

# Oracle® ZFS Storage Appliance 管理ガイド

ORACLE®

Part No: E54238-02  
2014年6月

Copyright © 2009, 2014, Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

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

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

# 目次

---

このドキュメントの使用方法 .....	17
<b>1 Oracle ZFS Storage Appliance の概要 .....</b>	<b>19</b>
ZFSSA の主な機能 .....	19
サポートされているプロトコル .....	20
ZFSSA データサービス .....	20
データ可用性 .....	21
ZFSSA の構成 .....	21
ブラウザユーザーインターフェイス (BUI) .....	22
メインウィンドウ .....	23
一般的な使用法 .....	28
サポートされるブラウザ .....	33
コマンド行インターフェイス (CLI) .....	34
CLI へのログイン .....	35
CLI コンテキスト .....	35
以前のコンテキストに戻る .....	37
親コンテキストへの移動 .....	38
コンテキストとタブ補完 .....	38
コンテキスト固有のコマンドの実行 .....	39
コミットされていないコンテキスト .....	39
プロパティ .....	40
<b>2 ステータス .....</b>	<b>45</b>
ダッシュボード .....	46
リンク .....	46
CLI .....	53
▼ ダッシュボードの連続実行 .....	54
設定 .....	55
概要 .....	55
BUI .....	55
CLI .....	57
タスク .....	57
NDMP ステータス .....	58
NDMP のステータス - BUI .....	58

NDMP のステータス - CLI .....	61
<b>3 初期構成 .....</b>	<b>63</b>
前提条件 .....	63
BUI を使用して初期構成を実行する .....	63
▼ 初期構成を実行する .....	64
管理ポートの構成 .....	64
CLI を使用して初期構成を実行する .....	65
<b>4 ネットワーク構成 .....</b>	<b>71</b>
「ネットワーク構成」ページ .....	71
デバイス .....	73
データリンク .....	73
ネットワークインタフェース .....	76
ネットワーク IP マルチパス (IPMP) .....	77
ネットワークのパフォーマンスと可用性 .....	79
ネットワークルーティングの構成 .....	79
BUI を使用したネットワーク構成 .....	83
「ネットワーク構成」ページ .....	85
ネットワークアドレス .....	85
「ネットワークルーティング」ページ .....	86
CLI を使用したネットワーク構成 .....	86
BUI を使用したネットワーク構成タスク .....	89
▼ シングルポートインタフェースの作成 .....	89
▼ インタフェースの変更 .....	89
▼ シングルポートインタフェースの作成、ドラッグ & ドロップ .....	89
▼ LACP 集合リンクインタフェースの作成 .....	90
▼ プロブベースのリンク状態障害検出を使用した IPMP グループの作成 .....	90
▼ リンク状態のみの障害検出を使用した IPMP グループの作成 .....	91
▼ LACP アグリゲーションの拡張 .....	92
▼ IPMP グループの拡張 .....	92
▼ InfiniBand パーティションのデータリンクとインタフェースの作成 .....	92
▼ クラスタ化されたコントローラでの VLAN ID なしの VNIC の作成 .....	93
▼ クラスタ化されたコントローラでの同じ VLAN ID の VNIC の作成 .....	94
▼ 静的ルートの追加 .....	95
▼ 静的ルートの削除 .....	95
CLI を使用したネットワーク構成タスク .....	95
▼ 静的ルートの追加 .....	95
▼ 静的ルートの削除 .....	95
▼ マルチホーミングプロパティの「厳しい」への変更 .....	96

<b>5 ストレージ構成</b> .....	97
ストレージ構成プロファイル .....	98
ストレージ構成の規則とガイドライン .....	99
ストレージの検証 .....	99
SAS-2 システムでのストレージの割り当て .....	100
データプロファイル構成 .....	100
既存のストレージプールのインポート .....	103
ストレージの追加 .....	103
ストレージの構成解除 .....	104
ストレージプールのスクラブ .....	104
BUI を使用したストレージの構成 .....	104
▼ ストレージプールの構成 .....	104
▼ 既存のプールへのキャッシュデバイスの追加 .....	105
CLI を使用したストレージの構成 .....	105
▼ 既存のプールへのキャッシュデバイスの追加 .....	105
<b>6 Storage Area Network の構成</b> .....	109
SAN のターゲットとイニシエータ .....	109
SAN のターゲットグループとイニシエータグループ .....	109
BUI を使用した SAN の構成 .....	110
CLI を使用した SAN の構成 .....	111
SAN の用語 .....	111
SAN ファイバチャネル .....	114
FC ポートターゲットの構成 .....	114
FC イニシエータの構成 .....	115
パフォーマンスの考慮事項 .....	115
FC のトラブルシューティング .....	116
BUI を使用した FC の構成 .....	117
CLI を使用した FC の構成 .....	121
iSCSI .....	124
ターゲットの構成 .....	124
イニシエータの構成 .....	126
クライアント構成の計画 .....	126
iSCSI のトラブルシューティング .....	127
iSCSI のパフォーマンスのモニタリング .....	127
BUI を使用した iSCSI の構成 .....	127
CLI を使用した iSCSI の構成 .....	130
SRP .....	132
SRP ターゲットの構成 .....	132
イニシエータの構成 .....	133
SRP のパフォーマンスの監視 .....	133
BUI を使用した SRP ターゲットの構成 .....	133
CLI を使用した SRP ターゲットの構成 .....	134

<b>7 ユーザー構成</b> .....	137
ユーザーロール .....	137
ユーザー承認 .....	138
ユーザープロパティの管理 .....	139
ユーザープロパティ .....	139
ロールのプロパティ .....	140
BUI の「ユーザー」ページ .....	140
BUI を使用したユーザーの構成 .....	141
▼ 管理者の追加 .....	141
▼ ロールの追加 .....	141
▼ ロールへの承認の追加 .....	142
▼ ロールからの承認の削除 .....	142
▼ ダッシュボードのみを表示できるユーザーの追加 .....	142
CLI を使用したユーザーの構成 .....	143
CLI のユーザー構成の例 .....	143
▼ 管理者の追加 .....	144
▼ ロールの追加 .....	145
▼ ロールへの承認の追加 .....	145
▼ ロールからの承認の削除 .....	146
<b>8 ZFSSA の設定</b> .....	147
設定のプロパティ .....	147
CLI を使用した設定 .....	148
CLI を使用した SSH 公開鍵の設定 .....	148
<b>9 警告の構成</b> .....	151
警告のカテゴリ .....	151
サポートされている警告アクション .....	152
しきい値警告 .....	154
BUI を使用した警告の構成 .....	155
CLI を使用した警告の構成 .....	156
<b>10 クラスタ構成</b> .....	159
クラスタの機能と利点 .....	159
クラスタのデメリット .....	160
クラスタの用語 .....	161
クラスタ化の理解 .....	162
クラスタ相互接続 I/O .....	163
クラスタリソース管理の理解 .....	165
クラスタのテイクオーバーとフェイルバック .....	169
クラスタ化された環境での構成変更 .....	171
クラスタ化におけるストレージの考慮点 .....	172

クラスタ化におけるネットワークの考慮点 .....	174
プライベートのローカル IP インタフェース .....	176
クラスタ化における Infiniband の考慮点 .....	176
クラスタ化における冗長パスのシナリオ .....	177
「スプリットブレイン」状態の回避 .....	178
テイクオーバーの影響の見積もりと削減 .....	181
BUI を使用したクラスタ構成 .....	183
▼ クラスタ化の構成 .....	184
▼ クラスタ化の構成解除 .....	185
CLI を使用したクラスタ化の構成 .....	186
▼ クラスタ構成をシャットダウンする .....	186
▼ スタンバイヘッドをシャットダウンする .....	187
▼ クラスタ化の構成解除 .....	188
クラスタノードの配線 .....	189
ストレージシェルフの配線 .....	190
クラスタ構成の BUI ページ .....	190
<b>11 ZFSSA サービス .....</b>	<b>193</b>
使用可能なサービス .....	193
データサービス .....	194
ディレクトリサービス .....	194
サービス設定 .....	195
リモートアクセスサービス .....	196
セキュリティーサービス .....	196
最小限必要なポート .....	196
BUI を使用したサービスの構成 .....	197
▼ 特定のサービス画面の表示 .....	197
▼ 特定のサービス画面の表示 .....	198
▼ サービスの有効化 .....	198
▼ サービスの無効化 .....	198
▼ プロパティの定義 .....	198
▼ サービスのログの表示 .....	198
CLI を使用したサービスの構成 .....	199
▼ サービスの選択 .....	201
▼ サービスの状態の表示 .....	201
▼ サービスの有効化 .....	201
▼ サービスの無効化 .....	201
▼ プロパティの設定 .....	201
▼ サービスのヘルプの表示 .....	202
NFS .....	202
プロパティ .....	203
Kerberos レルム .....	205
サービスのログ .....	206

NFS の分析 .....	206
NFS の BUI および CLI のプロパティ .....	207
▼ NFS を介したファイルシステムのシェア .....	207
iSCSI サービス .....	208
iSCSI サービスのプロパティ .....	208
iSCSI サービスの認証 .....	209
iSCSI サービスの承認 .....	209
iSCSI サービスのターゲットとイニシエータ .....	209
iSCSI のトラブルシューティング .....	209
SMB サービス .....	210
SMB サービスのプロパティ .....	210
SMB のシェアプロパティ .....	211
NFS と SMB の相互運用性 .....	212
SMB の DFS 名前空間 .....	213
▼ 例: DFS ネームスペースの操作 .....	215
SMB の自動ホームサービス .....	215
SMB のローカルグループ .....	217
SMB のローカルアカウント .....	218
SMB の MMC の統合 .....	218
BUI を使用した SMB の構成 .....	223
FTP サービス .....	226
FTP のプロパティ .....	226
FTP のログ .....	228
BUI を使用した FTP の構成 .....	228
HTTP サービス .....	228
HTTP のプロパティ .....	229
HTTP の認証とアクセス制御 .....	229
HTTP のログ .....	230
HTTP の構成 .....	230
NDMP サービス .....	230
NDMP のローカル構成とリモート構成の違い .....	231
NDMP のバックアップの形式とタイプ .....	231
NDMP の増分バックアップ .....	235
NDMP のプロパティ .....	237
NDMP のログ .....	239
リモートレプリケーション .....	239
シャドウ移行 .....	240
シャドウ移行のプロパティ .....	240
SFTP サービス .....	240
SFTP のプロパティ .....	240
SFTP ポート .....	241
SFTP のログ .....	241
SFTP の構成 .....	241



SRP サービス .....	243
TFTP サービス .....	244
TFTP のプロパティ .....	244
TFTP の構成 .....	244
ウイルススキャンサービス .....	245
ウイルススキャンのプロパティ .....	245
ウイルススキャンのログ .....	247
ウイルススキャンの構成 .....	247
NIS サービス .....	248
NIS のプロパティ .....	248
NIS のログ .....	249
NIS の構成 .....	249
LDAP サービス .....	250
LDAP のプロパティ .....	250
LDAP のカスタムマッピング .....	251
LDAP のログ .....	253
LDAP の構成 .....	253
Active Directory .....	254
Active Directory のプロパティ .....	254
Active Directory のドメインとワークグループ .....	255
Active Directory の LDAP 署名 .....	256
Active Directory の Windows Server 2012 でのサポート .....	256
Active Directory の Windows Server 2008 でのサポート .....	256
BUI を使用した Active Directory の構成 .....	258
CLI を使用した Active Directory の構成 .....	258
アイデンティティマッピングサービス .....	260
アイデンティティマッピングのプロパティ .....	260
アイデンティティマッピング規則 .....	261
アイデンティティマッピングのマッピング .....	262
アイデンティティマッピングのログ .....	262
アイデンティティマッピングのベストプラクティス .....	263
アイデンティティマッピングの概念 .....	263
アイデンティティマッピングの例 .....	265
アイデンティティマッピングの構成 .....	266
DNS サービス .....	267
DNS のプロパティ .....	268
DNS の構成 .....	268
DNS のログ .....	268
Active Directory と DNS .....	269
DNS 以外での解決 .....	269
DNS を使用しない操作 .....	269
動的ルーティングサービス .....	270
RIP および RIPng 動的ルーティングプロトコル .....	270

動的ルーティングのログ .....	270
IPMP サービス .....	270
IPMP のプロパティ .....	271
IPMP のログ .....	271
NTP サービス .....	271
NTP のプロパティ .....	272
NTP の BUI のクロック .....	273
NTP のヒント .....	273
BUI を使用した NTP の構成 .....	274
CLI を使用した NTP の構成 .....	274
フォンホームサービス .....	275
Oracle シングルサインオンアカウント .....	276
フォンホームのプロパティ .....	276
アプライアンスの登録 .....	277
フォンホームのステータス .....	278
フォンホームの状態 .....	278
フォンホームのログ .....	278
REST .....	279
RESTful API .....	279
サービスタグ .....	279
サービスタグのプロパティ .....	279
SMTP サービス .....	280
SMTP のプロパティ .....	280
SMTP のログ .....	281
SNMP サービス .....	281
SNMP のプロパティ .....	281
SNMP の MIB .....	282
Sun FM MIB .....	283
Sun AK MIB .....	283
SNMP の構成 .....	284
Syslog サービス .....	285
Syslog のプロパティ .....	286
Classic Syslog: RFC 3164 .....	286
最新の Syslog: RFC 5424 .....	286
syslog メッセージの書式 .....	287
レシーバーの構成例 .....	289
システム ID .....	291
システム ID のプロパティ .....	291
システム ID のログ .....	291
SSH サービス .....	291
SSH のプロパティ .....	292
SSH のログ .....	292
SSH の構成 .....	292

<b>12 シェア、プロジェクト、およびスキーマ</b> .....	293
シェアについて .....	294
ストレージプール .....	294
シェアの使用 .....	295
シェアのプロパティ .....	295
シェアのスナップショット .....	296
シェアのクローン .....	297
シェアのスペース管理 .....	298
シェアのスペースに関する用語 .....	298
スナップショットについて .....	299
ファイルシステムとプロジェクトの設定 .....	300
ユーザーとグループの設定 .....	302
ファイルシステムの名前空間 .....	306
名前空間の入れ子のマウントポイント .....	306
名前空間のマウントポイントへのプロトコルアクセス .....	307
「シェア」>「シェア」 .....	309
BUI での「シェア」>「シェア」の操作 .....	309
CLI での「シェア」>「シェア」の操作 .....	315
「シェア」>「シェア」>「一般」- BUI ページ .....	320
領域の使用 .....	321
マウントポイント .....	322
読み取り専用 .....	322
読み取り時のアクセス時間の更新 .....	322
ブロック不可の必須ロック .....	323
データ複製解除 .....	323
データ圧縮 .....	324
チェックサム .....	325
キャッシュデバイスの使用状況 .....	325
同期書き込みバイアス .....	326
データベースレコードサイズ .....	327
追加レプリケーション .....	328
ウイルススキャン .....	328
破棄の防止 .....	328
所有権の変更の制限 .....	329
カスタムプロパティ .....	329
「シェア」>「シェア」>「プロトコル」- BUI ページ .....	329
シェアのプロトコル .....	329
シェアのプロトコル - NFS .....	330
シェア - SMB .....	335
シェア - iSCSI .....	336
シェア - HTTP .....	337
シェア - FTP .....	337
シェア - SFTP .....	338

「シェア」>「シェア」>「アクセス」 .....	338
アクセス制御 .....	338
シェア - ルートディレクトリアクセス .....	338
シェア - ACL の動作 .....	340
ルートディレクトリ ACL .....	342
シェア - スナップショット .....	346
シェア - スナップショットのプロパティ .....	346
BUI を使用したスナップショットの一覧表示 .....	347
BUI を使用した手動スナップショット .....	348
BUI を使用した定期スナップショット .....	351
CLI を使用した手動スナップショット .....	352
プロジェクト .....	356
BUI を使用したプロジェクトの操作 .....	356
CLI を使用したプロジェクトの操作 .....	358
プロジェクト - 一般 .....	363
プロジェクトのプロトコル .....	365
プロジェクトのアクセス .....	366
プロジェクトのスナップショット .....	366
スキーマ .....	368
シェアのカスタムプロパティ .....	368
BUI でのスキーマの操作 .....	368
CLI を使用したスキーマの操作 .....	370
▼ CLI を使用したスキーマの構成 .....	371
<b>13 レプリケーション</b> .....	<b>373</b>
レプリケーションの概要 .....	373
レプリケーションの理解 .....	375
レプリケーションの用語 .....	375
プロジェクトレプリケーションターゲット .....	375
プロジェクトレプリケーションアクションおよびパッケージ .....	376
プロジェクトレプリケーションのストレージプール .....	378
プロジェクトレベルのレプリケーションとシェアレベルのレプリケーションの比較 .....	379
プロジェクトレプリケーションの構成 .....	380
ターゲットの作成と編集 .....	380
▼ BUI でのターゲットの作成および編集 .....	380
▼ CLI を使用したターゲットの作成および編集 .....	381
アクションの作成と編集 .....	381
▼ BUI でのアクションの作成および編集 .....	383
▼ CLI でのアクションの作成および編集 .....	384
レプリケーションモード: 「定期」または「連続」 .....	386
レプリケーション - 中間スナップショットを含める .....	386
レプリケーション - 更新の送信と取り消し .....	386

レプリケーションパッケージの管理 .....	387
BUI でのレプリケーションパッケージの管理 .....	388
CLI を使用したレプリケーションパッケージの管理 .....	389
レプリケーション更新の取り消し .....	390
パッケージの無効化 .....	391
パッケージまたは個々のシェアのクローニング .....	391
レプリケートされたファイルシステムのエクスポート .....	392
レプリケーションの切断 .....	393
レプリケーションの方向を逆にする .....	394
レプリケーションパッケージの破棄 .....	396
レプリケーションタスク .....	396
レプリケーションを逆向きにする - レプリケーションの確立 .....	396
▼ レプリケーションを逆向きにする .....	396
レプリケーションを逆向きにする - 障害からの回復のシミュレート .....	397
▼ レプリケーションを逆向きにする .....	398
レプリケーションを逆向きにする - 本番システムからのレプリケーションの再開 .....	399
▼ レプリケーションを逆向きにする .....	399
レプリケーションでの静的ルートの強制的な使用 .....	400
▼ レプリケーションでの静的ルートの強制的な使用 .....	401
受信したレプリケーションプロジェクトのクローニング .....	403
リモートレプリケーションの詳細 .....	403
承認 .....	403
警告 .....	404
レプリケーション監査イベント .....	405
レプリケーションとクラスタ化 .....	405
スナップショットとデータ整合性 .....	406
スナップショットの管理 .....	406
iSCSI 構成のレプリケーション .....	408
クローンのレプリケーション .....	408
レプリケーションの監視 .....	409
レプリケーションの失敗 .....	409
レプリケーションの互換性 .....	412
2009.Q3 以前からのアップグレード .....	412
<b>14 シャドウ移行 .....</b>	<b>415</b>
データの移行 .....	415
従来のデータ移行 .....	415
シャドウ移行 .....	417
シャドウ移行の動作 .....	418
シャドウソースの制限事項 .....	418
移行時のシャドウファイルシステムのセマンティクス .....	418
ID と ACL の移行 .....	419

シャドウ移行の管理 .....	420
シャドウファイルシステムの作成 .....	420
バックグラウンド移行の管理 .....	420
移行エラーの処理 .....	421
移行の進行状況のモニタリング .....	421
移行の取り消し .....	422
シャドウファイルシステムのスナップショット .....	422
シャドウファイルシステムのバックアップ .....	423
シャドウファイルシステムのレプリケート .....	423
シャドウ移行の分析 .....	423
ローカルファイルシステムの移行 .....	424
シャドウ移行のタスク .....	425
▼ シャドウ移行の可能性のテスト .....	425
▼ アクティブな NFS サーバーからのデータの移行 .....	425
<b>15 CLI のスクリプト化 .....</b>	<b>427</b>
アクセスの自動化 .....	427
コマンドのバッチ処理 .....	427
コマンドのスクリプト化 .....	428
スクリプト環境 .....	428
システムとの対話 .....	429
出力の生成 .....	433
エラーの処理 .....	434
<b>16 保守のワークフロー .....</b>	<b>435</b>
ワークフローの使用 .....	435
ワークフローの実行コンテキスト .....	436
ワークフローのパラメータ .....	436
制約付きのパラメータ .....	438
オプションのパラメータ .....	439
ワークフローのエラー処理 .....	439
ワークフローの入力の検証 .....	440
ワークフローの実行の監査 .....	441
ワークフローの実行の報告 .....	441
バージョン管理 .....	443
アプライアンスのバージョン管理 .....	443
ワークフローのバージョン管理 .....	444
警告アクションとしてのワークフロー .....	444
警告アクションの実行コンテキスト .....	444
警告アクションの監査 .....	445
スケジュールされたワークフローの使用 .....	446
CLI の使用 .....	446

