のホストを追加すると、セットが作成されます。ホスト名は、/etc/nodename にあるものと同じ名前です。

3. オプションなしで `metaset(1M)` コマンドを使用して、ホストがディスクセットに追加されたことを確認する。

```
# metaset
```

例 - ディスクセットに他のホストを追加

```
red# metaset -s relo-red -a -h blue
red# metaset -s relo-red
Set name = relo-red, Set number = 1

Host                Owner
  red                Yes
  blue

Drive               Dbase
  c1t0d1             Yes
  c1t2d0             No
  c1t3d0             No
  c1t4d1             No
  c2t2d0             Yes
  c3t0d1             Yes
  c3t2d0             No
  c3t3d0             No
  c3t4d1             No
```

この例では、ディスクセット `relo-red` にホスト `blue` を追加します。

DiskSuite オブジェクトの変更

この章では、DiskSuite ツールのグラフィカルユーザーインターフェースとコマンド行ユーティリティの両方による、DiskSuite デバイスの変更方法について説明します。

DiskSuite ツールを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 223ページの「DiskSuite 構成をディスクに保存する方法 (DiskSuite ツール)」
- 224ページの「ディスクから DiskSuite 構成を復元する方法 (DiskSuite ツール)」
- 226ページの「状態データベースの複製の変更方法 (DiskSuite ツール)」
- 229ページの「ミラーのオプションの変更方法 (DiskSuite ツール)」
- 233ページの「ロギングデバイスをファイルシステム間で共有する方法 (DiskSuite ツール)」

コマンド行インターフェースを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 231ページの「ミラーのオプションの変更方法 (コマンド行)」
- 235ページの「ロギングデバイスをファイルシステム間で共有する方法 (コマンド行)」

DiskSuite オブジェクトの変更の概要

DiskSuite オブジェクトを作成した後、それを変更したり、再構成したり、そのパラメータを変更したりする必要もできます。この節では、次のような作業について説明します。

- 未確定の変更を DiskSuite 構成に保存および復元する
- 状態データベースの複製を交換および復元する
- ミラーのオプションを変更

DiskSuite の概要については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

DiskSuite オブジェクトを変更するための前提条件

この章で説明する手順の前提条件を次に示します。

- 現在の時点で、すべてのデータをバックアップしている。
- ルート権限があることを確認する。
- グラフィカルユーザーインターフェースを使用する場合は、DiskSuite ツールを起動する。

ローカルなメタデバイス (ディスクセット構成に含まれないメタデバイス) を操作するには、次のように入力します。

```
# metatool &
```

ディスクセット内のメタデバイスを操作するには、自分がディスクセットの所有者であることを確認し、次のように入力します。

```
# metatool -s <ディスクセット名> &
```

DiskSuite 構成の操作

この節では、作業している現在の DiskSuite 構成を保存および復元する方法について説明します。

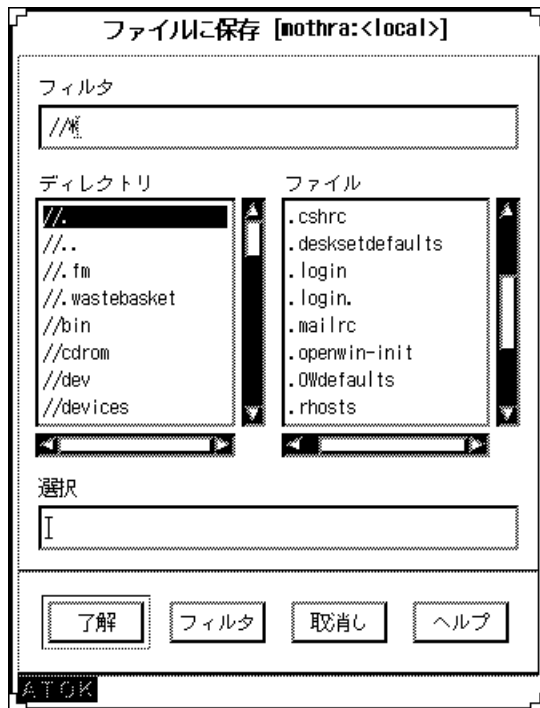
DiskSuite 構成のための予備情報

- 構成を保存すると、メタデバイスエディタのキャンバス上で未確定状態である変更が保存されます。これでは、現在メモリー上にある構成は保存されません。これは状態データベースの複製によって保守します。
- 保存と復元の構成オプションにより、実際にテンプレート構成を作成できます。これは機械的に構成する際に便利です。1つのマシン上に構成を作成し、別のマシンにコピーすることによって、2番目の構成を作成するための時間を節約できます。
- 作業を中断する必要がある、しかもこれまでに構成した内容 (まだ未確定) を失いたくない場合にも、保存と復元の構成オプションが便利です。

▼ DiskSuite 構成をディスクに保存する方法 (DiskSuite ツール)

この作業は、メタデバイスエディタのキャンバスに表示される未確定の変更を保存するために使用します。なお、この方法では未確定の変更が保存されるだけであり、構成の「バックアップ」は保存されません。

1. **前提条件** (222ページの「DiskSuite オブジェクトを変更するための前提条件」) を満たし、**予備情報** (223ページの「DiskSuite 構成のための予備情報」) を読んだことを確認する。
2. 「ファイル」メニューから「ファイルに保存」を選択する。
次のダイアログボックスが表示されます。



3. 「選択」テキストフィールドに保存するファイル名を入力し、「了解」をクリックする。

「ファイルに保存」ウィンドウのファイルリストから、ファイル名を選択することもできます。ファイルがすでに存在する場合、それを上書きしたり、あるいはキャンセルして別のファイル名を使用することができます。

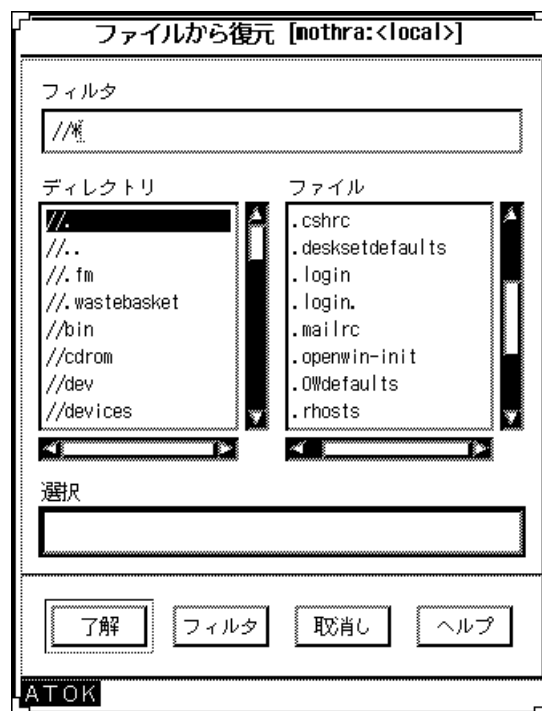
4. **DiskSuite** 構成を保存すると、構成が保存されたことを示すダイアログボックスが表示されます。ダイアログボックスで「了解」をクリックする。

▼ ディスクから DiskSuite 構成を復元する方法 (DiskSuite ツール)

この作業は、前もってディスクに保存されていた構成を、メタデバイスエディタのキャンバスに戻すために使用します。この場合、バックアップからの復元という意

味では、構成を復元しません。その代わりに、キャンバスに前もって保存された未確定変更の構成を開きます。

1. 前提条件 (222ページの「DiskSuite オブジェクトを変更するための前提条件」) を満たし、予備情報 (223ページの「DiskSuite 構成のための予備情報」) を読んだことを確認する。
2. 「ファイル」メニューから「ファイルから復元」を選択する。
次のダイアログボックスが表示されます。



3. 「ファイルから復元」ウィンドウのファイルリストからファイルを選択する。
「選択」テキストフィールドにファイル名を入力することもできます。

4. 構成を復元する場合、「ファイルから復元」ダイアログボックスが表示され、構成がキャンバスに復元されたことを知らせます。ダイアログボックスで「了解」をクリックする。



注意 - ファイルから復元すると、現在の構成に対して行なった未確定の変更が取り消されます。

状態データベースの複製の変更

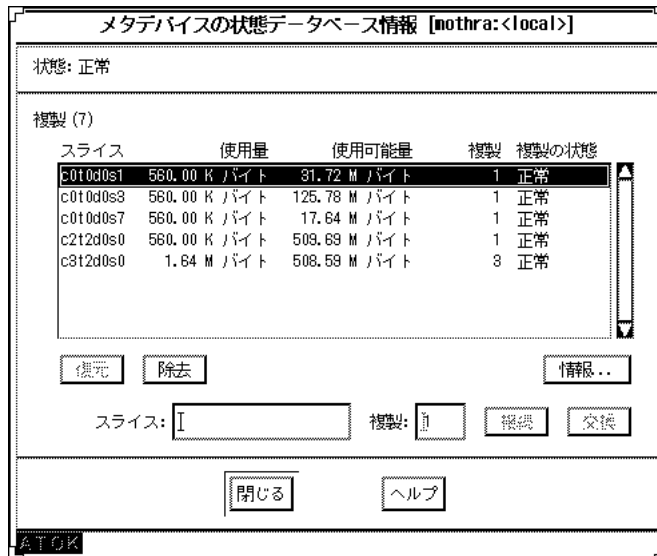
この節では、「メタデバイスの状態データベース情報」ウィンドウを使用して、状態データベースの複製を表示および変更する方法について説明します。

状態データベースの複製を変更するための予備情報

- システムには、状態データベースの複製が少なくとも3つ必要です。これらの複製を別のコントローラ上に置くことによって、システムのアキレス腱となる部分を解消してください。
- 状態データベースの複製の最大数は50です。この数は、ディスクセットの一部となる複製にも適用されます。

▼ 状態データベースの複製の変更方法 (DiskSuite ツール)

1. 前提条件 (222ページの「DiskSuite オブジェクトを変更するための前提条件」) を満たし、予備情報 (226ページの「状態データベースの複製を変更するための予備情報」) を読んだことを確認する。
2. 「メタデバイスの状態データベース情報」ウィンドウを表示する。
MetaDB オブジェクトをキャンバスにドラッグしてから、ダブルクリックします。「メタデバイスの状態データベース情報」ウィンドウが表示されます。



3. 必要な情報をクリックまたは入力することによって、状態データベースの複製に希望の変更を行う。終了したら、「閉じる」をクリックする。
4. [変更を行なった場合] 変更内容を有効にするために、**MetaDB** オブジェクトを確定する。
5. コンフィグレーションログを表示して、**MetaDB** オブジェクトが確定されたことを確認する。

表 4-1 「メタデバイスの状態データベース情報」ウィンドウの領域に関連付けられた機能を、表 4-1 に示します。

表 4-1 「メタデバイスの状態データベース情報」ウィンドウの機能

フィールド	機能
状態	メタデバイスの状態データベースの状態を説明します。これらの値については、134ページの「状態データベースの複製の状態のチェック方法 (DiskSuite ツール)」を参照してください。
複製の操作領域	この領域では次の情報を表示し、操作を行うことができます。 <ul style="list-style-type: none"> ■ 複製 - 複製の数。 ■ スクロールリスト - 複製を含むスライスのスクロールリスト。ここには、スライスの名前、使用される領域のサイズ、使用可能な領域、スライス上の複製の数、複製の状態があります。 ■ 除去 - 選択したスライスを削除します。 ■ 復元 - 選択したスライスにエラーがある場合、これを復元します。 ■ 情報 - 選択したスライスの「スライス情報」ウィンドウを表示します。 ■ スライス - MetaDB に接続する新しいスライス、または選択したスライスと交換する新しいスライスを指定します。 ■ 複製 - スライス上で作成される複製の数を示します。デフォルトは 1。 ■ 接続 - 「スライス」フィールドに入力したスライスを「複製」リストに追加します。このボタンは、スライス名が入力されたときにだけ有効です。

ミラーオプションの変更

この節では、「ミラー情報」ウィンドウを使用して、ミラーオプションの表示と変更を行う方法について説明します。

ミラーオプションを変更するための予備情報

- ミラーのパス番号や読み書きのオプションを変更することができます。
- ミラーオプションは、ミラーの実行中でも変更が可能です。

注・サブミラーとして使用されるストライプの飛び越し値を変更する必要がある場合は、367ページの「ミラー内のストライプの飛び越し値を変更する方法 (DiskSuite ツール)」を参照してください。

▼ ミラーのオプションの変更方法 (DiskSuite ツール)

設定可能なミラーオプションを、表 4-2 に示します。

表 4-2 ミラーオプション

オプション	機能
パス番号	ミラーには、0～9の範囲のパス番号を割り当てられます。パス(再同期)番号は、システムのレポート中にそのミラーが再同期される順序を決定します。デフォルトは1。小さなパス番号が最初に再同期されます。パス番号が0の場合、再同期はスキップされます。0を使用するのは、読み取り専用としてマウントされたミラーの場合です。複数のミラーが同じパス番号をもつ場合、それらのミラーは同時に再同期されます。
読み取り	<p>ミラーの3つの読み取りオプションは、「ラウンドロビン(巡回的)」、「ジオメトリック(幾何学的な配置順)」、「先頭のディスクから」である。デフォルトは「ラウンドロビン(巡回的)」であり、バランスロードとも呼ばれます。</p> <p>ラウンドロビン(巡回的) - すべての読み取りは、ミラーのすべてのサブミラーからラウンドロビン方式で行われます。つまり、最初の読み取りは最初のサブミラーから行われ、次の読み取りは2番目のサブミラーから行われます。</p> <p>ジオメトリック(幾何学的な配置順) - 順次的読み取りを行う場合やトラックバッファリングを行うディスクを使用する場合に、パフォーマンスが向上します。ジオメトリック方式の読み取りを使用すれば、論理ディスクのブロックアドレスにもとづいて、読み取り操作を複数のサブミラーに分散させることができます。たとえば、3面のミラーの場合、ミラー上のディスク領域は3つの(同じサイズの)論理アドレス範囲に分割されます。この3つの領域からの読み取りは、それぞれ別のサブミラーによって実行されます(たとえば、最初の領域の読み取りは最初のサブミラーによって実行されます)。</p> <p>先頭のディスクから - 最初のサブミラーだけからの読み取りを指定します。このオプションは、2番目のサブミラーの入出力特性が劣っている場合にだけ指定します。</p>
書き込み	サブミラーは、並列書き込みまたは逐次書き込みに設定できます。デフォルト設定は並列書き込みであり、書き込みはすべてのサブミラーに対して同時に行われます。逐次オプションでは、1つのサブミラーへの書き込みが終了してから、次のサブミラーへの書き込みを開始するよう指定します。逐次オプションは、たとえば電源障害によって、サブミラーの読み取りが不可能になった場合に備えて提供されます。

「ミラー情報ウィンドウ」を使用すれば、読み書きオプション、レポート中にミラーが再同期される順序、サブミラーにホットスペアがあるかどうかなど、ミラーに関するオプションを設定できます。

1. 前提条件 (222ページの「DiskSuite オブジェクトを変更するための前提条件」) を満たし、予備情報 (228ページの「ミラーオプションを変更するための予備情報」) を読んだことを確認する。
2. オブジェクトリストから既存のミラーオブジェクトをダブルクリックする。
ミラーオブジェクトがキャンバスに表示されます。
3. ミラーオブジェクトのポップアップメニューを表示し、「情報」を選択する。
「ミラー情報」ウィンドウが表示されます。
4. 希望のオプションを変更する。
ミラーオプションについては、表 4-2 を参照してください。ミラーオプションを変更したら、「閉じる」をクリックします。
5. ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
6. コンフィグレーションログを表示して、変更内容が確定されたことを確認する。

▼ ミラーのオプションの変更方法 (コマンド行)

222ページの「DiskSuite オブジェクトを変更するための前提条件」に記載の前提条件と 228ページの「ミラーオプションを変更するための予備情報」の予備情報をチェックしてから、`metaparam(1M)` コマンドを使用して、ミラーオプションの表示と変更を行います。ミラーオプションについては、表 4-2 を参照してください。詳細については、`metaparam(1M)` のマニュアルページを参照してください。

ミラーの読み取りオプションの変更

```
# metaparam -r geometric d30
# metaparam d30
d30: mirror current parameters are:
    Pass: 1
    Read option: geometric (-g)
    Write option: parallel (default)
```

`-r` オプションは、ミラーの読み取りオプションを `geometric` (ジオメトリック) に変更します。

例 - ミラーの書き込みオプションの変更

```
# metaparam -w serial d40
# metaparam d40
d40: mirror current parameters are:
    Pass: 1
    Read option: roundrobin (default)
    Write option: serial (-S)
```

-w オプションは、ミラーの書き込みオプションを serial (逐次的) に変更します。

例 - ミラーのパス番号の変更

```
# metaparam -p 5 d50
# metaparam d50
d50: mirror current parameters are:
    Pass: 5
    Read option: roundrobin (default)
    Write option: parallel (default)
```

-p オプションは、ミラーのパス番号を 5 に変更します。

ロギングデバイスをファイルシステム間で共有

この節では、トランスメタデバイスを設定して、ロギングデバイスを共有する方法について説明します。

ロギングデバイスを共有するための予備情報

- トランスメタデバイスを設定したら、ロギングデバイスをファイルシステム間で共有できます。
- ロギングデバイスを共有する場合、ロギングデバイスをミラー化することをお勧めします。
- システム内に使用できるスライスが多くない場合、またはロギングデバイスを共有するファイルシステムがほとんど読み取りアクセスされる場合、ロギングデバイスをファイルシステム間で共有することを検討してください。



注意 - 共有されたロギングデバイスの1つのマスターデバイスがエラー状態になった場合、ロギングデバイスはその変化をロールフォワードできません。そのため、そのロギングデバイスを共有するすべてのマスターデバイスが、ハードエラー状態になります。

▼ ロギングデバイスをファイルシステム間で共有する方法 (DiskSuite ツール)

この作業では、他のファイルシステム用のロギングデバイスにより、すでにトランスメタデバイスが設定されているものと想定します。

1. **前提条件** (222ページの「DiskSuite オブジェクトを変更するための前提条件」) を満たし、**予備情報** (232ページの「ロギングデバイスを共有するための予備情報」) を読んだことを確認する。
2. ファイルシステムを含むスライスまたはメタデバイスを識別する。
マスターデバイスを作成するときに、これが必要となります。
必要ならば、「スライス」をクリックして「スライスブラウザ」ウィンドウを表示して、ファイルシステムによって使用されるスライスを識別します。
3. トランスメタデバイスのテンプレートをクリックする。
キャンバスには、未割り当てで未確定のトランスメタデバイスオブジェクトが表示されます。メタデバイス名は自動的に割り当てられます。
4. **[オプション]** デフォルトのメタデバイス名を変更する。
 - a. オブジェクトのポップアップメニューを表示し、「情報」を選択する。
 - b. 「デバイス名」フィールドに新しいメタデバイス名を入力し、「接続」をクリックする。
 - c. 「閉じる」をクリックする。
5. オブジェクトリストから、既存のトランスメタデバイスオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。新しいトランスメタデバイスは、このオブジェクトとログを共有します。

6. このトランスメタデバイスによって使用されるロギングデバイス名をメモする。
7. ログに記録されるファイルシステムを含んだスライスやメタデバイスを、それぞれ「スライスブラウザ」ウィンドウやオブジェクトリストから、新しいトランスメタデバイスのマスター矩形にドラッグする。
警告ダイアログボックスが表示されたら「継続」をクリックします。

注 - /etc/vfstab ファイルにファイルシステムのエントリが存在し、そのファイルシステムが現在マウントされている場合、DiskSuite ツールはこれを自動的に更新して、トランスメタデバイスの名前を使用します。

8. 新しいトランスメタデバイスの「情報」ウィンドウを表示し、既存のトランスメタデバイスについてメモした (234 ページの手順 6) スライス名またはメタデバイス名を、「ログ」テキストフィールドに入力する。
9. 「接続」をクリックする。
評価に対する警告ダイアログボックスが表示されることがあります。その場合は、「適用」をクリックします。「情報」ウィンドウの「閉じる」をクリックして、「情報」ウィンドウを終了する。
10. トランスメタデバイスオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
表示される警告ダイアログボックスの「確定」をクリックします。これによって、新しいトランスメタデバイスが作成され、既存のトランスメタデバイスのロギングデバイスを共有します。
11. コンフィグレーションログを表示して、トランスメタデバイスが確定されたことを確認する。
12. システムをリブートし、ファイルシステムのロギングを有効にする。
それ以降のリブート時には、ファイルシステムをチェックする代わりに、ログに記録されたファイルシステムごとに fsck(1M) が次のようなメッセージを表示します。

```
/dev/md/rdisk/trans: is logging.
```

▼ ロギングデバイスをファイルシステム間で共有する方法 (コマンド行)

この作業では、他のファイルシステム用のログにより、すでにトランスメタデバイスが設定されているものと想定します。

222ページの「DiskSuite オブジェクトを変更するための前提条件」の前提条件と 232ページの「ロギングデバイスを共有するための予備情報」の予備情報をチェックしてから、`metainit(1M)` を使用してログを共有します。詳細については、`metainit(1M)` のマニュアルページを参照してください。

例 - ロギングデバイスの共有

```
# umount /xyzfs
# metainit d64 -t c0t2d0s4 d10
d64: Trans is setup
(/xyzfs 用のエントリがトランスメタデバイス d64 を参照するよう、/etc/vfstab ファイルを編集する)
# mount /xyzfs
# metastat
...
d10: Logging device for d63 d64
...
```

この例では、前のトランスメタデバイス用のログとして定義されたロギングデバイス (d10) を、新しいトランスメタデバイス (d64) と共有します。マスターデバイスとして設定されるファイルシステムは `/xyzfs` であり、スライス `/dev/dsk/c0t2d0s4` を使用しています。`metainit -t` は、設定がトランスメタデバイスであることを指定します。`/etc/vfstab` ファイルを編集して、ファイルシステムがトランスメタデバイスを参照するように変更 (または新規に入力) しなければなりません。たとえば、次の行は、

```
/dev/dsk/c0t2d0s4 /dev/rdisk/c0t2d0s4 /xyzfs ufs 2 yes -
```

次のように変更してください。

```
/dev/md/dsk/d64 /dev/md/rdsk/d64 /xyzfs ufs 2 yes -
```

`metastat` コマンドは、ログが共有されていることを確認します。システムをリブートすると、ファイルシステムのロギングが有効になります。

それ以降のリブート時には、ファイルシステムをチェックする代わりに、`fsck(1M)` が 2 つのファイルシステムに対して次のメッセージを表示します。

```
/dev/md/rdisk/d63: is logging.  
/dev/md/rdisk/d64: is logging.
```

DiskSuite オブジェクトの除去

この章では、DiskSuite ツールのグラフィカルユーザーインターフェースとコマンド行ユーティリティの両方による、DiskSuite オブジェクトの除去方法について説明します。

DiskSuite ツールを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 239ページの「状態データベースの複製の除去方法 (DiskSuite ツール)」
- 241ページの「ストライプ、連結、またはストライプ方式の連結の除去方法 (DiskSuite ツール)」
- 244ページの「ミラーとサブミラーを除去する方法 (DiskSuite ツール)」
- 247ページの「RAID5 メタデバイスの除去方法 (DiskSuite ツール)」
- 249ページの「トランスメタデバイスの除去方法 (DiskSuite ツール)」
- 251ページの「マウント解除不可能なファイルシステムからトランスメタデバイスを除去する方法 (DiskSuite ツール)」
- 255ページの「ホットスペアをホットスペア集合から除去する方法 (DiskSuite ツール)」
- 256ページの「ホットスペア集合の除去方法 (DiskSuite ツール)」

コマンド行インターフェースを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 240ページの「状態データベースの複製を除去する方法 (コマンド行)」
- 242ページの「ストライプ、連結、またはストライプ方式の連結の除去方法 (コマンド行)」

- 245ページの「ミラーとサブミラーの除去方法 (コマンド行)」
- 248ページの「RAID5 メタデバイスの除去方法 (コマンド行)」
- 250ページの「トランスメタデバイスの除去方法 (コマンド行)」
- 253ページの「マウント解除不可能なファイルシステムからトランスメタデバイスを除去する方法 (コマンド行)」
- 255ページの「ホットスペアをホットスペア集合から除去する方法 (コマンド行)」
- 257ページの「ホットスペア集合の除去方法 (コマンド行)」
- 259ページの「ディスクセットからホストを除去する方法 (コマンド行)」
- 260ページの「ディスクセットからドライブを除去する方法 (コマンド行)」
- 261ページの「ディスクセットの除去方法 (コマンド行)」

DiskSuite オブジェクトの削除の概要

この章では、次に示すような DiskSuite オブジェクトを除去する手順について説明します。

- 状態データベースの複製
- メタデバイス
- ホットスペアとホットスペア集合
- ディスクセットのドライブとホスト
- ディスクセット

トラブルシューティングや再構成を実行したり、あるいはシステムからオブジェクトを単に削除する場合、オブジェクトの除去が必要になることもあります。DiskSuite の概要については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

DiskSuite オブジェクトを除去するための前提条件

この章の手順に対する前提条件を次に示します。

- 現在の時点で、すべてのデータをバックアップしている。
- ルート権限があることを確認する。

- グラフィカルユーザーインターフェースを使用する場合は、DiskSuite ツールを起動する。

ローカルなメタデバイス (ディスクセット構成に含まれないメタデバイス) を操作するには、次のように入力します。

```
# metatool &
```

ディスクセット内のメタデバイスを操作するには、自分がディスクセットの所有者であることを確認し、次のように入力します。

```
# metatool -s <ディスクセット名> &
```

状態データベースの複製を除去

この節では、状態データベースの複製をシステムから除去する方法について説明します。

状態データベースの複製を除去するための予備情報

- 状態データベースの複製をすべて除去することは可能ですが、実際にはメタデバイスがまだ構成されている間は除去してはなりません。状態データベースの複製をすべて除去すると、メタデバイスが動作不能になります。
- 状態データベースの複製のエントリを md.tab ファイルから除去しても、複製はシステムから削除されません。DiskSuite ツールを使用するか、metadb -d コマンドを実行してください。
- 状態データベースの複製を削除する前に、それがもう不要であることを確認してください。

▼ 状態データベースの複製の除去方法 (DiskSuite ツール)

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (239ページの「状態データベースの複製を除去するための予備情報」) を読んだことを確認する。

2. **MetaDB** オブジェクトを、オブジェクトリストから「メタデバイスエディタ」ウィンドウのキャンバスにドラッグする。
3. 削除する状態データベースの複製を、**MetaDB** オブジェクトからドラッグする。選択したスライスを「メタデバイスエディタ」のキャンバスにドラッグします。
4. **MetaDB** オブジェクトの先頭の矩形内部をクリックし、「確定」をクリックする。
状態データベースの複製を含むスライスの状態が、「スライスブラウザ」ウィンドウ内で「未割り当て」に変化します。
5. スライスをキャンバスから移動させる。
MetaDB オブジェクトから外にドラッグされたスライスを選択し、「キャンバスから移動」をクリックします。スライスは「スライスブラウザ」に返されます。
6. コンフィグレーションログを表示して、**MetaDB** オブジェクトが確定されたことを確認する。

注 - 状態データベースの複製をすべて削除しようとする、警告メッセージが表示されます。

▼ 状態データベースの複製を除去する方法 (コマンド行)

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と 239ページの「状態データベースの複製を除去するための予備情報」の予備情報をチェックしてから、`-d` オプション付きの `metadb(1M)` コマンドを使用して、状態データベースの複製を削除します。状態データベースの複製がもう不要であることを確認します。詳細については、`metadb(1M)` のマニュアルページを参照してください。

例 - 状態データベースの複製を 2 つ除去する

```
# metadb -d c0t2d0s0 c0t1d0s0
```


この例では、スライス /dev/dsk/c0t2d0s0 と /dev/dsk/c0t1d0s0 に置かれた、状態データベースの複製を 2 つ除去します。

例 - 状態データベースの複製をすべて強制的に除去する

```
# metadb -f -d c0t1d0s3 c4t1d0s3
```

この例では、システムの残り 2 つの状態データベースの複製を除去します。-f オプションが必要です。



ストライプと連結の除去

この節では、ストライプと連結をシステムから除去する方法について説明します。

注意 - メタデバイスをシステムから恒久的に除去し、その配下にあるスライスを再使用すると、そのメタデバイス上のデータはすべて失われます。データを保存する必要がある場合、データをバックアップしてください。

ストライプと連結を除去するための予備情報

- ストライプや連結を削除する前に、それがもう不要であることを確認します。
- ストライプや連結を削除し、削除されたメタデバイスの一部であったスライスを再使用した場合、そのメタデバイスにあったすべてのデータはシステムから失われます。したがって、メタデバイスを削除する前に、メタデバイス上のデータをバックアップしておいてください。

▼ ストライプ、連結、またはストライプ方式の連結の除去方法 (DiskSuite ツール)

1. **前提条件** (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、**予備情報** (241ページの「ストライプと連結を除去するための予備情報」) を読んだことを確認する。
2. メタデバイスの最新のバックアップがあることを確認する。

3. メタデバイスへのアクセスを停止する。
たとえば、メタデバイス上のファイルシステムをマウント解除します。データベースなど、非 UFS アプリケーションの場合、アプリケーションによるメタデバイスの使用を停止させるために必要な手順を実行します。
4. 削除するストライプや連結をオブジェクトリストからダブルクリックする。
メタデバイスオブジェクトがキャンバスに表示されます。
5. オブジェクトのポップアップメニューを表示し、「削除」を選択する。
6. 表示されるダイアログボックスで「削除」をクリックする。
7. [オプション] このメタデバイスの /etc/vfstab ファイルにエントリが存在する場合、そのエントリを削除する。
存在しないメタデバイスにファイルシステムをマウントするよう要求して、システムを混乱させることがないようにします。
8. コンフィグレーションログを表示して、オブジェクトが削除されたことを確認する。

注 - メタデバイスに割り当てられたホットスペア集合は削除されません。

▼ ストライプ、連結、またはストライプ方式の連結の除去方法 (コマンド行)

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と 241ページの「ストライプと連結を除去するための予備情報」の予備情報をチェックし、`metaclear(1M)` コマンドを使用して、メタデバイスを除去します。詳細については、`metaclear(1M)` のマニュアルページを参照してください。

例 - 連結の除去

```
# umount d8
# metaclear d8
d8: Concat/Stripe is cleared
```

```
(/etc/vfstab ファイルを編集する)
```

この例では、マウントされたファイルシステムを含む、連結 d8 を除去します。メタデバイスを除去するためには、その前に、ファイルシステムをマウント解除しなければなりません。システムは、連結が除去されたことを示す確認メッセージを表示します。このメタデバイスの /etc/vfstab ファイルにエントリが存在する場合、そのエントリを削除します。存在しないメタデバイスにファイルシステムをマウントするよう要求して、システムを混乱させることがないようにします。

ミラーの除去

この節では、ミラーをシステムから除去する方法について説明します。

ルート (/)、swap、/opt、/usr など、マウント解除できないファイルシステムによって使用されるミラーを除去することは、本質的に、ファイルシステムの「ミラー化解除」を行い、ミラーを構成するいずれかのサブミラーの配下にあるスライスにそれをマウントする操作を伴います。206ページの「マウント解除不可能なファイルシステムをミラー化解除する方法 (コマンド行)」を参照してください。

注・ミラーを除去し、マウントデバイスと同じメタデバイス名を維持するには、358ページの「メタデバイス名の切り替え」を参照してください。

ミラーを除去するための予備情報

- ミラーのサブミラーを削除し、削除されたサブミラーの一部であったスライスを再使用した場合、そのデータはすべてシステムから失われます。したがって、ミラーとそのサブミラーを削除する前に、ミラー上のデータをバックアップしておいてください。
- ミラーを除去する場合、最初にミラーをそのサブミラーに「分解」します。データをいずれかのサブミラーに保存するよう選択したり、サブミラーが単一のスライスから成る場合には、データをスライス上に保存するよう選択できます。

▼ ミラーとサブミラーを除去する方法 (DiskSuite ツール)

この作業は、ルート (/)、swap、/opt、または /usr であるファイルシステムを除いて、ミラーを使用する非 UFS アプリケーションや任意のファイルシステムに使用できます。これらのファイルシステムのいずれかによって使用されるミラーを除去するには、206ページの「マウント解除不可能なファイルシステムをミラー化解除する方法 (コマンド行)」を参照してください。

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (243ページの「ミラーを除去するための予備情報」) を読んだことを確認する。
2. メタデバイスの最新のバックアップがあることを確認する。
3. メタデバイスへのアクセスを停止する。
たとえば、ミラー化されたファイルシステムのマウント解除が必要です。データベースなど、非 UFS アプリケーションの場合、アプリケーションによるメタデバイスの使用を停止させるために必要な手順を実行します。
4. オブジェクトリストから、ミラーオブジェクトをダブルクリックする。
ミラーオブジェクトがキャンバスに表示されます。
5. ミラーオブジェクトのポップアップメニューを表示し、「削除」を選択する。
6. 「削除」ボタンをクリックする。
ミラーは、その配下のサブミラーに分割されます (連結方式メタデバイス)。
7. [オプション] ミラーがファイルシステムとして使用され、/etc/vfstab ファイル内にミラーのエントリが存在した場合、次に示すいずれかの方法を使用して、ミラーのエントリを整理する。

- /etc/vfstab エントリを変更して、いずれかのサブミラーにファイルシステムをマウントする。
 - いずれかのサブミラーが 1 面の連結から成る場合、/etc/vfstab エントリを変更して、配下のスライスにファイルシステムをマウントできる。
 - もうデータにアクセスする予定がない場合、ファイルシステムの /etc/vfstab エントリを完全に除去できる。
8. 連結方式オブジェクトを整理する。
不要となったサブミラーメタデバイスを削除します。
 9. コンフィグレーションログを表示して、オブジェクトが削除されたことを確認する。

▼ ミラーとサブミラーの除去方法 (コマンド行)

この作業は、ルート (/)、swap、/opt、または /usr であるファイルシステムを除いて、ミラーを使用する非 UFS アプリケーションや任意のファイルシステムに使用できます。これらのファイルシステムによって使用されるミラーを除去するには、206ページの「マウント解除不可能なファイルシステムをミラー化解除する方法 (コマンド行)」を参照してください。

この作業の手順を次に示します。

- ファイルシステムをマウント解除する
- ミラーとそのいずれかのサブミラー上で `metadetach(1M)` を実行する
- ミラー上で `metaclear(1M)` を実行する
- オプション: /etc/vfstab ファイルを編集して、非ミラーデバイスを使用する (ファイルシステムエントリがここに表示される場合)
- オプション: ファイルシステムを再マウントする

例 - ミラーを除去し、データをサブミラーに保存

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と243ページの「ミラーを除去するための予備情報」の予備情報をチェックしてから、`metadetach(1M)` と `metaclear(1M)` のコマンドを使用して、メタデバイスを

削除します。詳細については、metadetach(1M) と metaclear(1M) のマニュアルページを参照してください。

```
# metastat d2
d2: Mirror
   Submirror 0: d0
   State: Okay
   Submirror 1: d1
   State: Okay
...
# umount /news
# metadetach d2 d0
# metaclear d2
d2: Mirror is cleared
(/news がサブミラー d0 を参照するよう、/etc/vfstab ファイルを編集する)
# mount /news
# metaclear d1
d1: Concat/Stripe is cleared
```

この例では、マウントされたファイルシステムも含むミラー d2 を除去します。ミラーは、サブミラー d0 と d1 から構成されます。metastat コマンドは、両方のサブミラーが「Okay (正常)」状態であることを通知します。metadetach コマンドがサブミラー d0 をミラー d2 から切り離すためには、その前に、ファイルシステムがマウント解除されなければなりません。その後、ミラーが除去されます。

サブミラー d0 上のデータへのアクセスを続けるために、/etc/vfstab ファイル内のファイルシステムのエントリーは、ミラーから連結 (サブミラー) d0 に変更されます。

サブミラー (連結) d0 を参照するよう /etc/vfstab ファイルを整理した後、ファイルシステムが再マウントされます (d0 に再マウントされます)。別のサブミラー d1 は、metaclear コマンドで除去されます。

RAID5 メタデバイスの除去

この節では、RAID5 メタデバイスをシステムから除去する方法について説明します。



注意 - RAID5 メタデバイスがシステムから永久に除去され、その配下にあるスライスは再使用されると、RAID5 メタデバイスにあったデータはすべて失われます。データを保存する必要がある場合は、データをバックアップしてください。

RAID5 メタデバイスを除去するための予備情報

- RAID5 メタデバイスを削除し、削除されたデバイスの一部であったスライスを再使用した場合、そのデータはすべて、システムから失われます。したがって、RAID5 メタデバイスを削除する前に、そこにあるデータをバックアップしてください。

▼ RAID5 メタデバイスの除去方法 (DiskSuite ツール)

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (247ページの「RAID5 メタデバイスを除去するための予備情報」) を読んだことを確認する。
2. メタデバイスの最新のバックアップがあることを確認する。
3. **RAID5** メタデバイスに対するアクセスを停止する。
たとえば、ファイルシステムをマウント解除します。データベースなど、非UFS アプリケーションの場合、アプリケーションによるメタデバイスの使用を停止させるために必要な手順を実行します。
4. 削除する **RAID5** オブジェクトを、オブジェクトリストからダブルクリックする。
RAID5 オブジェクトがキャンバスに表示されます。
5. **RAID5** オブジェクトのポップアップメニューを表示し、「削除」を選択する。
6. 表示されるダイアログボックスで、「削除」をクリックする。
7. [オプション] このメタデバイスの /etc/vfstab ファイルにエントリが存在する場合、そのエントリを削除する。
存在しないデバイスにファイルシステムのマウント要求を行なって、システムを混乱させることがないようにします。
8. コンフィグレーションログを表示して、オブジェクトが削除されたことを確認する。

注 - メタデバイスに割り当てられたホットスペア集合は、削除されません。

▼ RAID5 メタデバイスの除去方法 (コマンド行)

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と 247ページの「RAID5 メタデバイスを除去するための予備情報」の予備情報をチェックしてから、`metaclear(1M)` コマンドを使用して、メタデバイスを削除します。詳細については、`metaclear(1M)` のマニュアルページを参照してください。

例 - RAID5 メタデバイスの除去

```
# umount /nfs
# metaclear d80
d80: RAID is cleared
(/etc/vfstab ファイルを編集する)
```

この例では、マウントされたファイルシステム `/nfs` も含む、RAID5 メタデバイス `d80` を除去します。`d80` へのアクセスは、そのファイルシステムをマウント解除することによって停止されます。システムは、RAID5 メタデバイスが除去されるという確認メッセージを表示します。このメタデバイスの `/etc/vfstab` ファイルにエントリが存在する場合は、それを削除してください。存在しないメタデバイスにファイルシステムをマウントしようとして、システムを混乱させることがないようにします。