スケジュールのコーディング .....	447
例: デバイスタイプの選択 .....	449
BUI .....	451
CLI .....	452
ワークフローのダウンロード .....	452
ワークフローの表示 .....	453
ワークフローの実行 .....	453
<b>17 統合 .....</b>	<b>455</b>
Oracle Exadata Database Machine Backup .....	456
Sun ZFS Storage Appliance の手動構成 .....	456
ネットワーク、プール、およびシェアの構成 .....	457
Oracle RMAN および Oracle Database インスタンスの構成 .....	459
次のステップ .....	460
Sun ZFS Storage Appliance 用の Oracle Exadata の構成 .....	460
Exadata の構成 Sun ZFS Storage Appliance 用の Oracle Exadata の構成 .....	461
詳細な実装手順 .....	461
Oracle SPARC SuperCluster のバックアップ .....	465
バックアップ用の ZFS Storage Appliance の構成 .....	466
ZFS Storage Appliance InfiniBand データリンクの構成 .....	466
ZFS Storage Appliance を追加する Oracle SPARC SuperCluster InfiniBand スイッチの構成 .....	467
シングル IP 接続用の ZFS Storage Appliance ネットワークの構成 ....	470
アクティブ/アクティブ構成用の ZFS Storage Appliance ネットワークの 構成 .....	471
ZFS Storage Appliance ストレージプールの構成 .....	473
ZFS Storage Appliance のシェアの構成 .....	473
ZFS Storage Appliance DTrace Analytics の構成 .....	474
クライアント NFS マウントの構成 .....	475
Solaris 11 ネットワークおよびカーネルのチューニング .....	476
Oracle Direct NFS (dNFS) の構成 .....	476
Oracle RMAN バックアップおよび復元用の Oracle Database インスタ ンスの調整 .....	477
Oracle RMAN 操作の専用サービスの作成 .....	479
Oracle RMAN の構成 .....	480
次のステップ .....	484
ZFS Storage Appliance のバックアップのための Oracle SPARC SuperCluster の構成 .....	484
SSC の構成 ZFS Storage Appliance のバックアップのための Oracle SPARC SuperCluster の構成 .....	484
詳細な実装手順 .....	485
Oracle Intelligent Storage Protocol .....	488
最適なファイルレコードサイズを設定します .....	489

---

要求ごとに ZFS の「待機時間」または「スループット」書き込みモードのい ずれかを使用します .....	489
Oracle Solaris Cluster 用 Sun ZFS Storage Appliance ネットワークファイ ルシステムプラグイン .....	489
Oracle Solaris Cluster Geographic Edition 用 Sun ZFS Storage Appliance プラグイン .....	489
Oracle Enterprise Manager Grid Controller 用 Sun ZFS Storage Management プラグイン .....	490
Oracle Enterprise Manager Grid Controller 用 Oracle Grid Controller Sun ZFS Storage Management プラグイン .....	491
Sun ZFS Storage Appliance 用 Oracle Virtual Machine Storage Connect プラグイン .....	492
ボリュームシャドウコピーサービスソフトウェア用 Sun ZFS Storage Appliance プロバイダ .....	492
Symantec の「DMP」/Storage Foundation での FC サポート .....	493
次の OS バージョンにおける Symantec Storage Foundation 5.1RP2 以降の FC サポート .....	493
Sun ZFS Storage 7000 Storage Replication Adapter for VMware Site Recovery Manager .....	494
索引 .....	497



## このドキュメントの使用方法

---

- 概要 – Oracle ZFS Storage Appliance を管理する方法について説明します
- 対象読者 – 技術者、システム管理者、および認定サービスプロバイダ
- 前提知識 – Oracle ZFS Storage Appliance の操作経験

### 製品ドキュメントライブラリ

製品ドキュメントライブラリについては、<http://www.oracle.com/goto/ZFSStorage/docs> で Oracle ZFS Storage Appliance ドキュメントライブラリを参照してください。

ホワイトペーパーを含む関連ドキュメントについては、<http://www.oracle.com/technetwork/server-storage/sun-unified-storage/overview/index.html> にアクセスして、「Documentation」タブをクリックしてください。この製品に関する最新の情報と既知の問題については、My Oracle Support (<http://support.oracle.com>) にアクセスしてください。

### Oracle サポートへのアクセス

Oracle のお客様は、My Oracle Support を通じて電子サポートにアクセスできます。詳細は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> にアクセスしてください。聴覚に障害をお持ちの場合は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> にアクセスしてください。

### フィードバック

このドキュメントに関するフィードバックを <http://www.oracle.com/goto/docfeedback> からお聞かせください。



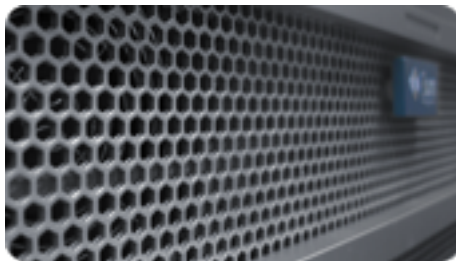
# ◆◆◆ 第 1 章

## Oracle ZFS Storage Appliance の概要

---

Oracle ZFS Storage Appliance (ZFSSA) 製品ファミリーは、ネットワーク経由のクライアントに対して効果的なファイルサービスおよびブロックデータサービスを提供し、システムに保存されたデータに対して適用できる豊富な一連のデータサービスを提供します。

### ZFSSA の主な機能



Oracle ZFS Storage システムには、最高クラスのストレージ価格性能比と、本番ワークロードに対する過去に例を見ない可観測性を実現するために、次のテクノロジーが組み込まれています。

- 『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」は、システムの動作をリアルタイムで動的に観察し、データをグラフィカルに表示するためのシステムです
- ZFS ハイブリッドストレージプールは、読み書きを高速化するオプションのフラッシュメモリーデバイス、低消費電力で大容量のディスク、および DRAM メモリーで構成され、これらすべてが単一のデータ階層として透過的に管理されます
- さまざまな『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「ハードウェアビュー」のサポート
- さまざまな『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「ハードウェアビュー」のサポート

## サポートされているプロトコル

ZFSSA は、次のような業界標準のさまざまなクライアントプロトコルをサポートしています。

- [210 ページの「SMB」](#)
- [202 ページの「NFS」](#)
- [228 ページの「HTTP および HTTPS」](#)
- [228 ページの「WebDAV」](#)
- [208 ページの「iSCSI」](#)
- [114 ページの「SAN ファイバチャネル」](#)
- [132 ページの「SRP」](#)
- [128 ページの「iSER ターゲットの構成」](#)
- [226 ページの「FTP」](#)
- [240 ページの「SFTP」](#)

## ZFSSA データサービス

これらのプロトコルを使用してエクスポートするデータを管理するために、次のような組み込み済みの高度なデータサービスのコレクションを使用して、ZFSSA を構成できます。

---

ライセンスについて: リモートレプリケーションとクローニングは無償で評価することが許可されていますが、本稼動環境で使用するには、各機能のライセンスを個別に購入する必要があります。評価期間を過ぎたら、これらの機能のライセンスを購入するか、機能を非アクティブ化する必要があります。オラクル社は、ライセンスが遵守されているかどうかをいつでも監査する権利を保持しています。詳細は、「オラクル社のソフトウェアライセンス契約書 (SLA)」および「ハードウェアシステムと組み込みのソフトウェアオプションの権利書」を参照してください。

---

- RAID-Z (RAID-5 および RAID-6)、ミラー化、およびストライプ化された[第5章「ストレージ構成」](#)
- 無制限の読み取り専用および読み書き用[346 ページの「シェア - スナップショット」](#) (スナップショットのスケジューリング機能付き)
- [320 ページの「データ複製解除」](#)
- 組み込み式の[320 ページの「データ圧縮」](#)
- 障害回復のためのデータの[第13章「レプリケーション」](#)
- 高可用性のためのアクティブ/アクティブ[第10章「クラスタ構成」](#)
- [208 ページの「iSCSI」「LUN」](#)のシンプロビジョニング
- [245 ページの「ウイルススキャンと隔離」](#)

- [230 ページの「NDMP バックアップと復元」](#)

## データ可用性

本番データの可用性を最大限に高めるために、ZFSSA には、スタックの各レベルでの冗長性など、データの完全性のための完全なエンドツーエンドアーキテクチャーが組み込まれています。主な機能は次のとおりです。

- CPU、DRAM、I/O カード、ディスク、ファン、電源など、すべてのシステムハードウェアの障害に対する予測的自己修復および診断
- すべてのデータおよびメタデータについての ZFS エンドツーエンドのデータチェックサムにより、スタック全体でデータを保護
- RAID-6 (ダブルパリティおよびトリプルパリティ) およびオプションのディスクシェルフ全体での RAID-6
- 高可用性のためのアクティブ/アクティブ [第10章「クラスタ構成」](#)
- ネットワーク障害からの保護のための [第4章「ネットワーク構成」](#)
- コントローラとディスクシェルフの間の I/O マルチパス
- すべてのシステムの [第11章「ZFSSA サービス」](#)の統合ソフトウェア再起動
- すべてのソフトウェアおよびハードウェアの問題に対する [275 ページの「フォンホーム」](#)による遠隔監視
- リモート電源制御およびコンソールアクセスのための各システムの Lights-out 管理

## ZFSSA の構成

ZFSSA を構成するには、以降のセクションを使用します。

- [第3章「初期構成」](#) - 初期構成
- [第4章「ネットワーク構成」](#) - ネットワーク
- [第11章「ZFSSA サービス」](#) - データサービス
- [第6章「Storage Area Network の構成」](#) - Storage Area Network の構成
- [第10章「クラスタ構成」](#) - クラスタ化
- [第7章「ユーザー構成」](#) - ユーザーアカウントとアクセス制御
- [第7章「ユーザー構成」](#) - ユーザー設定
- [第9章「警告の構成」](#) - カスタム警告
- [第5章「ストレージ構成」](#) - ストレージデバイスの再構成

- 第12章「シェア、プロジェクト、およびスキーマ」

## ブラウザユーザーインターフェース (BUI)



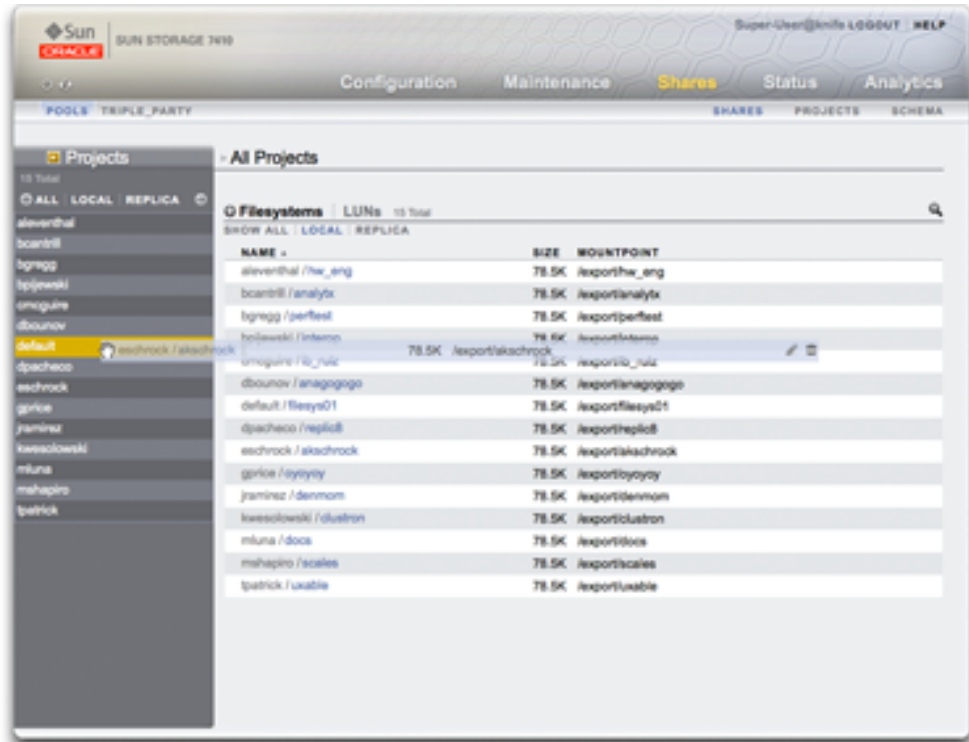
ZFSSA ブラウザユーザーインターフェース (BUI) は、アプライアンスの管理のためのグラフィカルツールです。BUI によって、管理タスク、概念の視覚化、およびパフォーマンスデータの解析のための直感的な環境が提供されます。BUI によって、システム動作を視覚化したり、アプライアンスでのパフォーマンスの問題を識別したりするための整理された環境が提供されます。

ブラウザでシステムにアクセスするには、初期構成中に NET-0 ポートに割り当てた IP アドレスまたはホスト名のいずれかを使用します。たとえば、`https://ipaddress:215` または `https://hostname:215` と指定します。ログイン画面が表示されます。

BUI の右上にリンクが表示されるオンラインヘルプは、コンテキストヘルプです。BUI のトップレベルとセカンドレベルの各画面で「ヘルプ」ボタンをクリックすると、関連するヘルプページが表示されます。

- 23 ページの「メインウィンドウ」 - BUI の要素と設計の概要
- 28 ページの「一般的な使用法」 - アイコンの参照
- 33 ページの「サポートされるブラウザ」 - サポートされるブラウザ

## メインウィンドウ



ファイルシステムのプロパティを、「プロジェクト」サイドパネルを使用して別のプロジェクトに移動することによって変更します。

## マストヘッド

マストヘッドには、ナビゲーションや通知のためのいくつかのインターフェース要素、および主要な機能が含まれています。左側には、上から下に、Sun/Oracle のロゴ、ハードウェアのモデルバッジ、ハードウェアの電源切断/再起動ボタンが配置されています。右側に移ると、ここでも上から下に、ログイン ID、ログアウト、ヘルプ、メインナビゲーション、サブナビゲーションが配置されています。



## 警告

システム警告は、トリガーされたときにマストヘッドに表示されます。複数の警告が連続してトリガーされた場合は、[46 ページの「ダッシュボード」](#)画面に表示される最新の警告のリスト、または『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「[ログ](#)」画面に表示される完全なログを参照してください。

## ナビゲーション

メインのナビゲーションリンクを使用して、BUI の[第4章「ネットワーク構成」](#)、『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「[保守](#)」、[第12章「シェア、プロジェクト、およびスキーマ」](#)、[第2章「ステータス」](#)、および『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」領域を表示します。

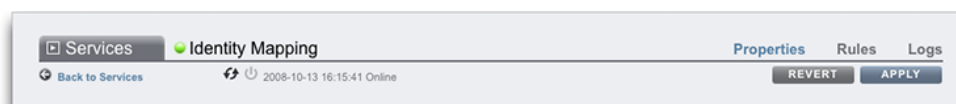
各領域内の機能や関数にアクセスするには、サブナビゲーションリンクを使用します。

## セッションの注釈

セッションの注釈を提供した場合、その注釈は、ログイン ID とログアウトコントロールの下に表示されます。ログアウトすることなく、以降の管理アクションのためにセッションの注釈を変更するには、テキストリンクをクリックします。セッションの注釈の詳細は、[第7章「ユーザー構成」](#)を参照してください。

## タイトルバー

タイトルバーはマストヘッドの下に表示され、ローカルナビゲーションや、現在のビューによって異なる機能を提供します。




たとえば、アイデンティティマッピングサービスのタイトルバーは、次のものを有効にします。

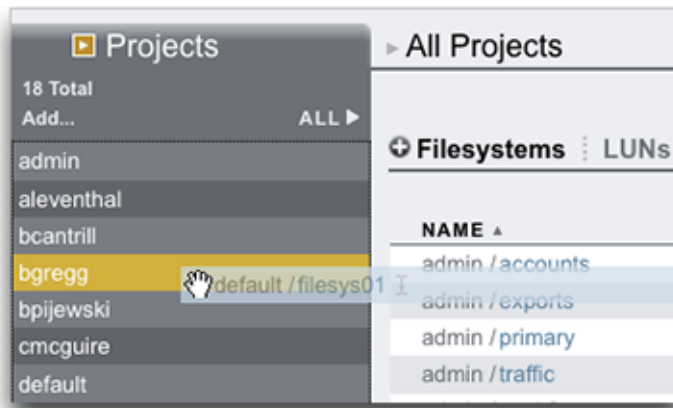
- サイドパネルを通じたサービスの完全なリストへのナビゲーション
- アイデンティティマッピングサービスを有効または無効にするためのコントロール
- アイデンティティマッピングの稼働時間のビュー



- アイデンティティマッピングサービスの「プロパティ」、「規則」、および「ログ」画面へのナビゲーション
- 現在の画面で実行された構成の変更を適用するためのボタン
- 現在の画面で適用された構成の変更を元に戻すためのボタン

## サイドパネルとメニュータイトル

「サービス」ビューと「プロジェクト」ビューの間をすばやく移動するには、タイトルまたは展開  
 矢印をクリックして、サイドパネルを開いたり閉じたりします。




## メインウィンドウのサイドパネルとメニュータイトル

### プロジェクトの追加


プロジェクトを追加するには、サイドバーの「追加...」リンクをクリックします。

### シェアの移動

プロジェクト間でシェアを移動するには、移動  アイコンをクリックし、ファイルシステムのシェアをサイドパネルにある適切なプロジェクトにドラッグします。

シェアを別のプロジェクトにドラッグすると、プロパティが親プロジェクトから継承されるように設定されている場合は、そのシェアのプロパティが変更されることに注意してください。

## オブジェクト名

シェア名を変更するには、そのシェアの強調表示されている表の行にある名前の変更  アイコンをクリックします。

## 非標準の BUI コントロールのガイド

ほとんどの BUI コントロールは標準の Web フォーム入力を使用しますが、注意する必要のある重要な例外がいくつか存在します。

表 1-1 主な Web フォームの例外

BUI コントロールのサマリー	
プロパティを変更する	編集  アイコンをクリックし、ダイアログを完了する
リスト項目またはプロパティエントリを追加する	追加  アイコンをクリックする
リスト項目またはプロパティエントリを削除する	削除  アイコンをクリックする
変更を保存する	「適用」ボタンをクリックする
保存された変更を元に戻す	「戻す」ボタンをクリックする
リストから項目を削除する	ごみ箱  アイコンをクリックする (マウスを項目の行の上に移動すると、このアイコンが表示される)
リスト内の項目を検索する	リストの右上にある検索  アイコンをクリックする
リストの見出しでソートする	リストを再ソートするには、太字のサブ見出しをクリックする
項目を移動またはドラッグする	移動  アイコンをクリックする
項目の名前を変更する	名前の変更  アイコンをクリックする
システムに関する詳細を表示する	使用しているモデルの oracle.com Web ページに移動するには、Oracle のロゴまたはモデルバッジをクリックする
サイドパネルを自動的に開く	項目をサイドパネルにドラッグする

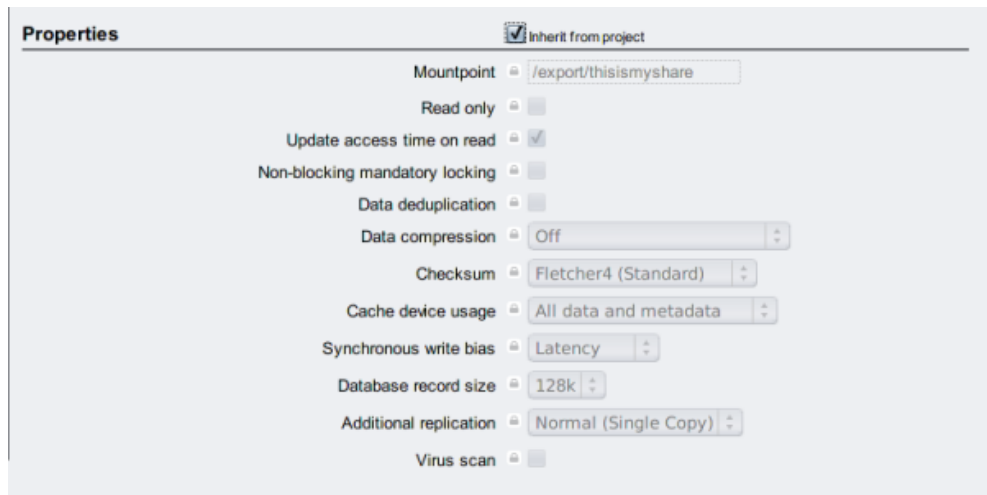
## アクセス権

アクセス権を設定する場合は、RWX ボックスがクリック可能なターゲットになります。アクセスグループのラベル (「ユーザー」、「グループ」、「その他」) をクリックすると、そのラベルのすべてのアクセス権のオン/オフが切り替わります。



## シェアプロパティの編集

シェアプロパティを編集するには、「プロジェクトから継承」チェックボックスのチェックマークを外します。



## リスト項目コントロールの表示

リスト内の項目のコントロールを表示するには、マウスをその行の上に移動します。

NAME ^	SIZE	MOUNTPOINT
admin / accounts	78.5K	/export/accounts
admin / exports	78.5K	/export/exports
admin / primary	78.5K	/export/primary
admin / traffic	78.5K	/export/traffic
admin / workflow	78.5K	/export/workflow

## モダルダイアログ

すべてのモダルダイアログでは、上に現在のアクションを識別するタイトルと、そのアクションのコミットまたは取り消しを行うためのボタンがあり、下にコンテンツが配置されています。モダルダイアログのコンテンツ領域はメインのコンテンツ領域と同じインターフェース規約に従いますが、ほかのアクションを実行するには、その前にタイトルバーにあるボタンを使用して破棄する必要がある点が異なります。

**Add Threshold Alert** [CANCEL] [APPLY]

**Threshold**

CPU: percent utilization exceeds 95 percent

**Timing**

for at least 5 minutes  only between 00:00 and 00:00  only during weekdays

Repost alert every 5 minutes while this condition persists.

Also post alert when this condition clears for at least 5 minutes

**Alert actions**

Send email TEST Send to admin@hostname.com Subject Alert! CPU has exceeded threshold

## 一般的な使用法

アイコンはシステムのステータスを示し、機能へのアクセスを提供するほか、ほとんどの場合は、クリックされたときにアクションを実行するボタンとして機能します。マウスをインターフェースアイコンの上に移動してツールチップを表示すると役立ちます。下の表は、ユーザーインターフェースの規約の基本を示しています。

## ステータス

ステータスの光は、システムの健全性やサービスの状態の基本的なインジケータです。

表 1-2 ステータスインジケータ

アイコン	説明	アイコン	説明
	オン		警告
	オフ		無効

## 基本的な使用法

次のアイコンはユーザーインターフェース全体にわたって現れ、基本的な機能のほとんどをカバーしています。

表 1-3 BUI のアイコン

アイコン*	説明	アイコン*	説明		
--		名前の変更 (テキスト編集)	--		切断
--		移動	--		クローン
		編集	--		ロールバック
		破棄	--		アプライアンス電源
		追加	--		適用
		削除	--		戻す
		取り消し/閉じる	--		情報
--		エラー	--		リスト列のソート (下)
--		alert	--		リスト列のソート (上)
		オン/オフの切り替え			最初のページ
		再起動			前のページ

アイコン*		説明	アイコン*		説明
--		検索			次のページ
		無効/オフライン			最後のページ
		ロック	--		検索
--		待機スピナー			メニュー
--		逆方向			パネル

\* 左側には無効になったアイコンが示されています。

## ネットワーク

次のアイコンは、ネットワークデバイスの状態とネットワークデータリンクのタイプを示します。

表 1-4 ネットワークのアイコン

アイコン	説明	アイコン	説明
	アクティブなネットワークデバイス		アクティブな Infiniband ポート
	アクティブでないネットワークデバイス		アクティブでない Infiniband ポート
	ネットワークデータリンク		ネットワークデータリンク (IB パーティション)
	ネットワークデータリンク VLAN		
	ネットワークデータリンクアグリゲーション		
	ネットワークデータリンクアグリゲーション VLAN		

## ダッシュボードしきい値

次のアイコンは、55 ページの「設定」内で設定されたユーザーが構成できるしきい値に関連した、モニターされている統計情報の現在の状態を示します。

表 1-5 ダッシュボードのアイコン

アイコン	説明	アイコン	説明
	快晴		ハリケーン
	部分的に曇り		ハリケーンクラス 2
	曇り		ハリケーンクラス 3
	雨		ハリケーンクラス 4
	嵐		ハリケーンクラス 5

## 分析

この一連のアイコンは、「分析」ワークシート内の情報の表示を操作するためにツールバーで使用されます。

表 1-6 「分析」ツールバーのアイコン

アイコン	説明	アイコン	説明
	戻ります		最小値を表示します
	進めます		最大値を表示します
	現在まで進めます		折れ線グラフを表示します
	一時停止します		山型グラフを表示します
	ズームアウトします		外れ値を除外します
	ズームインします		ワークシートとこの統計を同期します
	1 分間を表示します		ワークシートと統計の同期を解除します
	1 時間を表示します		ドリルダウンします

アイコン	説明	アイコン	説明
	1 日を表示します		統計データをエクスポート (クライアントにダウンロード) します
	1 週間を表示します		統計データを保存します
	1 か月を表示します		データセットをアーカイブ します
			ワークシートをサポートバ ンドルとともに送信します

## アイデンティティマッピング

次のアイコンは、Windows と UNIX の間でユーザーとグループをマップするときに適用されるロールのタイプを示します。

表 1-7 アイデンティティマッピングのアイコン


アイコン*	説明	アイコン*	説明
	 Windows から UNIX を許可し ます		 UNIX から Windows を許 可します
	 Windows から UNIX を拒否し ます		 UNIX から Windows を拒 否します
	 双方向を許可し ます		

\* 左側には無効になったアイコンが示されています。

## その他のアイコン

次のアイコンは、異なるタイプのオブジェクトを区別するとともに、二次的に重要な情報を 提供するために使用されます。

表 1-8 その他のアイコン

アイコン	説明	アイコン	説明
	許可		SAS



アイコン	説明	アイコン	説明
	拒否		SAS ポート
	ストレージプール		

## サポートされるブラウザ

このセクションでは、BUI のブラウザサポートを定義します。最適な結果を得るには、層 1 のブラウザを使用します。

### 層 1

BUI ソフトウェアは、次の層 1 のブラウザ上で完全に機能するように設計されています。

- Firefox 3.x 以降
- Internet Explorer 7 以降
- Safari 3.1 以降
- Google Chrome (安定)
- WebKit 525.13 以降

### 層 2

層 2 のブラウザでは、BUI の要素が見かけ上不完全になることがあるほか、必要な機能はすべて正しく機能しますが、一部の機能が使用できない場合があります。次のいずれかの層 2 のブラウザを使用している場合は、ログイン中に警告メッセージが表示されます。

- Firefox 2.x
- Solaris 10 上の Mozilla 1.7
- Opera 9

### サポートされていないブラウザ

Internet Explorer 6 以前のバージョンはサポートされていません。問題があることがわかっているため、ログインが完了しません。

## コマンド行インタフェース (CLI)

CLI は、BUI の機能をミラー化する一方で、繰り返しタスクを行うための強力なスクリプト環境も提供するように設計されています。コマンド行は、繰り返しの管理タスクのための効率的で強力なツールです。このアプライアンスは、『Oracle ZFS Storage Appliance インストールガイド』の「コンソール」、または 291 ページの「SSH」から使用可能な CLI を提供します。システムとの推奨される対話の方法が CLI である状況が、次に示すようにいくつか存在します。

- ネットワークが使用不可 - ネットワークが使用できない場合、ブラウザベースの管理は不可能です。テキストベースのインタフェースにしか対応していない『Oracle ZFS Storage Appliance インストールガイド』の「コンソール」が、管理のための唯一の手段になります。
- 便宜性 - システムの特定の側面を検査したり、構成をすばやく変更したりしたいだけの場合は特に、ブラウザの起動に非常に長い時間がかかることがあります
- 精度 - 状況によっては、ブラウザで提供される情報はその性質上、定量的というより定性的であることがあるが、より正確な答えを必要としている場合
- 自動化 - ブラウザベースの対話は、容易には自動化できません。繰り返しのタスクや厳密に定義されたタスクがある場合は、それらのタスクをスクリプト化します
- タブ補完が幅広く使用されます。特定のコンテキストで何を入力すればよいかわからない場合、Tab キーを押すと、入力可能なオプションが表示されます。ドキュメント全体を通して、Tab キーを押す場合は「tab」という文字を太字の斜体で表現します。
- ヘルプは常に使用できます。help コマンドによって、コンテキスト固有のヘルプが提供されます。特定のトピックのヘルプは、たとえば help commands のように、そのトピックを help の引数として指定することによって表示できます。使用可能なトピックは、help コマンドをタブ補完するか、または help topics を入力することによって表示されます。

CLI 内を移動する場合に注意する原則が 2 つあります。

- タブ補完が幅広く使用されます。特定のコンテキストで何を入力すればよいかわからない場合、Tab キーを押すと、入力可能なオプションが表示されます。ドキュメント全体を通して、Tab キーを押す場合は「tab」という文字を太字の斜体で表現します。
- ヘルプは常に使用できます。help コマンドによって、コンテキスト固有のヘルプが提供されます。特定のトピックのヘルプは、たとえば help commands のように、そのトピックを help の引数として指定することによって表示できます。使用可能なトピックは、help コマンドをタブ補完するか、または help topics を入力することによって表示されます。

この 2 つの原則を、次のように組み合わせることができます。

```
dory:> help tab
builtins  commands  general  help  properties  script
```

## CLI へのログイン

CLI を使用してリモートからログインするには、ssh クライアントを使用します。このアプライアンスを管理する、[第7章「ユーザー構成」](#)がない場合は、root としてログインする必要があります。ログインすると、次に示すように、ホスト名とそれに続くコロン、さらにそれに続く大なり記号で構成されるプロンプトが CLI に表示されます。

```
% ssh root@dory
Password:
Last login: Mon Oct 13 15:43:05 2009 from kiowa.sf.fishpo
dory:>
```

## CLI コンテキスト

CLI における中心的な原則は、コマンドが実行されているコンテキストです。このコンテキストによって、システムのどの要素を管理できるか、およびどのコマンドを使用できるかが規定されます。コンテキストには、各コンテキスト自体に入れ子のコンテキストが含まれている可能性のあるツリー構造があり、一般には、この構造によって BUI 内のビューの構造がミラー化されています。

### ルートコンテキスト

ログイン時の初期コンテキストはルートコンテキストであり、すべてのコンテキストの親または上位コンポーネントとして機能します。コンテキストに移動するには、そのコンテキストの名前をコマンドとして実行します。たとえば、ブラウザ内の [第4章「ネットワーク構成」](#)ビューで使用可能な機能は、CLI の configuration コンテキストで使用できます。次のように、ルートコンテキストから直接入力してこのコンテキストにアクセスできます。

```
dory:> configuration
dory:configuration>
```

プロンプト内のコロンと大なり記号の間にコンテキストが表示されており、プロンプトがコンテキストを反映して変更されることに注意してください。

### 子コンテキスト

show コマンドは子コンテキストを表示します。たとえば、configuration コンテキストから実行すると、次のようになります。

```
dory:configuration> show
```

Children:

```

net => Configure networking
services => Configure services
version => Display system version
users => Configure administrative users
roles => Configure administrative roles
preferences => Configure user preferences
alerts => Configure alerts
storage => Configure Storage

```

これらの子コンテキストは、第4章「ネットワーク構成」、第11章「ZFSSA サービス」および第7章「ユーザー構成」、「設定」第8章「ZFSSA の設定」などの、ブラウザ内の第6章「Storage Area Network の構成」ビューの下で使用可能なビューに対応しています。これらの子コンテキストのいずれかを選択するには、その名前を入力します。

```

dory:configuration> preferences
dory:configuration preferences>

```

中間のコンテキストを空白で区切って指定することによって、上位コンテキストから直接下位コンテキストに移動できます。たとえば、ルートコンテキストから直接 configuration preferences に移動するには、単に次のように入力します。

```

dory:> configuration preferences
dory:configuration preferences>

```

## 動的な子コンテキスト

一部の子コンテキストは、ブラウザ内の固定されたビューにではなく、ユーザーまたはシステムのどちらかによって作成された動的なエンティティに対応しているという点で動的です。これらのコンテキストに移動するには、select コマンドを使用し、そのあとに動的コンテキストの名前を指定します。特定のコンテキスト内に含まれている動的コンテキストの名前は、list コマンドを使用して表示されます。たとえば、users コンテキストは静的コンテキストですが、各ユーザーは独自の動的コンテキストです。

```

dory:> configuration users
dory:configuration users> list
NAME                USERNAME            UID                TYPE
John Doe            bmc                 12345             Dir
Super-User          root                 0                 Loc

```

bmc という名前のユーザーを選択するには、コマンド select bmc を発行します。

```

dory:configuration users> select bmc
dory:configuration users bmc>

```

一部のコンテキストでは select と destroy を交互に使用すると、プロパティに基づいてエンティティを選択できます。たとえば、次のコマンドを発行することによって、maintenance logs system コンテキストで reboot モジュールによって発行されたログエントリを選択できます。

```

dory:maintenance logs system> select module=reboot

```

```
dory:maintenance logs system entry-034> show
Properties:
  timestamp = 2010-8-14 06:24:41
  module = reboot
  priority = crit
  text = initiated by root on /dev/console syslogd: going down on signal 15
```

ほかのコマンドと同様に、`select` をコンテキスト変更コマンドに追加することもできます。たとえば、ルートコンテキストから `bmc` という名前のユーザーを選択するには、次のコマンドを実行します。

```
dory:> configuration users select bmc
dory:configuration users bmc>
```

## 最後のコンテキスト

`last` コマンドを使用すると、前に選択または作成したコンテキストに移動できます。このコマンドは現時点では、レプリケーションアクションのコンテキストにのみ実装されています。

次の例ではレプリケーションアクションを作成し、`last` および `get id` コマンドを使用してレプリケーションアクション ID を取得します。その後別のアクションを選択し、`last` および `get id` コマンドを使用して、最後に実行したレプリケーションアクションの ID を取得します。

```
dory:shares p1/share replication> list
      TARGET      STATUS      NEXT
action-000 oakmeal      idle       Sync now
action-001 dory         idle       Sync now
dory:shares p1/share replication> create
dory:shares p1/share action (uncommitted)> set target=dory
      target = dory (uncommitted)
dory:shares p1/share action (uncommitted)> set pool=p0
      pool = p0 (uncommitted)
dory:shares p1/share action (uncommitted)> commit
dory:shares p1/share replication> last
dory:shares p1/share action-002> get id
      id = 7034367a-d4d8-e26f-fa93-c3b454e3b595
dory:shares p1/share action-002> done
dory:shares p1/share replication> select action-000
dory:shares p1/share action-000> get id
      id = 9895d9f4-7b23-ebel-faf2-d85a581e3dff
dory:shares p1/share action-000> done
dory:shares p1/share replication> last get id
      id = 9895d9f4-7b23-ebel-faf2-d85a581e3dff
dory:shares p1/share replication>
```

## 以前のコンテキストに戻る

以前のコンテキストに戻るには、`done` コマンドを使用します。

```
dory:configuration> done
dory:>
```

次に示すように、これによって戻る以前のコンテキストは、必ずしも親コンテキストではないことに注意してください。

```
dory:> configuration users select bmc
dory:configuration users bmc> done
dory:>
```

done コマンドを複数回使用すると、以前のコンテキストに 1 つずつ戻ることができます。

```
dory:> configuration
dory:configuration> users
dory:configuration users> select bmc
dory:configuration users bmc> done
dory:configuration users> done
dory:configuration> done
dory:>
```

## 親コンテキストへの移動

親コンテキストに移動するには、cd コマンドを使用します。従来の UNIX コマンドと同様に、親コンテキストへの移動を示すには、cd に「..」の引数を指定します。

```
dory:> configuration users select bmc
dory:configuration users bmc> cd ..
dory:configuration users>
```

また、UNIX コマンドと同様に、「cd /」を指定するとルートコンテキストに移動します。

```
dory:> configuration
dory:configuration> users
dory:configuration users> select bmc
dory:configuration users bmc> cd /
dory:>
```

さらに、UNIX コマンドと同様に、「cd ../..」を使用すると親の親のコンテキストに移動できます。

```
dory:> configuration
dory:configuration> users
dory:configuration users> select bmc
dory:configuration users bmc> cd ../..
dory:configuration>
```

## コンテキストとタブ補完

コンテキスト名は、静的コンテキスト (通常のコマンド補完を介して) か動的コンテキスト (select コマンドのコマンド補完を介して) にかかわらずタブ補完されます。次に示すのは、ルートコンテキストから bmc という名前のユーザーを、タブ補完を使用しない場合に必要になる 31 個ではなく、わずか 15 個のキーストロークで選択する例です。

```
dory:> configtab
dory:> configuration utab
```

```
dory:> configuration users setab
dory:> configuration users select tab
bmc root
dory:> configuration users select btab
dory:> configuration users select bmcenter
dory:configuration users bmc>
```

## コンテキスト固有のコマンドの実行

コンテキスト内では、コンテキスト固有のコマンドを実行します。たとえば、現在のユーザーの設定を取得するには、`configuration preferences` コンテキストから `get` コマンドを実行します。

```
dory:configuration preferences> get
      locale = C
      login_screen = status/dashboard
      session_timeout = 15
      session_annotation =
      advanced_analytics = false
```

コンテキストを変更するコマンドのあとに入力が存在する場合は、ターゲットのコンテキストでそのコマンドが実行されますが、制御は呼び出し側のコンテキストに戻ります。たとえば、コンテキストを変更せずにルートコンテキストから設定を取得するには、コンテキストナビゲーションコマンドに `get` コマンドを追加します。

```
dory:> configuration preferences get
      locale = C
      login_screen = status/dashboard
      session_timeout = 15
      session_annotation =
      advanced_analytics = false
```

## コミットされていないコンテキスト

システム内で新しいエンティティを作成した場合は通常、新しいエンティティに関連付けられたコンテキストが、**コミットされていない**状態で作成されます。たとえば、`configuration alerts threshold` コンテキストから `create` コマンドを実行することによって、[第9章「警告の構成」](#)を作成します。

```
dory:> configuration alerts thresholds create
dory:configuration alerts threshold (uncommitted)>
```

プロンプト内の `(uncommitted)` は、これがコミットされていないコンテキストであることを示します。コミットされていないエンティティは、`commit` コマンドを使用してコミットされます。コミットされていないコンテキストから移動しようとする、確認を求めるプロンプトが表示されます。

```
dory:configuration alerts threshold (uncommitted)> cd /
Leaving will abort creation of "threshold". Are you sure? (Y/N)
```

コミットされていないエンティティをコミットするときは、新しいエンティティに関連付けられたプロパティが検証され、エンティティを作成できない場合はエラーが生成されます。たとえば、新しいしきい値警告の作成には統計名の指定が必要であり、これを設定できない場合はエラーが生成されます。

```
dory:configuration alerts threshold (uncommitted)> commit
error: missing value for property "statname"
```