トランスメタデバイスの除去

この節では、トランスメタデバイス (UFS ロギング) の除去方法について説明します。

注・トランスメタデバイスを除去し、マウントデバイスと同じメタデバイス名を維持するには、358ページの「メタデバイス名の切り替え」を参照してください。

トランスメタデバイスを除去するための予備情報

- トランスメタデバイスを除去すると、適切なファイルシステムからロギングが除去されます。そのメタデバイスやスライスも除去しない限り、ファイルシステム

上の配下のデータ (マスターがメタデバイスから成る場合は、メタデバイスも) は、まだ保存されています。

- そのマスターデバイスに関連する情報は、トランスメタデバイスを除去する前に、ログデバイスから元に戻されます。

▼ トランスメタデバイスの除去方法 (DiskSuite ツール)

この作業は、マウント解除できるファイルシステムから UFS ロギングを除去するために使用します。

1. **前提条件** (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、**予備情報** (248ページの「トランスメタデバイスを除去するための予備情報」) を読んでいることを確認する。
2. メタデバイスの最新のバックアップがあることを確認する。
3. ファイルシステムをマウント解除する。
4. オブジェクトリストから、トランスメタデバイスオブジェクトをダブルクリックする。
トランスメタデバイスオブジェクトがキャンバスに表示されます。
5. トランスメタデバイスオブジェクトのポップアップメニューを表示し、「削除」を選択する。
6. 「削除」ボタンをクリックする。
トランスメタデバイスは、その配下のマスターデバイスとロギングデバイスに分割されます。
7. 次のいずれかの方法によって、`/etc/vfstab` ファイルを編集する。
 - マスターデバイスがメタデバイスである場合は、マスターデバイスとして機能するメタデバイスのファイルシステムをマウントするよう、`/etc/vfstab` ファイルのエントリを変更できます。
 - マスターデバイスが1つのスライスから成るメタデバイスである場合は、そのメタデバイスの配下にあるスライスのファイルシステムをマウントするよう、`/etc/vfstab` ファイルのエントリを変更できます。

- マスターデバイスがスライスである場合は、そのスライスのファイルシステムをマウントするよう、`/etc/vfstab` ファイルのエントリを変更できます。
- これ以上データにアクセスする予定がない場合は、ファイルシステム用の `/etc/vfstab` ファイルのエントリを完全に除去します。

8. コンフィグレーションログを表示して、トランスメタデバイスオブジェクトが削除されたことを確認する。

9. `fsck(1M)` コマンドを実行する。

ファイルシステムはもうロギングデバイスではないため、ファイルシステムをマウントするには、その前に `fsck` を実行する必要があります。`fsck` の実行は、マスターデバイスの構成に応じて、`raw` メタデバイス上、またはスライス用の `raw` デバイス上で行います。次のプロンプトに対して `y` と答えます。

```
# fsck <raw デバイス名>
...
FILE SYSTEM STATE IN SUPERBLOCK IS WRONG; FIX? y
...
```

注 - ファイルシステムをメタデバイスにマウントする場合、そのメタデバイス用の `raw` デバイス上で `fsck` を実行します。そうでない場合、ファイルシステムをマウントするスライス用の `raw` デバイス上で `fsck` を実行します。

10. ファイルシステムをマウントする。

ファイルシステムは、もうログを記録していません。

11. [オプション] マスターデバイスとロギングデバイスを整理する。

マスターデバイスとロギングデバイスがメタデバイスであった場合は、不要となったメタデバイスを削除します。

▼ トランスメタデバイスの除去方法 (コマンド行)

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と 248ページの「トランスメタデバイスを除去するための予備情報」の予備情報をチェック

クしてから、`metaclear(1M)` コマンドを使用して、トランスメタデバイスを除去します。詳細は、`metaclear(1M)` のマニュアルページを参照してください。

例 - トランスメタデバイスの除去

```
# umount /abcfs
# metaclear d64
(/etc/vfstab ファイルを編集する)
# fsck /dev/rdsk/c0t2d0s6
...
FILE SYSTEM STATE IN SUPERBLOCK IS WRONG; FIX? y
...
# mount /abcfs
```

この例では、`/abcfs` ファイルシステムから UFS ロギングを除去し、トランスメタデバイス `d64` を使用します。マスターデバイスの配下のスライスは、`/dev/dsk/c0t2d0s6` です。トランスメタデバイスが除去されると、マスターデバイスに関連する情報は、デバイスを除去する前にログから元に戻されます。`/etc/vfstab` ファイルに含まれるファイルシステム用のエントリは、トランスメタデバイス用のメタデバイス名の代わりに、ファイルシステムを含むブロックデバイスと `raw` デバイスを参照するように変更しなければなりません。ファイルシステムはもうロギングデバイスではないため、マウントする前に `fsck(1M)` コマンドが実行されます。FIX? プロンプトに対して `y` と応答すると、ファイルシステムが配下のスライスにマウントされます。

▼ マウント解除不可能なファイルシステムからトランスメタデバイスを除去する方法 (DiskSuite ツール)

この作業は、ルート (`/`)、`/usr`、`swap` など、マウント解除できないファイルシステムから UFS ロギングを除去するために使用します。

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (248ページの「トランスメタデバイスを除去するための予備情報」) を読んでいることを確認する。
2. メタデバイスの最新のバックアップがあることを確認する。

- オブジェクトリストから、トランスメタデバイスオブジェクトをダブルクリックする。

オブジェクトがキャンバスに表示されます。

- ロギングデバイスを、トランスメタデバイスオブジェクトからキャンバスにドラッグする。

オブジェクトの状態は、「ログの切断 (予定設定済み)」に変化します。

- トランスメタデバイスオブジェクトの先頭の矩形内部をクリックしてから、「確定」をクリックする。

ダイアログボックスが表示され、トランスメタデバイスがマウント解除された後、または次のリブートの後で、ロギングデバイスが切断されるという警告が出されます。「確定」をクリックします。

トランスメタデバイスの状態が「ログの切断 (進行中)」に変化します。

注 - /etc/vfstab ファイルにファイルシステムのエントリが存在し、そのファイルシステムが現在マウントされている場合、DiskSuite ツールはこれを自動的に更新して、トランスメタデバイス名の代わりにスライス名を使用するようにします。

- リポートする。

ファイルシステムがチェックされていることを示すメッセージが表示されます。

```
...
The /usr file system (/dev/md/rdisk/d0) is being checked.
/dev/md/rdisk/d0: 11576 files, 198318 used, 42081 free
/dev/md/rdisk/d0: (737 frags, 5168 blocks, 0.3% fragmentation)
...
```

- /etc/vfstab ファイルを編集して、トランスメタデバイスを除去する。

ファイルシステムが、トランスメタデバイスではなく、ファイルシステムを含むブロックデバイスと raw デバイスを参照するよう、ファイルシステムのエントリを変更します。

- リポートする。

このリポートによって、システムは、ファイルシステムがトランスメタデバイスではなく、すでにその配下のスライスにマウントされていることを認識できません。

9. トランスメタデバイスを削除する。
 - a. オブジェクトリストから、トランスメタデバイスオブジェクトをダブルクリックする。オブジェクトがキャンバスに表示されます。
 - b. オブジェクトの先頭の矩形内部をクリックし、プルダウンメニューを表示する。
 - c. 「削除」オプションを選択する。
10. 確認ダイアログボックスの「削除」ボタンをクリックする。
11. コンフィグレーションログを表示して、トランスメタデバイスオブジェクトが削除されたことを確認する。

▼ マウント解除不可能なファイルシステムからトランスメタデバイスを除去する方法 (コマンド行)

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と248ページの「トランスメタデバイスを除去するための予備情報」の予備情報をチェックしてから、`metadetach(1M)` と `metaclear(1M)` のコマンドを使用して、トランスメタデバイスを除去します。詳細は、`metadetach(1M)` と `metaclear(1M)` のマニュアルページを参照してください。

この作業は、`/usr` など、通常のシステム操作中にはマウント解除できないファイルシステムから、トランスメタデバイスを除去するときに使用します。

例 - `/usr` からトランスメタデバイスを除去

```
# metadetach -f d20
d20: logging device c0t0d0s1 will be detached at unmount or reboot
# reboot
...
```

(続く)

続き

```
The /usr file system (/dev/rdisk/c0t3d0s3) is being checked.  
...  
(/etc/vfstab ファイルを編集する)  
# reboot  
...  
# metaclear d20  
d20: Trans is cleared
```

この例では、d20 は、/usr ファイルシステムに UFS ロギングを提供するトランス
メタデバイスです。ロギングを除去するため、-f オプションを付けて metadetch
コマンドを実行してロギングデバイスを強制的に切断し、システムをリブートしま
す。次に、ファイルシステムが、(トランスメタデバイスではなく)ファイルシステム
を含むスライスを参照するよう、/etc/vfstab ファイルを編集してファイルシス
テムのエントリを変更します。もう一度リブートすると、/usr ファイルシステム
は、その新しいマウントデバイス上に置かれます。metaclear コマンドは、トラン
スメタデバイス d20 をシステムから除去します。

ホットスペアとホットスペア集合の除去

この節では、ホットスペアとホットスペア集合をシステムから除去する方法につい
て説明します。

ホットスペアとホットスペア集合を除去するための予備 情報

- 使用中のホットスペアは削除できません。
- ホットスペア集合を削除するためには、その前に、サブミラーと RAID5 メタデ
バイスに対するホットスペア集合の関連付けをすべて削除します。

▼ ホットスペアをホットスペア集合から除去する方法 (DiskSuite ツール)

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (254ページの「ホットスペアとホットスペア集合を除去するための予備情報」) を読んでいることを確認する。
2. オブジェクトリストから、ホットスペア集合オブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
3. 除去したいホットスペアスライスを、ホットスペア集合オブジェクトからキャンバスにドラッグする。
4. ホットスペア集合オブジェクトの先頭の矩形内部をクリックし、「確定」をクリックする。
5. コンフィグレーションログを表示して、ホットスペア集合が確定されたことを確認する。

▼ ホットスペアをホットスペア集合から除去する方法 (コマンド行)

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と 254ページの「ホットスペアとホットスペア集合を除去するための予備情報」の予備情報をチェックしてから、`metahs(1M)` コマンドを使用して、ホットスペアを除去します。詳細は、`metahs(1M)` のマニュアルページを参照してください。

例 - ホットスペアをホットスペア集合から除去

```
# metahs -d hsp003 /dev/dsk/c2t1d0s2
hsp003: Hotspare is deleted
```

この例では、ホットスペア `/dev/dsk/c2t1d0s2` をホットスペア集合 `hsp003` から除去します。

例 - ホットスペアをすべてのホットスペア集合から除去

```
# metahs -d all /dev/dsk/c2t1d0s2
hsp003: Hotspare is deleted
hsp004: Hotspare is deleted
# metahs -i
...
hsp003: 2 hot spares
         c1t3d0s6           Available      912800 blocks
         c0t0d0s4           Available      5600 blocks

hsp004: 2 hot spares
         c1t3d0s6           Available      912800 blocks
         c0t0d0s4           Available      5600 blocks
```

この例では、ホットスペア /dev/dsk/c2t1d0s2 を、それに関連付けられたすべてのホットスペア集合から除去します。-i オプション付きの metahs コマンドは、ホットスペアスライスが、もう (削除元の) ホットスペア集合の一部ではないことを明らかにします。

▼ ホットスペア集合の除去方法 (DiskSuite ツール)

ホットスペア集合を除去するためには、その前に、サブミラーと RAID5 メタデバイスに対する関連付けをすべて除去しなければなりません。

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (254ページの「ホットスペアとホットスペア集合を除去するための予備情報」) を読んでいることを確認する。
2. 「ブラウズ」メニューから「ホットスペア集合」を選択し、「ホットスペア集合ブラウザ」ウィンドウを表示する。
3. スクロールリストから、削除したいホットスペア集合をダブルクリックする。
そのホットスペア集合の「ホットスペア集合情報」ウィンドウが表示されます。「関連のあるデバイス」リストには、そのホットスペア集合を使用するすべてのメタデバイスが表示されます。
4. 「関連のあるデバイス」リストに含まれるメタデバイス (サブミラーや RAID5 メタデバイス) ごとに、ホットスペア集合の関連付けを除去する。
 - a. 「関連のあるデバイス」リストに含まれるメタデバイスを、1つずつダブルクリックする。
適切な「情報」ウィンドウが表示されます。

- b. 「除去」をクリックする。「ホットスペア集合」フィールドが空白になるので、「閉じる」をクリックする。
 - c. 「関連のあるデバイス」リストに含まれるメタデバイスごとに、**4a** と **4b** を繰り返す。
 - d. 各メタデバイスのホットスペア集合の関連付けを除去し終わったら、「ホットスペア集合情報」ウィンドウで「閉じる」をクリックする。
5. ホットスペア集合の関連付けを除去した、それぞれのミラーや **RAID5** デバイスを確定する。
ホットスペア集合の「使用状況」状態が「なし」に変化します。
 6. オブジェクトリストから、ホットスペア集合オブジェクトをダブルクリックして、キャンバスに表示する。
 7. ホットスペア集合を削除する。
オブジェクトのポップアップウィンドウを表示し、「削除」を選択します。
 8. コンフィグレーションログを表示して、ホットスペア集合が削除されたことを確認する。

▼ ホットスペア集合の除去方法 (コマンド行)

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と 254ページの「ホットスペアとホットスペア集合を除去するための予備情報」の予備情報をチェックしてから、`metaparam(1M)` と `metahs(1M)` のコマンドを使用して、ホットスペア集合を除去します。詳細は、`metaparam(1M)` と `metahs(1M)` のマニュアルページを参照してください。

例 - ホットスペア集合を除去

```
# metastat
...
d30: Mirror
```

(続く)

```
State: Okay
...
d31: Submirror of d30
    Hot spare pool: hsp001
...
d32: Submirror of d30
    Hot spare pool: hsp001
...
# metaparam -h none d30
# metaparam -h none d31
# metahs -d hsp001
hsp001: Hotspare pool is cleared
```

この例では、ホットスペア集合 hsp001 の除去方法を示します。ホットスペア集合の関連付けを調べるには、metastat コマンドを使用します。metastat の出力には、ホットスペア集合 hsp001 を使用する 2 つのサブミラー (d31 と d32) が示されます。-h オプションと none オプションを付けた metaparam コマンドは、ホットスペア集合に対する関連付けを除去します。その際、まずサブミラー d31 の関連付けが除去されてから、d32 の関連付けが除去されます。ホットスペア集合に関連付けがなくなれば、metahs -d コマンドで除去されます。

ディスクセットの除去

この節では、ディスクセットをシステムから除去する方法について説明します。

ホストとディスクをディスクセットから除去するための予備情報

- ディスクセットからホストを除去することができます。ただし、最後のホストをディスクセットから除去するには、ディスクセット内のドライブがすべて除去されていなければなりません。
- 最後のホストをディスクセットから除去すると、ディスクセットが破壊されます。
- ディスクセットからドライブが除去されると、DiskSuite は、残りのドライブ全体を通じて、メタデバイス状態データベースの複製のバランスを再調整します。

▼ ディスクセットからホストを除去する方法 (コマンド行)

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (258ページの「ホストとディスクをディスクセットから除去するための予備情報」) を読んでいることを確認する。
2. `metaset(1M)` コマンドを使用して、ディスクセットからホストを除去する。

```
# metaset -s <ディスクセット> -d -h <ホスト>
```

このコマンドでは、

<code>-s <ディスクセット></code>	<code>metaset</code> が作用するディスクセットの名前を指定します。
<code>-d</code>	指定したホストを削除します。
<code>-h <ホスト></code>	ディスクセットから削除する 1 つ以上のホスト名を指定します。このホスト名は、 <code>/etc/nodename</code> にあるものと同じ名前です。

3. ホストがディスクセットから除去されたことを確認するには、オプションなしで `metaset(1M)` コマンドを使用する。

```
# metaset
```

例 - ホストをディスクセットから除去

```
red# metaset -s relo-red -d -h blue
red# metaset

Set name = relo-red, Set number = 1
Host          Owner
  red          Yes
...
```

この例では、ホスト `blue` をディスクセット `relo-red` から除去します。

▼ ディスクセットからドライブを除去する方法 (コマンド行)

1. 前提条件 (238ページの「DiskSuite オブジェクトを除去するための前提条件」) を満たし、予備情報 (258ページの「ホストとディスクをディスクセットから除去するための予備情報」) を読んでいることを確認する。
2. **metaset(1M)** コマンドを使用して、ディスクセットからドライブを除去する。

```
# metaset -s <ディスクセット> -d [-f] <ドライブ>...
```

このコマンドでは、

-s <ディスクセット>	metaset が作用するディスクセットの名前を指定します。
-d	指定したドライブ名を削除します。
-f	ディスクセット内の最後のドライブを強制的に削除します。このドライブには最後の状態データベースの複製が暗黙に含まれるためです。
<ドライブ>	ディスクセットから削除されるドライブを指定します。ドライブの指定形式は <code>cxtxdx</code> とし、スライスを指定しません。

注 - ディスクセット内の最後のドライブを削除するには、`-f` オプションを使用します。

3. **metaset(1M)** コマンドを使用して、ディスクセットからドライブが除去されたことを確認する。

例 - ディスクセットからドライブを除去

```
red# metaset -s relo-red -d c2t5d0
red# metaset
...
```

(続く)

Host	Owner
red	Yes
blue	
Drive	Dbase
c1t2d0	Yes
c1t3d0	Yes
c2t2d0	Yes
c2t3d0	Yes
c2t4d0	Yes

この例では、ディスクセット relo-red からドライブ c2t5d0 を除去します。metaset コマンドは、削除されたドライブがもうディスクセットの一部ではないことを確認します。

▼ ディスクセットの除去方法 (コマンド行)

ディスクセットを完全に除去するという事は、次のようなことを意味します。

- ディスクセット内のメタデバイスとホットスペア集合をすべて除去する
- ディスクセット内のホストを1つ残して、あとはすべて除去する
- ディスクセット内のドライブをすべて除去する
- 最後のホストを除去する

例 - ディスクセットの除去

238ページの「DiskSuite オブジェクトを除去するための前提条件」の前提条件と 258ページの「ホストとディスクをディスクセットから除去するための予備情報」の予備情報をチェックしてから、metaclear(1M) と metaset(1M) のコマンドを使用して、ディスクセットを完全に除去します。詳細は、metaclear(1M) と metaset(1M) のマニュアルページを参照してください。

```
red# metaclear -s relo-red -a
red# metaset -s relo-red -d -h blue
red# metaset -s relo-red -f -d c1t2d0 c1t3d0 c2t2d0 c2t3d0 c2t4d0 c2t5d0
red# metaset -s relo-red -d -h red
```

`metaclear -a` コマンドは、`relo-red` ディスクセットからすべてのメタデバイスとホットスペア集合を除去します。次に、ディスクセットからホスト `blue` が削除され、さらに共有ディスクドライブが削除されます。最後に、ディスクセットからはセット内の最後のホスト (この場合は `red`) が削除されます。この最後のコマンドは、システムからディスクセットを除去します。

システムの管理

この章では、DiskSuite に関連するシステム管理作業の実行方法について説明します。

DiskSuite ツールを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 268ページの「[「ディスク表示」ウィンドウでオブジェクトを選択する方法 (DiskSuite ツール)]
- 269ページの「[SPARCstorage Array ディスクの状態のチェック方法 (DiskSuite ツール)]
- 270ページの「[SPARCstorage Array コントローラのファンとバッテリーの状態をチェックする方法 (DiskSuite ツール)]
- 271ページの「[SPARCstorage Array コントローラの World Wide Name を表示する方法 (DiskSuite ツール)]
- 273ページの「[コントローラ、トレイ、ディスク上で NVRAM を有効にする方法 (DiskSuite ツール)]
- 274ページの「[コントローラ、トレイ、ディスクに対する同期書き込み用の NVRAM を有効にする方法 (DiskSuite ツール)]
- 275ページの「[コントローラ、トレイ、ディスク上で NVRAM を無効にする方法 (DiskSuite ツール)]
- 276ページの「[未処理の書き込みを NVRAM からフラッシュする方法 (DiskSuite ツール)]
- 276ページの「[高速書き込みデータを NVRAM からパージする方法 (DiskSuite ツール)]

- 277ページの「ディスクをホスト専用予約する方法 (DiskSuite ツール)」
- 278ページの「ホストによって予約されたディスクを解放する方法 (DiskSuite ツール)」
- 279ページの「ディスクの停止方法 (DiskSuite ツール)」
- 280ページの「ディスクの起動方法 (DiskSuite ツール)」
- 282ページの「デバイス統計情報の表示方法 (DiskSuite ツール)」
- 283ページの「デバイス統計情報をグラフ化する方法 (DiskSuite ツール)」
- 284ページの「「統計情報グラフ」ウィンドウにデバイスを追加する方法 (DiskSuite ツール)」
- 284ページの「「統計情報グラフ」ウィンドウからデバイスを除去する方法 (DiskSuite ツール)」
- 285ページの「SunNet Manager を有効にして DiskSuite ツールを起動する方法 (SunNet Manager)」
- 286ページの「SunNet Manager から DiskSuite ツールを起動する方法 (SunNet Manager)」
- 290ページの「ファイルシステムマネージャとディスクマネージャの起動方法 (DiskSuite ツール)」

コマンド行インタフェースを使用する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 287ページの「DiskSuite の SNMP サポートの構成方法 (コマンド行)」
- 289ページの「DiskSuite を有効にしてストレージマネージャを起動する方法 (コマンド行)」

システム管理の概要

DiskSuite を実行する管理システムに関連する業務には、次のものがあります。

- DiskSuite ツールを使用して、SPARCstorage Array の操作と管理を行う
- DiskSuite ツールを使用して、メタデバイスのパフォーマンスを監視する
- Solstice SunNet Manager™ との統合
- SNMP 警告との統合

- Solstice AdminSuite™ ストレージマネージャとの統合

DiskSuite の概要については、『*Solstice DiskSuite 4.2.1* リファレンス』を参照してください。

システムを管理するための前提条件

この章で説明する手順の前提条件を次に示します。

- 現在の時点で、すべてのデータをバックアップする。
- ルート権限があることを確認する。
- グラフィカルユーザーインターフェースを使用する場合、DiskSuite ツールを起動する。

ローカルなメタデバイス (ディスクセット構成に含まれないメタデバイス) を操作するには、次のように入力します。

```
# metatool &
```

ディスクセット内のメタデバイスを操作するには、自分がディスクセットの所有者であることを確認し、次のように入力します。

```
# metatool -s <ディスクセット名> &
```

SPARCstorage Array のグラフィック表示を操作

この節では、次の説明を行います。

- 「ディスク表示」ウィンドウにおけるオブジェクト (ディスク、トレイ、コントローラ) の選択
- SPARCstorage Array ディスクの状態チェック
- SPARCstorage Array コントローラのファンとバッテリーの状態チェック
- SPARCstorage Array の World Wide Number の表示

SPARCstorage Array をグラフィカルに操作するための予備情報

DiskSuite ツールの「ディスク表示」ウィンドウでは、SPARCstorage Array をグラフィック表示できます。「ディスク表示」ウィンドウを表示すると、ファイバチャネルアイコンに似た SPARCstorage Array コントローラ用のアイコンが、「ディスク表示」のキャンバスと「コントローラ」ウィンドウの両方に現われます。

SPARCstorage Array 内の各トレイは、トレイ番号を示すタイトルが描かれたフレームで表現されます。SPARCstorage Array 100 と 200 の違いを次に示します。

- SPARCstorage Array 100 - トレイ内の各 SCSI バス (2 つまで) は、トレイ番号を示すフレーム内で、(フレームの内部に) 描かれた 2 番目のフレームによって示されます。SSA100 トレイ内の各 SCSI バスに接続されたディスクは、SCSI フレーム内では連続して表示され、それらを連結する線はありません。

「ディスク表示」ウィンドウにおける SPARCstorage Array 100 の表示を図 6-1 に示します。

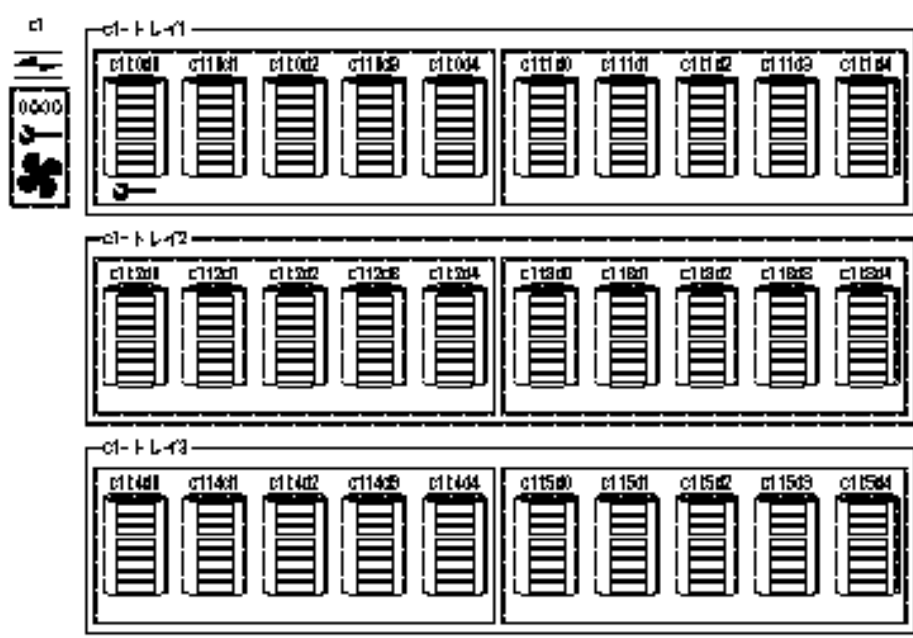


図 6-1 SPARCstorage Array 100

トレイ上のバス的一方または両方にディスクがない場合、バスは空のフレームを示すことによって「スタブアウト」されます。1つ以上のトレイにディスクがない場合、DiskSuite ツールはトレイをスタブアウトしません。

- **SPARCstorage Array 200** - トレイ内には SCSI バスが存在しないため、トレイフレームの内部にあるディスクだけが表示されます。これらのディスクは、連結する線なしで連続して表示されます。

1つ以上のトレイにディスクがない場合、DiskSuite ツールはトレイをスタブアウトしません。

「ディスク表示」ウィンドウに表示される SPARCstorage Array 200 を、図 6-2 に示します。

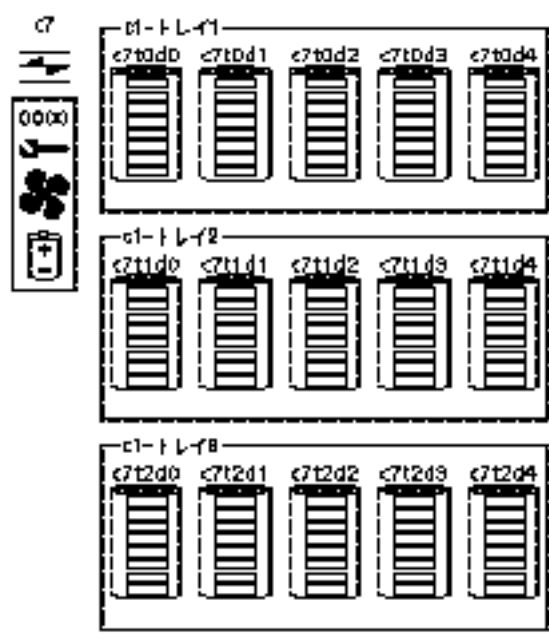


図 6-2 SPARCstorage Array 200

「ディスク表示」ウィンドウを使用すれば、World Wide Name と呼ばれる、SPARCstorage Array の一意な 12 桁の識別番号を表示することもできます。最後の 4 桁は、キャンバス (および SPARCstorage Array 自身の LCD パネル) のコントローラアイコンのすぐ下に表示されます。全体の数字を見るには、コントローラの「情報」ウィンドウを表示します。

「ディスク表示」ウィンドウを使用すれば、バッテリーやファンなど、SPARCstorage Array コンポーネントの状態を一目でチェックしたり、ディスクが起動しているか停止しているかを調べることができます。

▼ 「ディスク表示」ウィンドウでオブジェクトを選択する方法 (DiskSuite ツール)

この作業では、「ディスク表示」ウィンドウでオブジェクトを選択する方法について説明します。選択モデルは階層型になっています。1つのオブジェクトを選択すると、オブジェクトに加えて、その子オブジェクトもすべて選択することになります。

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (266ページの「SPARCstorage Array をグラフィカルに操作するための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックして、「ディスク表示」ウィンドウを表示する。
3. オブジェクトをクリックして選択する。

SPARCstorage Array の場合 :

- ディスク内のスライスをすべて選択すると、ディスクが選択されます。ディスクを選択すると、ディスク内のスライスがすべて選択されます。
- トレイ内のディスクをすべて選択すると、トレイが選択されます。トレイを選択すると、トレイ内のディスクがすべて選択されます。
- コントローラ上のトレイをすべて選択すると、コントローラが選択されます。コントローラを選択すると、コントローラ上のトレイがすべて選択されます。

非 SPARCstorage Array 構成の場合 :

- ディスク内のスライスをすべて選択すると、ディスクが選択されます。ディスクを選択すると、ディスク内のすべてのスライスが選択されます。

- コントローラ上のディスクをすべて選択すると、コントローラが選択されず。コントローラを選択すると、コントローラ上のディスクがすべて選択されます。

▼ SPARCstorage Array ディスクの状態のチェック方法 (DiskSuite ツール)

この作業は、SPARCstorage Array ディスクの状態 (起動または停止) を表示するために使用します。

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (266ページの「SPARCstorage Array をグラフィカルに操作するための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックして、「ディスク表示」ウィンドウを表示する。
3. キャンバスを見る。停止状態のディスクでは、ディスクの下に下向き矢印アイコンが表示される。

必要ならば、「コントローラ」ウィンドウから希望のコントローラを選択して、ディスクを含んでいるコントローラを表示させます。

注 - DiskSuite ツールを使用して、非 SPARCstorage Array ディスクを停止できます。しかし、「ディスク表示」ウィンドウは、SPARCstorage Array ディスクに対してだけ起動や停止の状態を示します。また、停止された非 SPARCstorage Array ディスク上で入出力を実行すると、そのディスクが自動的に起動されます。

次の作業

停止されていた SPARCstorage Array ディスクを起動するには、280ページの「ディスクの起動方法 (DiskSuite ツール)」を参照してください。

▼ SPARCstorage Array コントローラのファンとバッテリーの状態をチェックする方法 (DiskSuite ツール)

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (266ページの「SPARCstorage Array をグラフィカルに操作するための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックして、「ディスク表示」ウィンドウを表示する。
3. 各コントローラのそばに表示されるアイコンを調べる (必要ならば、「コントローラ」ウィンドウで選択して、希望のコントローラを表示する)。

DiskSuite ツールは、SPARCstorage Array LCD 画面に表示されるものと同じアイコンによって、SPARCstorage Array コントローラのファンとバッテリーの状態を表示します。

- スパナアイコン - オブジェクトには保守が必要。
- スパナアイコンとファンアイコン - ファンに障害が発生したか、内部温度が異常に上昇した。通常、SPARCstorage Array は自動的にオフラインになる。
- スパナアイコンとバッテリーアイコン - NVRAM のバッテリーが低下したか、取り付けてない。

注 - 「ディスク表示」ウィンドウには、非 SPARCstorage Array ディスクの障害も表示されます。これらのいずれかのディスクの下にスパナアイコンが表示された場合、ディスク障害が発生しています。

スパナアイコンとバッテリーアイコンが表示されたコントローラを、図 6-3 に示します。



図 6-3 SPARCstorage Array のスパナアイコンとバッテリーアイコン

注 - 1 つ以上の障害が同時に発生することがあります。その場合、複数のアイコンが表示されます。障害ログと同様、「コントローラ情報」ウィンドウにも障害に関する情報が記録されます。

次の作業

SPARCstorage Array での障害の解決については、第 7 章を参照してください。

▼ SPARCstorage Array コントローラの World Wide Name を表示する方法 (DiskSuite ツール)

1. 前提条件 (265 ページの「システムを管理するための前提条件」) を満たし、予備情報 (266 ページの「SPARCstorage Array をグラフィカルに操作するための予備情報」) を読んでいることを確認する。

2. 「ディスク表示」をクリックして、「ディスク表示」ウィンドウを表示する。

3. キャンバス上のコントローラを調べる (必要ならば、「コントローラ」ウィンドウで選択して希望のコントローラを表示する)。

World Wide Name の最後の 4 桁は、各 SPARCstorage Array コントローラの下に表示されます。

4. 12 桁の数字を全部表示するには、コントローラを選択し、そのポップアップメニューを表示し、「情報」を選択する。

World Wide Name の全体が、シリアル番号として「情報」ウィンドウに表示されます。

SPARCstorage Array の管理

この節では、次のような DiskSuite ツールの使用法について説明します。

- NVRAM の有効化と無効化
- NVRAM のフラッシュとパーージ

- ディスクの予約と解放
- ディスクの起動と停止

DiskSuite ツールは、SPARCstorage Array を操作するための統合化されたソリューションを提供するため、SPARCstorage Array ハードウェアの物理的なレイアウトを表示したり、さまざまな管理作業を実行することができます。これらの作業を実行するには、DiskSuite ツールの「ディスク表示」ウィンドウを使用します。DiskSuite は、このウィンドウに等価なコマンド行ユーティリティを提供していません。SPARCstorage Array をコマンド行から管理するには、`ssaadm(1M)` コマンドを使用します。



注意 - この機能の使用には注意が必要です。SPARCstorage Array を管理するための強力な方法が提供されます。最低限の対策として、DiskSuite ツールをこのようにして使用する前に、データの最新のバックアップをとってください。

NVRAM を有効または無効にするための予備情報

- 高速書き込みを設定できる : コントローラレベルでは、SPARCstorage Array 内のすべてのドライブに高速書き込みが設定されます。トレイレベルでは、1つのトレイ内のすべてのドライブに影響を与えます。ドライブレベルでは、個々のドライブに高速書き込みを設定します。
- 高速書き込みが有効になると、電源を再投入しても、SPARCstorage Array の設定の一部として保存されます。
- NVRAM バッテリーの容量が低下したり、残容量がなくなったり、故障した場合、コントローラ上では高速書き込みが無効になるため、メニューも無効になります。
- 高速書き込みを有効にする前に、コントローラ、トレイ、ディスクへの入出力をすべて停止します。SPARCstorage Array 内のディスク上に存在するメタデバイスへの入出力がすべて停止していることを確認します。特に、ディスクセットの所有権が維持されている間は暗黙の入出力ストリームが存在するため、すべてのディスクセットメタデバイスの所有権が解放されていることを確認します。ディスクセットの所有権の取得と解放については、214ページの「ディスクセットの操作」を参照してください。

▼ コントローラ、トレイ、ディスク上で NVRAM を有効にする方法 (DiskSuite ツール)

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (272ページの「NVRAM を有効または無効にするための予備情報」) を読んでいることを確認する。
2. 279ページの「ディスクの停止方法 (DiskSuite ツール)」を参照して、すべての入出力を停止する。
SPARCstorage Array 内のディスク上に存在するメタデバイスへの入出力を、すべて停止しておきます。特に、ディスクセットの所有権が維持されている間は暗黙の入出カストリームが存在するため、すべてのディスクセットメタデバイスの所有権が解放されていることを確認します。
3. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。
4. 「ディスク表示」ウィンドウでは、コントローラ、トレイ、またはディスクを選択する。
5. 「オブジェクト」メニューから「高速書き込み」->「有効」を選択する。
6. 「高速書き込み」ダイアログボックスが表示され、オブジェクトに対する入出力をすべて停止するよう指摘されます。
7. 「継続」をクリックする。
高速書き込みが有効になったことを示す確認が表示されます。ダイアログボックスで「OK」をクリックします。
8. 280ページの「ディスクの起動方法 (DiskSuite ツール)」を参照して、ディスクを再起動する。

▼ コントローラ、トレイ、ディスクに対する同期書き込み用の NVRAM を有効にする方法 (DiskSuite ツール)

この作業では、同期書き込み専用的高速書き込みを有効にします。すべての「書き込み」に対して高速書き込みを有効にするには、前の作業を使用してください。

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (272ページの「NVRAM を有効または無効にするための予備情報」) を読んでいることを確認する。

2. 279ページの「ディスクの停止方法 (DiskSuite ツール)」を参照して、すべての入出力を停止する。

SPARCstorage Array 内のディスク上に存在するメタデバイスへの入出力を、すべて停止しておきます。特に、ディスクセットの所有権が維持されている間は暗黙の入出力ストリームが存在するため、すべてのディスクセットメタデバイスの所有権が解放されていることを確認します。

3. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。
4. 「ディスク表示」ウィンドウでは、コントローラ、トレイ、またはディスクを選択する。
5. 「オブジェクト」メニューから「高速書き込み」を選択し、「同期」を選択する。
6. 「高速書き込み」ダイアログボックスが表示され、オブジェクトに対する入出力をすべて停止するよう指摘されます。
7. 「継続」をクリックする。
同期書き込み用の高速書き込みが有効になったことを示す確認メッセージが表示されます。「了解」をクリックします。
8. 280ページの「ディスクの起動方法 (DiskSuite ツール)」を参照して、ディスクを再起動する。

▼ コントローラ、トレイ、ディスク上で NVRAM を無効にする方法 (DiskSuite ツール)

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (272ページの「NVRAM を有効または無効にするための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。
3. 「ディスク表示」ウィンドウでは、コントローラ、トレイ、またはディスクを選択する。
4. 「オブジェクト」メニューから「高速書き込み」を選択し、「無効」を選択する。
5. 高速書き込みが無効にされたことを示す確認メッセージが表示されます。「了解」をクリックする。

NVRAM データをパージおよびフラッシュするための予備情報

- フラッシュオプションは、未処理の書き込みを NVRAM からディスクドライブにフラッシュします。
- データのフラッシュ中にエラーが発生した場合、そのデータをパージする必要があります。データのパージでは、NVRAM 内の未処理の書き込みを捨てます。
- 高速書き込みデータをパージすると、データを失う可能性があります。したがって、ドライブに障害が発生した場合にだけ、注意して実行してください。
- NVRAM バッテリーの容量が低下したり、残容量がなくなったり、故障した場合、NVRAM が機能しなくなり、そこにあったデータが失われるため、メニューも無効になります。

▼ 未処理の書き込みを NVRAM からフラッシュする方法 (DiskSuite ツール)

この作業では、選択したコントローラ (およびすべてのディスク)、トレイ (およびそのトレイ内のすべてのディスク)、または個々のディスクに対する未処理の書き込みを、NVRAM からディスクにすべてフラッシュします。

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (275ページの「NVRAM データをパージおよびフラッシュするための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。
3. 「ディスク表示」ウィンドウでは、コントローラ、トレイ、またはディスクを選択する。
4. 「オブジェクト」メニューから「**NVRAM** の同期処理」を選択する。
5. **NVRAM** が同期することを示す確認ダイアログボックスが表示されます。「了解」をクリックする。

▼ 高速書き込みデータを NVRAM からパージする方法 (DiskSuite ツール)

この作業は、ドライブが故障した場合など、SPARCstorage Array にアクセスできなくなった場合にだけ使用します。SPARCstorage Array 内のドライブにアクセスできる場合、276ページの「未処理の書き込みを NVRAM からフラッシュする方法 (DiskSuite ツール)」を参照してください。高速書き込みデータをパージすると、未処理の書き込みはすべて取り除かれます。

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (275ページの「NVRAM データをパージおよびフラッシュするための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。

3. 「ディスク表示」ウィンドウでは、コントローラ、トレイ、またはディスクを選択し、オブジェクトのポップアップメニューを表示する。
4. 「**NVRAM** をパージ」を選択する。**NVRAM** がパージされることを示す確認ダイアログボックスで「パージする」をクリックする。

ディスクを予約および解放するための予備情報

- SPARCstorage Array を複数のホストに接続したディスクセット構成では、1つのホストが個々のドライブを予約したり、アレイ内のすべてのドライブを予約したりすることが必要な場合もあります。ホストがドライブの予約を発行した場合、そのホストがドライブを解放するまでは、他のホストはそれらのドライブを使用できません。
- 予約コマンドを発行するホスト以外のホスト上で DiskSuite ツールを実行した場合、「情報」ウィンドウにはそれらのディスクに対して「予約済み」状態が表示され、「ディスク表示」キャンバスのディスクの下には「ロック」アイコンが表示されます。しかし、予約を発行したホストでは、ロックアイコンは表示されません。

注 - 現在のところ、この機能を使用できるのは、SPARCstorage Array ディスクを使用する SPARC システムだけです。

▼ ディスクをホスト専用予約する方法 (DiskSuite ツール)

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (277ページの「ディスクを予約および解放するための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。
3. コントローラ、トレイ、または **1** つのディスクを選択し、「オブジェクト」メニューから「ディスクの予約」を選択する。
4. ディスクの予約を確認するダイアログボックスが表示されます。「予約」をクリックする。

ディスクが予約されたという画像表示はありません。しかし、予約コマンドを発行するホスト以外のホスト上で DiskSuite ツールが実行された場合は、「情報」ウィンドウにはそれらのディスクに対して「予約済み」状態が表示され、「ディスク表示」キャンバスには「ロック」アイコンが表示されます。

▼ ホストによって予約されたディスクを解放する方法 (DiskSuite ツール)

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (277ページの「ディスクを予約および解放するための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。
3. コントローラ、トレイ、または 1 つのディスクを選択し、「オブジェクト」メニューから「ディスクの解放」を選択する。
4. ディスクが解放されたことを示す確認ダイアログボックスが表示されます。
ダイアログボックスで「了解」をクリックする。

ディスクを停止および起動するための予備情報

- DiskSuite では、SPARCstorage Array ディスクに加えて、非 SPARCstorage Array (SCSI) ディスクの起動と停止もサポートします。
- SPARCstorage Array ディスクが停止した場合、「ディスク表示」キャンバスのディスクの下に「下向き矢印」が表示されます。「ディスク情報」ウィンドウにもディスクの状態が表示される (現在「ディスク表示」ウィンドウは、非 SPARCstorage Array (SCSI) ディスクに対して「下向き矢印」を表示しません。非 SPARCstorage Array ディスク上で入出力が実行されると、自動的にそのディスクを起動します)。
- 通常、ディスクの停止は、ハードウェアの保守を行うときに一連の手順の一部として実行されます。通常、ディスクを単独で停止することはありません。
- 1 つの SPARCstorage Array トレイを停止することによって、SPARCstorage Array 全体の電源を切断することなく、ディスクを交換できます。

- 現在、ディスクの起動や停止を行うときに、DiskSuite はイベント通知を生成しません。つまり、同じホスト上で DiskSuite ツールの複数のインスタンスを実行している場合、1つの DiskSuite ツールでディスクの起動や停止を行なっても他のインスタンスは更新されません。このような障害が発生する可能性を回避するには、ディスクの起動や停止を行うときに、同じホスト上で DiskSuite ツールの複数のインスタンスを実行しないでください。



注意 - `ssaadm(1M)` コマンドを使用してディスクの起動や停止を行なった場合、DiskSuite 用のイベント通知は生成されません。したがって、ディスクの起動や停止を行う場合には、DiskSuite ツールと `ssaadm(1M)` を一緒に使用しないでください。一緒に使用した場合、DiskSuite ツールによるディスクの状態表示が正しく行われないことがあります。ディスクの起動と停止には、必ずいずれかの方法を使用してください。

▼ ディスクの停止方法 (DiskSuite ツール)

ディスクを停止するための前提条件 (SPARCstorage Array)

SPARCstorage Array ディスクはトレイの中にあるため、1つのディスクを停止する場合でも、適切にシャットダウンされることを保証するために、トレイ内のすべてのディスクを停止してください。これには次の操作が伴います。

- トレイ内のディスク上のファイルシステムをマウント解除
- トレイ内のディスクにアクセスするデータベースプロセスをすべて停止
- トレイ内のディスクにアクセスするその他のプロセスもすべて停止
- ディスクを停止

注 - ディスクを停止すると、`sync_cache` が NVRAM 内の未処理の書き込みをフラッシュします。

1. **前提条件** (265ページの「システムを管理するための前提条件」) を満たし、**予備情報** (278ページの「ディスクを停止および起動するための予備情報」) を読んでいることを確認する。

2. 影響を受けるコントローラ、ディスク、またはトレイ上の他の **DiskSuite** オブジェクトをチェックする。

「ディスク表示」ウィンドウでは、コントローラを表示します。コントローラの「情報」ウィンドウを表示し、「デバイスマップ」を選択して、どのオブジェク

ト (メタデバイス、ホットスペア、状態データベースの複製) が物理デバイスを使用しているのかを調べます。

3. ディスクに対するすべてのアクセスを停止する。
たとえば、ディスクに関連付けられたファイルシステムをマウント解除したり、ディスクを使用するデータベースに対する入出力をすべて停止します。
4. 「ディスク表示」ウィンドウでは、コントローラ、トレイ、または 1 つのディスクを選択し、「オブジェクト」メニューから「ディスクの利用停止」を選択する。
5. 「ディスクの利用停止」ダイアログボックスが表示され、ディスクに対する入出力をすべて停止するよう指摘されます。「継続」をクリックする。
6. ディスクが停止すると、システムがそのことを通知します。「了解」をクリックする。
SPARCstorage Array の場合、停止した各ディスクの下には、ディスクが停止したことを示す下向き矢印が表示されます。SCSI ディスクを停止した場合は、下向き矢印は表示されません。

▼ ディスクの起動方法 (DiskSuite ツール)

1. 前提条件 (265ページの「システムを管理するための前提条件」) を満たし、予備情報 (278ページの「ディスクを停止および起動するための予備情報」) を読んでいることを確認する。
2. 「ディスク表示」をクリックする。
「ディスク表示」ウィンドウが表示されます。停止した SPARCstorage Array ディスクは、キャンバスではディスクの下に下向き矢印と一緒に表示されます。
3. コントローラを選択して、それに接続されているすべてのディスクを起動するか、または **Control** キーを押しながらクリックして個々のディスクを選択する。
Control キーを押しながらクリックすると、複数のコントローラを選択することができます。
4. 「オブジェクト」メニューから「ディスクの利用開始」を選択する。

5. ディスクが起動されると、システムがそのことを通知します。「了解」をクリックする。

SPARCstorage Array ディスクの場合、ディスクの下にあった下向き矢印が消えて、起動したことを示します。SCSI ディスクの場合は、表示されません。

パフォーマンスの監視とグラフ化

この節では、次の方法について説明します。

- メタデバイス統計情報の表示
- メタデバイス統計情報のグラフ化
- grapher オプションの設定

パフォーマンスの監視と解析

DiskSuite ツールを使用すれば、メタデバイスや物理ディスクのパフォーマンス監視を簡単に実行できます。パフォーマンス監視の目標は、データの可用性を維持しながら少しずつ変更を加えることによって、システムパフォーマンスを最大限に高めることです。パフォーマンス監視では、簡単に再構成したり頻繁に再構成したりできない、特定の構成を使用しているものと想定します。

一方、パフォーマンス解析は、「最高のパフォーマンス」を達成できるまで、さまざまな構成を試すことによって、システムスループットを最大限に高めたり、応答時間を最小限に抑えることを目指します。パフォーマンス解析では、ユーザーはいろいろな構成を設定でき、さまざまなデータを収集できることを想定します。このため、方法論的に異なる上に、デバイス構成、ファイルシステムの管理、データ収集、テスト負荷の生成などの業務をカバーする一連のツール群が必要となります。

DiskSuite ツールに含まれるパフォーマンス監視機能は、DiskSuite の日常的な処理を通じてホットスポットやボトルネックに関連するパフォーマンス障害を検出するのに最も適しています。この機能を使用すれば、一般的なパフォーマンス傾向を監視したり、急激な変化を探したり、構成のさまざまな部分から収集されたデータを比較することができます。たとえば、RAID5 メタデバイスを監視中に、他のカラムよりもはるかに入出力の多いカラム (スライス) を発見することがあります。そ

のカラムを他のディスクに移動することによって、メタデバイス全体の入出力の均衡を調整することができます。

パフォーマンスの監視とグラフ化のための予備情報

- DiskSuite ツールは、kstat(3K) インタフェースによって提供された raw 統計データのサブセットを表示します (詳細は、kstat(3K) と iostat(1M) のマニュアルページを参照してください)。
- DiskSuite ツールでは kstat インタフェースを使用するため、スライスレベルではなく、ディスクレベルの統計情報だけを提示できます。
- オブジェクトを、DiskSuite ツールの任意のウィンドウから「統計情報グラフ」ウィンドウにドラッグ&ドロップできます。このオブジェクトには、「メタデバイスエディタ」キャンパス、オブジェクトリスト、「スライスブラウザ」ウィンドウ、「ディスク表示」ウィンドウが含まれます。
- スライスを「統計情報グラフ」ウィンドウにドラッグすると、ディスク全体を効果的にドラッグすることになります。スライスレベルの統計情報を得ることはできません。
- 状態データベースの複製、ホットスペア集合、および個々のスライスは、統計情報を表示できません。
- 統計情報を使用してみるための良い方法は、まず、メタデバイスの負荷値を見ることです。次に、読み取りや書き込みが原因であるかどうかを調査します。その後、メタデバイスを構成するディスク自身を追加し、各ディスクの負荷を表示します。
- 一般に、長期間監視するには、raw スループットデータと負荷値が最も便利な値です。

▼ デバイス統計情報の表示方法 (DiskSuite ツール)

この作業は、統計変数の現在の値の「スナップショット」を示すデバイス統計シートを表示するために使用します。

1. 「メタデバイスエディタ」ウィンドウまたは「ディスク表示」ウィンドウから、オブジェクトを選択する。
MetaDB オブジェクト、ホットスペア集合、スライスは、統計情報を表示できません。

2. どちらかのウィンドウの「オブジェクト」メニューから「統計情報」を選択し、オブジェクトの「デバイス統計情報」ウィンドウを表示する。
「デバイス統計情報」ウィンドウについては、オンラインヘルプを参照してください。
3. 最新の統計情報を表示するには、「更新」をクリックする。

注 - DiskSuite ツールは、「ディスク表示」ウィンドウで選択したディスクごとに、別個の「デバイス統計情報」ウィンドウを表示します。したがって、多数のディスクを備えたトレイやコントローラを選択した場合、選択したディスクと同じ数の「デバイス統計情報」ウィンドウが表示されます。

▼ デバイス統計情報をグラフ化する方法 (DiskSuite ツール)

この作業は、瞬間の統計情報をデバイス用のグラフという形式で表示するために使用します。

1. 「メタデバイスエディタ」ウィンドウの「ブラウズ」メニューから「統計情報グラフ」を選択して、「統計情報グラフ」ウィンドウを表示する。
2. オブジェクトを、「メタデバイスエディタ」ウィンドウまたは「ディスク表示」ウィンドウから「統計情報グラフ」ウィンドウにドラッグする。
「統計情報グラフ」ウィンドウには、オブジェクトのグラフ用の空白領域と統計情報が表示されます。MetaDB オブジェクト、ホットスペア集合、スライスは、統計情報をグラフ化できません。
3. グラフ化する統計情報を選択する。
「統計情報グラフ」ウィンドウには 2 種類の y 軸 (2 本の垂直スケール) があります。したがって、次のいずれかから 2 種類のグラフを作ることができます。
処理 / 秒 - 毎秒のディスク入出力操作の数であり、読み取り、書き込み、または合計として測定されます。
KB / 秒 - ディスク入出力のキロバイト数であり、読み取り、書き込み、または合計として測定されます。

負荷 – ディスク負荷は、ビジー時間のパーセンテージまたは待ち行列として測定され、処理が行なわれるまでの平均待ち時間となります。

4. 「統計情報グラフ」ウィンドウの使用を終了したら、「すべてのグラフ」メニューから「キャンバスから移動」を選択する。
デバイスを片付けない場合は、「統計情報グラフ」ウィンドウは、次に開かれたときにデバイスを自動的に組み込みます。

▼ 「統計情報グラフ」ウィンドウにデバイスを追加する方法 (DiskSuite ツール)

1. 「メタデバイスエディタ」ウィンドウの「ブラウズ」メニューから「統計情報グラフ」を選択し、「統計情報グラフ」ウィンドウを表示する。
2. オブジェクトを、「メタデバイスエディタ」ウィンドウまたは「ディスク表示」ウィンドウから「統計情報グラフ」ウィンドウにドラッグする。
「統計情報グラフ」ウィンドウには、オブジェクトのグラフ用の空白領域と統計情報が表示されます。MetaDB オブジェクト、ホットスペア集合、スライスは、統計情報をグラフ化できません。
3. 283ページの「デバイス統計情報をグラフ化する方法 (DiskSuite ツール)」の283ページの手順3を参照する。

▼ 「統計情報グラフ」ウィンドウからデバイスを除去する方法 (DiskSuite ツール)

1. 「メタデバイスエディタ」ウィンドウの「ブラウズ」メニューから「統計情報グラフ」を選択して、「統計情報グラフ」ウィンドウを表示する。
2. 1つのデバイスを除去するには、それを選択して「キャンバスから移動」ボタンをクリックする。
「統計情報グラフ」ウィンドウは、リストからデバイスをクリアします。
3. すべてのデバイスを除去するには、「統計情報グラフ」ウィンドウの「すべてのグラフ」メニューから「キャンバスから移動」を選択する。

SunNet Manager と DiskSuite の統合

DiskSuite と SunNet Manager を連携することによって使いやすい構成にできます。DiskSuite には、システムコンソールログを監視するデーモンを介して非同期 SNMP トラップを使用することにより、ドライバメッセージを転送する機能があります。これにより、SunNet Manager のコンソールは、障害やエラーなどのメッセージを検出できます。SunNet Manager の内部で DiskSuite メッセージを受信すると、DiskSuite ツールを起動できます。

SunNet Manager の詳細は、『*Site/SunNet/Domain Manager Administration Guide*』を参照してください。

▼ SunNet Manager を有効にして DiskSuite ツールを起動する方法 (SunNet Manager)

SunNet Manager は、DiskSuite のコンソールメッセージを自動的に受信します。SunNet Manager を有効にして DiskSuite ツールを起動するには、次の手順に従ってください。

1. **SunNet Manager** を実行した状態で、「ファイル」メニューから「ファイルに保存」を選択し、さらに「ファイル」メニューから「終了」を選択する。
2. `/opt/SUNWconn/snm/struct/elements.schema` ファイルを編集して、`elementCommand` 構造をサーバーに合わせて変更する。
エントリは、次のようになります。

```
instance elementCommand (  
...  
  
(component.sun-server ``DiskSuite Tool...``  
  ``/usr/sbin/metatool``)  
...  
)
```

このエントリを「コンポーネント」カテゴリに追加します。

3. `-i` オプションを付けて **SunNet Manager** を再起動する。

-i オプションは、変更された `elements.schema` ファイルを読み取り、SunNet Manager データベースを初期化し直します。

▼ SunNet Manager から DiskSuite ツールを起動する方法 (SunNet Manager)

この作業は、SunNet Manager から DiskSuite ツールを起動するために使用します。なお、285ページの「SunNet Manager を有効にして DiskSuite ツールを起動する方法 (SunNet Manager)」の手順に従っているものと想定します。

1. **SunNet Manager** キャンバスの内部で、**(DiskSuite をインストールした) ホストオブジェクト**を選択する。
2. ホストのポップアップメニューを表示し、「ツール」を選択する。
3. 「ツール」メニューから「DiskSuite Tool...」を選択する。
選択されたホスト用の DiskSuite ツールが表示されます。

SNMP 警告と DiskSuite の統合

DiskSuite は、SNMP メッセージを受信できる任意のネットワーク管理コンソールに対して、SNMP トラップデータ (警告) を送信できます。この操作は、インストールするよう選択した別個の DiskSuite デーモンである `mdlogd` を通じて行われます。構成ファイルを通じて、トラップ変数だけではなく、探すべきメッセージも指定します。

DiskSuite の SNMP デーモンを使用するには、オプションの `SUNWmdnr` および `SUNWmdnu` パッケージをインストールしなければなりません。『*Solstice DiskSuite 4.2.1 ご使用にあたって*』を参照してください。

▼ DiskSuite の SNMP サポートの構成方法 (コマンド行)

この作業では、DiskSuite の SNMP パッケージである SUNWmdnr および SUNWmdnu がインストールされているものと想定します。

DiskSuite を SNMP のサポート用に構成する手順を次に示します。

- SNMP デーモン mdlogd を設定する
- SNMP トラップ構成ファイル /etc/lvm/mdlogd.cf を編集する

詳細については、mdlogd(4) のマニュアルページを参照してください。

SUNWmdnr および SUNWmdnu パッケージをインストールした後でシステムをリブートすると、次のメッセージが表示されます。

```
Starting mdlogd ...  
  
/etc/lvm/mdlogd.cf: no configuration information
```

DiskSuite がこのメッセージを表示する理由は、mdlogd.cf ファイルがまだ構成されていないためです。

1. mdlogd デーモンがブート時に自動的にロードされるよう設定されていることを確認する。
パッケージを追加したとき、インストール後のスクリプトでこのための処理を行います。
2. **SNMP** トラップ構成ファイルである mdlogd.cf の編集と保存を行う。
 - a. ENTERPRISE = の行を、デーモンを実行するホストが所属する企業の **SNMP** 識別子に変更する。
 - b. OBJECTID = の行を、デーモンを実行するホストの **SNMP** 識別子に変更する。
 - c. デーモンが探す SubStrings を設定する。
これらは、DiskSuite によって生成されたメッセージです。
 - d. trap destination の記述例のところの **SNMP** トラップ、ポート、プロトコルを、受信するホストの名前に変更する。

ポート 162 とプロトコル udp は、SNMP トラップのデフォルトであるため、おそらく常にこのポートとプロトコルを使用することになります。

- e. Generic SNMP Trap # には、6 を設定したままにする。
- f. Specific Trap # には、独自のコーディング方式を使用できる。
たとえば、低優先度のメッセージには番号 1 を使用し、中優先度の場合は番号 2、高優先度の場合は 3 を使用します。

例 - mdlogd.cf ファイル

```
##ident "@(#)mdlogd.cf 1.1 96/02/15"
# DiskSuite SNMP Trap configuration file.
...
ENTERPRISE = 1.3.6.1.4.1.42
OBJECTID = 1.3.6.1.4.1.860
#
# SubString Trap Destination      SNMP Trap #   Specific Trap #
#          (host:port:protocol)   0 < n <= 6   0 < n
"NOTICE: md:" "spin:162:udp"      6             1
"WARNING: md:" "spin:162:udp"      6             2
```

この例では、/dev/console に書き込まれた DiskSuite エラーの SNMP トラップを、spin という名前のホストに振り分けます。

エラーメッセージ「WARNING: md: d6: /dev/dsk/c3t3d0s7 needs maintenance」によって生成されたトラップを、SunNet Manager のコンソールで受信したときの状態を次に示します。

```
Wed Feb 21 15:40:41 1996 [ spin ] : Trap:

sequence=2
receive-time=Wed Feb 21 15:40:41 1996
version=0
community=public
enterprise=Sun Microsystems
source-time=00:00:00.00
trap-type=enterprise specific trap: 2

1.3.6.1.4.1.860 = Feb 21 15:40:41 1996 spin WARNING:
md: d6: /dev/dsk/c3t3d0s7 needs maintenance
```


ストレージマネージャと DiskSuite の統合

DiskSuite とストレージマネージャを連携することによって使いやすい構成にできます。ストレージマネージャのアプリケーションには、ディスクマネージャとファイルシステムマネージャという 2 つのツールが組み込まれています。これらのツールを使用すれば、サーバーのディスク構成とファイルシステムを管理できます。たとえば、DiskSuite ツール内にメタデバイスを作成し、ファイルシステムマネージャにドラッグして、メタデバイス上に UFS ファイルシステムを作成することができます。

DiskSuite ツールの「ツール」メニューからストレージマネージャを起動できるように、DiskSuite を設定できます。

ストレージマネージャの詳細は、付録 A を参照してください。

▼ DiskSuite を有効にしてストレージマネージャを起動する方法 (コマンド行)

この作業では、ストレージマネージャのパッケージがインストールされているものと想定しています。詳細は、『*Solstice DiskSuite 4.2.1* ご使用にあたって』を参照してください。

DiskSuite を有効にしてストレージマネージャのツールを起動することは、`/usr/lib/lvm/metatool-toolsmenu` ファイルの設定を伴います。詳細は、`metatool-toolsmenu(4)` のマニュアルページを参照してください。

提供される `metatool-toolsmenu` ファイルでは、ファイルシステムマネージャとディスクマネージャを使用するよう、すでに 2 つの行が設定されています。この 2 つの行のコメントを解除 (先頭の # 記号を除去) して、ファイルを保存します。

例 - `metatool-toolsmenu` ファイル

次に `metatool-toolsmenu` ファイルの例を示します。このファイルには、DiskSuite ツールからファイルシステムマネージャとディスクマネージャを起動できるようにするための 2 つのエントリがあります。

```
#
# metatool 'Tools' menu registry file...
#
# Entries are of the form:
```

(続く)

```

#
# :0:<name>:<command>:
#
# 0 is a format specifier. '0' is the only valid specifier for metatool.
#
# <name> becomes a menu item1.5 in the Metadevice Editor and
# DiskView Tools menu
# <command> is passed to system() when item is chosen from the menu.
#
# ':' is a field delimiter, one of: '+' '|' ':' '^'
#
# sample entries:
+0+File System Manager...+/opt/SUNWadm/2.2/bin/stomgr -F+
:0:Disk Manager...:/opt/SUNWadm/2.2/bin/stomgr -D:
#
#

```

▼ ファイルシステムマネージャとディスクマネージャの起動方法 (DiskSuite ツール)

この作業では、前の作業の説明に従って、metatool-toolsmenu ファイルが設定されているものと想定します。

1. DiskSuite ツールを起動する。

```
# metatool &
```

現在、DiskSuite ツールのメニューオプションには「ツール」項目があり、ファイルシステムマネージャとディスクマネージャを選択できます。

2. 「ツール」メニューから適切な項目を選択して、ファイルシステムマネージャかディスクマネージャを起動する。

システムのトラブルシューティング

この章では、DiskSuite の問題を解決する方法について説明します。

特定の作業に対する手順ごとの指示を記載した節に直接進むためには、次の目次を使用してください。

- 293ページの「md.cf ファイルを使用して DiskSuite 設定を回復する方法」
- 295ページの「デフォルトのメタデバイス数を増やす方法 (コマンド行)」
- 296ページの「デフォルトディスクセットの数を増やす方法 (コマンド行)」
- 297ページの「大きな状態データベースの複製を追加する方法 (コマンド行)」
- 298ページの「メタデバイス内のスライスエラーのチェックを自動化する方法 (コマンド行)」
- 301ページの「不適切な /etc/vfstab エントリからの回復方法 (コマンド行)」
- 304ページの「状態データベースの複製の不足からの回復方法 (コマンド行)」
- 307ページの「ブートデバイス障害からの回復方法 (コマンド行)」
- 312ページの「代替ブートデバイスへのパスを記録する方法 (コマンド行)」
- 313ページの「SPARC: 代替デバイスからのブート方法 (コマンド行)」
- 314ページの「x86: 代替デバイスからのブート方法 (コマンド行)」
- 315ページの「障害の発生した SCSI ディスクの交換方法 (コマンド行)」
- 321ページの「ミラー内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール)」
- 327ページの「RAID5 メタデバイス内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール)」

- 327ページの「SPARCstorage Array トレイの除去方法 (コマンド行)」
- 328ページの「SPARCstorage Array トレイの交換方法」
- 329ページの「SPARCstorage Array の電源断からの回復方法 (コマンド行)」
- 331ページの「ホスト間で SPARCstorage Array ディスクを移動する方法 (コマンド行)」
- 333ページの「SPARCstorage Array ディスクをブートプロセスの初期段階で使用可能にする方法」

システムのトラブルシューティングの概要

この章では、DiskSuite に関するいくつかのトラブル、およびその適切な解決法について説明します。テーマを包括的に取り上げるのではなく、共通のシナリオと回復手順を提示することを目的としています。

システムのトラブルシューティングのための前提条件

この節に含まれる手順の前提条件を次に示します。

- ルート権限をもつ。
- 現在の時点で、すべてのデータをバックアップしている。

トラブルシューティングを行う時の注意

トラブルシューティングを行う前に、次の情報を調べてください。

- `/etc/vfstab` ファイルの内容
- DiskSuite ツール、または `metadb(1M)` コマンドと `metastat(1M)` コマンドの出力から得られる、状態データベースの複製、メタデバイス、ホットスペアの状態
- `prtvtoc(1M)` コマンドやストレージマネージャ (Solaris システム)、または `fdisk` コマンド (x86 システム) から得られる、ディスクパーティションに関する情報
- Solaris のバージョン
- Solaris のパッチ

- DiskSuite のパッチ

DiskSuite 設定の回復

/etc/lvm/md.cf ファイルは、ローカルディスクセット用の DiskSuite 設定のバックアップファイルです。設定を変更するたびに、md.cf ファイルは自動的に更新されます (ホットスペアリングを除く)。md.cf ファイルを直接編集しないでください。

システムがメタデバイスの状態データベース内に保存されていた情報を失った場合、その一方でメタデバイスの作成や変更が行われていない限り、md.cf ファイルを使用して、DiskSuite 設定を回復することができます。

注 - md.cf ファイルは、アクティブなホットスペアに関する情報を保持しません。したがって、DiskSuite の設定が失われたときにホットスペアが使用されていた場合、ホットスペアを使用しているメタデバイスは、おそらく破壊されます。

▼ md.cf ファイルを使用して DiskSuite 設定を回復する方法



注意 - この作業を行うのは、DiskSuite の設定が完全に失われた場合だけです。

1. 状態データベースの複製を再作成する。
状態データベースの複製の作成方法については、第 1 章を参照してください。
2. /etc/lvm/md.tab ファイルのバックアップコピーを作成する。
3. md.cf ファイルから md.tab ファイルに情報をコピーする。
4. 新しい md.tab ファイルを編集して、次のように設定する。
 - すべてのミラーを 1 面のミラーにします。ミラーのサブミラーが同じサイズでない場合は、この 1 面のミラーには最も小さいサブミラーを使用します。さもなければ、データが失われる可能性があります。

- デバイスの再初期化を防止するため、RAID5 メタデバイスを `-k` オプションで再作成します (このオプションの詳細は、`metainit(1M)` のマニュアルページを参照してください)。

5. `metainit(1M)` コマンドを実行して、`md.tab` ファイルのエントリの構文をチェックする。

```
# metainit -n -a
```

6. `md.tab` ファイルのエントリの構文が正しいことを確認してから、`metainit(1M)` コマンドを実行して、`md.tab` ファイルからホットスペア集合とメタデバイスを再作成する。

```
# metainit -a
```

7. `metattach(1M)` コマンドを実行して、1面のミラーを多面のミラーとする。
8. メタデバイス上のデータの妥当性をチェックする。

DiskSuite のデフォルトの変更

DiskSuite 設定のデフォルトでは、1034 ブロックのサイズをもつ 128 個のメタデバイスと状態データベースの複製があります。ディスクセットのデフォルト数は 4 です。必要ならば、これらの値はすべて変更可能です。この節ではそのための作業を説明します。

メタデバイスの予備情報

- システムに対するメタデバイスのデフォルト数は 128 です。最大 1024 までこの値を増やすことができます。
- 多数のメタデバイスを追加する場合、(DiskSuite ツールやコマンド行ユーティリティを使用して) メタデバイスを管理するときに、システムパフォーマンスが若

干低下し始めることがあります。多数のメタデバイスがあっても、通常のシステム操作には影響を与えません。

- メタデバイスの数を増やして、一定の数値範囲内でデバイスタイプをパーティション分割するための大きな名前空間を獲得しても、作成したデバイスの数が 128 未満である場合、パフォーマンスの低下は発生しません。この場合、状態データベースの複製をさらに追加する必要はありません。

▼ デフォルトのメタデバイス数を増やす方法 (コマンド行)

この作業では、メタデバイスの数をデフォルト値の 128 から増やす方法について説明します。



注意 - デバイスの数を減らした場合、元の数と新しい数の間に存在するメタデバイスは使用できなくなり、データが失われる可能性があります。「md: d20: not configurable, check /kernel/drv/md.conf」などのメッセージを受け取った場合、この作業で説明するように md.conf ファイルを編集する必要があります。

1. **前提条件** (292ページの「システムのトラブルシューティングのための前提条件」) と **予備情報** (294ページの「メタデバイスの予備情報」) をチェックしてから、/kernel/drv/mdoconf ファイルを編集する。
2. nmd フィールドの値を変更する。
1024 までの値を設定できます。
3. 変更内容を保存する。
4. 再構成するためにリポートを実行して、メタデバイス名を構築する。

```
# boot -r
```

例 - md.conf ファイル

次に、256 個のメタデバイス用に設定された md.conf ファイルの例を示します。

```
#  
#ident "@(#)md.conf 1.7 94/04/04 SMI"  
#  
# Copyright (c) 1992, 1993, 1994 by Sun Microsystems, Inc.
```

```
#
name="md" parent="pseudo" nmd=256 md_nsets=4;
```

ディスクセットのための予備情報

システムに対するディスクセットのデフォルト数は4です。この値を32まで増やすことができます。md_nsetsにはローカルセットが含まれるため、共有されるディスクセットの数は、常にmd_nsetsの値より1だけ小さな値です。

▼ デフォルトディスクセットの数を増やす方法 (コマンド行)

この作業では、ディスクセットの数をデフォルト値の4から増やす方法について説明します。



注意 - デバイスの数を減らした場合、元の数と新しい数の間に存在するディスクセットに影響がある可能性があります。

1. 292ページの「システムのトラブルシューティングのための前提条件」の前提条件をチェックしてから、/kernel/drv/md.conf ファイルを編集する。
2. md_nsets フィールドの値を変更する。
32までの値を設定できます。
3. 変更内容を保存する。
4. 再構成するためにリブートを実行して、メタデバイス名を構築する。

```
# boot -r
```

例 - md.conf ファイル

次に、5つのディスクセット用に設定されたmd.confファイルの例を示します。md_nsetsの値は6であり、5つのディスクセットと1つのローカルディスクセットで構成されます。


```
#
#ident "@(#)md.conf 1.7 94/04/04 SMI"
#
# Copyright (c) 1992, 1993, 1994 by Sun Microsystems, Inc.
#
name="md" parent="pseudo" nmd=255 md_nsets=6;
```

状態データベースの複製のための予備情報

- 多数のメタデバイスを作成した場合、状態データベースの複製のサイズは相対的に小さくなり、必要な情報をすべて収めることができなくなります。この場合、-1 オプション付きの `metadb(1M)` コマンドを使用して、大きな状態データベースの複製の追加してから、小さな状態データベースの複製を除去します。
- 一般的に、デフォルトのメタデバイス数を2倍にしたら、状態データベースの複製のサイズも2倍にします。

▼ 大きな状態データベースの複製を追加する方法 (コマンド行)

前提条件 (292ページの「システムのトラブルシューティングのための前提条件」) をチェックして予備情報 (297ページの「状態データベースの複製のための予備情報」) を読んでから、`metadb` コマンドを使用して大きな状態データベースの複製を追加し、古い小さな状態データベースの複製を削除します。詳細は、`metadb(1M)` のマニュアルページを参照してください。

例 - 大きな状態データベースの複製の追加

```
# metadb -a -l 2068 c1t0d0s3 c1t1d0s3 c2t0d0s3 c2t1d0s3
# metadb -d c1t0d0s7 c1t1d0s7 c2t0d0s7 c2t1d0s7
```

最初の `metadb` コマンドでは、-l 2068 オプション (2068 ブロック) でサイズを指定された、状態データベースの複製を追加します。これは、デフォルトの複製サイズである 1034 ブロックの2倍です。2番目の `metadb` コマンドでは、小さな状態データベースの複製をシステムから除去します。

エラーのチェック

DiskSuite に障害 (スライスレベルでの物理エラーによって、メタデバイスに書き込みできないなど) が発生すると、メタデバイスの状態を、たとえば「Maintenance (保守状態)」に変更します。しかし、DiskSuite ツールで継続的に調査していたり metastat(1M) を実行している場合を除き、このような状態変化をタイムリーに見ることはできません。

DiskSuite のエラーを自動的にチェックする方法が 2 つあります。

- SNMP トラップを使用する (286ページの「SNMP 警告と DiskSuite の統合」を参照)
- スクリプトを使用して、継続的にエラーをチェックする (DiskSuite エラーのチェックに使用できるスクリプトについては、次の節以降を参照)

▼ メタデバイス内のスライスエラーのチェックを自動化する方法 (コマンド行)

メタデバイス内の不良スライスを連続して自動的にチェックする 1 つの方法は、cron によって呼び出されるスクリプトを記述する方法です。次にその例を示します。

```
#
#ident "@(#)metacheck.sh 1.3 96/06/21 SMI"
#
# Copyright (c) 1992, 1993, 1994, 1995, 1996 by Sun Microsystems, Inc.
#
#
# DiskSuite コマンド
#
MDBIN=/usr/sbin
METADB=${MDBIN}/metadb
METAHS=${MDBIN}/metahs
METASTAT=${MDBIN}/metastat
#
# システムコマンド
#
AWK=/usr/bin/awk
DATE=/usr/bin/date
MAILX=/usr/bin/mailx
RM=/usr/bin/rm
#
```

(続く)

```

# 初期化
#
eval=0
date=`${DATE} '+%a %b %e %Y'`
SDSTMP=/tmp/sdscheck.${}$
${RM} -f ${SDSTMP}

MAILTO=${*:-"root"} # default to root, or use arg list

#
# 複製の障害をチェック。フラグ内の大文字はエラーを示す。
#
dbtrouble=`${METADB} | tail +2 | \
  ${AWK} '{ fl = substr($0,1,20); if (fl ~ /[A-Z]/) print $0 }'`
if [ "${dbtrouble}" ]; then
  echo "" >>${SDSTMP}
  echo "SDS replica problem report for ${date}" >>${SDSTMP}
  echo "" >>${SDSTMP}
  echo "Database replicas are not active:" >>${SDSTMP}
  echo "" >>${SDSTMP}
  ${METADB} -i >>${SDSTMP}
  eval=1
fi

#
# メタデバイスの状態をチェック。状態が正常でない場合、何かが発生。
#
mdtrouble=`${METASTAT} | \
  ${AWK} '/State:/ { if ( $2 != "Okay" ) print $0 }'`
if [ "${mdtrouble}" ]; then
  echo "" >>${SDSTMP}
  echo "SDS metadevice problem report for ${date}" >>${SDSTMP}
  echo "" >>${SDSTMP}
  echo "Metadevices are not Okay:" >>${SDSTMP}
  echo "" >>${SDSTMP}
  ${METASTAT} >>${SDSTMP}
  eval=1
fi

#
# ホットスペアが使用されているかどうかをチェック。
#
hstrouble=`${METAHS} -i | \
  ${AWK} '/blocks/ { if ( $2 != "Available" ) print $0 }'`
if [ "${hstrouble}" ]; then
  echo "" >>${SDSTMP}
  echo "SDS Hot spares in use ${date}" >>${SDSTMP}
  echo "" >>${SDSTMP}
  echo "Hot spares in usage:" >>${SDSTMP}
  echo "" >>${SDSTMP}
  ${METAHS} -i >>${SDSTMP}
  eval=1
fi

```

(続く)

```
#
# 何かのエラーが発生した場合、ルート、またはコマンド行で指定された相手に
# レポートをメールする。
#
if [ ${eval} -ne 0 ]; then
    ${MAILX} -s "SDS problems ${date}" ${MAILTO} <${SDSTMP}
    ${RM} -f ${SDSTMP}
fi
exit ${eval}
```

このようにスクリプトを呼び出す方法については、cron(1M) のマニュアルページを参照してください。

注 - このスクリプトは、DiskSuite のエラーチェックを自動化するための出発点として利用できます。このスクリプトは、使用する構成に合わせて変更する必要があります。

ブート障害

DiskSuite を使用すればルート (/)、swap、/usr をミラー化できるため、ハードウェアのエラーやオペレータのミスによって、システムのブート時に特殊な障害が発生することがあります。この節で説明する作業は、このような潜在的な障害に対する解決法です。

これらの障害と適切な解決法について表 7-1 に示します。

表 7-1 DiskSuite の一般的なブート障害

システムがブートしない理由	参照先
/etc/vfstab ファイルの情報に誤りがある。	301ページの「不適切な /etc/vfstab エントリからの回復方法 (コマンド行)」
状態データベースの複製が不足している。	304ページの「状態データベースの複製の不足からの回復方法 (コマンド行)」

表 7-1 DiskSuite の一般的なブート障害 続く

システムがブートしない理由	参照先
ブートデバイス (ディスク) が故障した。	307ページの「ブートデバイス障害からの回復方法 (コマンド行)」
ブートミラーが故障した。	313ページの「SPARC: 代替デバイスからのブート方法 (コマンド行)」または 314ページの「x86: 代替デバイスからのブート方法 (コマンド行)」