この問題を解決するには、エラーに対処してコミットを再試行します。

```
dory:configuration alerts threshold (uncommitted)> set statname=cpu.utilization
statname = cpu.utilization (uncommitted)
dory:configuration alerts threshold (uncommitted)> commit
error: missing value for property "limit"
dory:configuration alerts threshold (uncommitted)> set limit=90
limit = 90 (uncommitted)
dory:configuration alerts threshold (uncommitted)> commit
dory:configuration alerts thresholds> list
THRESHOLD      LIMIT      TYPE  STATNAME
threshold-000      90      normal cpu.utilization
```

## プロパティ

### CLI のプロパティ

プロパティとは、コンテキストに関連付けられている、型付けされた名前と値のペアです。特定のコンテキストのプロパティは、「help properties」コマンドを実行することによって確認できます。ユーザーの設定に関連付けられたプロパティを取得する例を次に示します。

```
dory:configuration preferences> help properties
Properties that are valid in this context:

locale           => Locality

login_screen     => Initial login screen

session_timeout  => Session timeout

session_annotation => Current session annotation

advanced_analytics => Make available advanced analytics statistics
```

### プロパティの取得

特定のコンテキストのプロパティは、get コマンドを使用して取得できます。get コマンドを使用してユーザーの設定を取得する例を次に示します。

```
dory:configuration preferences> get
locale = C
login_screen = status/dashboard
session_timeout = 15
```



```
session_annotation =
advanced_analytics = false
```

## 1 つのプロパティ値の取得

get コマンドは、引数として指定された任意のプロパティを返します。たとえば、login\_screen プロパティの値を取得するには、次のコマンドを実行します。

```
dory:configuration preferences> get login_screen
login_screen = status/dashboard
```

## タブ補完

get コマンドは、使用可能なプロパティの名前によってタブ補完されます。たとえば、[208 ページの「iSCSI」](#) サービスの使用可能なプロパティのリストを表示するには、次のコマンドを実行します。

```
dory:> configuration services iscsi get tab
<status>          isns_server          radius_secret          target_chap_name
isns_access        radius_access        radius_server          target_chap_secret
```

## プロパティの設定

set コマンドは、プロパティを指定された値に設定します。ここで、プロパティ名とその値は等号で区切られます。たとえば、login\_screen プロパティを「shares」に設定するには、次のコマンドを実行します。

```
dory:configuration preferences> set login_screen=shares
login_screen = shares (uncommitted)
```

アプライアンス上の状態を構成するプロパティの場合は、プロパティを設定してもその値は変更されず、代わりに、設定された値が記録され、そのプロパティの値はコミットされていないことが示されます。

## 設定済みのプロパティ値のコミット

設定済みのプロパティ値を強制的に有効にするには、それらのプロパティ値を明示的にコミットして、複数の値を 1 つの整合性のある変更として変更できるようにする必要があります。いずれかのコミットされていないプロパティ値をコミットするには、commit コマンドを使用します。

```
dory:configuration preferences> get login_screen
login_screen = shares (uncommitted)
dory:configuration preferences> commit
dory:configuration preferences> get login_screen
login_screen = shares
```

コミットされていないプロパティを含むコンテキストから移動しようとする、この移動によって設定済みのプロパティ値が破棄される旨の警告が表示され、移動を確認するよう求められます。次に例を示します。

```
dory:configuration preferences> set login_screen=maintenance/hardware
      login_screen = maintenance/hardware (uncommitted)
dory:configuration preferences> done
You have uncommitted changes that will be discarded. Are you sure? (Y/N)
```

## 暗黙のコミットによるプロパティ値の設定

コンテキスト内のプロパティが別のコンテキストから設定された場合、つまり、コンテキストを変更するコマンドに `set` コマンドが追加された場合、そのコミットは暗黙的に行われ、制御が発元元のコンテキストに返される前に実行されます。次に例を示します。

```
dory:> configuration preferences set login_screen=analytics/worksheets
      login_screen = analytics/worksheets
dory:>
```

## プロパティの値のリストへの設定

一部のプロパティには、値のリストを指定します。これらのプロパティでは、リストの要素をコンマで区切るようにしてください。たとえば、[271 ページの「NTP」](#) の `servers` プロパティを、次のように NTP サーバーのリストに設定できます。

```
dory:configuration services ntp> set servers=0.pool.ntp.org,1.pool.ntp.org
      servers = 0.pool.ntp.org,1.pool.ntp.org (uncommitted)
dory:configuration services ntp> commit
```

## プロパティの特殊文字を含む値への設定

プロパティ値にコンマ、等号、引用符、または空白が含まれている場合は、値全体を引用符で囲む必要があります。たとえば、デフォルトのプロジェクトの `sharenfs` シェアプロパティを読み取り専用にするが、ホスト「kiowa」には読み取り/書き込み権を提供するように設定するには、次のコマンドを実行します。詳細は、[第12章「シェア、プロジェクト、およびスキーマ」](#)を参照してください。

```
dory:> shares select default
dory:shares default> set sharenfs="ro,rw=kiowa"
      sharenfs = ro,rw=kiowa (uncommitted)
dory:shares default> commit
```

## 変更不可能なプロパティ

一部のプロパティは変更不可能です。その値は取得できますが、設定することはできません。変更不可能なプロパティを設定しようとする、エラーが生成されます。たとえば、デ

フォルトのプロジェクトの変更不可能な `space_available` プロパティを設定しようとすると、次のようになります。詳細は、[第12章「シェア、プロジェクト、およびスキーマ」](#)を参照してください。

```
dory:> shares select default
dory:shares default> get space_available
      space_available = 1.15T
dory:shares default> set space_available=100P
error: cannot set immutable property "space_available"
```

ほかの一部のプロパティは、特定の条件でのみ変更不可能です。これらのプロパティに対して、`set` コマンドは有効ではありません。たとえば、`bmc` という名前のユーザーがネットワークユーザーである場合、`fullname` プロパティは変更不可能です。

```
dory:> configuration users select bmc set fullname="Rembrandt Q. Einstein"
error: cannot set immutable property "fullname"
```



## ◆◆◆ 第 2 章

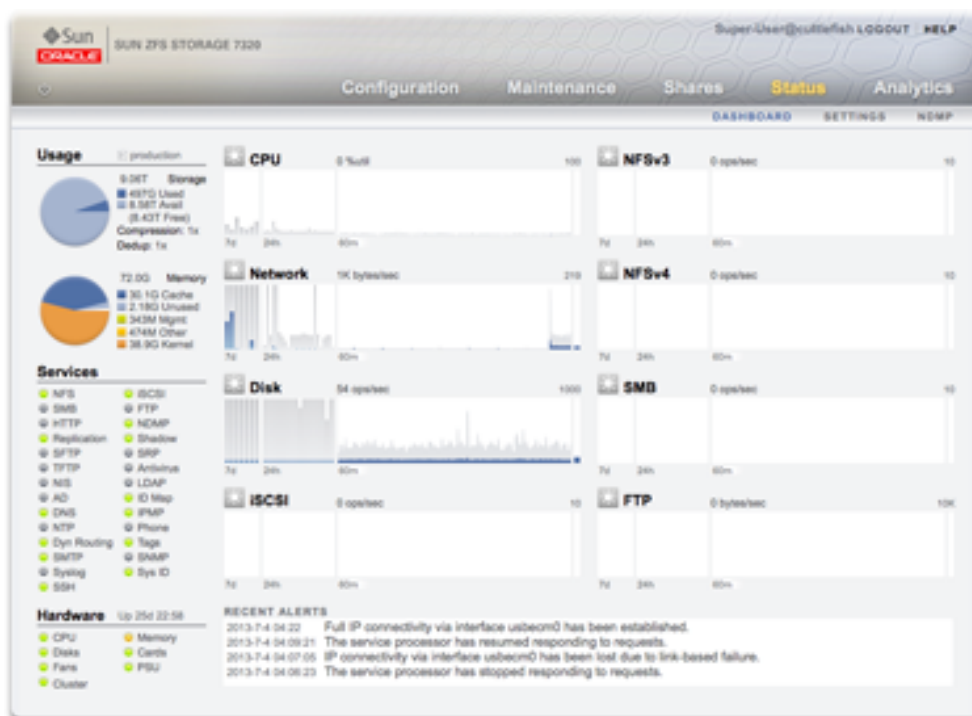
# ステータス

---

「ステータス」セクションには、アプライアンスステータスおよび構成オプションのサマリーが表示されます。アプライアンスステータスのビューや関連するサービス構成の概念的な情報および手続きに関する情報については、以降のセクションを使用してください。

- 46 ページの「[ステータス](#)」>「[ダッシュボード](#)」画面には、ストレージ、メモリー、サービス、ハードウェア、アクティビティ、および最新の警告のビューが表示されます。
- 55 ページの「[ステータス](#)」>「[設定](#)」画面を使用すると、「[ダッシュボード](#)」に表示されるグラフを変更したり、「[ダッシュボード](#)」上のグラフごとに示される天気アイコンに関連付けられたしきい値設定をカスタマイズしたりできます。
- 58 ページの「[ステータス](#)」>「[NDMP](#)」画面には、構成されているすべての NDMP デバイス、および NDMP セッションごとの最新のアクティビティのビューが表示されます。

## ダッシュボード



「ダッシュボード」にはアプライアンスステータスのサマリーが表示されます

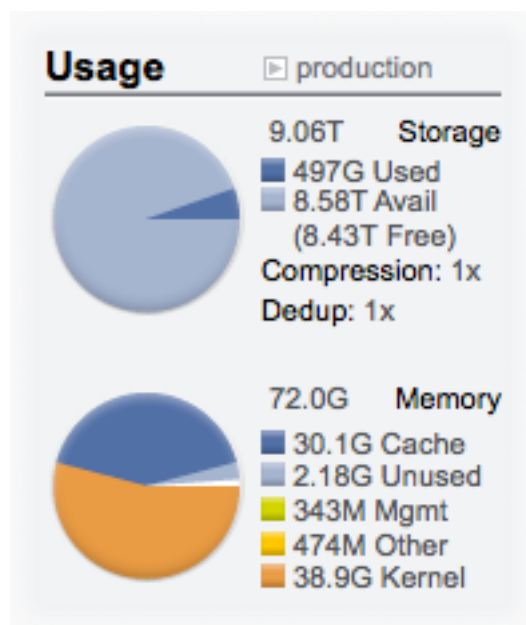
## リンク

「ステータス」「ダッシュボード」には、ブラウザユーザーインターフェース (BUI) のすべてのメイン画面へのリンクが表示されます。「ダッシュボード」に表示されている 100 を超える項目が、マウスを上に移動すると表示される枠または強調表示されたテキストによって示される関連付けられた BUI 画面にリンクされています。以降のセクションでは、「ダッシュボード」の各領域について詳細に説明します。

## 使用状況

「ダッシュボード」の使用状況の領域には、ストレージプールとメインメモリーの使用状況のサマリーが表示されます。使用状況の領域の右上には、プールの名前が表示されます。複数のプールが構成されている場合は、プルダウンリストを使って目的のプールを選択して表示します。

図 2-1 「ステータス」 「ダッシュボード」 使用状況



## ストレージ

プール容量の合計は、この領域のいちばん上に表示されます。「ストレージ」の円グラフに、使用済み、使用可能、および空いている領域の詳細が表示されます。プールの「シェア」画面に移動するには、「ストレージ」の円グラフをクリックします。

## メモリー

システム物理メモリーの合計は、この領域の最上部に表示されます。左側には、コンポーネントごとのメモリー使用量を示す円グラフが表示されます。アプリケーション名で分類された動的なメモリー使用量の Analytics ワークシートに移動するには、「メモリー」の円グラフをクリックします。

表 2-1 プール使用状況のサマリー

プール使用状況のサマリー	
使用中	このプールで使用されている領域 (データやスナップショットを含む)。
使用可能	使用可能な物理ディスクの容量。ファイルシステムメタデータの消費のために、ファイルデータで使用できる

プール使用状況のサマリー	
	容量 (「Shares」画面で報告される) はこれより少なくなります。
空き	使用可能な容量 (LUN の容量のうち、プロジェクトにより予約され、プール内部で共有されている未使用領域を減らします)。ディスク容量が事前に予約により割り当てられるか、LUN が作成される場合、または両方の場合に、使用可能な空きディスク容量を提供します。
圧縮	このプールによって達成されている現在の圧縮率。圧縮が無効になっている場合は、1x と表示されます。
複製解除	このプールによって達成されている現在のデータ複製解除率。データ複製解除が無効になっている場合は、1x と表示されます。

表 2-2 メインメモリー使用量のサマリー

メインメモリー (RAM) 使用量のサマリー	
キャッシュ	パフォーマンスを向上させるためにファイルシステムキャッシュで使用されているバイト数。
未使用	現在使用されていないバイト数。ブート後、領域がファイルシステムキャッシュで使用されるに従ってこの値は減少します。
管理	アプライアンス管理ソフトウェアで使用されているバイト数。
その他	その他のオペレーティングシステムソフトウェアで使用されているバイト数。
カーネル	オペレーティングシステムカーネルで使用されているバイト数。

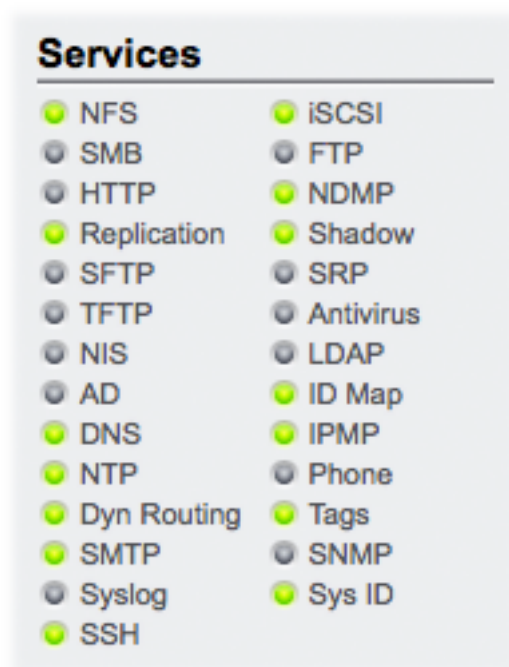
メモリー使用量を表示するユーザーには、`analytics/component create+read` 承認が必要なことに注意してください。この承認がない場合、「ダッシュボード」にメモリーの詳細は表示されません。

## サービス

「ダッシュボード」のこの領域には、アプライアンス上のサービスのステータスが表示され、光のアイコンが各サービスの状態を示します。



図 2-2 「ダッシュボード」「サービス」



## アイコン

ほとんどのサービスは、サービスがオンラインであることを示す緑色か、またはサービスが無効になっていることを示す灰色です。可能性のあるすべての状態とアイコンの色を参照するには、[28 ページの「一般的な使用法」](#)のセクションを参照してください。

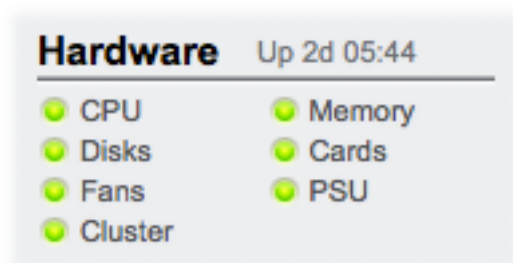
## リンク

関連付けられた構成画面に移動するには、サービス名をクリックします。「プロパティ」画面が表示されます。ここには、構成可能なフィールド、再起動、有効化、および無効化アイコン、このサービスに関連付けられた「ログ」画面へのリンクが含まれています。

## ハードウェア

「ダッシュボード」のこの領域には、アプライアンス上のハードウェアの概要が表示されません。

図 2-3 「ダッシュボード」「ハードウェア」



## 障害

既知の障害が存在する場合は、オレンジ色の障害  アイコンが表示されます。

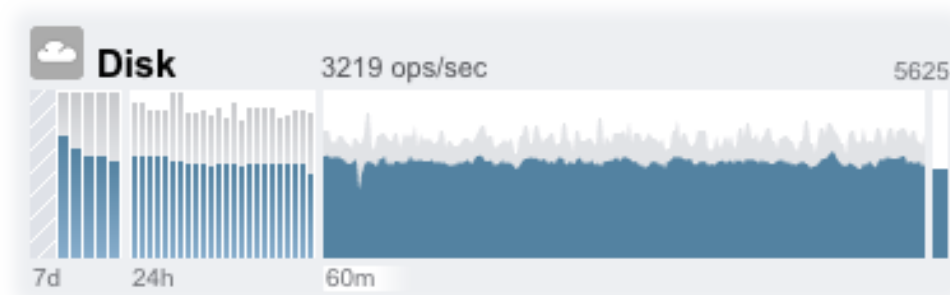
## リンク

ハードウェアの状態の詳細を表示するために『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「ハードウェア」画面に移動するには、ハードウェアコンポーネントの名前をクリックします。

## アクティビティ

「ダッシュボード」のアクティビティ領域には、デフォルトでは 8 つのパフォーマンス統計のグラフが表示されます。このセクションにある例は、1 秒あたりのディスク操作の数を示しています。統計の平均が青色でプロットされ、最大が明るい灰色で表示されます。

図 2-4 ディスクアクティビティダッシュボード



アクティビティの『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」ワークシートに移動するには、評価する統計の 4 つのグラフ (日、時間、分、秒) のいずれかをクリックします。

グラフごとの平均を表示するには、マウスをグラフの上に移動すると、平均がツールチップに表示されます。左上にある天気アイコンは、「55 ページの「ステータス設定」」画面で統計ごとにカスタマイズできるしきい値に従ったアクティビティのレポートを示します。

## グラフ

表 2-3 統計グラフのサマリー

統計グラフのサマリー	
7 日間グラフ (7d)	各棒が 1 日を表す棒グラフ。
24 時間グラフ (24h)	各棒が 1 時間を表す棒グラフ。
60 分間グラフ (60m)	1 時間にわたるアクティビティを表す折れ線グラフ (24 時間グラフ内の最初の 1 時間の棒としても表示される)。
1 秒間グラフ	瞬時のアクティビティレポートを表す折れ線グラフ。

## 平均

選択されたプロットの平均が、グラフの上に数値で表示されます。表示される平均を変更するには、目的の平均 (7d、24h、60m のいずれか) を選択します。

## 垂直スケール

すべてのグラフの垂直スケールが右上に出力され、すべてのグラフがこの同じ高さにスケールリングされます。高さは、選択されたグラフから計算されます (プラス余白)。高さが 100 % に固定されている使用率グラフを除き、高さは、選択されたグラフ内のアクティビティーに基づいて再スケールリングされます。

高さは再スケールリングが可能のため、アイドル状態アクティビティーの 60 分がビジー状態アクティビティーの 60 分と同様に表示される可能性があります。グラフが意味する内容を解釈しようとする前に、グラフの高さを常に確認してください。

統計情報によっては、明白に理解できない場合があります。環境内の特定のアプライアンスについて、毎秒 1000 個の NFSv3 操作をビジーまたはアイドル状態のどちらと見なせばいいか迷うことがあります。このような場合、比較のために現在のアクティビティーの横に履歴データを表示する 24 時間と 7 日間のプロットが役立ちます。

プロットの高さは、選択されたプロットから計算されます。デフォルトでは、60 分間のプロットが選択されます。そのため、高さは、その 60 分間隔中の最大のアクティビティー (プラス余白) になります。すべてのプロットを前の 7 日間のもっとも高いアクティビティーにわたるように再スケールリングするには、7d を選択します。これにより、現在のアクティビティーを前日または先週と容易に比較して確認できます。

## 天気

天気アイコンは、何かが異常にビジーまたはアイドル状態にあるときに、ユーザーの注意を引き付けるように考慮されています。天気のしきい値の構成ページに移動するには、天気アイコンをクリックします。良いしきい値や悪いしきい値は存在せず、BUI によって、アクティビティー統計情報ごとのレベルのグラデーションが表示されます。天気アイコンの基になっている統計情報は、次に示すように、ワークロードに合わせてカスタマイズすべきアプライアンスパフォーマンスに対する概略の理解を提供します。

- 環境が異なれば許容可能なレベルのパフォーマンス (待機時間) も異なり、1 つですべてに間に合うしきい値は存在しません。
- 「ダッシュボード」上の統計情報は 1 秒あたりの操作数や 1 秒あたりのバイト数に基づいているため、システムパフォーマンスを正確に理解するには『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」ワークシートを使用してください。

## 最新の警告

図 2-5 最新の警告

```
RECENT ALERTS
2010-2-22 16:53:51 Replication of 'default' to 'tuna' failed.
2010-2-22 16:29:23 Finished replicating 'default' to appliance 'tuna'.
2010-2-22 16:29 Began replicating 'default' to appliance 'tuna'.
2010-2-22 15:59:28 Finished replicating 'default' to appliance 'tuna'.
```

このセクションには、最後の 4 つのアプライアンス警告が表示されます。最近のすべての警告の詳細を確認するには、ボックスをクリックして『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「ログ」画面に移動します。

## CLI

「ステータス」>「ダッシュボード」画面のテキストバージョンを CLI から使用できるようにするには、「status dashboard」と入力します。

```
cuttlefish:> status dashboard
Storage:
  pool_0:
    Used      497G bytes
    Avail    8.58T bytes
    Free     8.43T bytes
    State    online
    Compression 1x

Memory:
  Cache      30.1G bytes
  Unused    2.18G bytes
  Mgmt       343M bytes
  Other     474M bytes
  Kernel    38.9G bytes

Services:
  ad          disabled      smb          disabled
  dns        online       ftp          disabled
  http       online       identity    online
  idmap      online       ipmp        online
  iscsi      online       ldap        disabled
  ndmp       online       nfs         online
  nis        online       ntp         online
  routing    online       scrk        maintenance
  snmp       online       ssh         online
  tags       online       vscan       online

Hardware:
  CPU        online       Cards        online
  Disks      faulted    Fans         online
  Memory     online     PSU          online
```

```
Activity:
CPU          1 %util          Sunny
Disk         32 ops/sec       Sunny
iSCSI        0 ops/sec          Sunny
NDMP         0 bytes/sec        Sunny
NFSv3        0 ops/sec          Sunny
NFSv4        0 ops/sec          Sunny
Network      13K bytes/sec      Sunny
SMB          0 ops/sec          Sunny
```

```
Recent Alerts:
2013-6-15 07:46: A cluster interconnect link has been restored.
```

46 ページの「BUI」セクションにある前の説明が適用されますが、次の違いがあります。

- アクティビティのプロットはテキストでは描画されません (ただし、aalib の使用について検討しました)。
- CLI では、ストレージ使用状況のセクションに、使用可能なすべてのプールの詳細が一覧表示されます。これに対して、BUI には 1 つのサマリーを表示する余裕しかありません。

別のビューも使用できます。たとえば、`status activity show` では次のようになります。

```
caji:> status activity show
Activity:
CPU          10 %util          Sunny
Disk         478 ops/sec       Partly Cloudy
iSCSI        0 ops/sec          Sunny
NDMP         0 bytes/sec        Sunny
NFSv3        681 ops/sec        Partly Cloudy
NFSv4        0 ops/sec          Sunny
Network      22.8M bytes/sec    Partly Cloudy
SMB          0 ops/sec          Sunny
caji:>
```

## ▼ ダッシュボードの連続実行

ブラウザで「ダッシュボード」画面を連続して (毎日 24 時間) 開いたままにすると、ブラウザメモリーの問題が発生することがあります。ブラウザのサイズが増加するため (メモリーリーク)、ブラウザを閉じて、ふたたび開く必要があります。ブラウザは、さまざまな Web サイトを参照している (また、タブを開いたり閉じたりしている) ときは、きわめて適切にメモリーを管理できます。問題は、「ダッシュボード」画面が実行されたままで閉じられないと、アクティビティのプロットのためにイメージを何回も開きなおすことになり、これによってイメージ描画のパフォーマンスが低下する点です。

Firefox の使用中にこの問題が発生する場合は、次のようにメモリーキャッシュを無効にします。

1. `about:config` を開きます
2. 「memory」でフィルタリングします
3. `browser.cache.memory.enable = false` を設定します

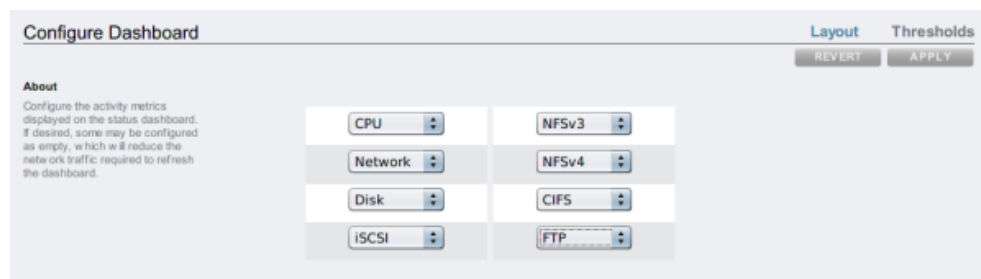
## 設定

### 概要

「ステータス」>「設定」画面を使用すると、表示される統計情報や、天気アイコンを通してアクティビティを示すしきい値など、[46 ページの「ステータス」「ダッシュボード」](#)をカスタマイズできます。

### BUI

図 2-6 ダッシュボードの設定



### レイアウト

次の表で定義された、[46 ページの「ダッシュボードアクティビティ」](#)領域に表示されるグラフを選択するには、レイアウトタブを使用します。

表 2-4 ステータスのレイアウトの設定

名前	単位	説明
<empty>	-	この場所にグラフは表示されません。

名前	単位	説明
SMB	操作数/秒	SMB 操作の平均数。
CPU	使用率	アプライアンス CPU がビジー状態である平均サイクル数。CPU サイクルには、メモリー待機サイクルが含まれます。
ディスク	操作数/秒	物理ストレージデバイスへの操作の平均数。
HTTP	操作数/秒	HTTP 操作の平均数。
iSCSI	操作数/秒	iSCSI 操作の平均数。
FC	操作数/秒	ファイバチャネル操作の平均数。
ネットワーク	バイト数/秒	すべての物理ネットワークインタフェースにわたる平均バイト数/秒。
NDMP	バイト数/秒	平均の NDMP ネットワークバイト数。
NFSv2	操作数/秒	NFSv2 操作の平均数。
NFSv3	操作数/秒	NFSv3 操作の平均数。
NFSv4	操作数/秒	NFSv4 操作の平均数。
FTP	バイト数/秒	FTP バイトの平均数。
SFTP	バイト数/秒	SFTP バイトの平均数。

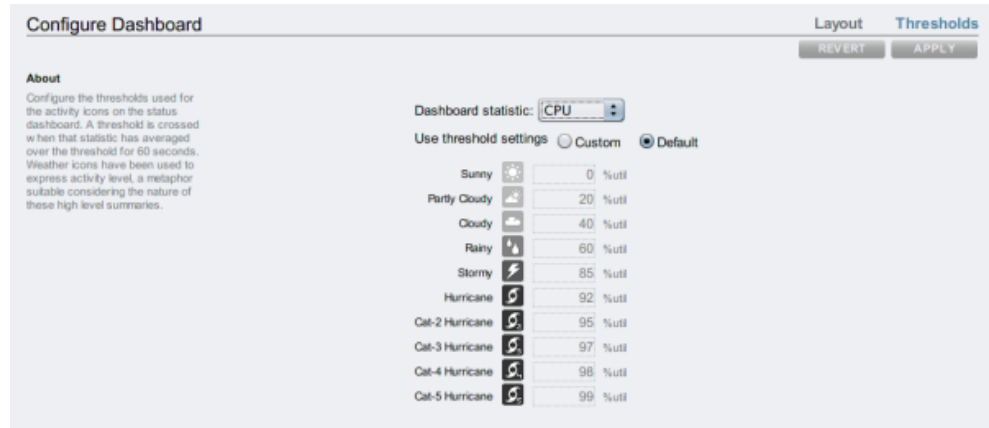
「ダッシュボード」をリフレッシュするために必要なネットワークトラフィックを削減するには、一部のアクティビティグラフを「<empty>」として構成してください。

## しきい値

46 ページの「ダッシュボードアクティビティ」の天気アイコンを構成するには、「しきい値」画面を使用します。提供されるデフォルト値は負荷の高いワークロードに基づいているため、実際の環境に適していないことがあります。



図 2-7 ダッシュボードアクティビティの設定



46 ページの「ダッシュボード」に表示される天気アイコンは、現在のアクティビティに対するしきい値の設定にもっとも近い (60 秒の平均として測定される) アイコンです。たとえば、CPU 使用率が 41% であった場合、デフォルトでは、しきい値が 40% (実際のアクティビティにもっとも近い) である曇りの天気アイコンが表示されます。しきい値を構成し、しかも必ず画面に表示される順番に構成するようにするには、「カスタム」ラジオボタンを選択します。

## CLI

現在、ダッシュボードを CLI から構成することはできません。BUI で保存された設定は、CLI から表示されるダッシュボードに適用されます。

## タスク

このトピックでのタスクの例を、手順を列挙して次に示します。

## BUI

### ▼ 表示されるアクティビティ統計情報の変更

1. 「ステータス」>「設定」>「レイアウト」画面に移動します。

2. ドロップダウンメニューから、「ダッシュボード」に表示する統計情報を選択します。
3. 選択を保存するには、「適用」ボタンをクリックします。

#### ▼ アクティビティーのしきい値の変更

1. 「ステータス」>「設定」>「しきい値」画面に移動します。
2. ドロップダウンメニューから、構成する統計を選択します。
3. 「カスタム」ラジオボタンをクリックします。
4. リスト内の値を表示される順番にカスタマイズします。一部の統計情報では、「Kilo」/「Mega」/「Giga」を選択できるように「単位」ドロップダウンが表示されます。
5. 構成を保存するには、「適用」ボタンをクリックします。

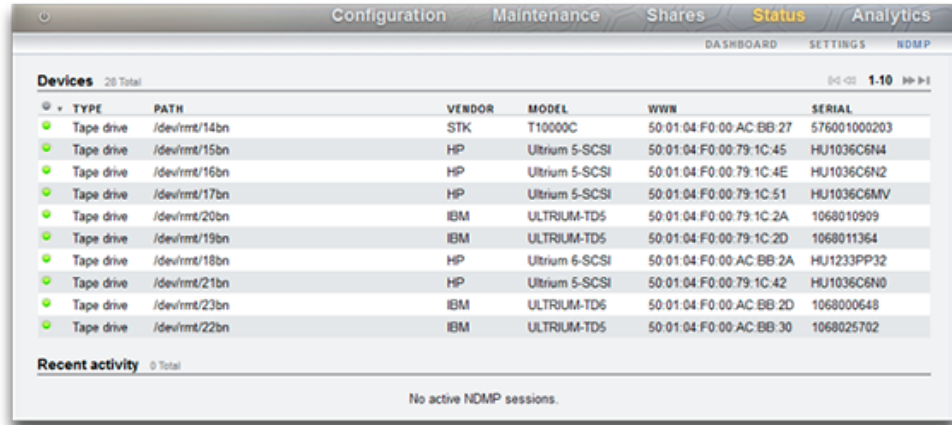
## NDMP ステータス

[230 ページの「NDMP サービス」](#)が構成されていて、アクティブである場合、「ステータス」=>「NDMP」ページには NDMP デバイスと最新のクライアントアクティビティーが表示されます。緑色のインジケータはデバイスがオンラインであることを示し、グレーのインジケータはデバイスがオフラインであることを示しています。

## NDMP のステータス - BUI

NDMP デバイスリストを再ソートするには、「デバイス」列見出しをクリックします。デバイスの詳細を表示するには、そのデバイスをダブルクリックします。

図 2-8 NDMP のステータスの BUI



## NDMP のステータス - デバイス

ここには NDMP デバイスが一覧表示されます。

表 2-5 NDMP のステータス - デバイス

フィールド	説明	例
タイプ	NDMP デバイスのタイプ	ロボット、テープドライブ
パス	NDMP デバイスのパス	/dev/rmt/14bn
ベンダー	デバイスのベンダー名	STK
モデル	デバイスのモデル名	T1000C
WWN	World Wide Name	50:01:04:F0:00:AC:BB:27
シリアル	デバイスのシリアル番号	576001000203

## NDMP のステータス - 最新のアクティビティー

このセクションには、最新の NDMP アクティビティーのサマリーが表示されます。

表 2-6 NDMP のステータス - 最新のアクティビティー

フィールド	説明	例
ID	NDMP のバックアップ ID	49

フィールド	説明	例
アクティブ	現在アクティブなバックアップ	いいえ
リモートクライアント	NDMP クライアントのアドレスとポート	192.168.1.219:4760
認証済み	クライアントがすでに認証を完了したかどうかを示す	はい、いいえ
データの状態	「データの状態」を参照	アクティブ、アイドル、...
ムーバーの状態	「ムーバーの状態」を参照	アクティブ、アイドル、...
現在の操作	現在の NDMP 操作	バックアップ、復元、なし
進行状況	このバックアップのための進行状況バー	

## NDMP データの状態

このフィールドは、バックアップまたは復元操作の状態を示します。可能性のある値は次のとおりです。

- 「アクティブ」: データがバックアップまたは復元されています。
- 「アイドル」: バックアップまたは復元がまだ開始されていないか、あるいはすでに完了しています。
- 「接続完了」: 接続は確立されていますが、バックアップまたは復元がまだ開始されていません。
- 「停止」: バックアップまたは復元が正常に完了したか、あるいは失敗または異常終了しました。
- 「待機」: 操作はリモート接続の受信を待機しています。

## NDMP ムーバーの状態

このフィールドは、NDMP デバイスサブシステムの状態を示します。テープデバイスの例を次に示します。

- 「アクティブ」: データがテープから読み取られているか、またはテープに書き込まれています。
- 「アイドル」: テープ操作がまだ開始されていないか、あるいはすでに完了しています。
- 「一時停止」: テープが最後に達したか、または変更されるのを待機しています。
- 「停止」: 読み取り/書き込み操作が正常に完了したか、あるいは失敗または異常終了しました。

- 「待機」: 操作はリモート接続の受信を待機しています。

## NDMP のステータス - CLI

現在、NDMP ステータスは CLI から使用できません。



# ◆◆◆ 第 3 章

## 初期構成

---

初期構成は、次の 6 つのセクションで構成されます。

- [第4章「ネットワーク構成」](#)
- [267 ページの「DNS」](#)
- [271 ページの「時間」](#)
- ネームサービス  
([248 ページの「NIS」](#)、[250 ページの「LDAP」](#)、[254 ページの「Active Directory」](#))
- [第5章「ストレージ構成」](#)
- [275 ページの「登録およびサポート」](#)

## 前提条件

初期システム構成は、「[「設置」](#)」に記載されているように、システムにはじめて電源を投入して接続を確立したあとに実行します。

---

**注記 - 注:** クラスタの初期構成を実行するオプションは、BUI でのみ使用できません。このオプションを選択する場合、初期構成を開始する前に[第10章「クラスタ構成」](#)を読み、クラスタを正しく設定するために必要な追加の詳細手順を参照してください。[174 ページの「クラスタ化におけるネットワークの考慮点」](#)のセクションに十分注意してください。あるいは、次の手順を使用して、クラスタ対応のアプライアンスを最初にスタンドアロン動作用に構成し、あとでクラスタ動作用に再構成できます。

---

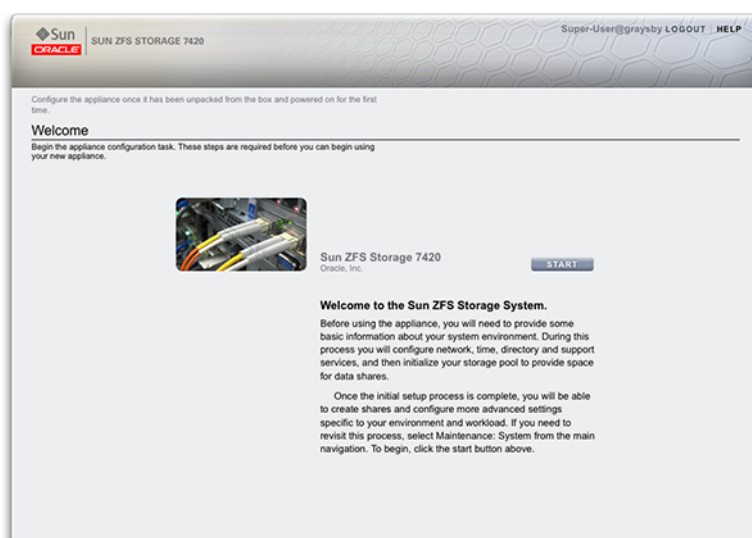
## BUI を使用して初期構成を実行する

初期構成では、スタンドアロン動作用のネットワーク接続、いくつかのクライアントネットワークサービス、およびストレージプールのレイアウトを構成します。完了すると、アプライアンスは使用できるようになりますが、リモートクライアントがアクセスするためのシェアは構成されていません。シェアを作成したり、設定をやり直すには、[第12章「シェア、プロジェクト、およびスキーマ」](#)を参照してください。

初期構成は「『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「システム」画面で「初期セットアップ」ボタンをクリックするか、CLI の maintenance system setup コンテキストに入ること、あとで繰り返すことができます。

BUI 初期構成は推奨される方式で、初期構成の手順ごとに画面が提供されています。

図 3-1 ZFSSA の「ようこそ」ページ



## ▼ 初期構成を実行する

1. 初期構成を開始するには、「ようこそ」ページで「開始」をクリックします。
2. 各ページで変更内容をコミットして次の画面に移動するには、「コミット」をクリックします。
3. 前の画面に移動するには矢印ボタンを使用します。

## 管理ポートの構成

すべてのスタンドアロンコントローラは、最低 1 つの NIC ポートを管理インターフェースとして構成する必要があります。BUI の「Admin を許可」オプションを選択して、ポート 215 の BUI 接続を有効化し、ssh ポート 22 の CLI 接続を有効化します。



上で説明したように、すべてのクラスタインストールでは、各コントローラの最低 1 つの NIC ポートを管理インタフェースとして構成する必要があります。さらに、各コントローラの NIC インスタンス番号は固有である必要があります。

## CLI を使用して初期構成を実行する

CLI を使用して初期構成セクションを実行できます。各ステップはステップのヘルプを出力して開始し、ヘルプは `help` と入力して出力し直すことができます。各ステップを完了するには `done` コマンドを使用します。

『[Oracle ZFS Storage Appliance インストールガイド](#)』の「[設置](#)」時に指定したパスワードを使用してログインします。

```
caji console login: root
Password:
Last login: Sun Oct 19 02:55:31 on console
```

```
To setup your system, you will be taken through a series of steps; as the setup
process advances to each step, the help message for that step will be
displayed.
```

```
Press any key to begin initial configuration ...
```

この例では、既存の設定 (DHCP サーバーから取得される) をチェックし、「`done`」と入力して受け入れます。この時点で設定をカスタマイズするには、各コンテキスト (データリンク、デバイス、およびインタフェース) を開始して「`help`」と入力し、そのコンテキストで使用できるアクションを表示します。追加のドキュメントについては、[第4章「ネットワーク構成」](#)のセクションを参照してください。クラスタを構成する場合は、[174 ページの「クラスタ化におけるネットワークの考慮点」](#)のセクションに十分注意してください。

```
aksh: starting configuration with "net" ...
```

```
Configure Networking. Configure the appliance network interfaces. The first
network interface has been configured for you, using the settings you provided
at the serial console.
```

```
Subcommands that are valid in this context:
```

```
  datalinks          => Manage datalinks
  devices            => Manage devices
  interfaces         => Manage interfaces
  help [topic]       => Get context-sensitive help. If [topic] is specified,
                       it must be one of "builtins", "commands", "general",
                       "help" or "script".
  show               => Show information pertinent to the current context
```

```

abort                => Abort this task (potentially resulting in a
                        misconfigured system)

done                 => Finish operating on "net"

caji:maintenance system setup net> devices show
Devices:

  DEVICE UP      MAC                SPEED
  igb0 true      0:14:4f:8d:59:aa    1000 Mbit/s
  igb1 false     0:14:4f:8d:59:ab     0 Mbit/s
  igb2 false     0:14:4f:8d:59:ac     0 Mbit/s
  igb3 false     0:14:4f:8d:59:ad     0 Mbit/s

caji:maintenance system setup net> datalinks show
Datalinks:

  DATALINK CLASS      LINKS      LABEL
  igb0 device         igb0       Untitled Datalink

caji:maintenance system setup net> interfaces show
Interfaces:

  INTERFACE STATE CLASS LINKS      ADDR          LABEL
  igb0 up     ip   igb0       192.168.2.80/22  Untitled Interface

caji:maintenance system setup net> done

```

DNS に関する追加のドキュメントについては、「[267 ページの「DNS」](#)」のセクションを参照してください。

Configure DNS. Configure the Domain Name Service.