ブート障害のための予備情報

- エラーにより、メタデバイスドライバがメタデバイスをオフラインにした場合は、障害の発生したディスク上のファイルシステムをすべてマウント解除します。各ディスクスライスは独立しているため、1つのディスクに複数のファイルシステムがマウントされていることがあります。メタディスクドライバに障害が発生した場合は、同じディスク上の他のスライスにも、まもなく障害が発生する可能性があります。ディスクスライスに直接マウントされたファイルシステムには、メタディスクドライバのエラー処理という保護機能がないため、このようなファイルシステムをマウントしたままで放置すると、システムのクラッシュによってデータを失う危険性があります。
- サブミラーを無効にしたりオフラインにした状態で実行する時間を最小限に抑えます。再同期やオンラインバックアップの処理中、ミラー化による保護は不完全になります。

▼ 不適切な /etc/vfstab エントリからの回復方法 (コマンド行)

たとえば、ルート (/) をミラー化するときなど、/etc/vfstab ファイル内に誤ったエントリを作成した場合、システムは、最初は適切にブーティングしているように見えても、障害が発生します。このような場合、シングルユーザーモードで /etc/vfstab を編集する必要があります。

不適切な /etc/vfstab ファイルのエントリから回復するための手順を次に示します。

- システムをシングルユーザーモードにブートする
- ミラーメタデバイス上で fsck(1M) を実行する
- ファイルシステムを読み書きモードで再マウントする
- オプション: ルート (/) ミラー用の metaroot(1M) コマンドを実行する
- /etc/vfstab ファイルがファイルシステムに記述されているメタデバイスを正しく参照することを確認する
- リブートする

例 - ルート (/) ミラーの回復

次の例では、ルート (/) は 2 面のミラー d0 でミラー化されます。/etc/vfstab 内のルート (/) エントリは、どういうわけかファイルシステムの元のスライスに戻りました。しかし、/etc/system 内の情報は、まだミラー d0 からのブートを示しています。通常考えられる理由としては、/etc/system と /etc/vfstab の保守に metaroot(1M) コマンドが使用されなかったか、または /etc/vfstab の古いコピーが戻されたことが挙げられます。

誤った /etc/vfstab ファイルは、次のようになります。

#device	device	mount	FS	fsck	mount	mount
#to mount	to fsck	point	type	pass	at boot	options
#						
/dev/dsk/c0t3d0s0	/dev/rdisk/c0t3d0s0	/	ufs	1	no	-
/dev/dsk/c0t3d0s1	-	-	swap	-	no	-
/dev/dsk/c0t3d0s6	/dev/rdisk/c0t3d0s6	/usr	ufs	2	no	-
#						
/proc	-	/proc	proc	-	no	-
fd	-	/dev/fd	fd	-	no	-
swap	-	/tmp	tmpfs	-	yes	-

エラーのため、マシンがブートされると自動的にシングルユーザーモードとなります。

```
ok boot
...
SunOS Release 5.5 Version Generic [UNIX(R) System V Release 4.0]
Copyright (c) 1983-1995, Sun Microsystems, Inc.
configuring network interfaces: le0.
Hostname: antero
```

続き

```
mount: /dev/dsk/c0t3d0s0 is not this fstype.
setmnt: Cannot open /etc/mnttab for writing

INIT: Cannot create /var/adm/utmp or /var/adm/utmpx

INIT: failed write of utmpx entry: " "

INIT: failed write of utmpx entry: " "

INIT: SINGLE USER MODE

Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance): <パスワードを入力>
```

この時点で、ルート (/) と /usr は読み取り専用でマウントされます。次の手順に従ってください。

1. ルート (/) ミラー上で **fsck(1M)** を実行する。

注 - ルートには正しいメタデバイスを使用するよう注意してください。

```
# fsck /dev/md/rdisk/d0
** /dev/md/rdisk/d0
** Currently 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
2274 files, 11815 used, 10302 free (158 frags, 1268 blocks,
0.7% fragmentation)
```

2. /etc/vfstab ファイルを編集できるよう、ルート (/) を読み書きモードでマウントする。

```
# mount -o rw,remount /dev/md/dsk/d0 /
mount: warning: cannot lock temp file </etc/.mnt.lock>
```

3. metaroot(1M) コマンドを実行する。

```
# metaroot d0
```

これは /etc/system と /etc/vfstab のファイルを編集して、ルート (/) ファイルシステムが現在メタデバイス d0 上にあることを指定します。

4. /etc/vfstab ファイルに正しいメタデバイスエントリが収められていることを確認する。

/etc/vfstab ファイル内のルート (/) エントリが次のようになっていると、ファイルシステムがミラーを正しく参照します。

#device	device	mount	FS	fscck	mount	mount
#to mount	to fsck	point	type	pass	at boot	options
#						
/dev/md/dsk/d0	/dev/md/rdisk/d0	/	ufs	1	no	-
/dev/dsk/c0t3d0s1	-	-	swap	-	no	-
/dev/dsk/c0t3d0s6	/dev/rdisk/c0t3d0s6	/usr	ufs	2	no	-
#						
/proc	-	/proc	proc	-	no	-
fd	-	/dev/fd	fd	-	no	-
swap	-	/tmp	tmpfs	-	yes	-

5. リブートする。

システムは通常の動作に復帰します。

▼ 状態データベースの複製の不足からの回復方法 (コマンド行)

たとえば、ドライブの障害など、何らかの理由によって状態データベースの複製が規定数に満たない場合、システムはリブートできません。DiskSuite の用語では、これを状態データベースが「無効」になったと表現します。ここでは、その回復方法について説明します。

この作業の手順を次に示します。

- 無効な状態データベースの複製を削除してリブートする
- 故障したディスクを修復する
- 状態データベースの複製を追加して戻す

例 - 無効な状態データベースの複製からの回復

次の例では、2つの複製を含むディスクが不良となりました。システムには正常な複製が2つしか残されておらず、システムはリブートできません。

1. マシンをブートして、どの状態データベースの複製が障害を受けているのかを判定する。

```
ok boot
...
Hostname: demo
metainit: demo: stale databases

Insufficient metadevice database replicas located.

Use metadb to delete databases which are broken.
Ignore any "Read-only file system" error messages.
Reboot the system when finished to reload the metadevice
database.
After reboot, repair any broken database replicas which were
deleted.

Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance): <パスワードを入力>
Entering System Maintenance Mode

SunOS Release 5.5 Version Generic [UNIX(R) System V Release 4.0]
```

2. **metadb(1M)** コマンドを使用してメタデバイスの状態データベースを調べ、状態データベースの複製のうち、使用できないものを判定する。

```
# metadb -i
flags      first blk      block count      /dev/dsk/c0t3d0s3
a m p lu   16             1034
a p l      1050           1034              /dev/dsk/c0t3d0s3
M p        unknown       unknown           /dev/dsk/c1t2d0s3
M p        unknown       unknown           /dev/dsk/c1t2d0s3
```

(続く)

続き

```
...
```

システムは、障害の発生したディスクに含まれるスライス /dev/dsk/c1t2d0s3 上では、もう状態データベースの複製を検出できません。metadb コマンドは、このスライス上の複製に対して、マスターブロックに障害があるというフラグを立てます。

3. -d オプション付きの metadb(1M) コマンドを使用して、不良ディスク上の状態データベースの複製を削除する。

この時点では、ルート (/) ファイルシステムは読み取り専用である。mddb.cf のエラーメッセージは無視できる。

```
# metadb -d -f c1t2d0s3
metadb: demo: /etc/lvm/mddb.cf.new: Read-only file
system
```

4. 複製が削除されたことを確認する。

```
# metadb -i
  flags          first blk      block count
  a m p l u      16            1034        /dev/dsk/c0t3d0s3
  a p l           1050          1034        /dev/dsk/c0t3d0s3
```

5. リポートする。
6. 交換用のディスクを用意できたらシステムを停止し、故障したディスクを交換し、もう一度システムをリポートする。format(1M) コマンドまたは fmthard(1M) コマンドを使用して、ディスクを故障の前と同じようにパーティション分割する。

```
# halt
...
ok boot
...
# format /dev/rdisk/c1t2d0s0
...
```

7. `metadb(1M)` コマンドを使用して、状態データベースの複製を追加して戻し、状態データベースの複製が正常であることを確認する。

```
# metadb -a -c 2 c1t2d0s3
# metadb
  flags          first blk      block count      dev/dsk/c0t3d0s3
  a m p luo      16             1034              dev/dsk/c0t3d0s3
  a   p luo      1050           1034              dev/dsk/c0t3d0s3
  a   u   16      1034              dev/dsk/c1t2d0s3
  a   u   1050    1034              dev/dsk/c1t2d0s3
```

-c 2 オプション付きの `metadb` コマンドは、同じスライスに状態データベースの複製を 2 つ追加します。

▼ ブートデバイス障害からの回復方法 (コマンド行)

ルート (/) ミラーがあり、ブートデバイスが故障した場合、代替のブートデバイスを設定する必要があります。

この作業での手順を次に示します。

- 代替ルート (/) サブミラーからのブート
- エラーの発生した状態データベースの複製とメタデバイスの判定
- 故障ディスクの修復
- メタデバイスの状態データベースとメタデバイスを元の状態に復元

例 - ブートデバイス障害からの回復

次の例では、6 つの状態データベースの複製のうち 2 つと、ルート (/)、`swap`、`/usr` の各サブミラーを含んだブートデバイスに障害が発生しました。

まず、ブートデバイスに障害が発生すると、次のようなメッセージが表示されます。このメッセージは、アーキテクチャによって異なります。

```
Rebooting with command:
Boot device: /iommu/sbus/dma@f,81000/esp@f,80000/
sd@3,0 File and args: kadb
kadb: kernel/unix
The selected SCSI device is not responding
Can't open boot device
...
```

このメッセージが表示されたら、デバイスをメモしてから、次の手順に従います。

1. 他のルート (/) サブミラーからブートする。

この例では、6つの状態データベースの複製のうち、エラーであるのは2つだけなので、まだブートが可能です。そうでない場合、シングルユーザーモードで無効な状態データベースの複製を削除する必要があります。この作業については、304ページの「状態データベースの複製の不足からの回復方法(コマンド行)」を参照してください。

ルート (/) ファイルシステム用のミラーを作成する場合、その作業の一部として、代替ブートデバイスを記録する必要があります。この例では、disk2 がその代替ブートデバイスです。

```
ok boot disk2
...
SunOS Release 5.5 Version Generic [UNIX(R) System V Release 4.0]
Copyright (c) 1983-1995, Sun Microsystems, Inc.

Hostname: demo
...
demo console login: root
Password: <パスワードを入力>
Last login: Wed Dec 16 13:15:42 on console
SunOS Release 5.1 Version Generic [UNIX(R) System V Release 4.0]
...
```

2. metadb(1M) コマンドを使用して、2つの状態データベースの複製が障害を受けていることを確認する。

```
# metadb
flags      first blk      block count      /dev/dsk/c0t3d0s3
M p        unknown    unknown
M p        unknown    unknown
a m p lu0  16         1034             /dev/dsk/c0t2d0s3
a p lu0    1050      1034             /dev/dsk/c0t2d0s3
a p lu0    16         1034             /dev/dsk/c0t1d0s3
a p lu0    1050      1034             /dev/dsk/c0t1d0s3
```

現在システムは、障害の発生したディスクの一部であるスライス /dev/dsk/c0t3d0s3 上の状態データベースの複製を検出することができません。

- 3. metastat(1M) コマンドを使用して、ルート (/)、swap、/usr の片方のミラーの半分が障害を受けていることを確認する。

```
# metastat
d0: Mirror
  Submirror 0: d10
    State: Needs maintenance
  Submirror 1: d20
    State: Okay
...

d10: Submirror of d0
  State: Needs maintenance
  Invoke: "metareplace d0 /dev/dsk/c0t3d0s0 <new device>"
  Size: 47628 blocks
  Stripe 0:
  Device          Start Block  Dbase State          Hot Spare
  /dev/dsk/c0t3d0s0      0      No   Maintenance

d20: Submirror of d0
  State: Okay
  Size: 47628 blocks
  Stripe 0:
  Device          Start Block  Dbase State          Hot Spare
  /dev/dsk/c0t2d0s0      0      No   Okay

d1: Mirror
  Submirror 0: d11
    State: Needs maintenance
  Submirror 1: d21
    State: Okay
...

d11: Submirror of d1
  State: Needs maintenance
  Invoke: "metareplace d1 /dev/dsk/c0t3d0s1 <new device>"
```

(続く)

```

Size: 69660 blocks
Stripe 0:
Device                Start Block  Dbase State      Hot Spare
/dev/dsk/c0t3d0s1      0           No   Maintenance

d21: Submirror of d1
State: Okay
Size: 69660 blocks
Stripe 0:
Device                Start Block  Dbase State      Hot Spare
/dev/dsk/c0t2d0s1      0           No   Okay

d2: Mirror
Submirror 0: d12
State: Needs maintenance
Submirror 1: d22
State: Okay
...

d2: Mirror
Submirror 0: d12
State: Needs maintenance
Submirror 1: d22
State: Okay
...

d12: Submirror of d2
State: Needs maintenance
Invoke: "metareplace d2 /dev/dsk/c0t3d0s6 <new device>"
Size: 286740 blocks
Stripe 0:
Device                Start Block  Dbase State      Hot Spare
/dev/dsk/c0t3d0s6      0           No   Maintenance

d22: Submirror of d2
State: Okay
Size: 286740 blocks
Stripe 0:
Device                Start Block  Dbase State      Hot Spare
/dev/dsk/c0t2d0s6      0           No   Okay

```

この例では、metastat は、次のサブミラーに保守が必要なことを示します。

- サブミラー d10、デバイス c0t3d0s0
- サブミラー d11、デバイス c0t3d0s1

- サブミラー d12、デバイス c0t3d0s6

4. システムを停止し、ディスクを修復し、format(1M) コマンドまたは fmthard(1M) コマンドを使用して、ディスクを障害を受ける前と同じようにパーティション分割する。

```
# halt
...
Halted
...
ok boot
...
# format /dev/rdisk/c0t3d0s0
```

5. リブートする。

なお、ルート (/) ミラーの残った片方からリブートしなければなりません。ミラーを作成する際に、代替ブートデバイスを記録しておいてください。

```
# halt
...
ok boot disk2
```

6. metadb(1M) コマンドを使用し、故障した状態データベースの複製を削除してから、追加して戻す。

```
# metadb
  flags          first blk      block count
M      p          unknown      unknown      /dev/dsk/c0t3d0s3
M      p          unknown      unknown      /dev/dsk/c0t3d0s3
a m    p  luo     16             1034         /dev/dsk/c0t2d0s3
a      p  luo     1050          1034         /dev/dsk/c0t2d0s3
a      p  luo     16             1034         /dev/dsk/c0t1d0s3
a      p  luo     1050          1034         /dev/dsk/c0t1d0s3
# metadb -d c0t3d0s3
# metadb -c 2 -a c0t3d0s3
# metadb
  flags          first blk      block count
a m    p  luo     16             1034         /dev/dsk/c0t2d0s3
a      p  luo     1050          1034         /dev/dsk/c0t2d0s3
```

(続く)

続き

```
a p luo 16 1034 /dev/dsk/c0t1d0s3
a p luo 1050 1034 /dev/dsk/c0t1d0s3
a u 16 1034 /dev/dsk/c0t3d0s3
a u 1050 1034 /dev/dsk/c0t3d0s3
```

7. **metareplace(1M)** コマンドを使用して、サブミラーを再び有効にする。

```
# metareplace -e d0 c0t3d0s0
Device /dev/dsk/c0t3d0s0 is enabled

# metareplace -e d1 c0t3d0s1
Device /dev/dsk/c0t3d0s1 is enabled

# metareplace -e d2 c0t3d0s6
Device /dev/dsk/c0t3d0s6 is enabled
```

しばらくすると、再同期が終了します。これで元のデバイスからブートできるようになります。

▼ 代替ブートデバイスへのパスを記録する方法 (コマンド行)

ルート (/) をミラー化するとき、後で一次デバイスが障害を受けた場合に、代替ブートデバイスへのパスが必要になることがあります。

例 - SPARC: 代替ブートデバイスパスの記録

この例では、ルート (/) ミラーに 2 番目のサブミラーとして接続されているスライス上で `ls -l` コマンドを使用することによって、代替ブートデバイスへのパスを調べます。


```
# ls -l /dev/rdisk/c1t3d0s0
lrwxrwxrwx 1 root root 55 Mar 5 12:54 /dev/rdisk/c1t3d0s0 -> ../.
./devices/sbus@1,f8000000/esp@1,200000/sd@3,0:a
```

ここでは、/devices ディレクトリに続く文字列を記録します。
/sbus@1,f8000000/esp@1,200000/sd@3,0:a

一部の新しい Sun ハードウェアでは、/devices ディレクトリ名を sd@ から disk@ に変更する必要があります。

OpenBoot PROM 付きのシステムを使用している DiskSuite ユーザーは、OpenBoot の nvalias コマンドを使用して、二次ルートミラー用の「バックアップルート」デバイス別名を定義できます。たとえば、

```
ok nvalias backup_root /sbus@1,f8000000/esp@1,200000/sd@3,0:a
```

一次ルートディスクに障害が発生した場合、次のように入力します。

```
ok boot backup_root
```

例 - x86: 代替ブートデバイスパスの記録

この例では、ルート (/) ミラーに 2 番目のサブミラーとして接続されているスライス上で ls -l コマンドを使用することによって、代替ブートデバイスへのパスを調べます。

```
# ls -l /dev/rdisk/c1t0d0s0
lrwxrwxrwx 1 root root 55 Mar 5 12:54 /dev/rdisk/c1t0d0s0 -> ../.
./devices/eisa/eha@1000,0/cmdk@1,0:a
```

ここでは、/devices ディレクトリに続く文字列を記録します。
/eisa/eha@1000,0/cmdk@1,0:a

▼ SPARC: 代替デバイスからのブート方法 (コマンド行)

代替ブートデバイスから SPARC システムをブートするには、次のように入力します。

```
# boot <代替ブートデバイス>
```

代替ブートデバイスの確認方法については、312ページの「代替ブートデバイスへのパスを記録する方法 (コマンド行)」を参照してください。

▼ x86: 代替デバイスからのブート方法 (コマンド行)

この作業は、代替ブートデバイスから x86 システムをブートするために使用します。

1. マルチデバイスブート (**MDB**) フロッピーディスクからシステムをブートする。
しばらくすると、次のような画面が表示されます。

```
Solaris/x86 Multiple Device Boot Menu
Code   Device   Vendor   Model/Desc           Rev
=====
10     DISK     COMPAQ   C2244                 0BC4
11     DISK     SEAGATE  ST11200N SUN1.05     8808
12     DISK     MAXTOR   LXT-213S SUN0207     4.24
13     CD       SONY     CD-ROM CDU-8812      3.0a
14     NET      SMC/WD   I/O=300 IRQ=5
80     DISK     First IDE drive (Drive C:)
81     DISK     Second IDE drive (Drive D:)

Enter the boot device code:
```

2. 画面に表示された選択項目の中から、代替ディスクのコードを入力する。
次の画面が表示されます。

```
Solaris 2.4 for x86           Secondary Boot Subsystem,vsn 2.11

<<<Current Boot Parameters>>>
Boot path:/eisa/eha@1000,0/cmdk@0,0:a
Boot args:/kernel/unix

Type b[file-name] [boot-flags] <ENTER>   to boot with options
or  i<ENTER>                               to enter boot interpreter
or  <ENTER>                                to boot with defaults

<<<timeout in 5 seconds>>>
```

3. i を入力してインタプリタを選択する。
4. 次のコマンドを入力する。

```
>setprop boot-path /eisa/eha@1000,0/cmdk@1,0:a
>^D
```

Control-D を入力して、インタプリタを終了します。

SCSI ディスクの交換

この節では、DiskSuite 環境において SPARCstorage Array に含まれない SCSI ディスクの交換方法について説明します。

▼ 障害の発生した SCSI ディスクの交換方法 (コマンド行)

SPARCstorage Array に含まれていない SCSI ディスクを交換するための手順を次に示します。

- 交換の必要なディスクを特定
- 障害の発生したディスク上にあるメタデバイスの状態データベースの複製を削除
- 障害の発生したディスク上で「使用可能」とマークされたホットスペアの削除
- 障害の発生したディスク上のスライスを使用するサブミラーの探索と切断
- システムを停止し、シングルユーザーモードへブート
- ディスクを物理的に交換
- 新しいディスクのパーティション再分割
- 削除された、メタデバイスの状態データベースの複製を追加
- 障害の発生したスライスの使用方法に応じて、次のいずれかを実行

単純スライスの場合：通常の回復手順を使用

ストライプや連結の場合：メタデバイス全体を `newfs` し、バックアップから復元

ミラーの場合：切断されたサブミラーを再接続 RAID5 メタデバイスの場合：影響を受けるスライスを再同期 (有効化)

トランスメタデバイスの場合: fsck(1M) を実行

■ 削除されたホットスペアをホットスペア集合に追加

1. /var/adm/messages と metastat の出力を調査することによって、交換すべきディスクを特定する。

2. 障害の発生したディスクに置かれた可能性のある、ローカルメタデバイスの状態データベースの複製を探索する。

複製を見つけ出すには、metadb コマンドを使用します。

障害の発生したディスク上に置かれた複製のエラーが通知されることがあります。この例では、c0t1d0 は障害デバイスです。

```
# metadb
  flags      first blk      block count      /dev/dsk/c0t0d0s4
a m        u          16                1034
a          u        1050                1034
a          u        2084                1034
W pc      lu0       16                1034
W pc      lu0       1050               1034
W pc      lu0       2084                1034
```

上の出力は、各ローカルディスク (c0t0d0 と c0t1d0) のスライス 4 にある、3つの状態データベースの複製を示します。c0t1d0s4 スライスのフラグフィールドにある W は、デバイスに書き込みエラーがあることを示します。c0t0d0s4 スライス上の 3 つの複製は、まだ正常です。



注意 - 不良の状態データベースの複製を削除して、正常な複製が 3 つ以下になった場合、操作を継続する前に、状態データベースの複製を追加してください。これによって、システムが正しくリブートされます。

3. 複製が存在するスライス名と複製の数を記録してから、状態データベースの複製を削除する。

複製の数を調べるには、316 ページの手順 2 の metadb 出力におけるスライスの出現回数をカウントします。この例では、c0t1d0s4 上に存在する 3 つの状態データベースの複製が削除されます。

```
# metadb -d c0t1d0s4
```

4. 障害の発生したディスク上のスライスを使用するサブミラーを探索して、切断する。

metastat コマンドは、影響を受けるミラーを表示できます。この例では、1つのサブミラー d10 が c0t1d0s4 も使用しています。ミラーは d20 です。

```
# metadetach d20 d10
d20: submirror d10 is detached
```

5. 障害の発生したディスク上のホットスペアを削除する。

```
# metahs -d hsp000 c0t1d0s6
hsp000: Hotspare is deleted
```

6. システムを停止し、シングルユーザーモードにブートする。

```
# halt
...
ok boot -s
...
```

7. 障害の発生したディスクを物理的に交換する。

8. 新しいディスクのパーティションを再分割する。

format(1M) コマンドまたは fmthard(1M) コマンドを使用して、障害の発生したディスクと同じスライス情報でディスクをパーティションに分割します。

9. 316ページの手順3で複製を削除した場合、適切なスライスに同じ数の複製を追加する。

この例では、/dev/dsk/c0t1d0s4 が使用されます。

```
# metadb -a c 3 c0t1d0s4
```

10. 次の表を使用し、ディスクの使用法に応じて、次に行うべき操作を決定する。

表 7-2 SCSIディスク交換のための決定表

デバイスの種類	操作内容
スライス	通常のリカバリ手順を使用します。
ミラー化解除された スト ライブや連結	ファイルシステムにストライプ / 連結が使用されている 場合、 <code>newfs(1M)</code> を実行し、ファイルシステムをマウン トしてから、バックアップからデータを復元します。 <code>raw</code> デバイスを使用するアプリケーションとしてストライプ / 連結が使用されている場合、そのアプリケーションに は独自の回復手順が必要です。
ミラー (サブミラー)	<code>metattach(1M)</code> を実行して、切断されたサブミラーを再 接続します。
RAID5 メタデバイス	<code>metareplace(1M)</code> を実行して、スライスを再び有効に します。これによって再同期が開始されます。
トランスメタデバイス	<code>fsck(1M)</code> を実行して、トランスメタデバイスを修復しま す。

11. 削除されたホットスペアを交換し、適切なホットスペア集合に追加する。

```
# metahs -a hsp000 c0t0d0s6
hsp000: Hotspare is added
```

12. データの妥当性をチェックする。

すべてのメタデバイス上のユーザーデータとアプリケーションデータをチェック
します。アプリケーションレベルの整合性チェック機能を実行したり、その他の
方法でデータをチェックする必要があります。

SPARCstorage Array の操作

この節では、DiskSuite を使用して SPARCstorage Array (SSA) のトラブルシューティングを行う方法について説明します。この節には次のような作業があります。

- ミラー内で障害の発生したディスクを交換
- RAID5 メタデバイス内で障害の発生したディスクを交換
- トレイを除去
- トレイを交換
- コントローラを交換
- 電源断からの回復

インストール

SPARCstorage Array は、SPARCstorage Array CD に添付の『*SPARCstorage Array Software*』マニュアルに従ってインストールします。DiskSuite だけを使用する場合は、SPARCstorage Array のボリューム管理機能をインストールする必要はありません。

デバイスの命名規則

DiskSuite は、SPARCstorage Array ディスクに対して、他のディスクと同じようにアクセスしますが、重要な例外が 1 つあります。つまり、非 SPARCstorage Array ディスクの場合とはディスク名が異なります。

SPARCstorage Array 100 ディスクの命名規則を次に示します。

`c[0-n]t[0-5]d[0-4]s[0-7]`

この名前では、

- `c` は、SSA ユニットに接続されたコントローラを示す
- `t` は、SSA 内部の 6 つの SCSI 列の 1 つを示す
- `d` は、内部 SCSI 列上の 5 つのディスクの 1 つを示す
- `s` は、ディスクのスライス番号を示す

- t0 と t1 はトレイ 1 に含まれ、t2 と t3 はトレイ 2 に含まれ、t4 と t5 はトレイ 3 に含まれる

SPARCstorage Array 200 ディスクの命名規則を次に示します。

c[0-n]t[0-5]d[0-6]s[0-7]

この名前では、

- c は、SSA ユニットに接続されたコントローラを示す
- t は、SSA ユニット内部の 6 つのターゲット (トレイ) の 1 つを示す
- d は、内部 SCSI 列上の 7 つのディスクの 1 つを示す
- s は、ディスクスライス番号を示す

注 - 古いトレイでは 6 つまでのディスクを保持し、新しいトレイでは 7 つまで保持できます。

SSA100 と SSA200 との主な違いは、SSA100 では 1 つのトレイに 2 つのターゲットが配置されていますが、SSA200 ではターゲットごとに別のトレイが割り当てられているという点です。

SPARCstorage Array コンポーネントを交換するための予備情報

交換可能な SPARCstorage Array コンポーネントには、ディスク、ファントレイ、バッテリー、トレイ、電源装置、バックプレーン、コントローラ、光モジュール、ファイバチャネルケーブルがあります。

いくつかの SPARCstorage Array コンポーネントは、SPARCstorage Array の電源を切断することなく交換できますが、電源を切断する必要があるコンポーネントもあります。詳細については、SPARCstorage Array のマニュアルを参照してください。

電源を切断する必要がある SPARCstorage Array コンポーネントを、サービスを中断せずに交換するには、電源を切断する前に SPARCstorage Array 内のすべてのトレイに対して、トレイの除去に必要な手順を実行します。これには、サブミラーをオフラインに設定、ホットスペアをホットスペア集合から削除、状態データベースの複製をドライブから削除、トレイの停止が含まれます。

これらの準備を行ってから、SPARCstorage Array の電源を切断し、コンポーネントを交換することができます。

注 - SPARCstorage Array コントローラは、Solaris から特定される一意な World Wide Name を持っています。そのため、SPARCstorage Array コントローラの交換には特別な作業が適用されます。技術的な支援に関しては、ご購入先にご連絡ください。

▼ ミラー内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール)

DiskSuite 環境において SPARCstorage Array ディスクを交換する手順は、ディスク上のスライスの使用方法、およびディスクとシステムのケーブル接続方法によって大きく異なります。また、ディスクスライスがそのまま使用されるのか、DiskSuite によって使用されるのか、それともその両方なのかによっても異なります。

注 - この作業は SPARCstorage Array 100 に適用されます。SPARCstorage Array 200 でディスクを交換するための手順もよく似ています。

この作業での手順を次に示します。

- 交換の必要なディスクを特定し、その位置を調査
- 取り出すべきトレイ内で「使用可能」とマークされたホットスペアを削除
- 取り出すべきトレイ内のディスク上にある状態データベースの複製を削除
- 取り出すべきトレイ内のディスクを使用するサブミラーを探索
- 交換中のディスク上にスライスをもつサブミラーを切断
- トレイ内のディスクを使用する他のサブミラーをオフライン設定
- トレイ内のディスクをすべて停止
- トレイを除去してディスクを交換
- トレイ内のディスクがすべて起動することを確認
- 新しいディスクをパーティションに再分割
- トレイ内のサブミラーをオンラインに戻す
- トレイ内の切断されたサブミラーを接続
- 削除されたホットスペアを交換
- 削除されたホットスペアをホットスペア集合に追加

■ 削除されたメタデバイスの状態データベースの複製を追加

注 - サブミラーが「保守」状態にある場合、ホットスペアによって交換された場合、またはときどきエラーが発生している場合には、この作業を使用できます。

ディスクを探索して交換するには、次の手順を実行します。

1. **DiskSuite** ツールを使用してオブジェクトの「状態」フィールドを調べるか、または `metastat` と `/var/adm/messages` の出力を調査することによって、交換するディスクを特定する。

```
# metastat
...
d50:Submirror of d40
    State: Needs Maintenance
...
# tail -f /var/adm/messages
...
Jun 1 16:15:26 host1 unix: WARNING: /io-
unit@f,e1200000/sbi@0.0/SUNW,pln@a0000000,741022/ssd@3,4(ssd49):
Jun 1 16:15:26 host1 unix: Error for command 'write(I)' Err
Jun 1 16:15:27 host1 unix: or Level: Fatal
Jun 1 16:15:27 host1 unix: Requested Block 144004, Error Block: 715559
Jun 1 16:15:27 host1 unix: Sense Key: Media Error
Jun 1 16:15:27 host1 unix: Vendor 'CONNER':
Jun 1 16:15:27 host1 unix: ASC=0x10(ID CRC or ECC error),ASCQ=0x0,FRU=0x15
...
```

`metastat` コマンドは、サブミラーが「Needs Maintenance」状態にあることを明らかにします。`/var/adm/messages` ファイルは、エラーのあるディスクドライブを通知します。ディスクドライブを探索するには、次のように `ls` コマンドを使用して、シンボリックリンクの名前と `/var/adm/messages` の出力からの名前を照合します。

```
# ls -l /dev/rdisk/*
...
lrwxrwxrwx 1 root root 90 Mar 4 13:26 /dev/rdisk/c3t3d4s0 -
> ../../devices/io-
unit@f,e1200000/sbi@0.0/SUNW,pln@a0000000,741022/ssd@3,4(ssd49)
...
```

上の情報と `metastat` の出力にもとづいて、ドライブ `c3t3d4` を交換しなければならぬことが決まります。

2. **DiskSuite** ツールを使用して、影響を受けるトレイを判定する。

障害の発生したディスクが存在する SPARCstorage Array トレイを見つけるには、「ディスク表示」ウィンドウを使用します。

a. 「ディスク表示」をクリックして、「ディスク表示」ウィンドウを表示する。

b. 障害の発生したメタデバイス (この例は、ミラー) を、オブジェクトリストから「ディスク表示」ウィンドウにドラッグする。

「ディスク表示」ウィンドウでは、メタデバイスを構成する物理スライスに色を割り当てることによって、論理デバイスから物理デバイスへのマップを表示します。障害の発生したディスクを含むトレイは、一目で判断できます。

c. **ssaadm(1M)** コマンドを使用する。

```
host1# ssaadm display c3
      SPARCstorage Array Configuration
Controller path: /devices/io-
unit@f,e1200000/sbi@0.0/SUNW,soc@0,0/SUNW,pln@a0000000,741022:ctlr
      DEVICE STATUS
      TRAY1          TRAY2          TRAY3
Slot
1      Drive:0,0      Drive:2,0      Drive:4,0
2      Drive:0,1      Drive:2,1      Drive:4,1
3      Drive:0,2      Drive:2,2      Drive:4,2
4      Drive:0,3      Drive:2,3      Drive:4,3
5      Drive:0,4      Drive:2,4      Drive:4,4
6      Drive:1,0      Drive:3,0      Drive:5,0
7      Drive:1,1      Drive:3,1      Drive:5,1
8      Drive:1,2      Drive:3,2      Drive:5,2
9      Drive:1,3      Drive:3,3      Drive:5,3
10     Drive:1,4      Drive:3,4      Drive:5,4

      CONTROLLER STATUS
Vendor:   SUNW
Product ID: SSA100
Product Rev: 1.0
Firmware Rev: 2.3
Serial Num: 000000741022
Accumulate performance Statistics: Enabled
```

コントローラ (c3) に対する **ssaadm** の出力によって、中央トレイを取り出すとき、Drive 3,4 (c3t3d4) が一番近い位置にあることがわかります。

3. [オプション] ディスクセットがある場合、影響を受けるドライブを含むディスクセットを探索する。

次のコマンドでは、ドライブ c3t3d4 を探索します。logicalhost2 でコマンドを実行したときには何の出力も表示されませんが、logicalhost1 の場合は、名前が存在することが通知されたことに注目します。通知された出力の yes フィールドは、ディスクに状態データベースの複製が収められていることを示します。

```
host1# metaset -s logicalhost2 | grep c3t3d4
host1# metaset -s logicalhost1 | grep c3t3d4
c3t3d4 yes
```

注 - Solstice HA サーバーを使用している場合、2つの論理ホストの所有権を1つの Solstice HA サーバーに切り替える必要があります。詳細については、Solstice HA のマニュアルを参照してください。

4. 影響を受けるトレイ上の他の **DiskSuite** オブジェクトを判定する。

ディスクを交換するにはトレイを取り出す必要があるため、このプロセスにおいて影響を受ける他のオブジェクトを確認します。

 - a. **DiskSuite** ツールで「ディスク表示」ウィンドウを表示する。トレイを選択し、「オブジェクト」メニューから「デバイスマップ」を選択する。
「物理デバイスから論理デバイスへのマッピング」ウィンドウが表示されません。
 - b. ウィンドウに表示されるホットスペア、メタデバイス、状態データベースの複製など、影響を受けるオブジェクトをすべて記録する。

5. 影響を受けるトレイに他の **DiskSuite** オブジェクトを作成することによって、ディスク交換の準備を行う。
 - a. 状態が「使用可能」であり、しかも障害の発生したディスクと同じトレイにあるホットスペアをすべて削除する。
交換作業が終了したらホットスペアをホットスペア集合に追加して戻せるよう、ホットスペアについての情報をすべて記録します。
 - b. 取り出すべきトレイ内のディスク上にある状態データベースの複製を削除する。

これらの複製は326ページの手順 14 で交換しなければならないため、この情報を記録しておいてください。同じディスク上に複数の複製がある場合もあります。各スライスから削除された複製の数を記録しておきます。

- c. トレイ内に存在するスライスを使用しているサブミラーを探索する。
- d. 交換中のディスク上のスライスをもつサブミラーをすべて切断する。
- e. トレイ内にスライスをもつ他のサブミラーをすべてオフラインにする。
これにより、DiskSuite はトレイ内のサブミラースライスの使用を停止するため、ドライブを停止できます。
オブジェクトを除去するには、第5章を参照してください。サブミラーを切断してオフラインにするには、202ページの「ミラーの操作」を参照してください。

6. **SPARCstorage Array** トレイ内のディスクをすべて停止する。

279ページの「ディスクの停止方法 (DiskSuite ツール)」を参照してください。

注 - トレイ上の LED が点灯している間は、SPARCstorage Array トレイの除去を行うべきではありません。また、トレイが停止している間は、DiskSuite コマンドを実行しないでください。これを実行した場合、その副作用により、トレイ内のドライブの一部または全部が起動する可能性があります。

7. トレイを取り出し、不良ディスクを交換する。

ハードウェアの作業については、『*SPARCstorage Array Model 100 Series Service Manual*』および『*SPARCcluster High Availability Server Service Manual*』を参照してください。

8. **SPARCstorage Array** のトレイ内のディスクがすべて起動したことを確認する。

SPARCstorage Array トレイ内のディスクは、ハードウェアの交換作業に続いて、自動的に起動します。トレイが2分以内の自動起動に失敗した場合は、次のコマンドを使用してアクションを強制します。

```
# ssaadm start -t 2 c3
```

9. `format(1M)`、`fmthard(1M)`、またはストレージマネージャを使用して、新しいディスクをパーティションに再分割する。新しいディスクのパーティション分割は、交換されたディスクとまったく同じにする。
障害が発生する前に、ディスクフォーマット情報を保存することが望ましいです。
10. オフラインにされていたすべてのサブミラーを、オンラインに戻す。
202ページの「ミラーの操作」を参照してください。
サブミラーがオンラインに復帰すると、DiskSuite はすべてのサブミラーを自動的に再同期し、データを最新の状態にします。
11. 切断されていたサブミラーを接続する。
202ページの「ミラーの操作」を参照してください。
12. 326ページの手順 11 で接続されたサブミラー内で使用中のホットスペアを交換する。
サブミラーを切断前に、使用中のホットスペアを交換されたサブミラーがあった場合、このホットスペア交換は、サブミラーが再接続されてから有効となります。この手順によって、ホットスペアは「使用可能」状態に戻ります。
13. 削除されたホットスペアをすべて追加する。
14. トレイ上のディスクから削除された状態データベースの複製をすべて追加する。
状態データベースの複製を交換するには、以前に保存した情報を使用します。
15. [オプション] **Solstice HA** サーバーを使用する場合、各論理ホストをそのデフォルトマスターに切り替える。
Solstice HA のマニュアルを参照してください。
16. データの妥当性をチェックする。
すべてのメタデバイスで、ユーザーデータとアプリケーションデータをチェックします。アプリケーションレベルの整合性チェック機能を実行したり、その他の方法でデータをチェックする必要があります。

▼ RAID5 メタデバイス内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール)

RAID5 メタデバイスをオンライン修復用に設定する場合、3つの最小幅の RAID5 のスライスを使用する必要があります。これは RAID5 にとって最適の構成ではありませんが、冗長データのオーバーヘッドという観点からは、ミラー化よりも若干負担の少ない方法です。各 RAID5 メタデバイスの3つのスライスは、それぞれ別のトレイに置いてください。SPARCstorage Array 内のすべてのディスクがこの方法で (または上述のミラーと組み合わせて) 構成された場合、どのデータに対するアクセスも失うことなく、障害の発生したディスクを含むトレイを除去できます。



注意 - 障害の発生したドライブを含むトレイ内の非複製ディスクを使用するアプリケーションは、最初に中断または終了してください。

1. 321ページの「ミラー内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール)」の322ページの手順 1 ~ 326ページの手順 9 を参照する。

これから障害の発生したディスクとトレイを探索し、影響を受ける他の DiskSuite オブジェクトを探索し、ディスクの交換準備を行なって、交換し、ドライブをパーティションに再分割します。

2. `metareplace -e` コマンドを使用して、トレイ内の新しいドライブを有効にする。
3. 321ページの「ミラー内で障害の発生した SPARCstorage Array ディスクを交換する方法 (DiskSuite ツール)」の326ページの手順 12 ~326ページの手順 16 を参照する。

▼ SPARCstorage Array トレイの除去方法 (コマンド行)

SPARCstorage Array トレイを除去する前に、すべての入出力を停止し、トレイ内のすべてのドライブを停止します。入出力要求が行われると、ドライブは自動的に起動します。したがって、ドライブを停止する前に、すべての入出力を停止することが必要です。

1. **DiskSuite** の入出力処理を停止する。

サブミラーをオフラインにする、`metaoffline(1M)` コマンドを参照します。トレイ上のサブミラーがオフラインにされると、ミラーは 3 面ミラーの場合を除いて、1 面ミラーだけで稼働します (つまり、データの冗長性が失われます)。サブミラーがオンラインに復帰すると、自動的に再同期が開始されます。

注 - サブミラーを含むドライブを交換する場合、`metadetach(1M)` コマンドを使用して、サブミラーを切断します。

2. `metastat(1M)` コマンドを使用して、除去されるトレイ上のスライスを含むサブミラーをすべて特定する。また、`metadb(1M)` コマンドを使用して、トレイ上の複製を特定する。`metahs(1M)` コマンドを使用して、使用可能なホットスペアデバイスと対応するサブミラーも特定する必要がある。

影響を受けるサブミラーをすべてオフラインにした状態で、トレイへの入出力が停止されます。

3. 279ページの「ディスクの停止方法 (DiskSuite ツール)」を参照する。

DiskSuite ツールか `ssaadm` コマンドを使用して、トレイを停止します。トレイのロックランプが消灯すると、そのトレイを除去して必要な作業を実行できます。

▼ SPARCstorage Array トレイの交換方法

SPARCstorage Array トレイに関する作業が終了したら、シャーシ内のトレイを交換します。ディスクは自動的に起動します。

しかし、ディスクが起動に失敗した場合は、DiskSuite ツール (または `ssaadm` コマンド) を使用して、トレイ全体を手作業で起動することができます。SPARCstorage Array 内の起動ドライブ間には、若干の遅延 (数秒) があります。

ディスクが起動したら、オフラインに設定されていたサブミラーをすべてオンラインにする必要があります。サブミラーをオンラインにすると、最適化された再同期動作によって、サブミラーは自動的に最新の状態となります。最適化された再同期では、サブミラーがオフラインであったときに変更されたディスク領域だけがコピーされます。これは一般に、サブミラー容量全体のごく一部にすぎません。状態データベースの複製もすべて交換して、ホットスペアに追加することが必要です。

注・metaoffline ではなく、metadetach(1M) を使用してサブミラーを切断した場合、サブミラー全体を再同期しなければなりません。この場合、一般的にデータ 1G バイトあたりおよそ 10 分かかります。

▼ SPARCstorage Array の電源断からの回復方法 (コマンド行)

1 つの SPARCstorage Array で電源断が発生した場合、次の現象が起こります。

- DiskSuite オブジェクトに対して入出力操作を行うと、エラーが生成される。
- エラーは、ドライブレベルではなく、スライスレベルで通知される。
- ディスクに入出力操作が行われるまで、エラーは通知されない。
- 影響を受けたデバイスがホットスペアを割り当てた場合、ホットスペアアクティビティが開始されることがある。

132ページの「DiskSuite オブジェクトの状態チェック」で説明したように、metastat(1M) コマンドを使用して、これらのイベントの構成を監視しなければなりません。

電源が回復したら、次の操作を実行する必要があります。

- エラーの発生したデバイスを metastat で特定する
 - エラーの発生したサブミラーや RAID5 メタデバイスを有効にする
 - 影響を受けた状態データベースの複製を削除して再作成する
1. 電源が回復したら、metastat コマンドを使用して、エラーの発生したデバイスを特定する。

```
# metastat
...
d10: Trans
    State: Okay
    Size: 11423440 blocks
    Master Device: d20
    Logging Device: d15

d20: Mirror
    Submirror 0: d30
    State: Needs maintenance
    Submirror 1: d40
```

(続く)

```

State: Okay
...
d30: Submirror of d20
State: Needs maintenance
...