Subcommands that are valid in this context:

```

help [topic]        => Get context-sensitive help. If [topic] is specified,
                        it must be one of "builtins", "commands", "general",
                        "help", "script" or "properties".

show                => Show information pertinent to the current context

commit              => Commit current state, including any changes

abort               => Abort this task (potentially resulting in a
                        misconfigured system)

done                => Finish operating on "dns"

get [prop]          => Get value for property [prop]. ("help properties"
                        for valid properties.) If [prop] is not specified,
                        returns values for all properties.

set [prop]          => Set property [prop] to [value]. ("help properties"
                        for valid properties.) For properties taking list
                        values, [value] should be a comma-separated list of
                        values.

caji:maintenance system setup dns> show
Properties:
<status> = online
domain = sun.com

```

```

servers = 192.168.1.4

caji:maintenance system setup dns> set domain=sf.fishworks.com
      domain = sf.fishworks.com (uncommitted)
caji:maintenance system setup dns> set servers=192.168.1.5
      servers = 192.168.1.5 (uncommitted)
caji:maintenance system setup dns> commit
caji:maintenance system setup dns> done
aksh: done with "dns", advancing configuration to "ntp" ...

```

時間情報プロトコル (NTP) を構成して、アプライアンスのタイムクロックを同期します。追加のドキュメントについては、[271 ページの「NTP」](#)のセクションを参照してください。

Configure Time. Configure the Network Time Protocol.

Subcommands that are valid in this context:

help [topic]	=> Get context-sensitive help. If [topic] is specified, it must be one of "builtins", "commands", "general", "help", "script" or "properties".
show	=> Show information pertinent to the current context
commit	=> Commit current state, including any changes
abort	=> Abort this task (potentially resulting in a misconfigured system)
done	=> Finish operating on "ntp"
enable	=> Enable the ntp service
disable	=> Disable the ntp service
get [prop]	=> Get value for property [prop]. ("help properties" for valid properties.) If [prop] is not specified, returns values for all properties.
set [prop]	=> Set property [prop] to [value]. ("help properties" for valid properties.) For properties taking list values, [value] should be a comma-separated list of values.

```

caji:maintenance system setup ntp> set servers=0.pool.ntp.org
      servers = 0.pool.ntp.org (uncommitted)
caji:maintenance system setup ntp> commit
caji:maintenance system setup ntp> done
aksh: done with "ntp", advancing configuration to "directory" ...

```

追加のドキュメントについては、[248 ページの「NIS」](#)、[250 ページの「LDAP」](#)、および[254 ページの「Active Directory」](#)のセクションを参照してください。

Configure Name Services. Configure directory services for users and groups. You can configure and enable each directory service independently, and you can configure more than one directory service.

Subcommands that are valid in this context:

nis	=> Configure NIS
-----	------------------

```

ldap          => Configure LDAP

ad            => Configure Active Directory

help [topic]  => Get context-sensitive help. If [topic] is specified,
                it must be one of "builtins", "commands", "general",
                "help" or "script".

show         => Show information pertinent to the current context

abort        => Abort this task (potentially resulting in a
                misconfigured system)

done         => Finish operating on "directory"

caji:maintenance system setup directory> nis
caji:maintenance system setup directory nis> show
Properties:
    <status> = online
    domain = sun.com
    broadcast = true
    ypservers =

caji:maintenance system setup directory nis> set domain=fishworks
    domain = fishworks (uncommitted)
caji:maintenance system setup directory nis> commit
caji:maintenance system setup directory nis> done
caji:maintenance system setup directory> done
aksh: done with "directory", advancing configuration to "support" ...

```

ベースとなるデータ冗長性によって特徴付けされるストレージプールを構成し、すべてのファイルシステムおよび LUN でシェアされるスペースを指定します。追加のドキュメントについては、[第5章「ストレージ構成」](#)のセクションを参照してください。

Configure Storage.

Subcommands that are valid in this context:

```

help [topic]  => Get context-sensitive help. If [topic] is specified,
                it must be one of "builtins", "commands", "general",
                "help", "script" or "properties".

show         => Show information pertinent to the current context

commit       => Commit current state, including any changes

done        => Finish operating on "storage"

config <pool> => Configure the storage pool

unconfig    => Unconfigure the storage pool

add         => Add additional storage to the storage pool

import      => Search for existing or destroyed pools to import

scrub <start|stop> => Start or stop a scrub

get [prop]   => Get value for property [prop]. ("help properties"
                for valid properties.) If [prop] is not specified,

```

```
returns values for all properties.

set pool=[pool]    => Change current pool

caji:maintenance system setup storage> show
Properties:
    pool = pool-0
    status = online
    profile = mirror
    log_profile = -
    cache_profile = -
caji:maintenance system setup storage> done
aksh: done with "storage", advancing configuration to "support" ...
```

リモートサポート構成に関する追加のドキュメントについては、(275 ページの「[フォンホーム](#)」) を参照してください。

Remote Support. Register your appliance and configure remote monitoring.

Subcommands that are valid in this context:

```
tags                => Configure service tags

scrk                 => Configure phone home

help [topic]        => Get context-sensitive help. If [topic] is specified,
                    it must be one of "builtins", "commands", "general",
                    "help" or "script".

show                 => Show information pertinent to the current context

abort                => Abort this task (potentially resulting in a
                    misconfigured system)

done                 => Finish operating on "support"
```

```
caji:maintenance system setup support> done
aksh: initial configuration complete!
```



# ◆◆◆ 第 4 章

## ネットワーク構成

---

ネットワーク構成機能により、物理的なネットワークポートを使用して、リンクアグリゲーション、仮想 NIC (VNIC)、仮想 LAN (VLAN)、マルチパスグループなどのさまざまな高度なネットワーク設定を作成できます。次に、システム上のさまざまなデータサービスへの接続で使用するために、これらの抽象化のための任意の数の IPv4 および IPv6 アドレスを定義できます。

システムのネットワーク構成には、次の 4 つのコンポーネントがあります。

- デバイス - 物理的なネットワークポート。デバイスは、物理的なネットワーク接続または InfiniBand (IPoIB) パーティション上の IP に対応します。
- データリンク - パケットを送受信するための基本的な構成体。データリンクは、デバイス (つまり、物理的なネットワークポート) または IB パーティションと 1:1 に対応させることができます。または、ほかのデバイスとデータリンクで構成されたアグリゲーション、VLAN、および VNIC データリンクを定義できます。
- インタフェース - IP の構成およびアドレス指定のための基本的な構成体。各 IP インタフェースは 1 つのデータリンクに関連付けられるか、またはほかのインタフェースで構成された IP マルチパス (IPMP) グループとして定義されます。
- ルーティング - IP ルーティングの構成。これにより、システムによる IP パケットの転送方法が制御されます。

### 「ネットワーク構成」ページ

ZFSSA モデルでは、ネットワークデバイスは使用可能なハードウェアを表し、構成可能な設定を持っていません。データリンクはレイヤー 2 のエンティティであり、これらのネットワークデバイスに LACP などの設定が適用されるように作成する必要があります。インタフェースは、IP 設定が含まれたレイヤー 3 のエンティティです。これらの設定は、データリンクを介して使用可能になります。このモデルでは、ネットワークインタフェースの設定が、レイヤー 2 の設定のためのデータリンクと、レイヤー 3 の設定のためのインタフェースという 2 つの部分に分離されています。

図 4-1 「ネットワーク構成」ウィンドウ



1つのポート上に1つのIPアドレスがある場合(一般的な構成)の例を次に示します。

表 4-1 例 - 1つのポート上の1つのIPアドレス

デバイス	データリンク	インタフェース
igb0	datalink1	deimos (192.168.2.80/22)

3ウェイリンクアグリゲーションのための次の構成を考えてみます。

表 4-2 例 - 3ウェイリンクアグリゲーションのための構成

デバイス	データリンク	インタフェース
igb1, igb2, igb3	aggr1 (LACP アグリゲーション)	phobos (192.168.2.81/22)

データリンクエンティティ(名前を「aggr1」とします)によって、ネットワークデバイスが構成可能な方法(LACPアグリゲーションポリシー)でグループ化されます。インタフェースエンティティ(名前を「phobos」とします)によって、構成可能なIPアドレス設定が提供されます。これらの設定は、データリンクを介してネットワーク上で使用可能になります。ネットワークデバイス(システムが付けた名前は「igb1」、「igb2」、...)には、直接の設定はありません。ネットワークデバイスに特定の設定が適用されるかどうかにかかわらず、データリンクは、ネットワーク構成を完了するために必要です。



## デバイス

デバイスは、使用可能なネットワークまたは InfiniBand ポートを表すために、システムによって作成されます。独自の構成設定は持っていません。

## データリンク

データリンクはデバイスを管理し、インタフェースによって使用されます。次のものをサポートします。

- LACP - 複数のネットワークデバイスを束ねて 1 つのネットワークデバイスとして動作させるためのリンクアグリゲーション制御プロトコル。これにより、パフォーマンス (帯域幅が増加する) および信頼性 (ネットワークポートの障害を乗り切ることができる) が向上しますが、LACP をサポートし、これらのポートに対して LACP が有効になっているスイッチにアプライアンスを接続する必要があります。
- IB パーティション - 論理的に分離された IB ファブリックドメインに接続するための InfiniBand パーティション。
- VLAN - ローカルネットワークのセキュリティおよび分離を向上させるための仮想 LAN。アプライアンスの管理には VLAN が推奨されており、それ以外の場合は VNIC を使用します。
- VNIC - 単一または集合 Ethernet データリンクを複数の仮想 (Ethernet) データリンクに分割可能な仮想ネットワークインタフェースカード。VNIC は、オプションとして VLAN ID でタグ付けでき、クラスタ内で物理ネットワークポートをシェアできます。手順については、次の [174 ページ](#)の「[クラスタ化におけるネットワークの考慮点](#)」のセクションを参照してください。

---

**注記** - VNIC ベースと VLAN ベースのデータリンクが、同じ VLAN ID をシェアすることはできません。

IEEE802.3ad (リンクアグリゲーション) の標準では、複数のスイッチにわたるアグリゲーションは明示的にサポートされていませんが、一部のベンダーは独自の拡張機能によって、マルチスイッチのサポートを提供しています。これらの拡張機能で構成されたスイッチが IEEE 標準に準拠しており、またエンドノードに対して透過的であれば、アプライアンスでこれらの拡張機能の使用がサポートされます。問題が発生した場合、Oracle サポートでは、単一スイッチ構成で問題を再現することが必要になる場合があります。

---

次のデータリンク設定が使用できます。

表 4-3 データリンクの設定

プロパティ	説明
名前	定義されたカスタム名を使用します。たとえば、「internal」、「external」、「adminnet」などがあります。
速度	定義された速度を使用します。有効な値は auto、10、100、1000、および 10000 であり、それぞれ、自動ネゴシエーション、強制 10M ビット/秒、強制 100M ビット/秒、強制 1G ビット/秒、および強制 10G ビット/秒を表します。速度と全二重/半二重は、両方が特定の値に強制されるか、または両方が自動ネゴシエーションに設定されるのどちらかである必要があります。すべてのネットワークデバイスが、速度と全二重/半二重の可能性のあるすべての組み合わせへの強制をサポートしているわけではありません。自動ネゴシエーションを無効にしないことが強く推奨されています。ただし、スイッチで自動ネゴシエーションが無効になっている場合は、データリンクが必ず予期された速度と全二重/半二重で動作するように、速度 (および全二重/半二重) の強制が必要になることがあります。
全二重/半二重	定義された転送方向を使用します。有効な CLI 値は auto、half、および full であり、それぞれ、自動ネゴシエーション、半二重、および全二重を表します。速度と全二重/半二重は、両方が特定の値に強制されるか、または両方が自動ネゴシエーションに設定されるのどちらかである必要があります。
VLAN	VLAN ヘッダーを使用します。
VLAN ID	定義された VLAN 識別子を使用します (VNIC の場合はオプション)。
VNIC	VNIC を使用します。
MTU	定義された最大転送単位 (MTU) サイズを使用します。デフォルト MTU は 1500 バイトです。パケットに余裕を持たせるには (トンネリングプロトコルなど)、指定する MTU (最小 1280) を小さくします。ネットワークのパフォーマンスを改善するには、指定する MTU (最大 9000) を大きくします。同じ LAN 上のすべてのシステムおよびスイッチが、選択された MTU を使用して構成されている必要があります。MTU 値が設定され、システムに対して新しいネットワーク構成がコミットされたあとは、ネットワーク画面に戻り、データリンクのステータスを表示して、選択された正確な MTU 値をバイト単位で確認できます。VLAN および VNIC

プロパティ	説明
	は、ベースとなるデータリンクより大きい MTU 値では構成できないことに注意してください。
LACP アグリゲーション	複数のネットワークデバイス LACP アグリゲーションを使用します。
LACP ポリシー	定義済みの LACP ポリシーをアウトバウンドポートの選択に使用します。L2 は発信元と宛先の MAC アドレスをハッシュし、L3 は発信元と宛先の IP アドレスを使用し、L4 は発信元と宛先のトランスポートレベルのポートを使用します
LACP モード	定義済みの LACP 通信モードを使用します。アクティブモードでは、接続のネゴシエーションを行うために LACP メッセージを送受信し、リンクステータスをモニターします。受動モードでは、LACP メッセージのみを待機します。オフモードでは、集合リンクを使用しますが、リンク障害やスイッチ構成の変更を検出しません。Cisco Etherchannel を含む一部のネットワークスイッチ構成では、LACP プロトコルは使用されません。ネットワーク内で LACP 以外のアグリゲーションを使用する場合は、LACP モードを「オフ」に設定してください。
LACP タイマー	LACP メッセージ間の定義済みの間隔をアクティブモードで使用します。
IB パーティション	IB パーティションを使用します。
パーティションキー	ベースとなるポートデバイスがメンバーであるパーティション (ファブリックドメイン) を使用します。パーティションキー (pkey) はサブネットマネージャーによって検出され、構成されます。サブネットマネージャーを構成する前に pkey を定義できますが、サブネットパーティションがポート GUID をメンバーとして使用して正しく構成されるまで、データリンクは「ダウン」したままになります。HCA ポートのパーティションメンバーシップを、サブネットマネージャー上の <a href="#">77 ページの「ネットワーク IP マルチパス (IPMP)」</a> や <a href="#">第10章「クラスタ構成」</a> の規則と整合性がある状態に維持することが重要です。
IB リンクモード	定義済みの IB リンクモードを使用します。「信頼できないデータグラム」と「接続完了」の 2 つのモードがあります。「信頼できないデータグラム」では、ローカルキューのペアが任意のホスト上のほかの複数のキューのペアと通信することができ、メッセージは IB 層で未確認のまま通信されます。「信頼できないデータグラム」モードでは、2044 の MTU を使用します。「接続完了」モードでは、IB キューのペアを使用し、専用のリモートキューのペアと通信するためのローカルキュー

プロパティ	説明
	のペアを専用に割り当てます。「接続完了」モードでは 65520 の MTU を使用するため、「信頼できないデータグラム」より高いスループットが可能になります。

## ネットワークインタフェース

ネットワークインタフェースは、データリンクを介して IP アドレスを構成します。次がサポートされています。

- IPv4 および IPv6 プロトコル。
- IPMP - IP アドレスが、障害の発生したデータリンクから正常なデータリンクに自動的に移行できるようにすることによってネットワークの信頼性を向上させる IP マルチパス。

次のインタフェース設定が使用できます。

表 4-4 インタフェースの設定

プロパティ	説明
名前	インタフェースのカスタム名
管理を許可	このインタフェースを介したアプライアンスの管理 BUI または CLI への接続を許可します。ネットワーク環境に個別の管理ネットワークが含まれていた場合は、セキュリティを向上させるために、このプロパティはその管理ネットワークに対してのみ有効にできます
インタフェースを有効化	このインタフェースを IP トラフィックに使用できるようにします。あるインタフェースが無効になっている場合、アプライアンスはそのインタフェースを介して IP トラフィックを送受信しなくなるか、またはそのインタフェース上に構成されている任意の IP アドレスを使用します。現時点では、IPMP グループ内のアクティブ IP インタフェースを無効にしても、スタンバイインタフェースのアクティブ化はトリガーされません。
IPv4 構成:	手動で入力される「静的アドレスリスト」か、または動的に要求するための「DHCP」のどちらか
IPv4 アドレス/マスク	CIDR 表記での 1 つ以上の IPv4 アドレス (192.168.1.1/24)
IPv6 構成:	手動で入力される「静的アドレスリスト」か、または自動的に生成されたリンクローカルアドレス (およ

プロパティ	説明
	び、IPv6 ルーターが応答する場合はサイトローカル) を使用するための「IPv6 自動構成」のどちらか
IPv6 アドレス/マスク	CIDR 表記での 1 つ以上の IPv6 アドレス (1080::8:800:200C:417A/32)
IP マルチパスグループ	冗長性のためにデータリンクのプールを使用できる IP マルチパスを構成します

## ネットワーク IP マルチパス (IPMP)

IP マルチパスグループは、IP インタフェースの障害 (物理的な回線の切断や、ネットワークデバイスとスイッチの間の接続の障害など) が発生した場合、またはシステムとネットワークゲートウェイの間のパスの障害が発生した場合でも引き続き使用できる IP アドレスを提供するために使用されます。システムは、下記で説明するように、IP インタフェースの基になるデータリンクのリンクアップおよびリンクダウン通知をモニタリングしたり、オプションでグループ内の各 IP インタフェースに割り当てることができるテストアドレスを使用してプローブしたりすることによって障害を検出します。IPMP グループには、すべてが同じリンク (LAN、IB パーティション、または VLAN) 上に存在するかぎり任意の数の IP インタフェースを設定でき、また任意の数の高可用性アドレスを割り当てることができます。

IPMP グループ内の各 IP インタフェースは、<i>アクティブ</i>または<i>スタンバイ</i>のどちらかに指定されます。

- **アクティブ:** この IP インタフェースは、IPMP によって正しく機能していると判断されるかぎり、データを送受信するために使用されます。
- **スタンバイ:** この IP インタフェースは、アクティブインタフェース (または、以前にアクティブ化されたスタンバイ) が機能を停止した場合にのみ、データを送受信するために使用されます。

アクティブおよびスタンバイ IP インタフェースは複数構成できますが、各 IPMP グループには、少なくとも 1 つのアクティブ IP インタフェースが構成されている必要があります。IPMP は、構成された数のアクティブインタフェースを保持するために、必要な数のスタンバイをアクティブにしようとします。たとえば、IPMP グループに 2 つのアクティブインタフェースと 2 つのスタンバイインタフェースが構成されており、すべてのインタフェースが正しく機能している場合は、2 つのアクティブインタフェースのみがデータを送受信するために使用されます。アクティブインタフェースに障害が発生した場合は、いずれかのスタンバイインタフェースがアクティブ化されます。ほかのアクティブインタフェースに障害が発生した (または、アクティブ化されたスタンバイに障害が発生した) 場合は、2 番目のスタンバイインタフェースがアクティブ化されます。そのあと、アクティブインタフェースが修復された場合は、スタンバイインタフェースがもう一度非アクティブ化されます。