```

2. `metareplace` コマンドを使用して、エラーの発生したデバイスをサービスに復帰させる。

```
# metareplace -e <メタデバイス> <スライス>
```

`-e` オプションは、スライスの状態を「使用可能」状態に移行し、障害の発生したスライスを再同期します。

注 - ホットスペアによって交換されたスライスは、`metareplace` コマンドを使用して一番最後に交換するデバイスにしてください。ホットスペアを最初に交換すると、これが使用可能になるとすぐに、サブミラー内の他のエラーの発生したスライスと交換されてしまうことがあります。

再同期は、一度にサブミラー (メタデバイス) の 1 つのスライスでしか実行できません。サブミラーのすべてのスライスが電源断による影響を受けた場合、各スライスを別個に交換しなければなりません。1.05G バイトのディスクでは、再同期の実行におよそ 10 分かかります。

サブミラーの数、およびこれらのサブミラーに含まれるスライスの数にもよりますが、再同期には相当な時間が必要なことがあります。1.05G バイトのドライブ 30 個で構成される 1 つのサブミラーでは、終了するのにおよそ 5 時間を要することがあります。5 つのスライスのサブミラーから構成されるような、通常使用される構成の場合は、終了するのにたった 50 分ですむ場合もあります。

3. 電源断の後、影響を受けた **SPARCstorage Array** シャーシ上の状態データベースの複製は、すべてエラー状態となります。これらの複製は次のリブート時点で再生されますが、削除してから追加して戻せば、手動でサービスに復帰させることができます。

```
# metadb -d <スライス>
# metadb -a <スライス>
```

注 - 各スライス上で削除された状態データベースの複製の数と同じ数だけ追加することが必要です。1つの metadb コマンドで、複数の状態データベースの複製を削除できます。1つの metadb -d で削除された複製を追加して戻すには、metadb -a を何回か呼び出さなければならないこともあります。その理由は、1つのスライス上に複製のコピーが複数個必要な場合、-c フラグを使用する metadb を 1 回呼び出して追加しなければならないためです。詳細は、metadb(1M) のマニュアルページを参照してください。

状態データベースの複製の障害回復は自動的に実行されないため、SPARCstorage Array がサービスに復帰した直後に、障害回復を手動で実行するのが最も安全です。さもなければ、新しい障害が引き起こされて状態データベースの複製の大半がサービスを提供できなくなり、カーネルのパニックを引き起こすことがあります。使用できる状態データベースの複製の数が少なすぎる場合、このように DiskSuite が動作することがあります。

▼ ホスト間で SPARCstorage Array ディスクを移動する方法 (コマンド行)

この作業では、DiskSuite オブジェクトを含むディスクを、ある SPARCstorage Array から別の SPARCstorage Array に移動する方法について説明します。

1. エラー状態のデバイス、または移動すべきディスク上のホットスベアによって交換されたデバイスを修復する。
2. metadb コマンドと metastat -p コマンドからの出力を使用して、移動すべきディスク上のホットスベア、メタデバイス、状態データベースの複製を特定する。
3. ディスクを新しいホストに物理的に移動する。その際、デバイス名が同じになるように、類似の方法で接続するよう注意する。

4. 状態データベースの複製を再作成する。

```
# metadb -a [-f] <スライス> ...
```

331ページの手順2で特定された状態データベースの複製を含むスライス名と同じ名前を使用してください。-f オプションを使用して、状態データベースの複製を強制的に作成する場合があります。

5. 331ページの手順2の `metastat -p` コマンドからの出力を `md.tab` ファイルにコピーする。

6. `md.tab` ファイルを編集して、次の変更を行う。

- 移動しなかったメタデバイスを削除する。
- 古いメタデバイス名を新しい名前に変更する。
- 当分の間、任意のミラーを1面のミラーにし、最も小さなサブミラーを選択する(適切ならば)。

7. `md.tab` ファイルの構文をチェックする。

```
# metainit -a -n
```

8. 移動したメタデバイスとホットスペア集合を再作成する。

```
# metainit -a
```

9. 必要に応じて `metattach(1M)` コマンドを使用して、1面ミラーを多面ミラーにする。

10. ブート時に自動的にマウントされるファイルシステムの `/etc/vfstab` ファイルを編集する。その後、必要に応じて、新しいメタデバイス上にファイルシステムを再マウントする。

SPARCstorage Array をシステムディスクとして使用

この節では、SPARCstorage Array をシステムディスク (ブートデバイス) として機能させる方法について説明します。

SPARCstorage Array をブート可能にする

SPARCstorage Array に関する最小限のブート条件を次に示します。

- Solaris 2.5
- ホストの SOC カードの Fcode のリビジョン : 1.52 以降
- SPARCstorage Array の Firmware のリビジョン : 3.12 以降

Fcode のリビジョンを更新またはチェックするためには、SPARCstorage Array CD の専用サブディレクトリに提供される、`fc_update` プログラムを使用します。

詳細については、SPARCstorage Array のマニュアルを参照してください。

▼ SPARCstorage Array ディスクをブートプロセスの初期段階で使用可能にする方法

SPARCstorage Array ディスクをブートプロセスの早い段階で使用可能にするには、次の `forceload` エントリを `/etc/system` ファイルに追加します。これは、SPARCstorage Array をシステムディスク (ブートデバイス) として機能させるために必要です。

```
*ident "@(#)system 1.15 92/11/14 SMI" /* SVR4 1.5 */
*
* SYSTEM SPECIFICATION FILE
*
* ...
* forceload:
*
* これらのモジュールは、最初の参照時ではなく、ブート時 (ルートファイル
* システムをマウントする直前) にロードさせる。なお、forceload は、ディレ
* クトリを含んだファイル名を要求する。また、モジュールをロードすること
* は、必ずしもモジュールがインストールされることを意味しない。
*
forceload: drv/ssd
```

(続く)

続き

```
forceload: drv/pln
forceload: drv/soc
...
```

注 - SPARCstorage Array ディスク上にルート (/) ミラーを作成する場合、`metaroot(1M)` コマンドを実行すると、上記のエントリが自動的に `/etc/system` ファイルに置かれます。

ヒントとテクニック

この章では、DiskSuite を使用するためのヒントについて説明します。

必要な情報を提供する節に直接進むためには、次の目次を使用してください。

- 336ページの「状態データベースの複製とトランスメタデバイス」
- 336ページの「DiskSuite と Prestoserve」
- 339ページの「DiskSuite 構成のガイドライン」
- 346ページの「ディスクドライブの操作」
- 347ページの「トランスメタデバイス (UFS ログイン) とディスクの制限」
- 347ページの「DiskSuite ツールの使用法」
- 357ページの「メタデバイスの命名規則」
- 358ページの「メタデバイス名の切り替え」
- 365ページの「ストライプの操作」
- 367ページの「ミラーの操作」
- 372ページの「ホットスペア」
- 372ページの「ディスクセットの操作」

状態データベースの複製とトランスメタデバイス

トランスメタデバイス (UFS ロギング) を作成すれば、UFS の可用性を簡単に高めることができます。トランスメタデバイスを使用する際に、スライスを効果的に使用するためのヒントを次に示します。

- 新しいシステムで、各ディスク上に2～3個の小さなスライス(それぞれ8～10M バイト)を作成します。これらのスライスを使用して、状態データベースの複製とロギングデバイスの両方を保管します。経験的には、新しいディスクをシステムに追加するときに、この方法を使用して、状態データベースの複製とトランスメタデバイスを構成します。このようにして、1つのスライスで2つの DiskSuite オブジェクトを管理できます。

状態データベースの複製の追加、およびトランスメタデバイスの作成については、第2章を参照してください。

DiskSuite と Prestoserve

Prestoserve™ は、ディスク書き込みの多いアプリケーションにおける応答時間の短縮のためのハードウェア/ソフトウェア製品です。この製品では、ディスクブロックデバイスの書き込み操作を選択的に不揮発性メモリーにキャッシュすることによって、パフォーマンスを加速し、ディスクの入出力ボトルネックを減少させます。

Prestoserve を使用すれば、NFS™ サーバー、ディスク入出力の多いアプリケーション、およびファイルシステムのパフォーマンスが向上します。

DiskSuite は、次の制約のもとで、Prestoserve と完全に互換性があります。

Prestoserve と互換性のある DiskSuite オブジェクト

- ストライプ/連結
- トップレベルのメタデバイス(ミラーは好ましくない。337ページの「Prestoserve をミラーと使用することが好ましくない理由」を参照)

Prestoserve と互換性のない DiskSuite オブジェクト

- 配下のコンポーネント (つまり、サブミラー)
- 状態データベースの複製
- トランスメタデバイス (および配下のマスターデバイスとロギングデバイス)

Prestoserve をミラーと使用することが好ましくない理由

簡単な理由として、Prestoserve でミラーを使用すると、入出力サブシステムにシステムのアキレス腱となる部分が生じます。これこそ、ミラーの設計目的に反するものです。Prestoserve を使用すると、ミラーの MTBF はおよそ単体ディスクと同レベルまで低下します。

Prestoserve をトランスメタデバイスと使用することが好ましくない理由

Prestoserve は、トランスメタデバイス上では使用できません。ロギング UFS 上で Prestoserve を使用すると、システムがハングアップしたりパニック状態になることがあります。Prestoserve は、デバイスからの入出力を NVRAM にリダイレクトすることによって動作します。このリダイレクションは、ロギング UFS とメタデバイス間の通信プロトコルを妨害します。

▼ Prestoserve と DiskSuite の設定方法 (コマンド行)

次の手順では、DiskSuite と一緒に使用する Prestoserve をロードし、有効にする方法について説明します。基本的には、DiskSuite ドライバの後から Prestoserve をロードするよう、`/etc/system` ファイルを編集します。

1. 次の行を `/etc/system` ファイルに追加する。

```
exclude: drv/pr
```

2. `/etc/init.d/SUNWmd.init` ファイルを編集し、`start` 節の最後に次の行を追加する。

```
'start')  
rm -f /tmp/.mdlock
```

```

if [ -x "$METAINIT" -a -c "$METADEV" ]; then
#echo "$METAINIT -r"
$METAINIT -r
error=$?
#echo "$error"
case "$error" in
0|1) ;;

66)
echo "Insufficient metadevice database replicas located."
echo ""
echo "Use metadb to delete databases which are broken."
echo "Ignore any \"Read-only file system\" error messages."
echo "Reboot the system when finished to reload the metadevice
database."
echo "After reboot, repair any broken database replicas which
were deleted."
    /sbin/sulogin < /dev/console
echo "Resuming system initialization. Metadevice database will
remain stale."
    ;;

*) echo "Unknown $METAINIT -r failure $error."
    ;;
esac
modload /kernel/drv/pr
presto -p >/dev/null
fi
;;

```

3. /etc/init.d/prestoserve ファイルを編集する。

次の行を、

```
presto -u
```

次のように変更します。

```
presto -u /<ファイルシステム>...
```

このコマンドで、<ファイルシステム>... は、Prestoserve により加速されるすべてのファイルシステムのリストです。次の項目は除外してください。

- ルート (/)
- /usr

- /usr/kvm
- /var
- /var/adm

DiskSuite 構成のガイドライン

DiskSuite 構成に不備があると、パフォーマンスが低下することがあります。この節では、DiskSuite を使用して高いパフォーマンスを得るためのヒントを紹介します。

一般的なガイドライン

- ディスクとコントローラ - ドライブは、別のドライブパス上のメタデバイスに置きます。SCSI ドライブの場合、これは別のホストアダプタを意味します。IPI ドライブの場合、これは別のコントローラを意味します。入出力の負荷を複数のコントローラに分散すれば、メタデバイスのパフォーマンスと可用性が向上します。

SPARCstorage Array の場合、可能ならば、異なるシャーシ上のミラーに含まれるドライブを使用してください。このような構成では、すべてのミラーデータが SPARCstorage Array のシャーシ障害に耐えることができます。ドライブを異なるシャーシに分散できない場合、異なるトレイに含まれるドライブを使用します。これによって、ミラーをオンライン状態に保ったままで、サブミラーをオフラインにし、保守のためにトレイを停止したり除去することができます。

たとえば、2 面のミラーで、各サブミラーが 3 つの SPARCstorage Array ディスクの連結によって構成されるものとします。一方のサブミラーは、トレイ 1 に含まれる 3 つのディスクから構成され、他方のサブミラーは、トレイ 2 に含まれるドライブから構成されます。この構成を初期化するためのコマンド行インタフェースは、次のようになります。

```
# metainit d1 3 1 c0t0d0s2 1 c0t0d1s2 c0t0d2s2
d1: Concat/Stripe is setup
# metainit d2 3 1 c0t2d0s2 1 c0t2d1s2 c0t2d2s2
d2: Concat/Stripe is setup
# metainit d0 -m d1
```

(続く)

```
d0: Mirror is setup
# metattach d0 d2
d0: Component d2 is attached
```

文字列 t0 と t1 はトレイ 1 に含まれ、t2 と t3 はトレイ 2 に含まれ、t4 と t5 はトレイ 3 に含まれます。したがって、上述のコマンドでは、異なるトレイにサブミラーを作成するため、最初のサブミラーに文字列 t0 を使用し、2 番目のサブミラーに文字列 t2 を使用します。

- システムファイル - /etc/lvm/mddb.cf ファイルや /etc/lvm/mdo.cf ファイルについては、編集も除去も行わないでください。
これらのファイルは、定期的にバックアップしてください。
- スライスがメタデバイスとして定義および起動されたら、これを他の目的に使用しないでください。
- 不良ディスクの再フォーマットが必要となる場合に備えて、prtvtoc(1M) コマンドの出力のハードコピーを保存しておきます。

状態データベースの複製のガイドライン

- メタデバイスの一部となるスライス上に状態データベースの複製が置かれた場合、メタデバイスの容量は、複製によって占有される領域分だけ減少します。複製によって使用される領域は次のシリンダ境界まで切り上げられ、この領域はメタデバイスによってスキップされます。しかし、状態データベースの複製のデフォルトサイズは 1034 ブロックにすぎないため、複製とメタデバイスをこのように結合することは、実際に DiskSuite のきわめて効果的な使用方法となります。

ストライプ化のガイドライン

- ストライプは、(他のメタデバイスではなく) スライスからのみ作成されます。
- ディスクジオメトリ (幾何学的配置) の異なるディスクのスライスを使用しません。
- スライスは、同じコントローラ上でもかまいませんが、別のディスクのものを使用します。それぞれ別のコントローラ上に置かれたストライプを使用する場合、同時に実行できる読み書きの数が増大します。

- システムやアプリケーションからの入出力要求に適合するように、ストライプの飛び越し値を設定します。
- 単一のディスク上にあるパーティションを使用したストライプ化を行いません。これは、同時アクセスができなくなり、パフォーマンス上の障害となるからです。
- 同じサイズのディスクコンポーネントを使用します。異なるサイズのディスクコンポーネントでストライプ化すると、ディスク領域に使用不可能な部分が生じます。
- ストライプ化する際に異なるサイズのスライスを使用した場合、ディスク容量は最小サイズのスライスの倍数に制限されます。

連結のガイドライン

- 連結は、(他のメタデバイスではなく)スライスからのみ作成されます。
- ディスクジオメトリ(幾何学的配置)の異なるスライスを使用しません。
- 可能ならば、連結方式メタデバイスのコンポーネントを、異なるコントローラとバス上に分散配置します。
- 連結では、ストライプ化よりもCPUサイクルが短いので、小さなランダム入出力や均一な入出力分布の場合でも、うまく機能します。

連結方式ストライプのガイドライン

上記のストライプと連結のガイドラインを参照してください。

ミラーのガイドライン

- ディスクとコントローラ-サブミラーのスライスは、サブミラーごとに異なるディスクとコントローラ上に保持します。同じミラー中の複数のサブミラーのスライスが、同じディスクに置かれている場合、データの保護能力は著しく低下します。同様に、コントローラとそれに接続しているケーブルは、ディスクよりも故障する傾向が高いため、サブミラーは異なるコントローラにまたがって構成します。これによってミラーのパフォーマンスも改善されます。
- 同じディスク-同じディスク上にはミラーを定義しません。同じドライブへの書き込みは同じ資源をめぐって競合し、1つのドライブに障害が発生すると、すべてのデータが失われることとなります。

- 読み書きパフォーマンス - ミラー化により、読み取りパフォーマンスは向上することがありますが、書き込みパフォーマンスは必ず低下します。ミラー化によって読み取りパフォーマンスが向上するのは、スレッド化された入出力や非同期入出力の場合に限られます。メタデバイスからシングルスレッド読み出しだけが行われる場合、パフォーマンスの向上は得られません。
- 同じサイズのサブミラー - 同じサイズのサブミラーを使用します。異なるサイズのサブミラーでは、ディスク領域に使用不可能な部分が生じます。
- 同じタイプのディスクとコントローラ - 1つのミラーでは同じタイプのディスクとコントローラを使用します。特に、古い SCSI や SMD の記憶装置の場合、モデルやブランドの異なるディスクやコントローラ間で、パフォーマンスが大幅に異なることがあります。1つのミラーに異なるパフォーマンスレベルの製品が混在すると、パフォーマンスが著しく低下することがあります。
- サブミラーに対する読み書きオプションの設定 - ミラーの読み取りオプションを試してみると、パフォーマンスが向上することがあります。たとえば、デフォルトの読み取りモードでは、ディスク間の読み取りをラウンドロビン方式 (巡回式) で切り替えます。この方法は、UFS のマルチユーザー・マルチプロセスアクティビティでは最適なことが多いため、デフォルトとなっています。

場合によっては、ジオメトリック (幾何学的配置) オプションを使用すると、ヘッドの移動時間とアクセス時間を最小限に抑えることによって、パフォーマンスが向上することもあります。このオプションが最も有効なのは、ディスクあたり1つのスライスしかない場合、一度に1つのプロセスだけがスライス / ファイルのシステムを使用する場合、入出力パターンがきわめて連続的に行われる場合、すべてのアクセスが読み取りである場合です。

ミラーオプションを変更するには、229ページの「ミラーのオプションの変更方法 (DiskSuite ツール)」を参照してください。

- ミラーのマウント - 必ずミラーデバイスを直接マウントしてください。オフラインであって読み取り専用でマウントされている場合を除いて、サブミラーを直接マウントしてはなりません。サブミラーの一部であるスライスも直接マウントしてはなりません。これを行うと、データの破壊やシステムクラッシュが発生することがあります。
- swap のミラー化 - すべての swap デバイスをチェックするには、`swap -l` を使用します。swap と指定されたスライスは、別々にミラー化しなければなりません。

RAID5 メタデバイスのガイドライン

- **20% 書き込みのルールに従う** - パリティ計算が複雑であるため、書き込みがおよそ 20% を超えるメタデバイスは、RAID5 メタデバイスのデバイスとはなりません。データの冗長性が必要な場合は、ミラー化を検討してください。
- 「スライス消費型」RAID5 メタデバイスの欠点 - RAID5 メタデバイスに含まれるスライスが増大するほど、コンポーネントが故障した場合に読み書き操作の時間が長くなります。
- RAID5 メタデバイスはミラー化できません。
- **RAID5 メタデバイスとストライプ化のガイドライン** - ストライプ化のガイドラインは、RAID5 メタデバイスの構成にも適用されます。340ページの「ストライプ化のガイドライン」を参照してください。
- **異なるコントローラを使用する** - RAID5 デバイスを作成する場合、コントローラとそれに接続しているケーブルはディスクに比べて故障する可能性が高いため、個々のコントローラにまたがってスライスを使用します。これにより、ミラーのパフォーマンスも向上します。
- **同じサイズのスライスを使用する** - 同じサイズのディスクスライスを使用します。異なるサイズのスライスから成る RAID5 メタデバイスを作成すると、ディスク領域に使用不可能な部分が生じます。
- **飛び越し値** - これはメタデバイスの作成時に設定できます。その後では、値を変更することはできません。デフォルトの飛び越し値は 16K バイトです。これは多くのアプリケーションに対して適当な値です。RAID5 メタデバイス内のさまざまなスライスがさまざまなコントローラに置かれており、メタデバイスに対するアクセスが主に多量の順次アクセスである場合、32K バイトの飛び越し値を指定するとパフォーマンスが向上することがあります。
- **RAID5 メタデバイスへの連結** - 既存の RAID5 に新しいスライスを連結すると、連結によるデータは連続しているため、メタデバイス全体のパフォーマンスに影響を与えません。データは、すべてのコンポーネントを通じてストライプ化されません。メタデバイスの元のスライスでは、すべてのスライスを通じてデータとパリティがストライプ化されます。連結方式スライスではこのストライプが失われるが、コンポーネントの入出力中にはパリティが使用されるため、データはエラーから回復することができます。

連結方式スライスの場合、どの領域でもパリティをストライプ化しないという意味で、事情が異なります。したがって、スライスの内容全体をデータに使用できません。

スライスが連結されると、多量の書き込みや順次書き込みに対するパフォーマンス上のメリットが失われます。

UFS ロギングのガイドライン

- ロギングデバイスとマスターデバイスの場合 - 同じトランスメタデバイスに属するロギングデバイスとマスターデバイスを、異なるドライブとコントローラに置きます。
- トランスメタデバイスと共有ロギングデバイスの場合 - トランスメタデバイスは、メタデバイスのロギングデバイスを共有できます。しかし、きわめて負荷の重いファイルシステムには、別のログをもたせてください。
- 小規模なファイルシステム - 読み取り操作が大部分の小規模なファイルシステムでは、おそらくログに記録する必要はありません。
- ロギングデバイスのミラー化 - 可能な限り、すべてのロギングデバイスをミラー化します。デバイスエラーによってログ中のデータが失われると、ファイルシステムが不整合な状態となり、fsck(1M) を使用しても、ユーザーの介入なしでは修復できない可能性があります。
- 大きなロギングデバイスを使用すると、大きな多重度 (concurrency) が得られません。

ホットスペアのガイドライン

- 一時的な修復手段としてのホットスペア - ホットスペアは、そのまま永続的に構成中に組み込まれて使用されるように設計されていません。修復が済んだスライスや新しいスライスと交換する必要があります。
- ホットスペアと状態データベースの複製 - ホットスペアには状態データベースの複製を置くことができません。
- コントローラをまたがった割り当て - 理想的には、ホットスペア集合に追加するスライスは、異なるコントローラに接続します。これによって、コントローラのエラーや障害に対するデータの可用性を保証します。
- 不適切なサイズのホットスペア - 不適切なサイズのホットスペアを、サブミラーや RAID5 メタデバイスに関連付けしないでください。
- 使用中とマークされたホットスペア - ホットスペア集合内のすべてのホットスペアに「使用中」のマークがないことを確認してください。

- 1面のミラーとホットスペア - 1面ミラーに含まれるサブミラーには、ホットスペア集合を割り当てないでください。
- ホットスペアは適合した順に使用される - 異なるサイズのホットスペアをホットスペア集合に追加する場合は、小さなスライスから追加していきます。

ファイルシステムのガイドライン

メタデバイスの配下にあるスライスには、ファイルシステムをマウントしないでください。任意のメタデバイスにスライスが使用された場合、そのスライスをファイルシステムとしてマウントしてはなりません。可能ならば、メタデバイスとして使用する予定の物理デバイスは、起動する前にマウント解除します。たとえば、UFS用のトランスメタデバイスを `/etc/vfstab` ファイルに作成した場合、そのトランスメタデバイス名はマウントおよび `fsck` を行うデバイスとして指定します。

ラベル付きパーティション

すべての物理デバイスにはディスクラベルが必要であり、通常は `install`、`format`、`fmthard` などのプログラムによって作成されます。このラベルは、ラベルに定義された複数の論理パーティションに表示できます。ラベルを含んだ物理パーティションは、ラベルを含んだブロックに対するユーザーの書き込みを許してはなりません。通常、これはブロック 0 である。UNIX のデバイスドライバでは、ユーザーがこのラベルに上書きできます。

機密保護対策

- DiskSuite は、システム上で実行できるメタデバイスの再構成に対しては、監査トレールを提供しません。つまり、DiskSuite は C2 セキュリティをサポートしません。

互換性

- Solstice DiskSuite 4.2.1 を実行するシステムは、Solaris 2.4, Solaris 2.5, Solaris 2.5.1, Solaris 7 あるいは Solaris 8 を実行している必要があります。
- UFS ログインとディスクセットの場合、Solaris 7 あるいは Solaris 8 を実行する必要があります。

- DiskSuite は、Solstice Backup™ 5.5.1 製品と互換性があります。
- ディスクセットは、x86 システムではサポートされません。

ディスクドライブの操作

たとえば、ディスクを交換した後で、ディスクドライブのパーティションを再分割する必要がある場合、fmthard(1M) コマンドを使用してスクリプトを作成し、ディスク上に VTOC(ボリューム目録) 情報をすばやく再作成することができます。

▼ fmthard(1M) の使用方法

1. prtvtoc(1M) コマンドを使用して、ディスクに関するパーティション情報の一覧を取得する。

```
# prtvtoc /dev/rdsk/c2t0d0s0 > /tmp/vtoc
```

この例では、ディスク c2t0d0 の情報は、ディスク上のファイルにリダイレクトされます。

2. fmthard(1M) コマンドを使用して、次に示すようなスクリプトを作成して実行する。

```
for i in 1 2 3 5
do
fmthard -s /tmp/vtoc /dev/rdsk/c2t${i}d0s2
done
```

トランスメタデバイス (UFS ロギング) とディスクの制限

ディスク領域のサイズやユーザーが使用できる i ノードの数 (ファイルの数にほぼ等しい) を制限することができます (これは DiskSuite の機能ではなく、Solaris の機能です)。これらの制限は、ファイルシステムがマウントされるたびに自動的に起動されます。

ファイルシステムがロギング用にも設定されている場合、制限用のファイルシステムの設定は、quotacheck を使用して高速にチェックできます。このような設定により、quotacheck の実行に必要な時間を減らすことができます。

トランスメタデバイスを作成するには、第 2 章を参照してください。制限の詳細は、『Solaris のシステム管理 (第 2 巻)』を参照してください。

DiskSuite ツールの使用法

この節では、DiskSuite ツールの高度な使用法 (および制約) について説明します。

制約

- カラーマップ - DiskSuite ツールは、アプリケーションを終了するときに、「ディスク表示」ウィンドウのカラーマップを保存できません。DiskSuite ツールの使用を終了すれば、そのカラーマップは無効になり、保存することはできません。
- メタデバイスの論理名 - 現在、DiskSuite ツールには table1 や log1 などの論理名をメタデバイスに割り当てる機能が存在しません。
- スライスブラウザウィンドウの「使用状況」カラム - スライスが raw デバイスとして使用されている場合、スライスブラウザウィンドウの「使用状況」カラムは「未割り当て」から変化しません。現在、raw デバイスとして使用されるメタデバイスに限らず、すべての raw デバイスがこの問題を共有しています。使用するデバイスは、ファイルシステムや swap として登録するしか方法がありません。DiskSuite ツールには、このための独自の方法がありません。

メタデバイスエディタの使用法

メタデバイスエディタのキャンバスにおいて、画面の表示領域の管理に役立つ3つのヒントを紹介します。

- オブジェクトのポップアップメニューから「拡大表示解除」を選択すると、より多くのオブジェクトをキャンバスに収容できます。
- キャンバスに多数のオブジェクトがあり、マウスを使ってオブジェクトの再配置を行ったり、そのうちのいくつかを片付けたりする場合、「編集」メニューから「キャンバスを整理」を選択すると便利です。「キャンバスを整理」オプションは、キャンバス上のオブジェクトをグリッド単位に再配列し、表示を見やすくします。
- キャンバスのサイズ変更にはサッシ (枠) を使用します。キャンバス領域を広げるには、「メタデバイスエディタ」ウィンドウの下部にあるサッシ上でマウスの SELECT ボタン (通常は左ボタン) を押し、右方向にドラッグします。

スライス表示、ディスク表示、フィルタの使用法

「スライス表示」ウィンドウと「ディスク表示」ウィンドウの内部でフィルタを設定すると、当面の作業に対して適当なスライスをすばやく探索するのに役立ちます。

▼ スライスサイズの選別方法 (DiskSuite ツール)

多数のディスク (とスライス) を備えたシステムの場合、特定のサイズで使用できるスライスを探すことは面倒な仕事です。「スライスフィルタ」ウィンドウを使用すれば、この作業に要する時間を節約できます。

この作業では、200M バイトを超える使用可能なスライスに対して「スライスブラウザ」ウィンドウにフィルタを作成し、これらのスライスを「ディスク表示」ウィンドウにドラッグ&ドロップして、その位置を調べる方法について説明します。

1. 「スライス」をクリックして「スライスブラウザ」ウィンドウを表示する。
「スライスブラウザ」ウィンドウが表示されます。
2. 「スライスブラウザ」ウィンドウの「フィルタ」メニューから、「フィルタの設定」を選択する。
「スライスフィルタ」ウィンドウが表示されます。

3. 使用可能なスライスを検索するには、「使用可能な種類」ラジオボタンがチェックされており、プルダウンメニューで「任意」が選択されていることを確認する。
4. **200M** バイトを超えるスライスを選別するには、「サイズ」ラジオボタンをチェックし、最初のプルダウンで「指定数値より大きい」を選択し、テキストボックスに **200** を入力し、**2** 番目のプルダウンメニューで「**M** バイト」を選択する。
5. 「適用」をクリックし、「スライスブラウザ」ウィンドウで結果を見る。
必要ならば、「スライスフィルタ」ウィンドウの値を変更し、「適用」をクリックしてフィルタ方式を変更します。
6. 希望に合わせてフィルタ方式を調節してから、「了解」をクリックして「スライスフィルタ」ウィンドウを閉じる。
7. 「ディスク表示」をクリックして、「ディスク表示」ウィンドウを表示する。
「ディスク表示」ウィンドウが表示されます。
8. 「スライスブラウザ」ウィンドウで「すべてを選択」をクリックする。
選択したスライスを「ディスク表示」ウィンドウのドロップ領域の色の部分にドラッグする。
9. 「ディスク表示」ウィンドウで結果を見る。
DiskSuite ツールは、「ディスク表示」ウィンドウにドラッグされたすべてのスライスに対して、選択したドロップ領域の色を使用します。これで、339ページの「一般的なガイドライン」で概説した内容に従って、(たとえばサブミラーを作成するために) スライスを選択できます。

▼ 交換用スライスの選別方法 (DiskSuite ツール)

この作業では、DiskSuite ツールを使用することによって、サブミラー内のエラーの発生したスライスに対して、適切なサイズの交換用スライスを見つける方法を示します。

注・この手法はミラーだけに限定されません。この作業を使用すれば、どのタイプのメタデバイスに対しても、交換スライスを見つけることができます。

1. 「ディスク表示」をクリックして「ディスク表示」ウィンドウを表示する。
「ディスク表示」ウィンドウが表示されます。
2. エラーの発生したミラーオブジェクトを、オブジェクトリストからキャンバスにドラッグする。
3. ミラーの内部で1つのサブミラー(連結方式オブジェクト)を選択し、「ディスク表示」ウィンドウにドラッグする。さらに、2番目のサブミラーについても同じ操作を行う(3面ミラーの場合は、3番目のサブミラーについても同様)。
「ディスク表示」ウィンドウでは、ミラーオブジェクト内のサブミラーに対応する別のカラーでスライスが着色されます。このため、たとえばコントローラが異なっても、スライスの位置を探すために役立ちます。
4. 「スライス」をクリックして「スライスブラウザ」ウィンドウを表示する。
「スライスブラウザ」ウィンドウが表示されます。
5. 「スライスブラウザ」ウィンドウで「フィルタの設定」をクリックする。
「スライスフィルタ」ウィンドウが表示されます。
6. 使用可能なスライスを検索するには、「使用可能な種類」ラジオボタンがチェックされており、プルダウンで「メタバイコンポーネント」が選択されていることを確認する。
7. エラーの発生したスライスを交換するためのスライスを選別する。
これを行う1つの方法としては、エラーの発生したスライスよりも少しだけ小さいサイズを基準にして、それより大きなスライスを見つけるフィルタを設定します。この方法では、エラーの発生したスライスと同じサイズのスライスを検索するフィルタを設定する場合に比べて、広範囲なスライスが表示されます。
8. サイズを設定する。「サイズ」ラジオボタンをチェックし、最初のプルダウンで「指定数値より大きい」を選択する。テキストボックスにスライスのサイズ(Mバイト単位で、エラーの発生したスライスより少しだけ小さなサイズ)を入力する。2番目のプルダウンで「Mバイト」を選択する。

9. 「適用」をクリックし、「スライスブラウザ」ウィンドウで結果を見る。
必要ならば、「スライスフィルタ」ウィンドウの値を変更し、「適用」をクリックしてフィルタ方式を変更します。
10. 「スライスブラウザ」ウィンドウで、「すべてを選択」をクリックする。選択したスライスを「ディスク表示」ウィンドウのドロップ領域の色の部分にドラッグする。
11. 「ディスク表示」ウィンドウで結果を見る。
DiskSuite ツールは、「ディスク表示」ウィンドウにドラッグされたすべてのスライスに対して、この色を使用します。
12. 交換用のスライスを選択する。
これで、339ページの「一般的なガイドライン」で概説したガイドラインに従って、DiskSuite オブジェクトに対するスライスを選択できます。十分な大きさのある交換スライスを選び、(別のコントローラ上で、あるいは少なくとも別のディスク上で) ミラーのガイドラインに従います。
13. 交換用のスライスを、「ディスク表示」ウィンドウからエラーの発生したスライスをもち連結方式オブジェクトの矩形までドラッグする。
14. ミラーを確定する。
ミラーオブジェクトの先頭の内側をクリックしてから、「確定」をクリックすると、ミラーの再同期が始まります。