IP インタフェースの障害は、リンクベースの検出またはプローブベースの検出 (つまり、テストアドレスが構成されている) で発見できます。

IP インタフェース上でプローブベースの障害検出が有効になっている場合、システムは、プローブするターゲットシステムを動的に決定します。最初に、IP インタフェースのテストアドレスと同じサブネット上にゲートウェイ (ルーター) がないかどうかルーティングテーブルがスキャンされ、最大 5 つが選択されます。同じサブネット上にゲートウェイが見つからない場合、システムはマルチキャスト ICMP プローブを (IPv4 の場合は 224.0.0.1、IPv6 の場合は ff02::1 に) 送信し、同じサブネット上の応答した最初の 5 つのシステムを選択します。そのため、IPMP を使用したネットワーク障害検出および修復のために、各リンク上の少なくとも 1 つの近傍ノードまたはデフォルトゲートウェイが ICMP エコーリクエストに応答することを確認してください。IPMP は、IPv4 と IPv6 の両方のアドレス構成で動作します。IPv6 の場合は、インタフェースのリンクローカルアドレスがテストアドレスとして使用されます。

---

**注記** - IPMP テストのアドレスと同じサブネット上に、ICMP エコーリクエストに응答するように構成されている (クラスタピア以外の) システムがない場合、プローブベースの障害検出を使用しないでください。

---

システムは、選択されたターゲットシステムをラウンドロビン方式でプローブします。5 回のプローブに連続して応答がない場合、その IP インタフェースは障害が発生していると考えられます。逆に、10 回のプローブに連続して応答があった場合、システムは、以前に障害が発生した IP インタフェースを修復されたと考えられます。システムの IPMP プローブ障害検出時間を、「[270 ページの「IPMP」](#)」画面から設定できます。この時間によって、プローブの頻度と修復間隔が間接的に制御されます。たとえば、障害検出時間が 10 秒である場合、システムはおおよそ 2 秒間隔でプローブを送信すること、およびシステムがプローブベースのインタフェース修復を検出するには 20 秒かかることを示します。システムが選択したターゲットシステムを直接制御することはできませんが、ルーティングテーブルを介して間接的に制御することは可能です。

システムはルーティングテーブルをモニターし、必要に応じて、選択したターゲットシステムを自動的に調整します。たとえば、システムではマルチキャスト検出ターゲットを使用しているが、そのあと IP インタフェースのテストアドレスと同じサブネット上にゲートウェイを含むルートが追加された場合、システムはそのゲートウェイのプローブに自動的に切り替えます。同様に、マルチキャスト検出ターゲットがプローブされている場合、システムは一連の選択されたターゲットを定期的に取り替えます (たとえば、以前に選択されたターゲットの一部が応答しなくなっているため)。

IPMP グループを作成する手順については、[77 ページの「ネットワーク IP マルチパス \(IPMP\)」](#)を参照してください。

プライベートローカルインタフェースの詳細は、[第10章「クラスタ構成」](#)を参照してください。

## ネットワークのパフォーマンスと可用性

IPMP とリンクアグリゲーションは、ネットワークパフォーマンスの向上やネットワーク可用性の維持を実現するためにアプライアンスで使用可能な異なるテクノロジーです。一般に、リンクアグリゲーションはより優れたネットワークパフォーマンスを得るために配備するのに対して、IPMP は高可用性を保証するために使用します。この 2 つのテクノロジーは互いを補完するものであり、ネットワークパフォーマンスと可用性の組み合わせられた利点を実現するために一緒に配備できます。

リンクアグリゲーションでは、受信したトラフィックは、アグリゲーションを構成する複数のリンクにわたって分散されます。そのため、アグリゲーションにリンクを追加するために多くの NIC がインストールされているほど、ネットワークパフォーマンスは向上します。IPMP のトラフィックは、使用可能なアクティブインタフェースにバインドされている IPMP インタフェースのデータアドレスを使用します。たとえば、すべてのデータトラフィックが、必ずしも同じ接続上にはない 2 つの IP アドレス間だけを転送される場合は、多くの NIC を追加しても引き続き 2 つの IP アドレスだけが使用可能であるため、IPMP に関するパフォーマンスは向上しません。

パフォーマンスは、特定のデバイスのデータリンク上に構成された VNIC/VLAN の数、および VLAN ID の使用に影響されることがあります。特定のデバイスで複数の VNIC を構成する場合、VNIC が使用されていない場合でも、そのデバイス上のすべてのデータリンクのパフォーマンスが最大 5 パーセント影響を受ける可能性があります。特定のデータリンク上で 8 つ以上の VNIC/VLAN が構成されている場合は、パフォーマンスが大幅に低下する可能性があります。また、データリンクで VLAN ID が使用されている場合、そのデバイスのすべてのデータリンクパフォーマンスがさらに 5 パーセント影響を受ける可能性があります。

## ネットワークルーティングの構成

システムは、ルーティングテーブルエントリのコレクションから成る 1 つの IP ルーティングテーブルを提供します。IP パケットを特定の宛先に送信する必要がある場合、システムは、そのパケットの宛先アドレスに宛先がもっとも緊密に一致するルーティングエントリを選択します (システムのマルチホーミングポリシーに従います。下を参照)。次に、そのルーティングエントリ内の情報を使用して、パケットを送信するための IP インタフェースと、宛先に直接到達できない場合に使用する次のホップのゲートウェイを決定します。宛先に一致するルーティングエントリがない場合、そのパケットはドロップされます。複数のルーティングエントリが同程度のもっとも緊密な一致を示している (かつ、それ以外の場合はマルチホーミングポリシーによって優先付けされていない) 場合、システムは、接続単位にこれらのエントリにわたって負荷分散します。

システムは、ルーターとしては機能しません。

## ネットワークルーティングのエントリ

ルーティングテーブルは、それぞれが次のフィールドを含むルーティングエントリで構成されています。

表 4-5 ルーティングエントリのフィールド

フィールド	説明	例
宛先	ルートに一致する可能性のある IP 宛先アドレス (CIDR 表記) の範囲	192.168.0.0/22
ゲートウェイ	パケットの送信先のネクストホップ (IP アドレス) («system») ルートを除く。下を参照)	192.168.2.80
ファミリー	インターネットプロトコル	IPv4, IPv6
タイプ	ルートの起点	dhcp, static, system
インタフェース	パケットが送信されるとき IP インタフェース	igb0

0.0.0.0/0 の「宛先」フィールドを含むルーティングエントリは、すべてのパケットに一致するため (より正確に一致するほかのルートがない場合)、「デフォルト」ルートと呼ばれます。BUI では、デフォルトルートは、次の追加のプロパティによってデフォルト以外のルートと区別されます。

表 4-6 デフォルトルートとデフォルト以外のルートの区別

種類	ルートの種類	デフォルト、ネットワーク
----	--------	--------------

上に示すように、特定の packets は、ルーティングエントリの「インタフェース」フィールドで指定された IP インタフェース上で送信されます。IPMP インタフェースが指定されている場合は、IPMP グループ内のいずれかのアクティブ IP インタフェースが接続単位にランダムに選択され、選択された IP インタフェースがその後使用不可になった場合は自動的にリフレッシュされます。逆に、特定の IP インタフェースが IPMP グループに含まれている場合、このようなルートは可用性が高くないため、この IP インタフェースを「インタフェース」フィールドで指定することはできません。

ルーティングエントリは、「タイプ」フィールドで識別された、いくつかの異なる起点から来ています。ルーティングエントリの起点はシステムでのルーティングエントリの使用方法とは関係ありませんが、各エントリの編集や削除が可能かどうかや、その方法はこの起点によって制御されます。システムは、次のタイプのルートをサポートしています。



表 4-7 サポートされているルートのタイプ

タイプ	説明
静的	アプライアンス管理者によって作成および管理されます。
システム	IP インタフェースの有効化の一部として、アプライアンスによって自動的に作成されます。システムルートは、アプライアンスから直接到達できる IP サブネットごとに作成されます。これらのルートは直接到達可能であるため、代わりに「ゲートウェイ」フィールドが、そのサブネット上のアプライアンスの IP アドレスを識別します。
DHCP	DHCP を使用するように構成されている IP インタフェースの有効化の一部として、アプライアンスによって自動的に作成されます。DHCP ルートは、DHCP サーバーによって提供されるデフォルトルートごとに作成されます。
動的	RIP および RIPng ダイナミックルーティングプロトコルを介して、アプライアンスによって自動的に作成されます (これらのプロトコルが有効になっている場合)。

次の 1 つの追加のタイプによって、現在使用できない静的ルートが識別されます。

表 4-8 使用不可の静的ルートタイプ

非アクティブ	無効になっているか、またはオフラインの IP インタフェースに関連付けられた、以前に作成された静的ルート。
--------	---

## ネットワークルーティングのプロパティ

表 4-9 ルーティングのプロパティ

プロパティ	説明
マルチホーミングモデル	複数の IP インタフェースが同時に有効になっている場合に、IP パケットを受け入れて送信するためのシステムポリシーを制御します。許可される値は、「ルーズ」(デフォルト)、「適応可能」、および「厳格」です。下の説明を参照してください。

システムに複数の IP インタフェースが構成されている場合は、特定の宛先への同等のルートが複数存在する可能性があるため、パケットを送信するための IP インタフェースをシ

システムが強制的に選択するようにします。同様に、パケットがある IP インタフェースに到着したが、別の IP インタフェース上にホストされている IP アドレスに転送される可能性もあります。このような状況でのシステムの動作は、選択されたマルチホーミングポリシーによって決定されます。次の 3 つのポリシーがサポートされています。

表 4-10 マルチホーミングポリシー

ポリシー	説明
ルーズ	IP パケットと、その IP パケットを送受信するために使用される IP インタフェースの間にバインドを強制的に適用しません。1) IP パケットは、その宛先 IP アドレスがアプライアンス上で有効であるかぎり、IP インタフェース上で受け入れられます。2) IP パケットは、IP パケットの宛先アドレスにもっとも明確に一致するルートに関連付けられた IP インタフェースを介して転送されます。その IP インタフェース上にホストされている IP アドレスはまったく考慮されません。使用可能なルートが存在しない場合は、そのパケットをドロップします。
適応可能	パケットの発信元 IP アドレスと同じサブネット上にゲートウェイアドレスを含むルートが優先される点を除き、「ルーズ」と同じです。1) IP パケットは、その宛先 IP アドレスがアプライアンス上で有効であるかぎり、IP インタフェース上で受け入れられます。2) IP パケットは、IP パケットの宛先アドレスにもっとも明確に一致するルートに関連付けられた IP インタフェースを介して転送されます。複数のルートが均等に一致する場合は、パケットの発信元アドレスと同じサブネット上にゲートウェイアドレスを含むルートが優先されます。使用可能なルートが存在しない場合は、そのパケットをドロップします。
厳しい	IP パケットと、その IP パケットを送受信するために使用される IP インタフェースの間に厳格なバインドが必要です。1) IP パケットは、その宛先 IP アドレスがある IP インタフェース上で有効であるかぎり、その IP インタフェース上で受け入れられます。2) IP パケットは、その発信元 IP アドレスがある IP インタフェース上で有効な場合にのみ、その IP インタフェースを介して転送されます。このポリシーを強制的に適用するために、アプライアンスは使用可能なルートに対する一致を調べる場合、パケットの発信元アドレスとは異なるサブネット上にゲートウェイアドレスを含むルートすべてを無視します。使用可能なルートが残っていない場合は、そのパケットをドロップします。

マルチホーミングポリシーを選択する場合は、アプライアンスのいずれかの IP インタフェースが管理専用（たとえば、BUI アクセス専用）に使用され、そのために個別の管理ネットワークを介してアクセスされるかどうかを考慮することが重要です。特に、管理ネットワークにリモートアクセスを提供するためにデフォルトルートが作成され、またストレージプロトコルにリモートアクセスを提供するために別のデフォルトルートが作成される場合は、「ルーズ」のデフォルトのシステムポリシーを選択すると、管理用のデフォルトルートがストレージトラフィックに使用される可能性があります。ポリシーを「適応可能」または「厳しい」に切り換えることにより、アプライアンスは、要求に関連付けられた IP アドレスを応答のためのルートの選択の一部として見なすようになります。同じ IP インタフェース上にルートを見つけることができない場合、「適応可能」ポリシーでは使用可能な任意のルートが使用されるのに対して、「厳しい」ポリシーではパケットがドロップされます。

## BUI を使用したネットワーク構成

BUI を使用してネットワークを再構成する場合、システムは、ユーザーのブラウザへの現在のネットワーク接続を保持するためにあらゆる努力を行います。ただし、ブラウザの接続先の特定のアドレスを削除する場合など、ネットワーク構成の変更の中には、ブラウザが接続を失うことが避けられないものがあります。このため、管理者が使用するための特定の IP アドレスとネットワークデバイスを割り当てておき、常にそのアドレスを構成された状態に保つことをお勧めします。また、必要に応じて、シリアルコンソールを介して CLI から特に複雑なネットワーク再構成タスクを実行することもできます。

次のアイコンは、「構成」->「ネットワーク」セクションで使用されます。


表 4-11 ネットワーク構成のアイコン

アイコン	説明
	新しいデータリンク/インタフェース/ルートの追加
	データリンク/インタフェース/ルート設定の編集
	編集無効
	データリンク/インタフェース/ルートの破棄
	破棄無効
	ドラッグ & ドロップアイコン
	接続されたネットワークポート


アイコン	説明
	I/O アクティビティにより接続されたネットワークポート
	切断されたネットワークポート (リンク停止、ケーブルの問題か)
	アクティブな InfiniBand ポート
	I/O アクティビティによりアクティブな InfiniBand ポート
	アクティブでない InfiniBand ポート (停止、初期化、またはアラーム状態)
	InfiniBand パーティションデバイスが稼働中
	InfiniBand パーティションデバイスが停止中 (サブネットマネージャーの問題)
	ネットワークデータリンク
	ネットワークデータリンク VLAN または VNIC
	ネットワークデータリンクアグリゲーション
	ネットワークデータリンクアグリゲーション VLAN または VNIC
	ネットワークデータリンク IB パーティション
	インタフェースがパケットの送受信に使用されている (稼働中または縮退のどちらか)
	インタフェースがユーザーによって無効にされている
	インタフェースがオフライン (クラスタピアによって所有されている)
	インタフェースに障害が発生したか、または重複した IP アドレスが構成されている


右上には「構成」、「アドレス」、「ルーティング」のためのローカルナビゲーションが配置され、これらの構成ビューが交互に表示されます。

## 「ネットワーク構成」ページ

デフォルトでは「構成」ページが表示され、「デバイス」、「データリンク」、および「インタフェース」のリストが管理のためのボタンとともに表示されます。マウスをエントリの上に移動すると追加の  アイコンが表示され、任意のエントリをクリックすると、そのエントリに関連付けられたほかのコンポーネントが強調表示されます。

「デバイス」リストには、右側にリンクステータスのほか、ネットワークポートの状態が反映されたアイコンも表示されます。ポートの切断が表示される場合は、ポートがネットワークに正しく接続されていることを確認してください。

ネットワークデバイスで IP アドレスを構成するには、最初にデータリンクを作成し、次にそのデータリンクを使用するインタフェースを作成します。 アイコンを使用するとその両方を実行ことができ、データリンクプロパティとインタフェースプロパティのダイアログが表示されます。

ネットワークインタフェースの構成には複数の方法があります。デバイスの  アイコンをクリックし、それをデータリンクテーブルにドラッグしてみてください。次に、そのデータリンクをインタフェーステーブルにドラッグします。その他の移動も可能です。この方法は、有効な移動が強調表示されている複雑な構成に役立つことがあります。

## ネットワークアドレス

このページには、現在のネットワーク構成のサマリー表が表示されます。次のフィールドがあります。

表 4-12 現在のネットワーク構成のサマリー

フィールド	説明	例
ネットワークデータリンク	データリンク名と詳細なサマリー	datalink1 (igb0 経由)
ネットワークインタフェース	インタフェース名と詳細なサマリー	IPv4 DHCP, datalink1 経由
ネットワークアドレス	このインタフェースでホストされているアドレス	192.168.2.80/22
ホスト名	ネットワークアドレスの解決されたホスト名	caji.sf.example.com

## 「ネットワークルーティング」ページ

このページには、IP ルーティングテーブルの構成と、上で説明した関連付けられたプロパティが表示されます。デフォルトでは、ルーティングテーブル内のすべてのエントリが表示されますが、サブナビゲーションバーを使用すると、このテーブルをタイプでフィルタ処理できます。

特定のルートをチェックするには、CLI で `traceroute` を使用します。

```
zfssa-source:> traceroute 10.80.198.102
traceroute: Warning: Multiple interfaces found; using 10.80.198.101 @ igb3
traceroute to 10.80.198.102 (10.80.198.102), 30 hops max, 40 byte packets
 1 10.80.198.1 (10.80.198.1) 6.490 ms 0.924 ms 0.834 ms
 2 10.80.198.102 (10.80.198.102) 0.152 ms 0.118 ms 0.099 ms
zfssa-target:> traceroute 10.80.198.101
traceroute: Warning: Multiple interfaces found; using 10.80.198.102 @ igb3
traceroute to 10.80.198.101 (10.80.198.101), 30 hops max, 40 byte packets
 1 10.80.198.1 (10.80.198.1) 1.031 ms 0.905 ms 0.769 ms
 2 10.80.198.101 (10.80.198.101) 0.158 ms 0.111 ms 0.109 ms
```

## CLI を使用したネットワーク構成

ネットワーク構成は、`configuration net` の下にあります。これには、`devices`、`datalinks`、`interfaces`、および `routing` のためのサブコマンドがあります。各サブコマンドでは、`show` コマンドを使用して現在の構成を表示できます。

```
caji:> configuration net
caji:configuration net> devices show
Devices:

DEVICE      UP      SPEED      MAC
igb0        true    1000 Mbit/s 0:14:4f:9a:b9:0
igb1        true    1000 Mbit/s 0:14:4f:9a:b9:1
igb2        true    1000 Mbit/s 0:14:4f:9a:b8:fe
igb3        true    1000 Mbit/s 0:14:4f:9a:b8:ff

caji:configuration net> datalinks show
Datalinks:

        DATALINK CLASS      LINKS      LABEL
        igb0 device      igb0       dataLink1

caji:configuration net> interfaces show
Interfaces:

INTERFACE STATE CLASS LINKS      ADDR      LABEL
        igb0 up      ip      igb0       192.168.2.80/22  caji

caji:configuration net> routing show
Properties:

        multihoming = loose

Routes:
```

ROUTE	DESTINATION	GATEWAY	INTERFACE	TYPE
route-000	0.0.0.0/0	192.168.1.1	igb0	dhcp
route-001	192.168.0.0/22	192.168.2.142	igb0	system

データリンク、インタフェース、およびルートの作成や構成のための関連コマンドを表示するには、各セクションで `help` を入力します。このコンテキストで有効なサブコマンドは次のとおりです。

<code>help [topic]</code>	=> Get context-sensitive help. If [topic] is specified, it must be one of "builtins", "commands", "general", "help", "script" or "properties".
<code>show</code>	=> Show information pertinent to the current context
<code>commit</code>	=> Commit current state, including any changes
<code>abort</code>	=> Abort creation of "vnic"
<code>done</code>	=> Finish operating on "vnic"
<code>get [prop]</code>	=> Get value for property [prop]. ("help properties" for valid properties.) If [prop] is not specified, returns values for all properties.
<code>set [prop]</code>	=> Set property [prop] to [value]. ("help properties" for valid properties.) For properties taking list values, [value] should be a comma-separated list of values.
<code>available</code>	=> Get values that can be assigned to the links parameter when creating a network component.