DiskSuite ツールの色とフォントの変更

デフォルトでは、DiskSuite ツールは、OpenWindows™ のデスクトップアプリケーションと互換性のある色とフォントを使用します。この節では、これらの色とフォントの変更方法について説明します。

DiskSuite ツールと色

DiskSuite ツールでは多数の色を使用します。

- 標準のフォアグラウンドカラー - アプリケーション要素のほとんどすべてを表示するために使用する主要な色。標準では、ウィンドウ、ボタン、その他のコントロール用のデフォルトカラーを提供します。

- 標準のバックグラウンドカラー - ウィンドウ、ボタン、その他のコントロールで提示される情報の表示に使用される色。
- キャンパスのバックグラウンドカラー - データ領域のバックグラウンドカラー。たとえば、エディタ、「ディスク表示」ウィンドウ、スクロールリストの表示領域は、すべてキャンパスのバックグラウンドカラーを使用します。
- マッピングカラー - 論理デバイスから「ディスク表示」ウィンドウ内のスライスへのマッピングを表示する色。それぞれのディスク表示マッピングごとに1つずつ、全部で8つのマッピングカラーがあります。
- 状態カラー - 注意を必要とするオブジェクトの状態情報を強調する色。独自の色によって注意、緊急、重大な障害の3つの状態を表示します。

X ウィンドウシステムの RGB (赤、緑、青) 色指定機能を使用すれば、ほとんど無限に多彩な色を指定できます。もちろん、これらの色の多くは似ており、シェード (影の部分) や輝度がほんの少しずつ異なるだけです。

色の選択と指定に役立つよう、X ウィンドウシステムでは、RGB 値の代わりに名前指定できる、標準のデフォルトカラーセットを提供します。このカラー名のデータベースを調べるには、標準の X ユーティリティ `showrgb` を使用します。このユーティリティは、RGB 値と対応する記述エイリアスを表示します。たとえば、次のようになります。

```
# showrgb
199 21 133 medium violet red
176 196 222 light steel blue
102 139 139 paleturquoise4
159 121 238 mediumpurple2
141 182 205 lightskyblue3
  0 238 118 springgreen2
255 160 122 light salmon
154 205  50 yellowgreen
178  58 238 darkorchid2
 69 139 116 aquamarine4
...
107 107 107 gray42
 71  71  71 gray28
 61  61  61 gray24
255 255 255 white
  0 205 205 cyan3
  0  0  0 black
```

`/usr/openwin/lib/X11/rgb.txt` ファイルを見ても、デフォルトのカラー名データベースを調べることができます。

残念ながら、色をブラウズするための標準アプリケーションは存在しません。パブリックドメインのカラーブラウザを入手できない場合、試行錯誤しながら希望の色を探してください。

DiskSuite ツールのデフォルトカラー

DiskSuite ツールのデフォルトカラーを表 8-1 に示します。

表 8-1 DiskSuite ツールのデフォルトカラー

カラータイプ	色
標準のフォアグラウンド	black (黒)
標準のバックグラウンド	gray (グレー)
キャンバスのバックグラウンド	gray66 (グレー 66)
マッピングカラー :	
mappingColor1	blue (青)
mappingColor2	green (緑)
mappingColor3	magenta (マゼンタ)
mappingColor4	cyan (シアン)
mappingColor5	purple (紫)
mappingColor6	mediumseagreen (中間海緑色)
mappingColor7	firebrick (耐火レンガ)
mappingColor8	tan (黄褐色)
mappingColor9	white (白)
状態色 :	
重大な障害	red (赤)

表 8-1 DiskSuite ツールのデフォルトカラー 続く

カラータイプ	色
緊急	orange (オレンジ)
注意	yellow (黄)

DiskSuite ツールとフォント

DiskSuite ツールでは、4つのフォントを使用します。

- 標準フォント - ツール (たとえば、ボタンラベル、メニュー、ダイアログボックス) 内の大部分のテキストを表示します。
- モノスペース (固定幅) フォント - 一貫したカラム調整を可能にします (たとえば、さまざまなブラウザとスクロールリストで)。これは何回か指定する必要があります。
- ボールドフォント - 属性名とラベルを実際の属性値から特定します。「情報」ウィンドウの名前 / ラベルは標準フォントで表示され、対応する値がボールドフォントで表示される。このフォントは控え目に使用されます。
- スモールフォント - 「ディスク表示」ウィンドウで 50% のスケーリングレベルにある物理デバイスを示します。

DiskSuite で使用可能なフォント

使用可能なフォントは、アプリケーションの表示に使用する X ウィンドウシステムのサーバーによって異なります。標準の X ユーティリティである `xlsfonts(1)` では、サーバーで使用可能なフォントを表示します。たとえば、次のようになります。

```
# xlsfonts
--courier-bold-o-normal--0-0-0-0-m-0-iso8859-1
--courier-bold-r-normal--0-0-0-0-m-0-iso8859-1
--courier-medium-o-normal--0-0-0-0-m-0-iso8859-1
--courier-medium-r-normal--0-0-0-0-m-0-iso8859-1
--symbol-medium-r-normal--0-0-0-0-p-0--symbol
-symbol-medium-r-normal--0-0-0-0-p-0-sun-fontspecific
-adobe-courier-bold-i-normal--0-0-0-0-m-0-iso8859-1
...
utopia-bolditalic
```

(続く)

```

utopia-italic
utopia-regular
variable
vshd
vtbold
vtsingle
zapfchancery-mediumitalic
zapfdingbats

```

使用可能なフォントを表示するために便利なもう1つのユーティリティは `xftonsel(1)` です。詳細は、これらのユーティリティのマニュアルページを参照してください。

DiskSuite ツールのデフォルトフォント

C ロケールの場合

DiskSuite ツールのデフォルトフォントは、すべて Lucida フォントファミリーを基準にしています。

表 8-2 DiskSuite ツールのデフォルトフォント

フォントタイプ	フォント
標準フォント	lucidasans12
モノスペースフォント	lucidasans-typewriter12
ボールドフォント	lucidasans-bold12
スモールフォント	lucidasans8

DiskSuite ツールでは、X ウィンドウシステムの資源データベース機能を使用して、どのフォントを使用するかを決定します。デフォルトの資源指定を次に示します。

表 8-3 DiskSuite ツールのデフォルトのフォント資源指定

資源	フォント
Metatool*fontList:	Lucidasans12
Metatool*smallFontList:	Lucidasans8
Metatool*boldFontList:	Lucidasans-bold12
Metatool*fixedFontList:	Lucidasans-typewriter12
Metatool*XmList.fontList:	Lucidasans-typewriter12
Metatool*Help*helpsubjs.fontlist:	Lucidasans-typewriter12
Metatool*Help*helptext.fontlist:	Lucidasans-typewriter12

ja ロケールの場合

ja ロケールでは、Lucida フォントファミリーではなく、次のフォントをデフォルトに指定して使用しています。

```
-sun-gothic-medium-r-normal-14-120-75-75-c-120-jisx0208.1983-0
```

```
-sun-gothic-medium-r-normal-14-120-75-75-c-60-jisx0201.1976-0
```

▼ DiskSuite ツールのデフォルトカラーとフォントの変更方法

DiskSuite ツールのデフォルトカラーとフォントを変更するには、次の 4 つの方法のいずれかを使用します。

- DiskSuite ツールの 1 つの呼び出しに対しては、xrm ユーティリティを使用して、代替のフォント資源やカラー資源を指定する。

```
# metatool -xrm '<資源名>'
```

- DiskSuite ツールのすべての呼び出しに対しては、自分の .xdefaults ファイルを編集し、代替のカラー資源やフォント資源を指定する。

.Xdefaults ファイルは、一般にデスクトップセッションの起動時にロードされます。このファイルを編集したら、次にデスクトップセッションを起動したときに、新しい資源または変更された資源が使用されます。

- 現在のセッションに対しては、再起動することなく、xrdp ユーティリティを使用する。

```
# xrdp -merge <Xdefaults のパス名>
```

- DiskSuite ツールのすべてのユーザーに対しては、次のファイルを参照する。

ja ロケールの場合：

```
/usr/opt/sadm/lib/lvm/X11/ja/app-defaults/Metatool
```

C ロケールの場合：

```
/usr/opt/sadm/lib/lvm/X11/app-defaults/Metatool
```

このファイルに対する変更は、DiskSuite ツールが次に起動されたときに認識されます。

例 - フォントの変更

この例では、DiskSuite ツールの 1 つの呼び出しに対して、標準フォントを lucidasans16 に変更します。

```
# metatool -xrm 'Metatool*fontList: lucidasans16'
```

メタデバイスの命名規則

メタデバイスに対する命名規則を使用すると、DiskSuite の管理に役立ち、メタデバイスの種類を一目で簡単に特定できます。次に、いくつかの提案を示します。

- メタデバイスの種類ごとに範囲を指定する。

たとえば、ミラーには番号 0 ~ 20、ストライプと連結には 21 ~ 40 を割り当てます。

- ミラーの命名関係を使用する。

たとえば、ミラーにはゼロ (0) で終わる名前を付け、サブミラーには 1 と 2 で終わる名前を付けたり、ミラー d10 にはサブミラー d11 と d12、ミラー d20 にはサブミラー d21 と d22 などとします。

- スライス番号とディスク番号をメタデバイス番号にマップする命名方式を使用する。

注 - `metarename` コマンドを使用すれば、メタデバイス名を再編成できます。詳細については、`metarename(1M)` のマニュアルページを参照してください。

メタデバイス名の切り替え

DiskSuite の `metarename` コマンドは、メタデバイスのリネーム機能に加えて、「階層化」メタデバイスの切り替え機能を提供します。`metarename` に `-x` オプションを付けて使用すると、既存の階層化メタデバイスとそのサブデバイスの 1 つの名前を切り替え (交換) します。この操作には、ミラーとそのサブミラーの 1 つ、またはトランスメタデバイスとそのマスターデバイスを含みます。

注 - メタデバイスを交換するにはコマンド行を使用しなければなりません。この機能は DiskSuite ツールでは現在使用できませんが、コマンド行や DiskSuite ツールを使用して、メタデバイスをリネームすることはできます。

- メタデバイス名の切り替えをいつ行うか - `metarename -x` コマンドを使用すれば、既存のストライプや連結を簡単にミラー化およびミラー解除したり、既存のメタデバイスのトランスメタデバイスを簡単に作成および除去することができます。
- メタデバイス名の切り替えを行うメリット - メタデバイス名の切り替えは、メタデバイス名を管理する上で好都合です。たとえば、ファイルシステムのマウント先をすべて希望の数値範囲内で定めることができます。
- 切り替えできるメタデバイスの組み合わせ - `metarename -x` コマンドは、次のものの切り替えに使用できます。
 - ミラーとサブミラー (連結またはストライプ)
 - トランスメタデバイスとマスターデバイス。ここで、マスターデバイスは、連結、ストライプ、ミラー、または RAID5 メタデバイスのいずれか。

メタデバイス名の切り替えは、いずれの方向にも行えます。

- ミラー化されたマスターデバイスのあるトランスメタデバイス-マスターデバイスがミラーである場合、ミラーのサブミラーの1つをトランスメタデバイスと直接切り替えることはできません。ミラーとトランスメタデバイスの名前、あるいはミラーとそのサブミラーの1つを切り替えることはできます。切り替えの関係は、常に親子です。本質的には、次に紹介する2手順のプロセスを使用すれば、サブミラーとトランスメタデバイスとの交換と同じ結果が得られます。まず、サブミラーとミラーとを切り替え、さらにミラーとトランスメタデバイスとを切り替えます。
- トランスメタデバイスのコンポーネントを切り替えるときに「**Rename busy**」メッセージが表示された - このメッセージは、次の1つ以上の状況を意味します。
 - i. 最初にロギングデバイスを切断しなかった。
 - ii. トランスメタデバイスを使用してファイルシステムをマウント解除しなかった。
 - iii. `metarename` コマンドの `-f` (強制) フラグオプションを使用しなかった。

メタデバイス名の切り替えを使用するための前提条件

- 現在使用中のメタデバイスは切り替え(またはリネーム)できません。

これには、マウントされたファイルシステム、`swap`、またはアプリケーションやデータベースの有効な記憶領域として使用されるメタデバイスが含まれます。したがって、`metarename` コマンドを使用する前に、リネームされるメタデバイスに対するすべてのアクセスを停止します。たとえば、メタデバイスを使用してマウントされたファイルシステムをマウント解除します。アプリケーションやデータベースには、アクセスを停止するために指定された方法で行う必要があります。
- エラー状態のメタデバイスを切り替えたり、ホットスワップ交換を使用してメタデバイスを切り替えることはできません。
- 切り替えは、直接的な親と子の関係にあるメタデバイス間でのみ行えます。

たとえば、マスターデバイスであるミラー内のストライプとトランスメタデバイスを直接に交換することはできません。
- トランスデバイスのメンバーを切り替える場合は、`-f` (強制) フラグを使用する必要があります。

- ログインデバイスを切り替える (またはリネームする) ことはできません。

この回避策としては、ログインデバイスを切断してリネームしてからトランスデバイスに再接続するか、またはログインデバイスを切断して、希望する名前をもつ別のログインデバイスを接続します。

- メタデバイスだけを切り替えることができます。スライスやホットスベアは切り替えできません。

名前切り替えを使用したメタデバイスの作成

既存のストライプがある場合、`metarename -x` コマンドを使用して複合メタデバイスを作成することができます。これには、マスターデバイスとしてメタデバイスをもつトランスデバイス、または連結方式からのミラー作成が含まれます。

▼ 既存の連結方式からミラーを作成する方法 (コマンド行)

この例は、マウントされたファイルシステムをもつ連結 `d1` で始まり、`d1` という名前の 2 面のミラーにマウントされたファイルシステムで終わります。

```
# metastat d1
d1: Concat/Stripe
  Size: 5600 blocks
  Stripe 0:
    Device                Start Block  Dbase
    c0t0d0s1                0           No
# metainit d2 1 1 c1t3d0s1
d2: Concat/Stripe is setup
# metainit -f d20 -m d1
d20: Mirror is setup
# umount /fs2
# metarename -x d20 d1
d20 and d1 have exchanged identities
# metastat d1
d1: Mirror
  Submirror 0: d20
  State: Okay
...

d20: Submirror of d1
  State: Okay
...
# metattach d1 d2
d1: submirror d2 is attached
# metastat d1
d1: Mirror
```

(続く)


```

Submirror 0: d20
  State: Okay
Submirror 1: d2
  State: Okay
...
# mount /fs2

```

metastat コマンドは、連結 d1 が「Okay (正常)」状態であることを確認します。metainit コマンドを使用して 2 番目の連結 (d2) を作成し、さらに d1 からミラー d20 を強制的に (-f) 作成します。metarename -x を使用して d20 と d1 を切り替える前に、ファイルシステムをマウント解除しなければなりません。d1 はトップレベルのデバイス (ミラー) となり、metastat がそのことを確認します。d2 を 2 番目のサブミラーとして接続し、metastat でミラーの状態を確認してから、ファイルシステムを再マウントします。なお、/fs2 用のマウントデバイスは変化していないため、/etc/vfstab ファイルを編集する必要はありません。

▼ 既存のメタデバイスからトランスメタデバイスを作成する方法 (コマンド行)

この例は、マウントされたファイルシステムをもつミラー d1 から始まり、d1 という名前のトランスデバイスにマウントされたファイルシステムで終わります。

```

# metastat d1
d1: Mirror
  Submirror 0: d20
    State: Okay
  Submirror 1: d2
    State: Okay
...
# umount /fs2
# metainit d21 -t d1
d21: Trans is setup
# metarename -f -x d21 d1
d21 and d1 have exchanged identities
# metastat d1
d1: Trans
  State: Detached
  Size: 5600 blocks
  Master Device: d21
...
# metattach d1 d0
d1: logging device d0 is attached

```

(続く)

```
# mount /fs2
```

metastat コマンドは、ミラー d1 が「Okay (正常)」状態であることを確認します。metainit コマンドを使用して、マスターとして d1 をもつトランスデバイス d21 を作成する前に、ファイルシステムをマウント解除しなければなりません。metarename -f -x コマンドは、d21 と d1 を強制的に切り替えます。d1 は現在トップレベルのトランスメタデバイスであり、そのことは metastat コマンドによって確認されます。ロギングデバイス d0 は、metattach コマンドによって接続されます。それから /fs2 を再マウントします。なお、/fs2 用のマウントデバイスは変化していないため(まだ d1 です)、/etc/vfstab ファイルを編集する必要はありません。

名前切り替えを使用したメタデバイスの除去

既存のミラーやトランスメタデバイスがある場合、metarename -x コマンドを使用してミラーやトランスメタデバイスを除去し、配下のメタデバイスにデータを保持できます。トランスメタデバイスの場合、マスターデバイスがメタデバイス(ストライプ/連結、ミラー、または RAID5 メタデバイス)である限り、そのメタデバイス上にデータを保持できます。

このプロセスの一部として metarename -x を使用する場合、ファイルシステムのマウント先には変化がありません。

▼ ファイルシステムをミラー化解除し、マウントデバイスを保持する方法 (コマンド行)

この例は、マウントされたファイルシステムを含むミラー d1 で始まり、d1 という名前のストライプにマウントされたファイルシステムで終わります。

```
# metastat d1
d1: Mirror
   Submirror 0: d20
     State: Okay
   Submirror 1: d2
     State: Okay
```

(続く)

```

    Pass: 1
    ...
    # umount /fs2
    # metarename -x d1 d20
    d1 and d20 have exchanged identities
    # metastat d20
    d20: Mirror
        Submirror 0: d1
            State: Okay
        Submirror 1: d2
            State: Okay
    ...

    # metadetach d20 d1
    d20: submirror d1 is detached
    # metaclear -r d20
    d20: Mirror is cleared
    d2: Concat/Stripe is cleared
    # mount /fs2

```

metastat コマンドは、ミラー d1 が「Okay (正常)」状態であることを確認します。このファイルシステムは、ミラー d1 とそのサブミラー d20 を交換する前に、マウント解除されます。これによってミラーは d20 となり、そのことは metastat で確認されます。次に、d1 が d20 から切断され、ミラー d20 ともう一方のサブミラー d2 が削除されます。最後に、/fs2 が再マウントされます。なお、/fs2 用のマウントデバイスは変化していないため、/etc/vfstab ファイルを編集する必要はありません。

▼ トランスメタデバイスを除去し、マウントデバイスを保持する方法 (コマンド行)

この例は、マウントされたファイルシステムを含むトランスメタデバイス d1 で始まり、トランスメタデバイスの配下のマスターデバイス (d1 となる) 上にマウントされたファイルシステムで終わります。

```

# metastat d1
d1: Trans
    State: Okay
    Size: 5600 blocks
    Master Device: d21
    Logging Device: d0

```

(続く)

```
d21: Mirror
  Submirror 0: d20
  State: Okay
  Submirror 1: d2
  State: Okay
...

d0: Logging device for d1
  State: Okay
  Size: 5350 blocks
# umount /fs2
# metadetach d1
d1: logging device d0 is detached
# metarename -f -x d1 d21
d1 and d21 have exchanged identities
# metastat d21
d21: Trans
  State: Detached
  Size: 5600 blocks
  Master Device: d1

d1: Mirror
  Submirror 0: d20
  State: Okay
  Submirror 1: d2
  State: Okay
# metaclear 21
# fsck /dev/md/dsk/d1
** /dev/md/dsk/d1
** Last Mounted on /fs2
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups

FILE SYSTEM STATE IN SUPERBLOCK IS WRONG; FIX? y

3 files, 10 used, 2493 free (13 frags, 310 blocks, 0.5%
fragmentation)
# mount /fs2
```

metastat コマンドは、トランスメタデバイス d1 が「Okay (正常)」状態にあることを確認します。ファイルシステムは、トランスメタデバイスのロギングデバイスを切断する前に、マウント解除されます。トランスメタデバイスとそのミラー化されたマスターデバイスは、-f (強制) フラグを使用して交換されます。metastat を再実行すると、交換が行われたことを確認できます。必要ならば、トランスメタデバイスとロギングデバイス (この場合は、それぞれ d21 と d0) が除去されます。次

に、ミラー d1 上で fsck コマンドを実行し、入力要求に対して **y** と応答します。fsck コマンドが終了すると、ファイルシステムが再マウントされます。なお、/fs2 用のマウントデバイスは変化していないため、/etc/vfstab ファイルを編集する必要はありません。

ストライプの操作

この節では、障害の発生したコントローラ上に定義され、散発的なシステムパニックを引き起こすメタデバイスへのアクセスを取り戻す手法について説明します。システムに別の使用可能なコントローラがある場合、コントローラにディスクを移動してメタデバイスを再定義することによって、メタデバイスを新しいコントローラに事実上「移動する」ことができます。この手法では、データをバックアップしメタデバイスに戻す必要がなくなります。

▼ ストライプを別のコントローラに移動する方法 (コマンド行)

この例は、2つのスライスをもつ1つのディスクから構成されます。この2つのスライスは、それぞれ2つの別のストライプ方式メタデバイスである d100 と d101 の一部です。d100 と d101 は、それぞれファイルシステム /user6 と /maplib1 を含みます。影響を受けたコントローラは c5 でした。ディスクは空きコントローラ (c4) に移動されます。この例では md.tab ファイルも使用します。

1. 影響を受けるストライプへのアクセスを停止する。

たとえば、ストライプ方式メタデバイスに関連付けられたファイルシステムをマウント解除します。

```
# umount /user6
# umount /maplib1
```

2. metaclear を使用して、ストライプ方式メタデバイスを除去する。

```
# metaclear d100
d100: Concat/Stripe is cleared
# metaclear d101
d101: Concat/Stripe is cleared
```

3. サーバーをシャットダウンし、ディスクを新しいコントローラに移動する。
4. md.tab ファイルを編集して、メタデバイス名の中に新しいコントローラを指示する。
この例では、ディスクをコントローラ 4 に移動したので、ディスクに対しては c5 ではなく c4 を使用します。

```
(md.tab ファイルの変更前)
# Stripe /user6
/dev/md/dsk/d100 1 2 /dev/dsk/c5t0d0s3 /dev/dsk/c2t2d0s3
# Stripe /maplib1
/dev/md/dsk/d101 1 2 /dev/dsk/c5t0d0s0 /dev/dsk/c2t2d0s0

(md.tab ファイルの変更後)
# Stripe /user6
/dev/md/dsk/d100 1 2 /dev/dsk/c4t0d0s3 /dev/dsk/c2t2d0s3
# Stripe /maplib1
/dev/md/dsk/d101 1 2 /dev/dsk/c4t0d0s0 /dev/dsk/c2t2d0s0
```

5. metainit を使用してそのストライプ方式メタデバイスを初期化し、newfs でファイルシステムを再初期化せずにマウントする。

```
# metainit d100
d100: Concat/Stripe is setup
# metainit d101
d101: Concat/Stripe is setup
```

6. mountall を実行して、ファイルシステムを再マウントする。
7. metastat を実行して、メタデバイスがオンラインであることを確認する。



注意 - メタデバイスまたはその関連付けられたファイルシステム上では、`newfs` コマンドを実行しないでください。さもなければ大量のデータが失われ、テープから復元しなければなりません。

ミラーの操作

この節では、ミラーとその操作についていくつかのヒントを提供します。

高度なミラー手法

次に示す 2 つの作業では、ミラーを破壊せずにサブミラーの飛び越し値を変更する方法、およびオンラインバックアップ用にミラーを使用する方法を示します。

▼ ミラー内のストライプの飛び越し値を変更する方法 (DiskSuite ツール)

この作業では、ストライプ方式メタデバイスで構成される、ミラーの配下のサブミラーの飛び越し値を変更します。この方法を使用すれば、ミラーとサブミラーを再作成してデータを復元する必要がありません。

注 - コマンド行を使用してこの作業を実行するには、`metadetach(1M)`、`metainit(1M)`、および `metattach(1M)` のマニュアルページを参照してください。

この作業に含まれる手順の概要を次に示します。

- サブミラー 1 の切断
- サブミラー 1 の除去
- サブミラー 1 として使用される、新しい飛び越し値をもつ新しいストライプを作成
- サブミラー 1 をミラーに接続

- ミラー再同期の終了を待つ
 - サブミラー 2 に対して、上述の手順を繰り返す
1. **DiskSuite** ツールが起動されていることを確認する。
 2. オブジェクトリストからミラーオブジェクトをダブルクリックする。
オブジェクトがキャンバスに表示されます。
 3. 切断されるサブミラーの内部をクリックする。
 4. サブミラーをミラーオブジェクトからキャンバスにドラッグする。
これが 2 面のミラーである場合、ミラーの状態は「緊急」に変化します。
 5. ミラーオブジェクトの先頭の矩形をクリックし、「確定」をクリックする。
 6. 希望する飛び越し値をもつ新しいサブミラーを作成する。
63ページの「ストライプ方式メタデバイスの作成方法 (DiskSuite ツール)」を参照してください。
 7. 新しいサブミラーオブジェクトをミラーオブジェクトにドラッグする。さらに「確定」をクリックしてミラーを確定する。
ミラーの再同期が始まります。
 8. コンフィグレーションログを表示して、ミラーが確定されたことを確認する。
 9. ミラーの 2 番目 (さらには 3 番目) のサブミラーに対して、368ページの手順 3 ~ 368ページの手順 7 を繰り返す。

▼ ミラーを使用してオンラインバックアップを行う方法 (コマンド行)

DiskSuite は「バックアップ製品」ではありませんが、ミラーをマウント解除したりミラー全体をオフラインにすることなく、またシステムを停止したりデータへのユーザーアクセスを拒否することなく、ミラー化されたデータをバックアップするための手段を提供します。これは次のように行われます。サブミラーの 1 つをオフラインにして (一時的にミラーの機能を失います)、バックアップを実行します。

バックアップが終了すると、すぐにそのサブミラーをオンラインに戻し、再同期を行います。

この作業は、ルート (/) を除いて、どのファイルシステムでも使用できます。この種のバックアップでは、動作中のファイルシステムの「スナップショット」を取得していることに留意してください。ファイルシステムが書き込みロックされたときの使用方法に応じて、バックアップ上の一部のファイルとファイル内容が、ディスク上の実際のファイルと一致しないことがあります。

制約

- この作業を2面ミラーで行う場合、1つのサブミラーがバックアップ用にオフラインになっているとき、データの冗長性が失われることに注意してください。3面ミラーでは、この問題は発生しません。
- バックアップが終了してから、オフラインにされていたサブミラーがオンラインに復帰するとき、システムには若干の負荷が発生します。

注 - これらの作業を定期的に変更する場合、スクリプトにすると使いやすくなります。

この作業の手順を次に示します。

- ファイルシステムの書き込みロック (UFS のみ)。ルート (/) はロックしない。
 - `metaoffline(1M)` コマンドを使用して、1つのサブミラーをミラーからオフラインにする。
 - ファイルシステムのロック解除。
 - オフラインにしたサブミラー上のデータのバックアップ。
 - `metaonline(1M)` コマンドを使用して、オフラインにされたサブミラーをオンラインに戻す。
1. 作業を開始する前に、`metastat(1M)` コマンドを実行して、ミラーが「**Okay (正常)**」状態にあることを確認する。
「Maintenance (保守状態)」のミラーは、最初に修復してください。
 2. ルート (/) を除くすべてのファイルシステムで、ファイルシステムを書き込みからロックする。

```
# /usr/sbin/lockfs -w <マウント先>
```

UFS だけを書き込みロックする必要があります。メタデバイスが、データベース管理ソフトウェアやその他の特定アプリケーション用の raw デバイスとして設定された場合、lockfs(1M) の実行は必要ありません (しかし、他社提供の適切なユーティリティを実行して、バッファをフラッシュしたりアクセスをロックすることもできます)。



注意 - ルート (/) を書き込みロックすると、システムがハングアップすることがあります。したがって、これを行なってはいけません。

3. 1 つのサブミラーをミラーからオフラインにする。

```
# metaoffline <ミラー> <サブミラー>
```

このコマンドでは、

ミラー

ミラーのメタデバイス名。

サブミラー

オフラインにされるサブミラー (メタデバイス) のメタデバイス名。

読み取りは、他のサブミラーから続行されます。最初の書き込みが行われると、ミラーはすぐに同期外れとなります。この不一致は、オフラインにされたサブミラーが手順6 でオンラインに戻ると訂正されます。

オフラインにされたファイルシステムでは、fsck(1M) を実行する必要はありません。

4. ファイルシステムのロックを解除し、書き込みを継続させる。

```
# /usr/sbin/lockfs -u <マウント先>
```

上の 369 ページの手順 2 で使用された他社提供のユーティリティにもとづいて、必要なロック解除作業を実行する必要があります。

5. オフラインにされたサブミラーのバックアップを実行する。ufsdump(1M)、または通常使用しているバックアップユーティリティを使用する。

注 - 適切なバックアップを行うには、`/dev/md/rdisk/d4` などの *raw* メタデバイスを使用します。「`rdisk`」を使用すると、2G バイトを超えるアクセスが可能となります。

6. ミラーをオンラインに戻す。

```
# metaonline <ミラー> <サブミラー>
```

DiskSuite は、サブミラーとミラーの再同期を自動的に開始します。

例 - ミラーを使用して、オンラインバックアップを行う

この例では、サブミラー `d2` と `d3` から成る、`d1` という名前のミラーを使用します。`d3` はオフラインにされ、`d2` がオンラインである間にバックアップされます。ミラー上のファイルシステムは `/home1` です。

```
# /usr/sbin/lockfs -w /home1
# metaoffline d1 d3
d1: submirror d3 is offlined
# /usr/sbin/lockfs -u /home1
(/dev/md/rdisk/d3 を使用してバックアップを実行する)
# metaonline d1 d3
d1: submirror d3 is onlined
```

シングルユーザーモードへのブートがミラーに与える影響

ルート (`/`)、`/usr`、および `swap` 用のミラーを備えたシステム (いわゆる「ブート」ファイルシステム) がシングルユーザーモードでブートされる (`boot -s`) と、これらのミラーだけではなく、おそらくシステム上のすべてのミラーが、`metastat` コマンドで見ると「`Needing Maintenance`」状態として表示されます。その上、これらのスライスに書き込みが行われた場合、`metastat` はミラー上にダーティレージョンが増えることを明らかにします。

これは危険なように見えるかもしれませんが、心配する必要はありません。`metasync -r` コマンドは、通常はミラーを再同期するためにブート処理時に実行されますが、システムがシングルユーザーモードでブートされたときには、実行を中断します。システムがリブートされると、`metasync -r` が実行されて、すべてのミラーを再同期します。

これが心配な場合、`metasync -r` を手動で実行します。

ホットスペア

ホットスペア集合には、0 ~ n 個のホットスペアを含むことができます。1つのホットスペア集合は、複数のサブミラーや RAID5 メタデバイスと関連付けることができます。さまざまなサイズのスライスをもつホットスペア集合を1つ定義し、すべてのサブミラーや RAID5 メタデバイスと関連付けることができます。DiskSuite は、必要ならば正しいサイズのホットスペアの使用方法を判断します。

コントローラシステムの障害に耐えるために、コントローラ全体にまたがったホットスペアをホットスペア集合に置きます。この点では、サブミラーを作成する場合と同じガイドラインに従ってください。

ディスクセットの操作

この節では、ディスクセットを構成するためのヒントを紹介します。

注 - 現在、DiskSuite は、SPARCstorage Array ディスク上のディスクセットだけをサポートしています。

▼ ディスクセット用のディスクドライブデバイス名を設定する方法 (コマンド行)

ディスクセット構成で使用するハードウェアを構成するのは少々やっかいな場合があります。ディスクドライブは対称的でなければなりません。つまり、共有ドライブには同じデバイス番号が必要であるため、同じデバイス名と番号 (コントローラ / ターゲット / ドライブ) が必要になります。この作業では、この設定の構成方法について説明します。

注 - ハードウェアがあらかじめ構成されている新しいマシンセットでは、必要な対称性はデフォルトで実現されています。この作業を実行する必要はありません。

ディスクセット内にメタデバイスを作成する前に、デバイス名の設定を終了しなければなりません。非組み込みコントローラに置かれている他のドライブも、影響を受けることになります。

1. ディスクコントローラは、同じ順序で見つかるスロットに置かれていることを確認する。

このための最も良い方法は、特定の SPARCstorage Array に対するコントローラを、同一プロセッサモデルの同じスロットに置くことです。これが不可能な場合、スロットの順序が両方のプロセッサで同一になることを確認する必要があります。Sbus のプローブは規則正しく実施されるので、こうすることは可能ですが簡単ではありません。また、スロットは小さな番号のスロットから順に使用し、未使用のスロットはすべてハイエンド側に残すことをお勧めします。

注 - 構成システムは、同じ種類のコントローラ順に番号を付けます。この場合、「ディスクドライブ」がその種類にあたるため、ディスクドライブ用のすべてのコントローラは、デバイスが見つげ出される順序に影響を与えます。このため、共有されるすべてのデバイスは、システム内の他のディスクコントローラより前に置かれることになるため、正しい順序で検出されます。

これが終了すると、両方のホストマシンでインストールを完了するか、またはこの作業を続行することができます。後者の方がはるかに速くなります。

2. 1 つずつ、各ホスト上でスーパーユーザーになり、次の操作を実行します。

```
# rm /etc/path_to_inst*
# reboot -- '-rav'
reboot: rebooted by root
syncing file systems... [1] done
rebooting...
Resetting ...

Rebooting with command: -rav
Boot device: /iommu/sbus/espdma@f,400000/esp@f,800000/sd@3,0
File and args: -rav
Enter filename [kernel/unix]:
Size: 253976+126566+39566 Bytes
Enter default directory for modules [/kernel /usr/kernel]:
SunOS Release 5.4 Generic [UNIX(R) System V Release 4.0]
Copyright (c) 1983-1995, Sun Microsystems, Inc.
Name of system file [etc/system]:
The /etc/path_to_inst on your system does not exist or is empty.
Do you want to rebuild this file [n]? y
Using default device instance data
root filesystem type [ufs]:
```

(続く)

```
Enter physical name of root device
[/iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000/esp@f,800000
/sd@3,0:a]:
...
The system is ready.

console login: root
Password:<パスワードを入力>

# /usr/bin/rm -r /dev/*dsk/*
# /usr/sbin/disks
# ^D
```

ハードウェアが正しく設定されているものと仮定すると、これによってソフトウェアがその設定を反映します。/etc/path_to_inst ファイルは、ものがスライドする (コントローラの移動によってスライドしやすくなる) ことのないように使用されているため、コントローラが正しい位置にスライドするように除去されます。-rav オプション付きの reboot コマンドによって、カーネルはブート中にユーザーと対話処理を行い、再構成リブートを行います。/dev/*dsk/* の除去は、/usr/sbin/disks プログラムが実行されたときにシボリックリンクが正しく作成されるために使用されます。

注 - SPARCstorage Array コントローラには、Solaris への識別情報となる一意な World Wide Name が収められているため、SPARCstorage Array コントローラの交換には特別な作業が適用されます。詳細については、製品のご購入先にお問い合わせください。

▼ ディスクセット内で状態データベースの複製のサイズを変更する方法 (コマンド行)

ディスクセット内で状態データベースの複製のサイズを変更したい場合、基本的な手順としては、2つのディスクをディスクセットに追加し、新しいディスクの状態データベースの複製の1つを削除し、さらに他のディスクをディスクセットから削除します。その後、ディスクセットに追加したい他のディスクと一緒に、削除され

たディスクをディスクセットに追加して戻します。状態データベースの複製は、新しいサイズに合わせて自分自身を自動的にサイズ変更します。

例 - ディスクセット内で状態データベースの複製のサイズを変更

```
# metadb -s rmtic -d c1t0d0s7
# metadb -s rmtic -a -l 2068 c1t0d0s7
# metaset -s rmtic -d c1t1d0
# metaset -s rmtic -a c1t1d0
# metadb -s rmtic
```

この例では、ディスクセット rmtic に 2 つのディスクがすでに追加されており、複製が追加されるディスクの残りの部分にはデータが存在しないものと想定します。状態データベースの複製の新しいサイズは、`-l 2068` オプションで指定されるように、2068 ブロックとなります。metadb コマンドは、状態データベースの複製の新しいサイズを確認します。

ストレージマネージャの使用法

ストレージマネージャには、ディスクマネージャとファイルシステムマネージャという2つのツールがあり、ネットワーク上のサーバーでディスク構成とファイルシステムの管理を行うことができます。

この付録で取り扱う項目は次のとおりです。

- 380ページの「最初のコンテキストの読み込み」
- 381ページの「別のコンテキストの読み込み」
- 388ページの「UFS ファイルシステムの作成」
- 389ページの「マウント先の作成」
- 391ページの「マウント先またはディレクトリのプロパティの修正」
- 392ページの「ファイルシステムのマウントとマウント解除」
- 393ページの「ディレクトリの共有と共有解除」
- 395ページの「スタティックなクライアントファイルシステムの表示」
- 396ページの「アクティブなサーバーファイルシステムの表示」
- 397ページの「スタティックなサーバーファイルシステムの表示」
- 398ページの「/etc/vfstab ファイルからのマウント先の削除」
- 405ページの「表示フィルタの指定」
- 407ページの「ボリュームラベルの指定」
- 407ページの「fdisk パーティションの変更」
- 409ページの「スライス配置の変更」
- 410ページの「ディスクの複写」

「コンテキストの読み込み」プロパティブック

ストレージマネージャでは、「プロパティブック」という概念を使用しています。プロパティブックとは、ディスクやファイルシステムといった、編集可能な「オブジェクト」のプロパティを識別、表示、変更するためのメカニズムです。プロパティを各オブジェクトごとに章として扱うため、このメカニズムをプロパティブックと呼びます。すべての章について、プロパティを表示および変更することができます。

「コンテキストの読み込み」プロパティブックでは、ストレージマネージャで操作するコンテキストを設定することができます。一般的に、オブジェクトのプロパティを管理する方法をコンテキストと呼びます。ファイルシステムマネージャの場合、コンテキストには、どのホストを管理するか、どのネームサービスを変更するか、サーバー上のファイルシステムをどのようにして表示または変更するかも含まれます。ディスクマネージャの場合、システムに Solstice DiskSuite ソフトウェアがインストールされていれば、どのディスクセット (ディスクの論理グループ) を変更するかもコンテキストに含まれます。

「コンテキストの読み込み」プロパティブックは、Solstice 起動ツールからストレージマネージャを起動するときに表示されます。DiskSuite ツールの「ツール」メニューからディスクマネージャまたはファイルシステムマネージャを起動した場合は、「ファイル」メニューから「読み込み」を選択するか、どちらかのツールバー上のアイコンをクリックすると、このウィンドウを表示することができます。図 A-1 にストレージマネージャの「コンテキストの読み込み」プロパティブックを示します。

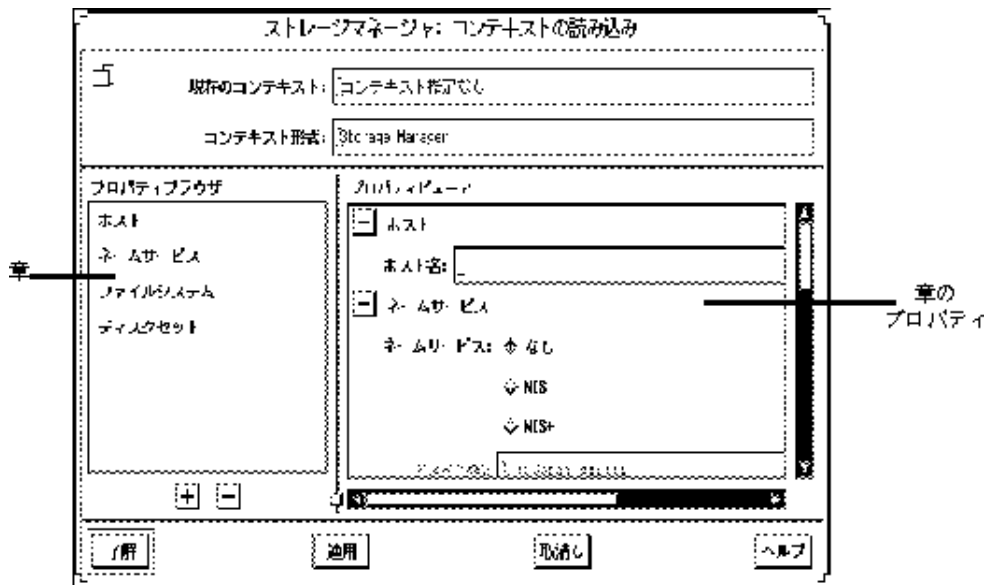


図 A-1 ストレージマネージャ: 「コンテキストの読み込み」プロパティブック
次に、「コンテキストの読み込み」プロパティブック内の各部分について簡単に説明します。

- 現在のコンテキスト

現在のコンテキストのホスト名が表示される読み取り専用フィールドです。デフォルトでは何のコンテキストも設定されていません。

- コンテキスト形式

これも読み取り専用フィールドで、現在のコンテキストと関連のあるツールの名前が示されます。図 A-1 では、ツールはストレージマネージャに設定されています。

- プロパティブラウザ

プロパティブラウザは、書籍の目次に相当します。プロパティブックにある(オブジェクトのプロパティを表わす)章と、副章(章の下位層にある章)がすべて表示されます。プロパティブラウザを使うと、希望する章に飛んで、必要に応じてその章を開いたり閉じたりすることができます。

プロパティブラウザで章をクリックすると、プロパティビューアに対応する章が開きます。また、プロパティブラウザで章をダブルクリックすると、プロパティビューア上でその章を開いたり閉じたりすることができます。

- 章

オブジェクトの共通プロパティセットを含みます。章の内容は表示および変更することができます。

- プラス (+) ボタンとマイナス (-) ボタン

プロパティブラウザの下にあるボタン。+ ボタンはプロパティビューアにあるすべての章を開きます。- ボタンは、プロパティビューアの中のすべての章を閉じます (章名のみが表示されます)。章のプロパティ全体または章の名前のみ表示したいときなどに便利です。

プロパティブックで章の名前のとりにあるボタンも上記の+ ボタンや- ボタンと同じような機能を持ちます。ただしこのボタンは、指定された章にのみ適用できます。

- プロパティビューア

現在のコンテキストを変更するときに設定できるプロパティが表示されます。デフォルトでは、このプロパティビューアにある章は開かれた状態なので、操作したいコンテキストを簡単に指定することができます。コンテキストを選択するときには、始めにホスト名を選択してください。詳細は、381ページの「別のコンテキストの読み込み」を参照してください。

- 章のプロパティ - 表示または変更したいコンテキストを設定することができます。

「コンテキストの読み込み」プロパティブックで使用できる章 (ホスト、ネームサービス、ファイルシステム、ディスクセット) の詳細は、オンラインヘルプを参照してください。

▼ 最初のコンテキストの読み込み

Solstice 起動ツールからストレージマネージャのアイコンをクリックします。または、ファイルシステムマネージャもしくはディスクマネージャの「ファイル」メニューから「読み込み」を選択して、ストレージマネージャを選択します。「コンテキストの読み込み」ウィンドウを表示してから、次の手順に従って行います。

1. **Solstice** 起動ツールやストレージマネージャを起動したシステムのコンテキストを表示または変更する場合は、「ネームサービス」、「ファイルシステム」、「ディスクセット」の章などを、必要に応じて変更する。

381ページの手順 5を参照してください。

2. **Solstice** 起動ツールやストレージマネージャを起動したシステム以外のシステムのコンテキストを表示または変更する場合は、「ホスト名」フィールドに表示さ

れている既存の名前を削除して、表示または変更したいコンテキストを持つホストの名前を指定する。

3. 「ホストを探索」をクリックする。
「システム構成の探索」ウィンドウが現われ、指定のホストから情報が更新中であることを知らせます。
4. 「ネームサービス」、「ファイルシステム」、「ディスクセット」の章などを、必要に応じて変更する。
5. 「了解」をクリックする。
「システム構成の探索」ウィンドウが現われ、指定のホストのデバイス、ディレクトリ、マウント先などの検出と一緒に、ストレージマネージャがコンテキストパラメータの妥当性を検査していることを知らせます。

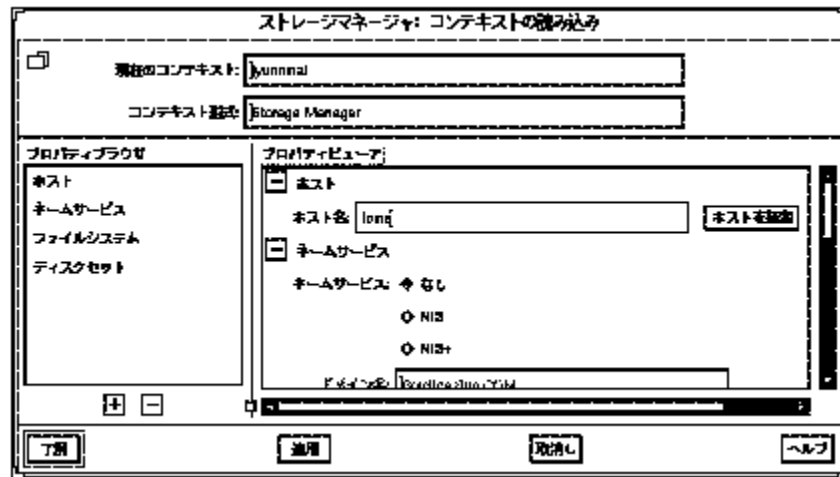
▼ 別のコンテキストの読み込み

ストレージマネージャに有効な現在のコンテキストが表示されている場合(つまり、ファイルシステムマネージャか、またはディスクマネージャのメインウィンドウが開いている場合)は、別のコンテキストを読み込むことができます。

1. 「ファイル」メニューから「読み込み」を選択するか、またはツールバーにあるアイコンをクリックする。
「コンテキストの読み込み」ウィンドウが現われ、現在のコンテキストのホスト名を表示します。
2. ホスト名フィールドに表示されている名前を削除し、表示または変更したいコンテキストをもつホスト名を入力する。
3. 「ホストを探索」をクリックする。
「システム構成の探索」ウィンドウが現われ、指定のホストから情報が更新中であることを知らせます。
4. 「ネームサービス」、「ファイルシステム」、「ディスクセット」の章などを、必要に応じて変更する。
5. 「了解」をクリックする。

「システム構成の探索」ウィンドウが現われ、指定のホストのデバイス、ディレクトリ、マウント先などの検出と一緒に、ストレージマネージャがコンテキストパラメータの妥当性を検査していることを知らせます。

例 — 別のコンテキストの読み込み



ファイルシステムマネージャの概要

ファイルシステムマネージャとは、メインウィンドウとプロパティブックという 2 種類のウィンドウを使ってファイルシステム、マウント先、ディレクトリを作成および表示するためのツールです。メインウィンドウには、ディレクトリとファイルシステム、および現在のコンテキストのマウント先と共有リソースが階層構造で表示されます。プロパティブックには、選択したディレクトリの章とそのプロパティが表示され、これを変更することもできます。

ファイルシステムマネージャで行うことのできる操作を以下に示します。

- 新しいファイルシステムの作成
- /etc/vfstab ファイルのファイルシステムオプションの変更
- 個々の、あるいはグループに属するディスククライアントや AutoClient システム上の /etc/vfstab ファイルの管理
- ファイルシステムのマウントとマウント解除

- ファイルシステムの共有と共有解除
- 既存のオートマウントマップへの、ファイルシステムの追加
- ディレクトリからマウント先への変換

上記の操作を実行する手順については、表 A-1 を参照してください。またファイルシステムマネージャのオンラインヘルプも合わせて参照してください。

ファイルシステムマネージャのメインウィンドウ

図 A-2 に、ファイルシステムマネージャのメインウィンドウの主要部分を示します。

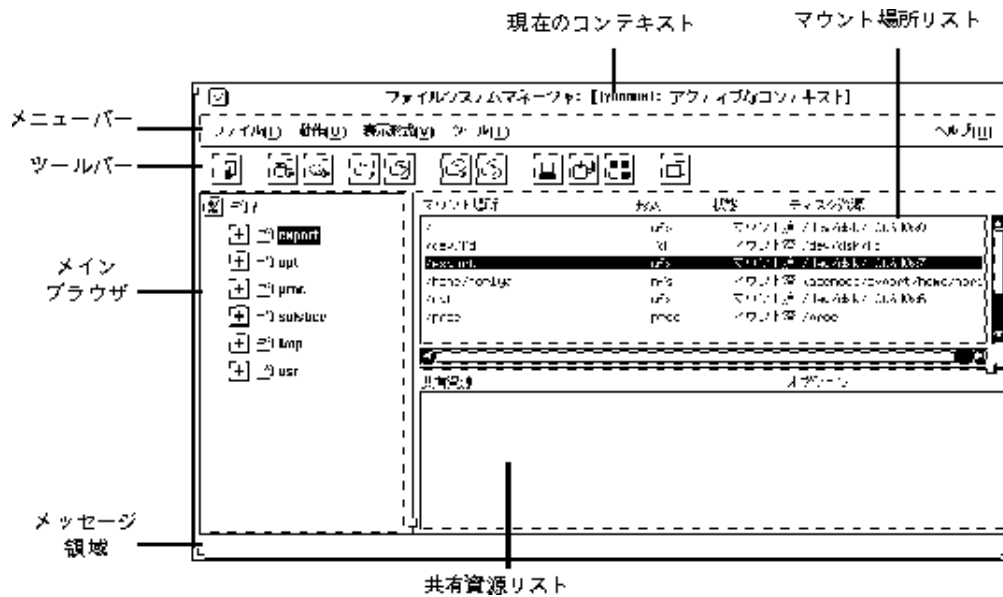


図 A-2 ファイルシステムマネージャ: メインウィンドウ

ファイルシステムマネージャのメインウィンドウの主要部分について、以下に簡単に説明します。

■ 現在のコンテキスト

ファイルシステムの管理を行なっているサーバー、またはサーバー上のクライアントを指定します。「ファイル」メニューから「読み込み」を選択するか、ツ

ルバーにあるアイコン をクリックすると現在のコンテキストを変更することができます。

- メニューバー

ファイルシステムマネージャの操作を行うためのメニューを表示します。メニューの詳細についてはオンラインヘルプを参照してください。

- ツールバー

マウント、マウント解除、他のツールの起動などメインメニューの操作の中でよく使われるものをアイコン表示して、簡単に選択できるようにしています。有効なアイコン上にマウスのポインタを動かすと、メッセージ領域にそのアイコンの機能が表示されます。「表示」メニューで「ツールバー」を選択すると、ツールバーの表示と非表示を切り換えることができます(デフォルトでは表示に設定されています)。

- メインブラウザ

現在のコンテキストのディレクトリ階層を表示します。+または- ボタンを使って、表示されているディレクトリおよびマウント先を開いたり閉じたりすることができます。ブラウザの初期画面には、ファイルシステムの最上位階層が表示されます。

- プラス (+) ボタンとマイナス (-) ボタン

表示されている階層を開く(下位層を表示する)または閉じる(上位層を表示する)ときに使います。次の3つの状態のいずれかに設定することができます。

- すべてが閉じた状態(+ が示されます)
- 管理されたオブジェクトだけが閉じた状態(+/- が示されます)
- すべてが開いた状態(- が示されます)

+/- の状態では、対応する項目が一部分だけ開いた、または閉じた状態になっています。この状態でボタンをクリックするとその項目のさらに下位の層が表示されます。

- メッセージ領域

マウスのポインタが置かれているメインウィンドウまたはアイコンについての説明が表示されます。

- マウント先リスト

現在のコンテキストで定義されている個々のマウント先の内容を表示します。各項目には、マウント先への絶対パス、マウント先にマウントされているファイル

システム、マウント先に関連する資源が現在マウントされているかどうかの情報、およびその資源名が含まれます。

- 共有資源リスト

現在のコンテキストで定義されている個々の共有資源(ディレクトリまたはマウント先)の内容が表示されます。各項目には、資源への絶対パス名と、その後でエクスポートを制御するオプション名が含まれます。

ファイルシステムマネージャのプロパティブック

ファイルシステムマネージャのプロパティブックを開く方法は3つあります。

- メインウィンドウでマウント先またはディレクトリを選択して、「動作」メニューの「プロパティ」を指定する。
- メインウィンドウでマウント先またはディレクトリを選択して、ツールバーにあるアイコンをクリックする。
- メインウィンドウで、マウント先またはディレクトリをダブルクリックする。

図 A-3 に、ファイルシステムマネージャのプロパティブックの主要部分を示します。

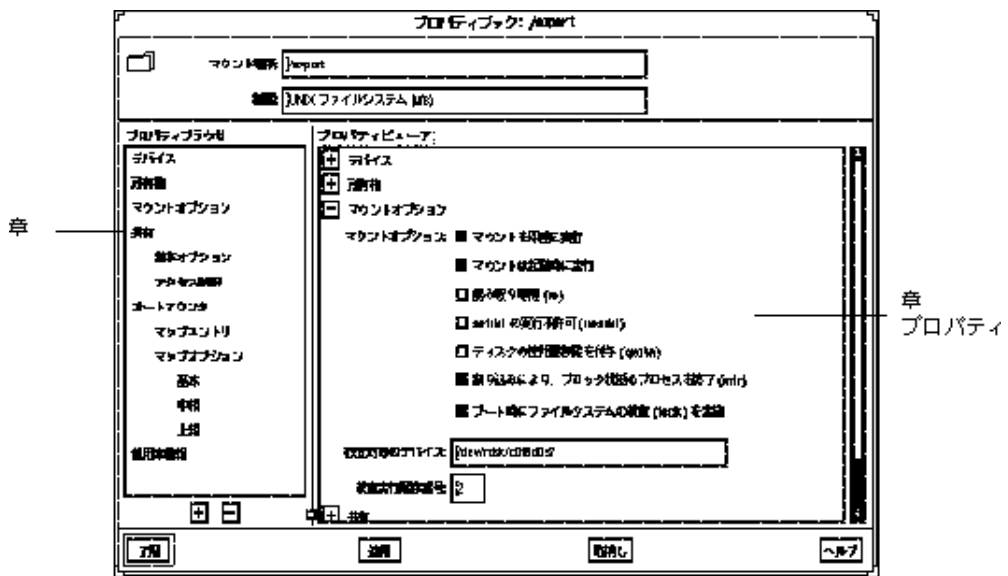


図 A-3 ファイルシステムマネージャ: プロパティブック

- マウント先

プロパティブックで管理するオブジェクトまたはオブジェクトグループを指定します。図 A-3 の例では、オブジェクトは /export マウント先です。

- 種類

マウント先の種類を指定します。

- プロパティブラウザ

ファイルシステムマネージャのプロパティブラウザは、ストレージマネージャの「コンテキストの読み込み」のプロパティブラウザとほとんど同じ働きをします。選択したディレクトリ、マウント先、またはファイルシステムのすべての章、さらに副章があればそれも含めたりリストが表示されます。

プロパティブラウザで章名をダブルクリックすると、プロパティビューアにその章が開き、副章があれば副章、なければオブジェクトのプロパティが表示されます。副章をダブルクリックするとその内容がプロパティビューアに表示されます。

- 章

オブジェクトの共通プロパティセットを含み、それを表示および変更することができます。

- プラス (+) ボタンとマイナス (-) ボタン

プロパティブラウザの下にあるボタン。+ ボタンはプロパティビューアにあるすべてのプロパティグループを開きます。- ボタンは、プロパティビューアの中のすべての章を閉じます (章名のみが表示されます)。章のプロパティ全体または章の名前のみ表示したいときなどに便利です。

プロパティブックで章の名前のとりにあるボタンも上記の+ ボタンや- ボタンと同じような機能を持ちます。ただしこのボタンは、指定された章にのみ適用できます。

- プロパティビューア

章の内容を開いたり閉じたりすることができます。章はオブジェクトプロパティのレベルまで開いて、オブジェクトのプロパティを表示および変更することができます。

- 章のプロパティ

表示または変更したいオブジェクトのプロパティを指定します。章は1つまたは複数のプロパティを持つことができます。

ファイルシステムプロパティブックで使用できる章の詳細は、オンラインヘルプを参照してください。

ファイルシステム、マウント先、ディレクトリの管理

表 A-1 操作の概要: ファイルシステムマネージャによるファイルの管理

操作	説明	参照先
UFS ファイルシステムの作成	指定のデバイス上に新しいファイルシステムを作成します。	388ページの「UFS ファイルシステムの作成」
マウント先の作成	ローカル (UFS) またはリモート (NFS) のマウント先を作成します。	389ページの「マウント先の作成」
マウント先やディレクトリのプロパティの変更	ファイルシステムのマウントやマウント解除、ディレクトリの共有や共有解除、オートマウントマップの変更などを行います。	391ページの「マウント先またはディレクトリのプロパティの修正」
ファイルシステムのマウントまたはマウント解除	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。	392ページの「ファイルシステムのマウントとマウント解除」
ディレクトリの共有または共有解除	ディレクトリを共有または共有解除します。	393ページの「ディレクトリの共有と共有解除」

表 A-1 操作の概要: ファイルシステムマネージャによるファイルの管理 続く

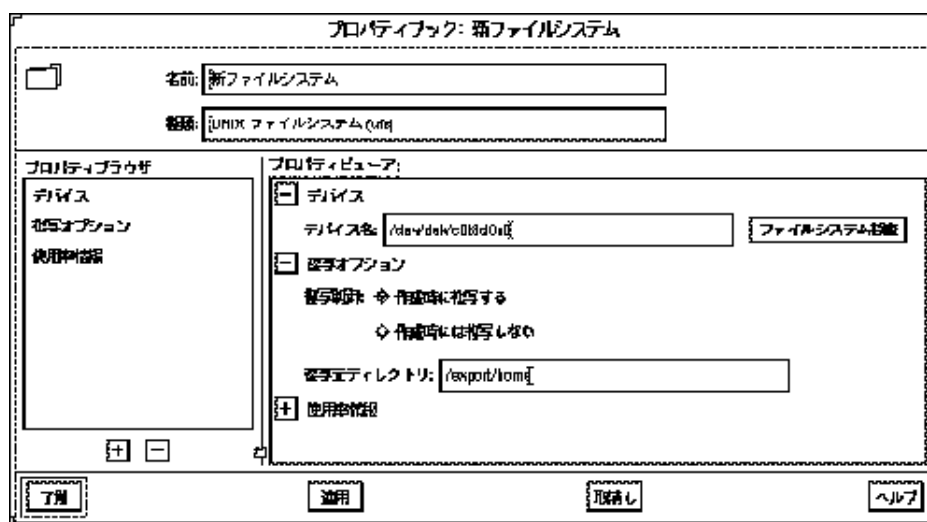
操作	説明	参照先
スタティックなクライアントファイルシステムの表示	ブート時にサーバーのディスククライアントや AutoClient システムにマウントされるファイルシステムや、共有されるディレクトリを表示します。	395ページの「スタティックなクライアントファイルシステムの表示」
アクティブなサーバーのファイルシステムの表示	サーバー上で、現在マウントされているファイルシステムや共有されているディレクトリを表示します。	396ページの「アクティブなサーバーファイルシステムの表示」
スタティックなサーバーのファイルシステムの表示	ブート時にマウントおよび共有されるファイルシステムやディレクトリを表示します。	397ページの「スタティックなサーバーファイルシステムの表示」
/etc/vfstab からのマウント先の削除	/etc/vfstab ファイルからマウント先を削除します。	398ページの「/etc/vfstab ファイルからのマウント先の削除」

▼ UFS ファイルシステムの作成

1. 「動作」メニューから「作成」->「ファイルシステム」を選択する。
「新ファイルシステム」プロパティブックが表示されます。
2. 「デバイス」の章を開く。
3. **UFS** ファイルシステムを作成する場所となる、未使用のスライスまたはメタデバイスのデバイス名を入力する。
デバイスの名前を直接入力するか、またはディスクマネージャや DiskSuite ツールからスライスまたはメタデバイスをそれぞれドラッグ & ドロップします。

4. 指定のデバイスが現在使用可能であるかどうかを検査する場合は、「ファイルシステム検査」をクリックする。
5. 既存のディレクトリの内容を新しいファイルシステムにコピーしたい場合は、「複写オプション」の章を開いて「作成時に複写」を選択する。コピーしたいファイルシステムの名前を入力する。
6. 「了解」をクリックする。

例 — UFS ファイルシステムの作成



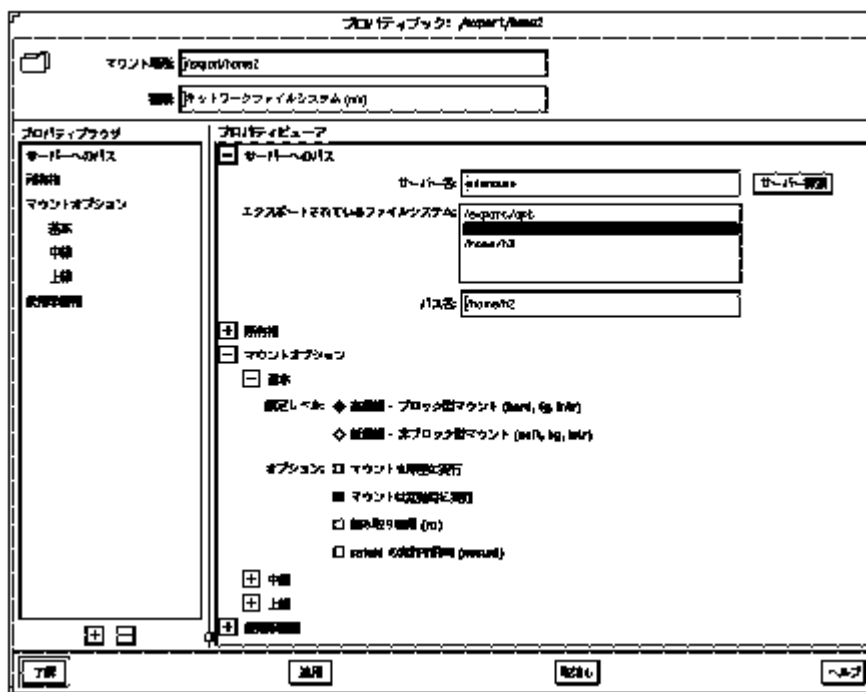
▼ マウント先の作成

1. ツールバーの適切なアイコンをクリックして **UFS** または **NFS** マウント先を作成するか、または「動作」メニューから適切なオプションを選択する。
マウント先名のウィンドウが表示されます。
2. 名前を入力して「了解」をクリックする。
マウント先のプロパティブックが表示されます。
3. **UFS** マウント先を作成する場合、390ページの手順 6に進んでください。**NFS** マウント先を作成する場合、「サーバーパス」の章を開いてください。
4. サーバーの名前を入力して「サーバー探索」をクリックする。

エクスポートされているファイルシステムのリストが、「エクスポートされているファイルシステム」リストに表示されます。

5. ファイルシステムの名前をクリックするか、「パス名」フィールドに名前を入力する。390ページの手順9に進んでください。
6. 「デバイス」の章を開く。
7. **UFS** マウント先を作成する先の、未使用スライスまたはメタデバイスの名前を入力する。
デバイス名を入力するか、またはディスクマネージャからスライスをドラッグ & ドロップします。
8. 指定のデバイスが現在使用可能であるかどうかを検査する場合は、「ファイルシステム検査」をクリックする。
9. 他の章を必要に応じて変更する。
たとえば「マウントオプション」の章では、マウントをすぐに、またはブート時に行うかを選択できます。
10. 「了解」をクリックする。
マウント先が「マウント先」リストに表示されます。

例 — マウント先の作成



▼ マウント先またはディレクトリのプロパティの修正

1. メインウィンドウからマウント先またはディレクトリを選択する。
ディレクトリやマウント先は、メインブラウザ、マウント先リスト、共有資源リストから選択できます。

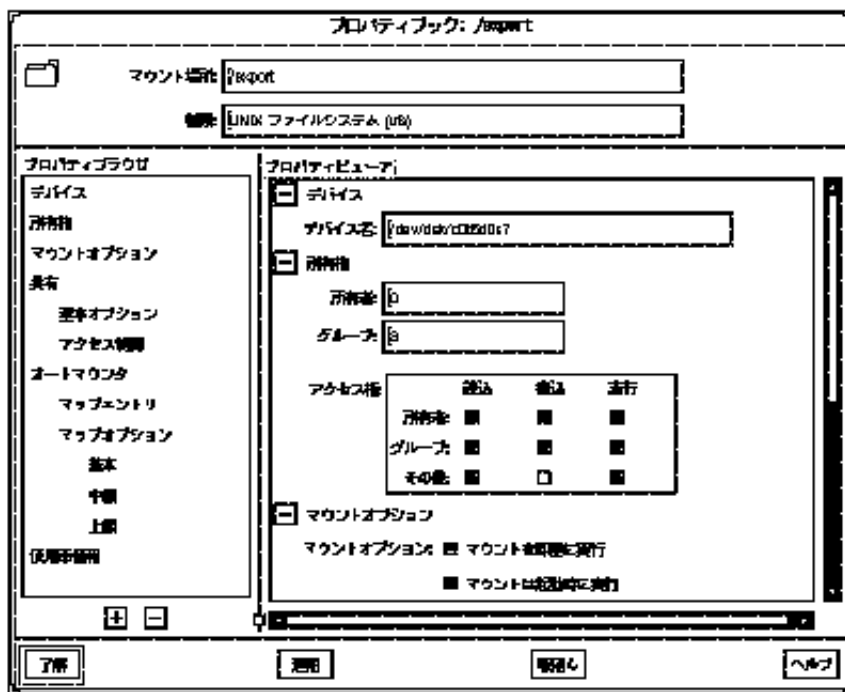
注 - 一度マウント先やディレクトリを選択すると、「動作」メニューから自動的にファイルシステムのマウントやマウント解除、ディレクトリの共有や共有解除などを行うことができます。「動作」メニューからの方がプロパティブックよりも早く処理することができます。

2. 「動作」メニューから「プロパティ」を選択する。
指定のファイルシステムやディレクトリに対応したプロパティブックが現われます。
3. 適切な章を開いて、マウント先やディレクトリのプロパティを変更する。

各章についての詳細は、オンラインヘルプを参照してください。

4. 「了解」をクリックする。

例 — ファイルシステムマネージャプロパティブック



▼ ファイルシステムのマウントとマウント解除

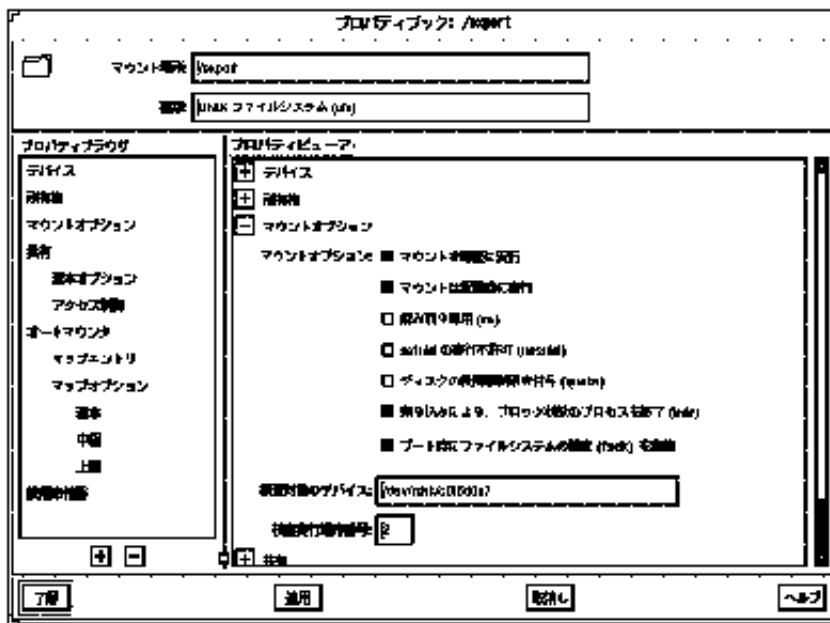
1. メインブラウザ、マウント先リスト、共有資源リストから、マウント先を選択する。

注 - 一度マウント先を選択すると、「動作」メニューから自動的にファイルシステムのマウントやマウント解除を行うことができます。「動作」メニューからの方がプロパティブックよりも早く処理することができます。

2. 「動作」メニューから「プロパティ」を選択する。
指定のファイルシステムに対応したプロパティブックが表示されます。

3. 「マウントオプション」の章を開いて、マウントオプション（たとえばファイルシステムのマウントやマウント解除）を変更する。
4. 「了解」をクリックする。

例 — ファイルシステムのマウント



▼ ディレクトリの共有と共有解除

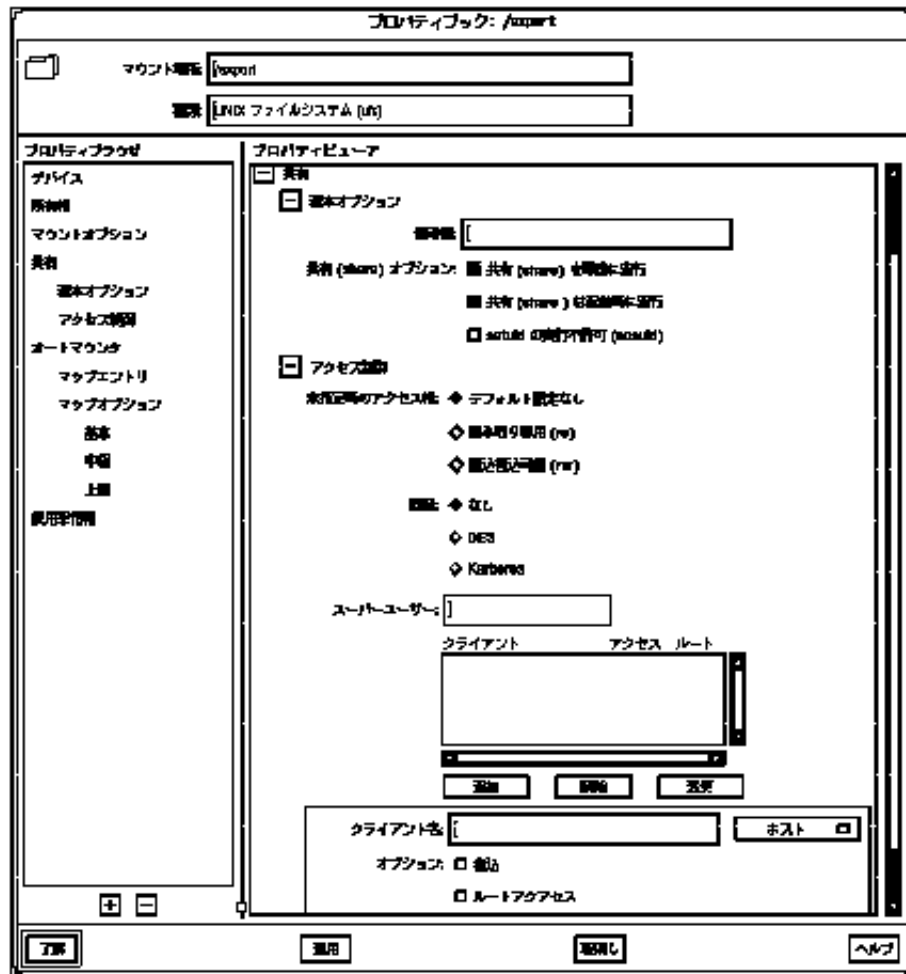
1. メインブラウザ、マウント先リスト、共有資源リストから、ディレクトリまたは **UFS** マウント先を選択する。

注 - 一度マウント先やディレクトリを選択すると、「動作」メニューから自動的に共有や共有解除などを行うことができます。「動作」メニューからの方がプロパティブックよりも早く処理することができます。

2. 「動作」メニューから「プロパティ」を選択する。
指定のファイルシステムに対応したプロパティブックが表示されます。

3. 「共有」の章を開いて、共有オプション (たとえばファイルシステムの共有や共有解除) を変更する。
4. 「了解」をクリックする。

例 — ディレクトリの共有



▼ スタティックなクライアントファイルシステムの表示

「スタティックなクライアントファイルシステム」とは、ブート時にサーバーの AutoClient またはディスクレスクライアントにマウントされるファイルシステムのことをさします。

1. ツールバーのアイコンをクリックするか、または「ファイル」メニューから「読み込み」を選択する。
「コンテキストの読み込み」プロパティブックが表示されます。
2. 「ファイルシステム」の章を開く。
3. 「スタティックなクライアント上のファイルシステム」をクリックする。

注 - このボタンは、システム上に AutoClient システムやディスクレスクライアントが構成されている場合にのみ有効になります。

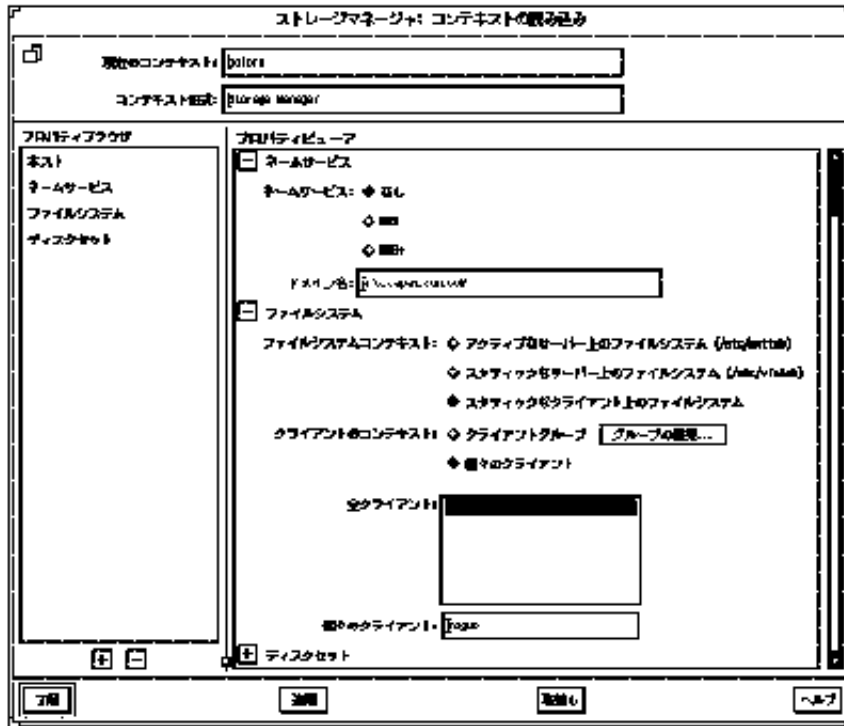
「クライアントのコンテキスト」フィールドが有効になります。

4. 「クライアントグループ」または「個々のクライアント」を選択する。
5. 「クライアントグループ」または「全クライアント」リストからシステムを選択する。
6. 「了解」をクリックする。

ブート時にクライアントにマウントされるファイルシステムは、メインウィンドウのマウント先リストに表示されます。

ブート時にクライアントに共有されるディレクトリは、メインウィンドウの共有資源リストに表示されます。

例 — スタティックなクライアントファイルシステムの表示



▼ アクティブなサーバーファイルシステムの表示

「アクティブなサーバーファイルシステム」とは、現在マウントまたは共有されているサーバー上にあるファイルシステムのことをさします。この情報は、`/etc/vfstab` ファイルにもあります。

1. ツールバーのアイコンをクリックするか、または「ファイル」メニューから「読み込み」を選択する。

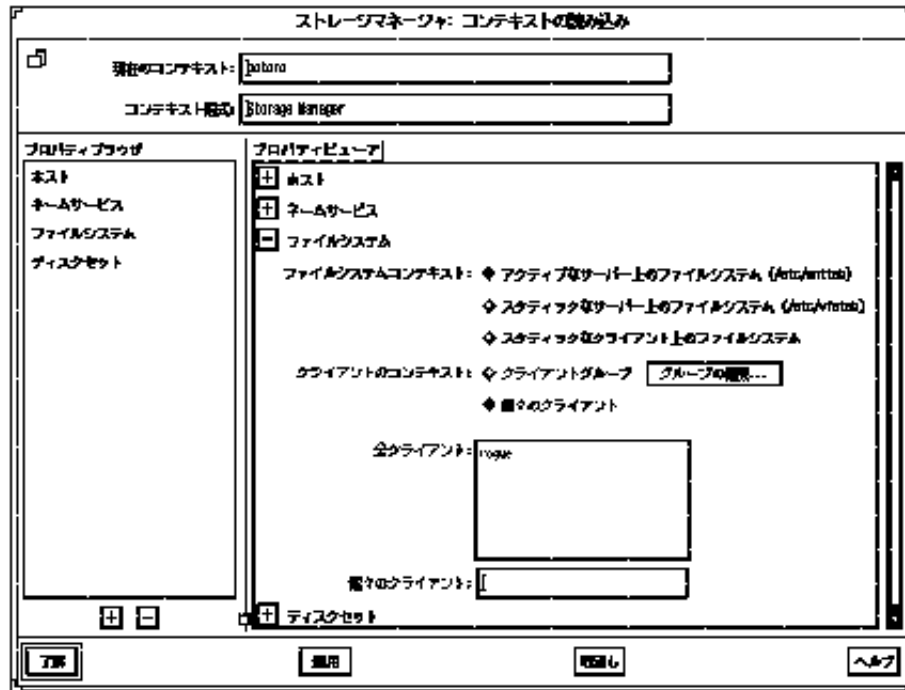
「コンテキストの読み込み」プロパティブックが表示されます。

2. 「ファイルシステム」の章を開く。
3. 「アクティブなサーバー上のファイルシステム」をクリックする。
4. 「了解」をクリックする。

現在サーバーにマウントされているファイルシステムは、メインウィンドウのマウント先リストに表示されます。

現在サーバー上で共有されているディレクトリは、メインウィンドウの共有資源リストに表示されます。

例 — アクティブなサーバーファイルシステムの表示



▼ スタティックなサーバーファイルシステムの表示

「スタティックなサーバーファイルシステム」とは、ブート時にマウントまたは共有化されるサーバー上のファイルシステムのことをさします。この情報は、`/etc/vfstab` ファイルにもあります。

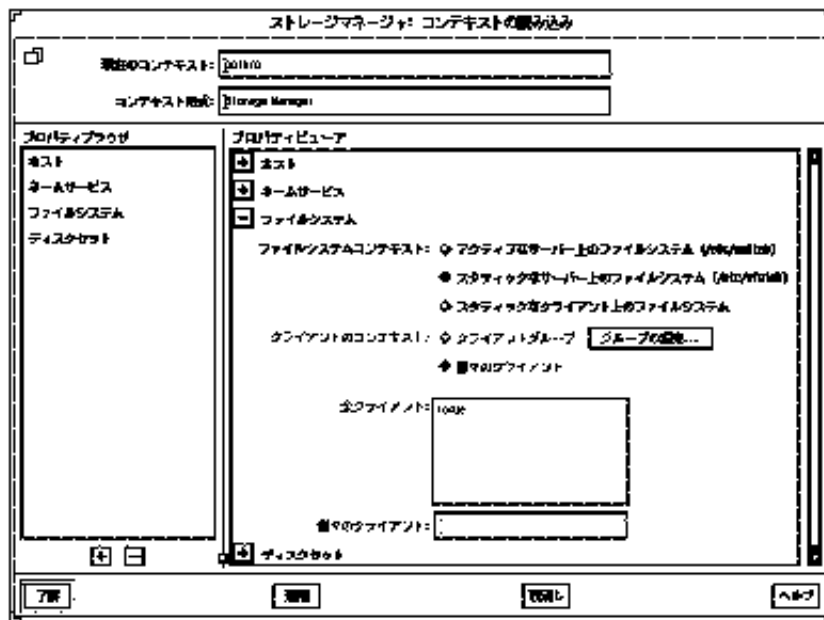
1. ツールバーのアイコンをクリックするか、または「ファイル」メニューから「読み込み」を選択する。
「コンテキストの読み込み」プロパティブックが表示されます。
2. 「ファイルシステム」の章を開く。
3. 「スタティックなサーバー上のファイルシステム」をクリックする。

4. 「了解」をクリックする。

ブート時にサーバーにマウントされるファイルシステムは、メインウィンドウのマウント先リストに表示されます。

ブート時にサーバー上で共有されるディレクトリは、メインウィンドウの共有資源リストに表示されます。

例 — スタティックなサーバーファイルシステムの表示

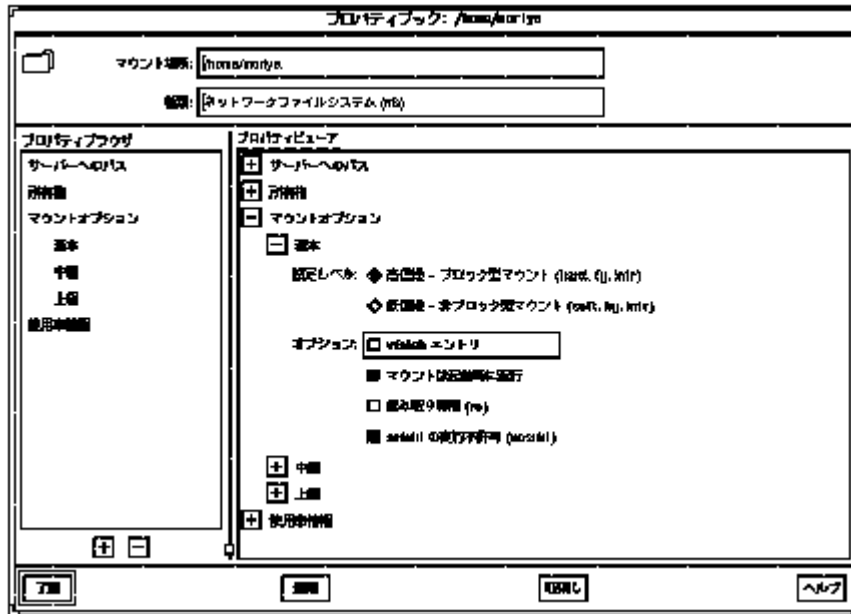


▼ /etc/vfstab ファイルからのマウント先の削除

1. 397ページの「スタティックなサーバーファイルシステムの表示」の手順を実行する。
2. メインブラウザ、マウント先リスト、共有資源リストからマウント先を選択する。
3. 「動作」メニューから「プロパティ」を選択する。
指定のファイルシステムに対応したプロパティブックが表示されます。
4. 「マウントオプション」の章を開く。

5. 「基本」の副章を開く。
6. 「**vfstab** エントリ」をクリックして、このオプションをオフにする。
7. 「了解」をクリックする。
この後、マウント先はマウント先リストに表示されなくなります。

例 — /etc/vfstab ファイルからのマウント先の削除



ディスクマネージャの概要

ディスクマネージャとは、メインウィンドウおよびプロパティブックという 2 種類のウィンドウを使って `fdisk` のパーティションとスライスを表示もしくは編集するためのツールです。メインウィンドウには、現在のコンテキストのコントローラ、ターゲット、ディスク、スライスが表示されます。プロパティブックには、指定したディスク(複数も可)の章とそのプロパティが表示され、編集することができます。

ディスクマネージャを利用して以下の操作を実行できます。

- ボリューム名をディスクに割り当てる。
- x86 と PowerPC 上で fdisk パーティションを表示、変更する。
- x86 と PowerPC 上で有効な fdisk パーティションを表示、設定する。
- SPARC、x86、PowerPC 上でスライスの配置を表示、変更する。
- ディスクの特性を同じタイプのディスクにコピーする。

注 - fdisk パーティションやスライスを変更する前に、重要なデータのバックアップをとっておいてください。

上記の操作を実行する手順については、表 A-2 を参照してください。またディスクマネージャのオンラインヘルプも合わせて参照してください。

ディスクマネージャのメインウィンドウ

図 A-4 に、ディスクマネージャのメインウィンドウの主要部分を示します。

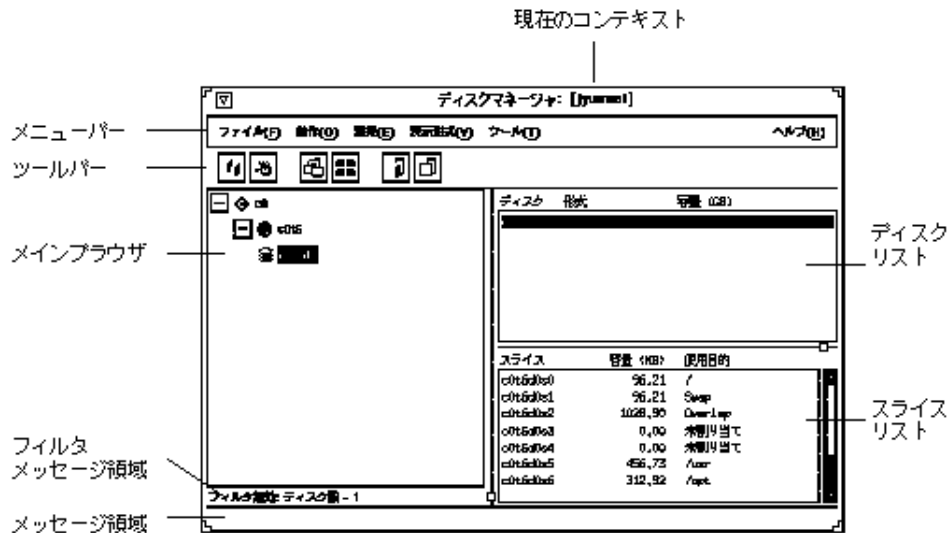


図 A-4 ディスクマネージャ: メインウィンドウ

ディスクマネージャのメインウィンドウの主要部分について以下に簡単に説明します。

- 現在のコンテキスト

ディスク管理を行なっているサーバーを指定します。「ファイル」メニューから「読み込み」を選択するか、ツールバーにあるアイコンをクリックすると現在のコンテキストを変更することができます。

- メニューバー

ディスクマネージャで操作を行うためのメニューが表示されます。メニューの詳細については、オンラインヘルプを参照してください。

- ツールバー

カット、ペースト、他のツールの起動など、メインメニューの操作の中でよく使われるものをアイコン表示して、容易に選択できるようにしています。有効なアイコン上にマウスのポインタを動かすと、メッセージ領域にそのアイコンの機能が表示されます。「表示」メニューで「ツールバー」を選択して、ツールバーの表示と非表示を切り換えることができます(デフォルトでは表示に設定されています)。

- メインブラウザ

ディスクコントローラ、SCSI ターゲット、現在のシステムコンテキストのディスクなどの階層構造が表示されます。+および- ボタンを使うと、SCSI ターゲットや各ディスクコントローラで管理されているディスクの表示状態を変更することができます。メインブラウザには、始めは現在のコンテキストで使用可能なすべてのディスクが階層表示されない状態で表示されます。

- プラス (+) ボタンとマイナス (-) ボタン

表示されている階層を開く(下位層を表示する)または閉じる(上位層を表示する)ときに使います。次の3つの状態のいずれかに設定することができます。

- すべてが閉じた状態(+ が示されます)
- 管理されたオブジェクトだけが閉じた状態(+/- が示されます)
- すべてが開いた状態(- が示されます)

+/- の状態では、対応する項目が一部分だけ開いた、または閉じた状態になっています。この状態でこれらのボタンをクリックするとその項目のさらに下位の層が表示されます。

- フィルタメッセージ領域

フィルタのオン、オフの状態が示され、オンの場合には何枚のディスクがフィルタされるかが表示されます。

- メッセージ領域

マウスのポインタが置かれているメインウィンドウまたはアイコンについての説明を表示します。

- ディスクリスト

メインブラウザで現在選択されているディスク (複数も可) の種類とサイズについての情報が表示されます。

- スライスリスト

ディスクリストで現在選択されているディスク (複数も可) のスライスとそのサイズが表示されます。

複数のディスクの選択

同じベンダー製の、物理的に同じ配置を持つディスクが複数存在する場合は、それらのディスクを同時に操作することができます。これをバッチ編集、またはバッチ処理と呼びます。メインブラウザやディスクリストから 2 台以上のディスクを選択するには、最初のディスクをクリックし、2 台め以降は **Shift** キーを押したままクリックします。

ディスクマネージャ・プロパティブック

ディスクマネージャ・プロパティブックを開く方法は 3 つあります。

- メインブラウザのディスクを選択して、「動作」メニューから「プロパティ」を選択する。
- メインブラウザのディスクを選択して、ツールバーのアイコン をクリックする。
- メインブラウザのディスクをダブルクリックする。

図 A-5 に、ディスクマネージャのプロパティブックの主要部分を示します。

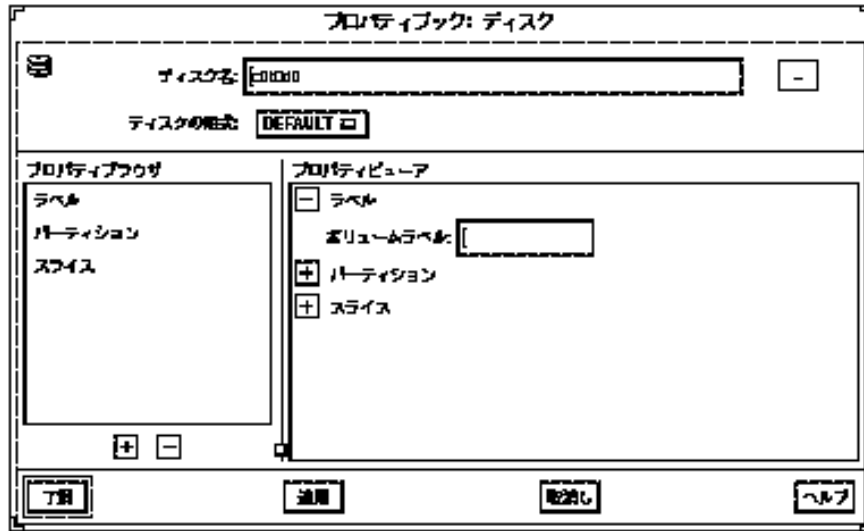


図 A-5 ディスクマネージャ: 「ディスク」プロパティブック

ディスクマネージャのメインウィンドウについて説明します。

- ディスク名

プロパティブックで管理するための1つまたは複数のオブジェクトを指定します。図 A-5 の例では、c0t6d0 という名前のディスクがオブジェクトとして指定されています。

- ... ボタン

システム上で編集されているディスクと同じタイプのものが、少なくとも1つ以上存在することを示します。このボタンをクリックすると、「編集するディスクの選択」ウィンドウが現われます。このウィンドウから、同時に変更可能な(同じタイプの)複数のディスクを選択できます。

- ディスクの形式

オブジェクトの形式を指定します。図 A-5 の例では、オブジェクトのデフォルトとして設定されたディスクの形式が指定されています。

- プロパティブラウザ

ディスクマネージャのプロパティブラウザは、ストレージマネージャの「コンテンツの読み込み」プロパティブラウザとほとんど同じ働きをします。このブラウザには、選択したディスク(複数も可)にあるすべての章、および副章があればそれも含めたリストが含まれます。

プロパティブラウザで章名をダブルクリックすると、プロパティビューアにその章が開き、副章があれば副章、なければオブジェクトのプロパティが表示されます。

- 章

オブジェクトの共通プロパティセットを含み、その内容を表示および変更することができます。

- プラス (+) ボタンとマイナス (-) ボタン

プロパティブラウザの下にあるボタンで、+ ボタンはプロパティビューアにあるすべてのプロパティグループを開きます。- ボタンは、プロパティビューアの中のすべての章を閉じます (章名のみが表示されます)。章のプロパティ全体または章の名前のみ表示したいときなどに便利です。

プロパティブックで章の名前のとりにあるボタンも上記の+ や- ボタンを同じような機能もちます。ただしこのボタンは、指定された章にのみ適用できます。

- プロパティビューア

章を開いたり閉じたりすることができます。章はオブジェクトプロパティのレベルまで開いて、オブジェクトのプロパティを表示および変更することができます。

- 章プロパティ

表示、変更したいオブジェクトのプロパティを指定します。章は1つまたは複数のプロパティを持つことができます。図 A-5 では、ラベルの章に、ディスクに名前を付けるための「ボリュームラベル」フィールドがあります。

具体的な手順については、オンラインヘルプを参照してください。

ディスクマネージャによるディスクの管理

表 A-2 操作の概要: ディスクマネージャによるディスクの管理

操作	説明	参照先
表示するフィルタの指定	ディスクマネージャのメインウィンドウに表示したいディスクの属性を指定します。	405ページの「表示フィルタの指定」
ボリュームラベルの指定	ディスクに割り当てる名前を指定します。	407ページの「ボリュームラベルの指定」
fdisk パーティションの変更	有効な fdisk パーティションの選択、サイズ変更、タイプの変更などを行います。	407ページの「fdisk パーティションの変更」
スライスの配置の変更	スライスサイズを変更します。	409ページの「スライス配置の変更」
ディスクの複写	ディスクの特性を同じタイプの別のディスクにコピーします。	410ページの「ディスクの複写」

▼ 表示フィルタの指定

1. 「表示」メニューから「フィルタ」を選択する。
「ディスクとスライスに対するフィルタ処理」ウィンドウが現われ、「有効な属性」リストに利用可能なディスクの属性をリストします。
2. メインウィンドウに表示したいディスクとディスクの属性を指定する。
 - a. 「有効な属性」リストにあるディスク属性をクリックして選択する。
 - b. 「>>」ボタンをクリックして、「ディスク表示」リストに指定の属性を移する。

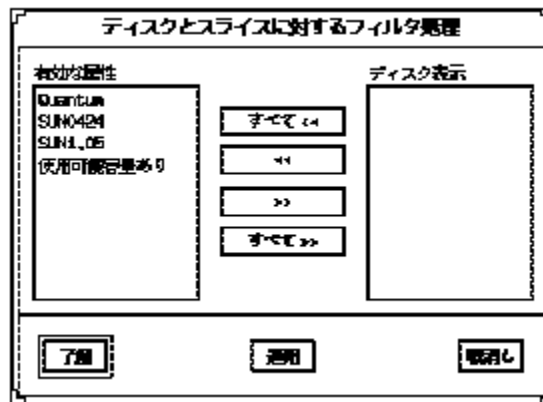
- c. メインウィンドウに表示されているディスクがもっているすべてのディスク属性が「ディスク表示」リストに入るまで、手順 **a** と手順 **b** を繰り返す。

注 - 「すべて >>」 ボタンをクリックすると、「有効な属性」リストに表示されている属性がすべて「ディスク表示」リストに移動します。また、「すべて <<」 ボタンをクリックすると、「ディスク表示」リストに表示されている属性がすべて「有効な属性」リストに移動します。

3. 「了解」をクリックする。

メインウィンドウが再表示され、「ディスク表示」リストに指定された条件に一致するディスクだけが表示されます。フィルタされたディスクの数が、メインブラウザのメッセージ領域に表示されます。

例 — 「ディスクとスライスに対するフィルタ処理」ウィンドウ

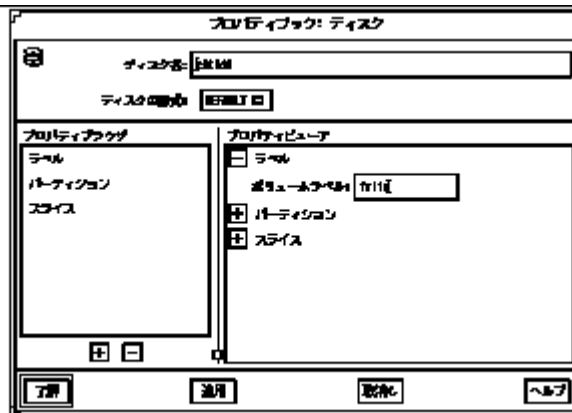


「有効な属性」フィールドには、現在のコンテキストに存在する各種類のディスクエントリ、および空き容量を持ったすべてのディスクとスライスに対応した「使用可能容量あり」エントリが表示されます。「ディスク表示」フィールドには、メインウィンドウに表示されるディスクの属性が一覧表示されます。デフォルトの状態では、すべてのディスクとその属性が表示されているので（つまりフィルタ機能はオフの状態）、このフィールドは空です。

▼ ボリュームラベルの指定

1. ディスクマネージャのメインウィンドウから、変更したいディスクを選択する。
2. 指定のディスクについてのプロパティブックを開く。
「ディスク」プロパティブック・ウィンドウが表示されます。詳細は、402ページの「ディスクマネージャ・プロパティブック」を参照してください。
3. 「ラベル」の章を開く。
詳細は、402ページの「ディスクマネージャ・プロパティブック」を参照してください。
4. 「ボリュームラベル」フィールドに名前が表示されている場合は、削除する。
5. ボリュームラベルの名前を入力する。
8文字以下の英数字を指定してください。
6. 「了解」をクリックする。

例 — ボリュームラベルの指定



▼ fdisk パーティションの変更

1. ディスクマネージャのメインウィンドウから、変更したいディスクを選択する。
2. 指定のディスクについてのプロパティブックを開く。

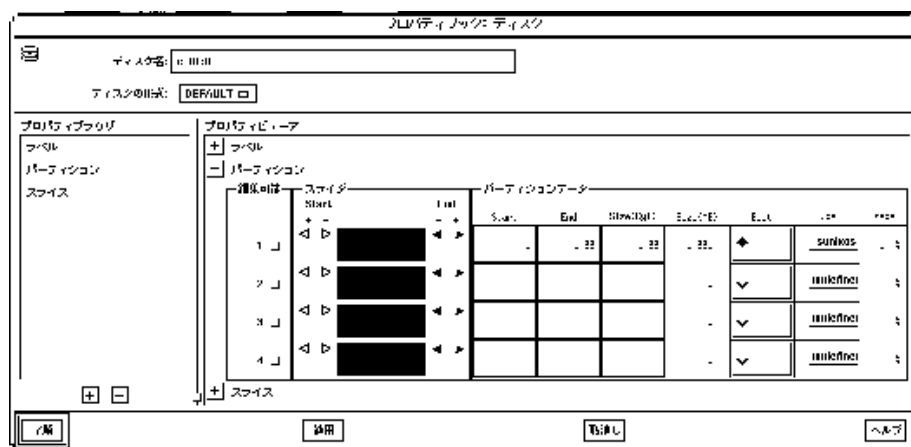
「ディスク」プロパティブック・ウィンドウが表示されます。詳細は、402ページの「ディスクマネージャ・プロパティブック」を参照してください。

3. パーティションの章を開く。
パーティションの配置状態が全体的に表示できるように、ウィンドウのサイズを調節してください。詳細は、402ページの「ディスクマネージャ・プロパティブック」を参照してください。
4. 変更したい fdisk パーティションに対応した「編集可能」ボックスをクリックして選択する。
5. fdisk パーティションのサイズを変更する。
サイズの値は「スライダ」ボックスの矢印ボタンをクリックするか、または「パーティションデータ」ボックスの Start または End フィールドをクリックして値を直接入力し、Return キーを押します。
手順についてはオンラインヘルプを参照してください。

注 - Intel プラットフォームでは、Solaris パーティション fdisk はシリンダ番号 1 以降から開始します。パーティションに重複する部分があってはなりません。

6. 必要に応じて、**Boot** カラムにあるボタンを選択して、fdisk パーティションを有効にする。つまり、その中のオペレーティングシステムを起動時に使用する。
7. fdisk パーティションのタイプを選択する。
Type カラムのメニューから、適切なタイプを選択します。
8. 「了解」をクリックする。

例 — fdisk パーティションの変更



▼ スライス配置の変更

1. ディスクマネージャのメインウィンドウから、変更したいディスクを選択する。

2. 指定のディスクについてのプロパティブックを開く。

「ディスク」プロパティブック・ウィンドウが表示されます。詳細は、402ページの「ディスクマネージャ・プロパティブック」を参照してください。

3. 「スライス」の章を開く。

スライスの配置状態が全体的に表示できるように、ウィンドウのサイズを調節してください。詳細は、402ページの「ディスクマネージャ・プロパティブック」を参照してください。

4. 変更したいスライスに対応した「編集可能」ボックスをクリックして選択する。

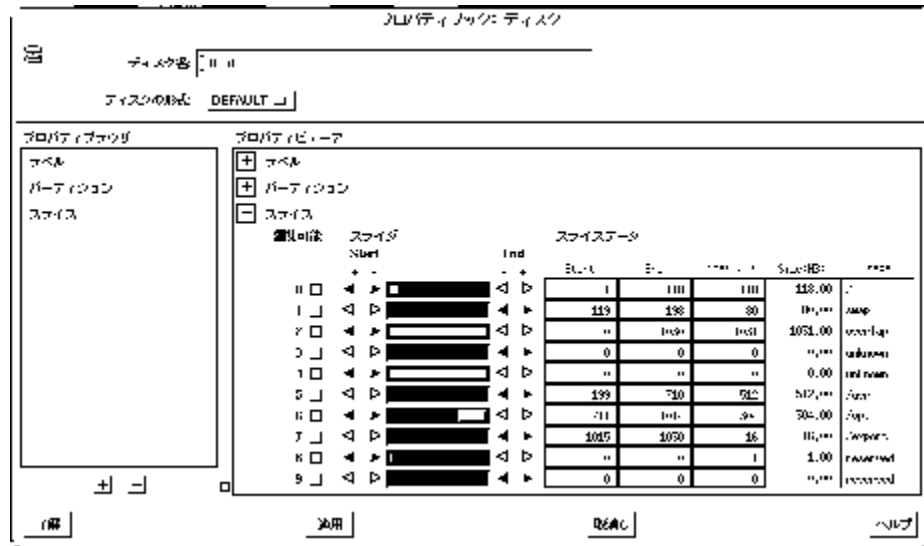
5. スライスのサイズを変更する。

サイズの値は「スライダ」ボックスの矢印ボタンをクリックするか、バーインジケータをドラッグするか、または「パーティションデータ」ボックスの Start または End フィールドをクリックして値を直接入力し、Return キーを押します。手順についてはオンラインヘルプを参照してください。

注 - Intel プラットフォームでは、スライスはシリンダ番号1以降から開始します。パーティションに重複する部分があることはありません。

6. 「了解」をクリックする。

例 — スライス配置の変更



▼ ディスクの複写

1. ディスクマネージャのメインブラウザから、複写したいディスクを選択する。
2. ツールバーのアイコンをクリックするか、または「編集」メニューから「コピー」を選択する。
3. 同じタイプの未使用ディスクを1つ選択するか、または同じタイプの未使用のディスク(複数でも可)を含んだコントローラを選択する。
4. ツールバーのアイコンをクリックするか、または「編集」メニューから「ペースト」を選択する。

注 - 手順2 から手順4 のほかに、複写したいディスクの上でマウスの中央ボタンを押したまま、同じタイプのディスクにドラッグ & ドロップしてもディスクを選択することができます。

索引

数字

3面のミラー 71

C

cron(1M) コマンド 300

D

DiskSuite

Prestoserve 336

SNMP サポートの構成 287

エラーのチェック 298

構成のガイドライン 339, 346

構成の回復 293

構成の計画 40

DiskSuite オブジェクトの状態チェック 132,
133

DiskSuite 設定の回復 293

DiskSuite ツール

SunNet Manager からの起動 286

開始 45

状態チェックに使用 132

ストレージマネージャとの統合 289

ストレージマネージャの起動 289

スライス交換の選別 349, 351

スライスサイズの選別 348, 349

制約 347

デフォルトカラーとフォントの変更 351,
357

未確定で保存された構成変更の復元 225

未確定の構成変更の保存 223

メタデバイスを使用するためのヒン
ト 348

連結方式オブジェクト 62

E

/etc/dfs/dfstabファイル 128

/etc/init.d/prestoserve ファイル 338

/etc/init.d/lvm.init ファイル 337

/etc/lvm/md.cf ファイル 293

/etc/lvm/mdlogd.cf ファイル 287

/etc/system ファイル 337

/etc/vfstabファイル 235

/etc/vfstabファイル 76, 77, 100, 106, 206, 254

ディスクセット 128

不適切なエントリからの回復 301

F

fdisk(1M)コマンド 88

fmthard(1M) コマンド 50, 52, 62, 307, 311,
346

format(1M) コマンド 50, 52, 62, 90, 307, 311

fsck(1M)コマンド 100, 103, 179, 181, 251

G

growfs(1M) コマンド 184, 185, 199

K

/kernel/drv/md.confファイル 295

L

lockfs(1M)コマンド 87, 88, 180, 370

M

md.cf ファイル

DiskSuite 設定の回復 293

md.tab ファイル 239, 293

mdlogd デーモン 287

metaclear(1M) コマンド 159, 169, 205, 243, 246, 248

metadb(1M) コマンド 48, 60, 136, 239, 297, 307, 311, 316

MetaDB オブジェクト 46, 59, 134

状態フィールド 134, 135

metadetach(1M) コマンド 169, 205, 211, 246, 253

metahs(1M) コマンド 174, 178, 255

metainit(1M) コマンド 74, 79, 86, 88, 101, 294

metaoffline(1M) コマンド 213

metaonline(1M) コマンド 214

metaparam(1M)コマンド 112, 118, 231, 258

metarename(1M) コマンド 200, 202

metareplace(1M) コマンド 165, 168, 171, 174, 312

metaroot(1M)コマンド 87, 88

metaset(1M)コマンド 121, 154, 259, 262

metastat(1M) コマンド 145, 148, 151

metatool(1M)コマンド 58, 132

metattach(1M) コマンド 74, 79, 87, 189, 192, 194, 197, 210, 294

mountall(1M) コマンド 366

N

newfs(1M)コマンド 125, 181

P

Prestoserve

DiskSuite との設定 336, 339

R

RAID5 パリティの計算 93

RAID5 メタデバイス

エラー状態の説明 162

エラーの発生したスライスの交換 171, 174

エラーの発生したスライスの有効化 170, 171

ガイドライン 343

拡張 193, 195

作成 94, 97

作成用の情報 93

状態 148, 150

初期化 96

除去 247, 248

スライス状態 149, 150

スライスの交換と有効化の概要 160

スライスを交換および有効にするための情報 163

飛び越し 94

飛び越しの設定 95

保守と最近にエラーが発生した時の状態 162

RAID5 メタデバイスでのスライスの有効化 170, 171

RAID 情報ウィンドウ 95

raw メタデバイス 66, 67, 69, 70, 73, 74, 96, 97

rename busy メッセージ 359

S

SCSI ディスク

交換 315, 319

SNMP 警告 138, 286

mdlogd デーモンの設定 287

Solaris パーティション

x86 システムに作成 88

SPARC

ルート(/)ミラーを作成 86

SPARCstorage Array

DiskSuite ツールでの表示 266

NVRAM からの書き込みをフラッシュ 276

NVRAM からの高速書き込みデータをページ 276

NVRAM の無効化 275

NVRAM の有効化 273, 275

NVRAM を有効または無効にするための情報 272

RAID5 メタデバイス内のディスクの交換 327

World Wide Name 267

World Wide Name の表示 271

コントローラのファンとバッテリーの状態をチェック 270
コンポーネントを交換するための情報 320
重大な障害状態 137
状態をチェック 269
ディスクの移動 331
ディスクの停止と起動 279, 280
ディスクを停止および起動するための情報 278
ディスクをホスト専用予約 277
デバイスの命名規則 319, 320
電源断からの回復 329, 331
トレイの交換 328
トレイの除去 327, 328
ブートデバイスにする 333
ホストによって予約されたディスクの解放 278
ミラー内のディスクの交換 321, 327
ssaadm(1M) コマンド 279
SunNet Manager
DiskSuite との統合 285
element.schema ファイル 285
swap
ミラー化 85
ミラー化解除 208
領域の割り当て 50

U
ufsdump(1M) コマンド 157
ufsrestore(1M) コマンド 158
UFS ロギング 97
/usr
ミラー化解除 206
/usr
ミラー化 84
ロギング 103, 106
ロギングの除去 251
/usr/lib/lvm/metatool - toolsmenu(4) ファイル 289

V
/var/adm/messages ファイル 316
/var/adm/messages ファイル 160

X

x86
ブート情報のインストール 92
ルート(/)ミラーを作成 88

え

エラー
スクリプト使用法をチェック 298

お

オンラインバックアップ 368

き

機密保護対策 345
共有ディスクセット 118

こ

構成の計画 40

さ

サブミラー 70
エラーの発生したスライスの交換 166, 168
エラーの発生したスライスを有効にする 164, 166
オフライン / オンライン設定 211, 214
状態 146, 150
状態キーワード 139, 140
状態をチェック 138
除去 244, 246
すべての交換 168, 169
スライス状態 147, 150
サブミラーでのスライスの有効化 164, 165
サブミラーの切断 210

し

状態 132, 155
状態キーワード
RAID5 メタデバイス 141, 143
ホットスペア集合 144, 150
状態データベースの複製 42

- 1つのスライス上に作成 49
- 新しいシステムに設定 45
- 大きな複製の追加 297
- 同じスライスに複製追加 59
- 最大数 43
- 作成用の情報 42
- 状態をチェック 133
- 初期作成 43, 50
- 除去 240, 241
- 追加作成 59, 60
- 変更 226
- 無効な複製からの回復 304
- 有効化 156
- 有効化用の情報 156
- 除去
 - SPARCstorage Array トレイ 327
 - シングルユーザーモードへのブート 371
- す
- ストライプ 61
- ストライプ方式の連結
 - ガイドライン 341
 - 最初から作成 190
 - 使用法 241, 243
 - 除去 187
- ストライプ方式メタデバイス
 - ガイドライン 340
 - 拡張 187, 190
 - 再作成 157, 160
 - 再作成用の情報 156
 - 作成 63, 67
 - 作成用の情報 61
 - 状態をチェック 138
 - 除去 241, 243
 - 別のコントローラへの移動 365
- ストレージマネージャ 62, 378
 - DiskSuite 289
 - オブジェクト定義 378
 - コンテキスト定義 378
 - 「コンテキストの読み込み」プロパティブック 378, 382
 - ディスクマネージャ説明 399
 - ファイルシステムマネージャ説明 382
 - プロパティブック定義 378
- スライス
 - RAID5 メタデバイスに追加 195
 - 拡張 184, 187

- た
- 代替ブートデバイス
 - SPARC 313
 - x86 314
- 代替ブートパス 87, 88
- 多面のストライプと連結 61
- 多面のミラーの警告 75

- て
- ディスククォータ
 - トランスメタデバイス 347
- ディスクセット
 - 解放 216, 217
 - 作成 120, 121
 - 作成のための情報 118
 - 状態のチェック 153
 - 除去 261, 262
 - 所有者の表示 154
 - 定義 118, 378
 - ディスクドライブ名の設定 372, 373
 - デフォルト数の増大 296
 - ドライブの除去 260, 261
 - ドライブの追加 122, 123, 217, 219
 - ファイルシステムの作成 128
 - 別のホストの追加 219, 220
 - ホストの除去 259
 - ミラーの作成 124
 - 予約 214, 216
- ディスクセットの解放 216
- ディスクセットの予約 214
- ディスク表示ウィンドウ
 - オブジェクトの選択 268, 269
- ディスクマネージャ
 - 説明 399
 - バッチ処理 402
 - 複数のディスクの選択 402
 - プロパティブック 402
 - メインウィンドウ 400
- 作業のまとめ 38
- デバイスの統計情報
 - グラフ化 283
 - 表示 282

- と
- 飛び越し

指定 64, 67
デフォルト 64, 94
ミラー化されたストライプ上での変更 367
飛び越し値の割り当て 65
トラブルシューティング
一般的なガイドライン 292
トランスmetaデバイス 98
/etc/vfstab ファイル 102
Prestoserve 337
metarename(1M) を使用する作成 361
metarename (1M) を使用する除去 363
rename busy エラー 359
エラーからの回復 178, 182
ガイドライン 344
拡張 195, 197
作成のための予備情報 97
状態 151, 152
状態キーワード 142, 143
状態データベースの複製 336
除去 249, 251
マウント解除できないファイルシステムからロギングを除去 251, 254
マウント解除できないファイルシステムの作成 103, 106
マウント解除できるファイルシステムの作成 99, 103
ミラーを使用する作成 106, 107

は

パス番号 230
パフォーマンス監視 281

ひ

表記上の規則

ふ

ファイルシステム
metarename (1M) を使用するミラー化解除 362
ガイドライン 345
拡張 198, 199
共有(エクスポートされた)資源として 128
パニック 178

ミラー化解除 202, 208
metaデバイス上での作成 125
連結の作成による拡張 186
ファイルシステムの拡張 198
ファイルシステムマネージャ 125
説明 382
プロパティブック 385
メインウィンドウ 383
ブート障害 300
ブートデバイス
障害からの回復 307
複製 42
複製の領域割り当て 50

ほ

ホットスペア 107
ガイドライン 344
定義 108
ホットスペア集合からの除去 255, 256
ホットスペア集合での交換 174, 176
ホットスペアへの追加 114, 116
有効化 176, 178
ホットスペア集合 107
関連付け 111, 113
関連付けの変更 116, 118
交換の情報 174
作成 109, 111
作成の情報 108
状態 153, 154
状態をチェック 133
除去 256, 258
定義 108
ホットスペア集合情報ウィンドウ 177
ホットスペア集合への追加 114, 116
ホットスペアの有効化 176, 178

ま

マスターデバイス 98, 99
ストライプ化metaデバイスを使用 102
metaデバイス名の切り替え 359

み

ミラー 70
2面のミラー 74
3面のミラー 71

- metarename(1M) による作成 360
 - エラー状態の説明 162
 - オプションの変更 229, 232
 - オンラインバックアップ 369
 - ガイドライン 341
 - 拡張 191, 192
 - 幾何学的配置 71
 - 作成の情報 70
 - サブミラーの接続 208, 210
 - サブミラーの切断 210
 - 状態キーワード 138, 139
 - 状態出力の例 145
 - 状態をチェック 138
 - 除去 244, 246
 - ストライプの飛び越し値変更 367, 368
 - スライスの交換と有効化の概要 160
 - スライスを交換および有効にするための情報 163
 - 切断とオフライン 203
 - パス番号 230
 - 保守と最近にエラーが発生した時の状態 162
 - ミラーオプション 230, 243
 - ミラー化
 - コマンド行を使用するルート(/) 86
 - マウント解除できないファイルシステム 80
 - マウント解除できるファイルシステム 74
 - 未使用のスライス 71, 73
 - ルート(/)、/usr と swap 80, 85
 - ミラー情報ウィンドウ 212
- め
- メタデバイス
 - 一般的な状態キーワード 137, 139
 - 拡張用の情報 183
 - 状態をチェック 133, 136, 151
 - デフォルト数の増大 295
 - デフォルト番号 294
 - 名前切り替え 358, 360
 - パフォーマンスの監視とグラフ化の情報 281
 - ファイルシステムを作成 125, 127
 - 命名規則 357
 - リネーム 199, 202
 - メタデバイス状態データベース 46
 - ガイドライン 340
 - 除去 239
 - 無効な複製からの回復 304
 - メタデバイスの状態データベース情報ウィンドウ 226
 - メタデバイスのリネーム 199, 200
 - メタデバイス名の切り替え 357, 360
- ら
- ラベル付きパーティション 345
- る
- ルート(/)
 - ミラー化 80, 83, 86, 87
 - ミラー化解除 206
 - ロギングの除去 251
 - ルート(/)ミラー
 - 回復 302
- れ
- 連結 61
 - 連結方式オブジェクト 62
 - 連結方式メタデバイス
 - ガイドライン 341
 - 拡張 187, 190
 - 再作成 157, 160
 - 再作成のための情報 156
 - 作成 67, 69
 - 作成のための情報 61
 - 状態チェック 138
 - 除去 241, 243
- ろ
- ローカルディスクセット 119
 - ロギングデバイス 98, 99
 - エラーからの回復 181
 - 共有 232, 236
 - 共有時の障害 180
 - 除去 248, 251
 - ハードエラー状態 179
 - 必要な領域 99