ネットワークコンポーネントの作成時に `links` パラメータに割り当てできる値を表示するには、`available` コマンドを使用します。次の例は、CLI コマンド `available` の出力を示しています。

```
caji:configuration net datalinks> device
caji:configuration net datalinks device (uncommitted)> available
igb7,igb6

caji:configuration net datalinks> vnic
caji:configuration net datalinks vnic (uncommitted)> available
igb5,igb4,aggr2,aggr1

caji:configuration net datalinks> vlan
caji:configuration net datalinks vlan (uncommitted)> available
igb5,igb4,aggr2,aggr1

caji:configuration net datalinks> aggregation
caji:configuration net datalinks aggregation (uncommitted)> available
igb7,igb6

caji:configuration net interfaces> ip
caji:configuration net interfaces ip (uncommitted)> available
aggr2,aggr1

caji:configuration net interfaces> ipmp
caji:configuration net interfaces ipmp (uncommitted)> available
vnic4,vnic3,igb5,igb4
```

次の例は、device コマンドを使用したデータリンクの作成と、ip コマンドを使用したインタフェースの作成を示しています。

```
caji:configuration net> datalinks
caji:configuration net datalinks> device
caji:configuration net datalinks device (uncommitted)> set links=igb1
      links = igb1 (uncommitted)
caji:configuration net datalinks device (uncommitted)> set label=datalink2
      label = datalink2 (uncommitted)
caji:configuration net datalinks device (uncommitted)> set mtu=9000
      mtu = 9000 (uncommitted)
caji:configuration net datalinks device (uncommitted)> commit
caji:configuration net datalinks> show
Datalinks:

      DATALINK CLASS      LINKS      LABEL
      igb0 device        igb0       datalink1
      igb1 device        igb1       datalink2

caji:configuration net datalinks> cd ..
caji:configuration net> interfaces
caji:configuration net interfaces> ip
caji:configuration net interfaces ip (uncommitted)> set label="caji2"
      label = caji2 (uncommitted)
caji:configuration net interfaces ip (uncommitted)> set links=igb1
      links = igb1 (uncommitted)
caji:configuration net interfaces ip (uncommitted)> set v4addrs=10.0.1.1/8
      v4addrs = 10.0.1.1/8 (uncommitted)
caji:configuration net interfaces ip (uncommitted)> commit
caji:configuration net interfaces> show
Interfaces:

      INTERFACE STATE  CLASS LINKS      ADDR5      LABEL
      igb0 up      ip    igb0      192.168.2.80/22  caji
      igb1 up      ip    igb1      10.0.1.1/8      caji2
```



次の例は、新しい igb1 IP インタフェースを介した 10.0.1.2 経由のデフォルトルートの作成を示しています。

```
caji:configuration net routing> create
caji:configuration net route (uncommitted)> set family=IPv4
      family = IPv4 (uncommitted)
caji:configuration net route (uncommitted)> set destination=0.0.0.0
      destination = 0.0.0.0 (uncommitted)
caji:configuration net route (uncommitted)> set mask=0
      mask = 0 (uncommitted)
caji:configuration net route (uncommitted)> set interface=igb1
      interface = igb1 (uncommitted)
caji:configuration net route (uncommitted)> set gateway=10.0.1.2
      gateway = 10.0.1.2 (uncommitted)
caji:configuration net route (uncommitted)> commit
```



## BUI を使用したネットワーク構成タスク


### ▼ シングルポートインタフェースの作成

1. データリンク  アイコンをクリックします。
2. オプションで、名前を設定し、カスタム MTU のラジオボタンを選択します (テキストボックスに「9000」と入力します)。
3. 「デバイス」リストからデバイスを選択します。
4. 「適用」をクリックします。データリンクが「データリンク」リストに表示されます。
5. インタフェース  アイコンをクリックします。
6. 目的のプロパティを設定し、前に作成されたデータリンクを選択します。
7. 「適用」をクリックします。インタフェースが「インタフェース」リストに表示されます。
8. 実行中のアプライアンスネットワーク構成はまだ変更されていません。インタフェースの構成を完了したら、いちばん上にある「適用」をクリックして、この構成をコミットします。

### ▼ インタフェースの変更



1. データリンクまたはインタフェースのどちらかにある編集アイコンをクリックします。
2. 設定を目的の値に変更します。
3. ダイアログにある「適用」をクリックします。
4. ページのいちばん上にある「適用」をクリックして、この構成をコミットします。

### ▼ シングルポートインタフェースの作成、ドラッグ & ドロップ

1. マウスをデバイスの上に移動し、ドラッグ & ドロップアイコン () をクリックします。

2. それを「データリンク」リストにドラッグして離します。
3. オプションで、名前とジャンボ MTU を設定します。
4. 「適用」をクリックします。
5. 次に、データリンクを「インタフェース」リストにドラッグします。
6. 目的のプロパティを設定し、「適用」をクリックします。
7. 画面のいちばん上にある「適用」をクリックして、この構成をコミットします。


## ▼ LACP 集合リンクインタフェースの作成

1. データリンク  アイコンをクリックします。
2. オプションで、データリンク名を設定します。
3. LACP アグリゲーションを選択します。
4. 「デバイス」リストから 2 つ以上のデバイスを選択し、「適用」をクリックします。
5. インタフェース  アイコンをクリックします。
6. 目的のプロパティを設定し、「データリンク」リストから集合リンクを選択して、「適用」をクリックします。
7. いちばん上にある「適用」をクリックして、この構成をコミットします。


## ▼ プロブベースのリンク状態障害検出を使用した IPMP グループの作成

IPMP テストのアドレスと同じサブネット上に、ICMP エコーリクエストに応答するように構成されている (クラスピア以外の) システムがない場合、プロブベースの障害検出を使用しないでください。


1. IPMP グループのコンポーネントとして使用される 1 つ以上の「基になる」IP インタフェースを作成します。各インタフェースには、プロブのソースとして使用される IP アドレスが必要です (上のシングルポートインタフェースを作成するための個別のタスクを参照)。

2. インタフェース  アイコンをクリックします。
3. オプションで、インタフェースの名前を変更します。
4. 「IP マルチパスグループ」チェックボックスにチェックマークを付けます。
5. 「IPv4 プロトコルを使用」または「IPv6 プロトコルを使用」、あるいはその両方をクリックし、IPMP インタフェースの IP アドレスを指定します。
6. 「インタフェース」リストから、最初の手順で作成されたインタフェースを選択します。
7. 選択した各インタフェースを、必要に応じて「アクティブ」または「スタンバイ」のどちらかに設定します。
8. 「適用」をクリックします。


## ▼ リンク状態のみの障害検出を使用した IPMP グループの作成

1. IPMP グループのコンポーネントとして使用される、IP アドレス 0.0.0.0/8 を持つ 1 つ以上の「基になる」IP インタフェースを作成します (上のシングルポートインタフェースを作成するための個別のタスクを参照)。
2. インタフェース  アイコンをクリックします。
3. オプションで、インタフェースの名前を変更します。
4. 「IP マルチパスグループ」チェックボックスにチェックマークを付けます。
5. 「IPv4 プロトコルを使用」または「IPv6 プロトコルを使用」、あるいはその両方をクリックし、IPMP インタフェースの IP アドレスを指定します。
6. 「インタフェース」リストから、最初の手順で作成されたインタフェースを選択します。
7. 選択した各インタフェースを、必要に応じて「アクティブ」または「スタンバイ」のどちらかに設定します。
8. 「適用」をクリックします。



## ▼ LACP アグリゲーションの拡張

1. マウスを「デバイス」リスト内のデバイスの上に移動します。
2.  アイコンをクリックし、そのデバイスをアグリゲーションのデータリンクにドラッグして離します。
3. ページのいちばん上にある「適用」をクリックして、この構成をコミットします。

## ▼ IPMP グループの拡張

1. マウスを「インタフェース」リスト内のインタフェースの上に移動します。
2.  アイコンをクリックし、そのデバイスを IPMP インタフェースにドラッグして離します。
3. ページのいちばん上にある「適用」をクリックして、この構成をコミットします。





## ▼ InfiniBand パーティションのデータリンクとインタフェースの作成

1. データリンク  アイコンをクリックします。
2. オプションで、名前を設定します。
3. 「IB パーティション」チェックボックスにチェックマークを付けます
4. 「パーティションデバイス」リストからデバイスを選択します。
5. 「適用」をクリックします。新しいパーティションデータリンクが「データリンク」リストに表示されます。
6. インタフェース  アイコンをクリックします。
7. 目的のプロパティを設定し、前に作成されたデータリンクを選択します。
8. 「適用」をクリックします。インタフェースが「インタフェース」リストに表示されます。

9. 実行中のアプライアンスネットワーク構成はまだ変更されていません。インタフェースの構成を完了したら、いちばん上にある「適用」をクリックして、この構成をコミットします。

## ▼ クラスタ化されたコントローラでの VLAN ID なしの VNIC の作成



この例は、ネットワークポートの半分がスタンバイ状態のアクティブ/アクティブ構成の場合です。このタスクでは、デバイスデータリンク上に IP インタフェースを作成して、ヘッドに割り当てます。VNIC は同じデータリンクの最上位に構築され、IP インタフェースは VNIC の最上位に構成されて、他方のヘッドに割り当てられます。特定のデータリンク上で VNIC を複数ではなく 1 つ構成すると、ピークパフォーマンスが確保されます。トラフィックは、一方のヘッド上の基盤となるアクティブレポート、および他方のヘッド上の基盤となるスタンバイレポートに関連付けられているケーブル上を転送されます。このため、アイドル状態にあるほかのスタンバイポートを VNIC で使用できます。

1. クラスタが AKCS\_CLUSTERED 状態にある場合、データリンク  アイコンをクリックします。
2. オプションで、名前と MTU を設定します。
3. 「デバイス」リストからデバイスを選択して、「適用」をクリックします。データリンクが「データリンク」リストに表示されます。
4. インタフェース  アイコンをクリックします。
5. 目的のプロパティを設定し、前に作成されたデータリンクを選択して、「適用」をクリックします。インタフェースが「インタフェース」リストに表示されます。
6. データリンク  アイコンをクリックします。
7. 「VNIC」チェックボックスにチェックマークを付け、オプションで名前と MTU (手順 2 の値以下) を設定して、「適用」をクリックします。新しい VNIC データリンクが「データリンク」リストに表示されます。
8. インタフェース  アイコンをクリックします。
9. 目的のプロパティを設定し、前に作成された VNIC データリンクを選択して、「適用」をクリックします。インタフェースが「インタフェース」リストに表示されます。

10. 実行中のアプライアンスネットワーク構成はまだ変更されていません。インタフェースの構成を完了したら、いちばん上にある「適用」をクリックして、この構成をコミットします。
11. 「クラスタ」タブをクリックします。新しく作成された 2 つのインタフェースが、デフォルトの所有者で「リソース」セクションに表示されます。
12. 「所有者」プルダウンリストを使用して 2 つのインタフェースの 1 つを他方のヘッドに割り当てて、「適用」をクリックします。

## ▼ クラスタ化されたコントローラでの同じ VLAN ID の VNIC の作成

この例は、ネットワークポートの半分がスタンバイ状態のアクティブ/アクティブ構成の場合です。このタスクでは、同じデバイスデータリンクの最上位に、同一の VLAN ID を持つ 2 つの VNIC を作成します。各 VNIC はインタフェースを使って構成され、各インタフェースは異なるヘッドに割り当てられます。トラフィックは、一方のヘッド上の基盤となるアクティブレポート、および他方のヘッド上の基盤となるスタンバイポートに関連付けられているケーブル上を転送されます。このため、アイドル状態にあるほかのスタンバイポートを VNIC で使用できます。

1. クラスタが AKCS\_CLUSTERED 状態にある場合、データリンク  アイコンをクリックします。
2. 「VNIC」チェックボックスにチェックマークを付け、オプションで名前と MTU を設定して、VLAN ID を設定してから、「デバイス」リストからデバイスを選択して、「適用」をクリックします。新しい VNIC データリンクが「データリンク」リストに表示されます。
3. インタフェース  アイコンをクリックします。
4. 目的のプロパティを設定し、前に作成された VNIC データリンクを選択して、「適用」をクリックします。インタフェースが「インタフェース」リストに表示されます。
5. 手順 1 と 2 の説明に従って、同じデバイスおよび VLAN ID で別の VNIC を作成してから、手順 3 と 4 の説明に従ってそのためのインタフェースを作成します。
6. 実行中のアプライアンスネットワーク構成はまだ変更されていません。インタフェースの構成を完了したら、いちばん上にある「適用」をクリックして、この構成をコミットします。
7. 「クラスタ」タブをクリックします。新しく作成された 2 つのインタフェースが、デフォルトの所有者で「リソース」セクションに表示されます。

8. 「所有者」プルダウンリストを使用して 2 つのインタフェースの 1 つを他方のヘッドに割り当てて、「適用」をクリックします。

### ▼ 静的ルートの追加

1. 「構成」->「ネットワーク」->「ルーティング」に移動します
2. 追加アイコンをクリックします。
3. 先に説明したように、プロパティを入力します。
4. 「追加」をクリックします。新しいルートがテーブルに表示されます。

### ▼ 静的ルートの削除

1. 「構成」->「ネットワーク」->「ルーティング」に移動します
2. マウスをルートエントリの上に移動し、右側にあるごみ箱アイコンをクリックします。

## CLI を使用したネットワーク構成タスク

### ▼ 静的ルートの追加

1. `configuration net routing` に移動します。
2. 「create」と入力します。
3. 「show」と入力して必要なプロパティを一覧表示し、それぞれ「set」と入力します。
4. 「commit」と入力します。

### ▼ 静的ルートの削除

1. `configuration net routing` に移動します。

2. 「show」と入力してルートとルート名 (たとえば、route-002) を一覧表示します。
3. 「destroy ルート名」と入力します。

## ▼ マルチホーミングプロパティの「厳しい」への変更

1. configuration net routing に移動します
2. 「set multihoming=strict」と入力します
3. 「commit」と入力します



## ストレージ構成

---

ストレージは、基になるデータ冗長性によって特徴付けされるプール内に構成され、すべてのファイルシステムおよび LUN でシェアされるスペースを指定します。ストレージプールが個々のファイルシステムまたは LUN にどのように関連しているかの詳細は、[294 ページの「シェアのセクション」](#)を参照してください。

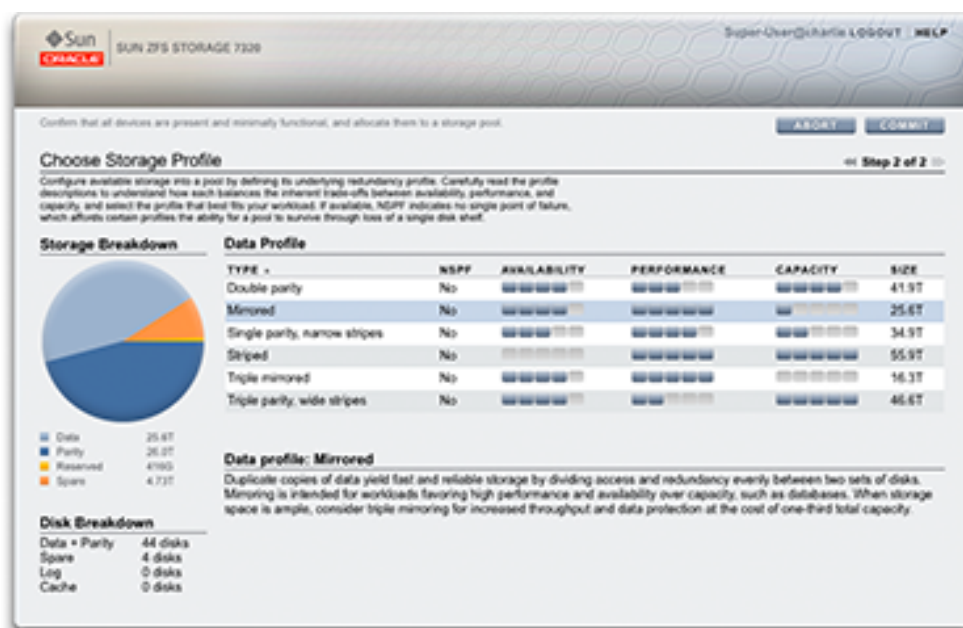
各ノードに任意の数のプールを含めることができ、各プールには、クラスタ内で独立して所有権を割り当てることができます。任意の数のプールがサポートされていますが、同じクラスタヘッドによって所有される同じ冗長性特性を持つ複数のプールを作成することはお勧めできません。それを行うと、パフォーマンスの低下、最善でないリソース割り当て、ストレージの人為的なパーティション分割、管理の一層の複雑さなどがもたらされます。同じホスト上での複数のプールの構成は、たとえば、ミラー化されたプールと RAID-Z プールのように、まったく異なる冗長性またはパフォーマンス特性が望ましい場合にのみ推奨されます。ログおよびキャッシュデバイスへのアクセスをシェア単位に制御できる機能を持つ、推奨される動作モードは 1 つのプールです。


プールは、新しいプールを構成したり、または既存のプールをインポートしたりすることによって作成できます。既存のプールのインポートは、以前に Sun Storage 7000 アプライアンス上で構成されたプールをインポートするためにのみ使用され、偶発的な再構成、ヘッドノード間でのプールの移動、致命的なヘッド障害などの場合に有効です。

プールに raw ストレージを割り当てる場合、特にシェアまたは LUN に書き込むときは、プールを完全にいっぱいになるとパフォーマンスが大幅に低下することに注意してください。これらの影響は通常、プールが 80% を超えていっぱいになると顕著になり、さらにプールが 90% を超えていっぱいになると重大になります。そのため、約 20% 過剰にプロビジョニングすることにより最適な結果が得られます。[294 ページの「シェアの UI」](#)を使用すると、現在どれだけの領域が使用されているかを判別できます。

## ストレージ構成プロファイル

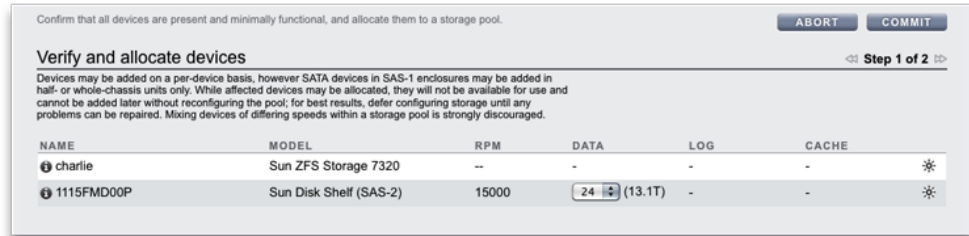
図 5-1 ストレージ構成プロファイル



このアクションによってストレージプールが構成されます。BUI でこれを行うには、プールのリストの横にある  ボタンをクリックします。ここに、新しいプールの名前を入力するように求められます。CLI でこれを行うには、config コマンドを使用します。このコマンドは、プールの名前を引数として受け取ります。

タスクが開始されると、ストレージ構成は、検証と構成という 2 つの異なるフェーズに入ります。

図 5-2 デバイスの検証と割り当て



## ストレージ構成の規則とガイドライン

パフォーマンスを最適化するため、次の点に注意してください。

規則 1 -- ヘッドノードまたは JBOD 内に含まれるすべての「データ」ディスクは、回転数 (メディアの回転速度) が同じである必要があります。ZFSSA ソフトウェアは、構成の誤りを検出し、条件に対する障害を生成します。

推奨 1 -- 予期しないパフォーマンスの問題のため、同じプール内で回転数が異なるディスクが混在しないようにします。

推奨事項 2 -- パフォーマンスを最適化するため、同じ SAS ファブリック (HBA 接続) 内で回転数が異なるディスクを JBOD と結合しないようにします。そのように混在させても正しく動作しますが、高速なデバイスのパフォーマンスが低下する可能性があります。

推奨事項 3 -- 容量が異なるデータディスクが含まれるストレージプールを構成するときに、場合によっては ZFS が容量がもっとも小さいディスクのサイズをストレージプール内の一部またはすべてのディスクに対して使用するため、想定される全体の容量が小さくなる場合があります。使用されるサイズは、デバイスのストレージプロファイル、配置、および組み合わせに依存します。同じプール内で容量が異なるディスクを混在しないようにします。

## ストレージの検証

検証により、すべてのストレージが接続され、機能していることが保証されます。ストレージデバイスを割り当てる前に、すべてのストレージデバイスが接続され、機能している必要があります。存在しないか障害の発生したデバイスを含むプールを割り当てる場合、存在しないか障害の発生したディスクをあとで追加することはできません。

接続ストレージのないシステムでは、使用可能なすべてのドライブがデフォルトで割り当てられます。拡張可能なシステムでは、ディスクシェルフがヘッドノードとともにリストに表示さ

れ、各ディスクシェルフ内で割り当てを制御できます。この動作は、ヘッドノードまたはディスクシェルフのモデルによって異なることがあります。

次を選択できます。

- デバイスサイズ - データデバイスを論理サイズでフィルタ処理します。デフォルトでは、Any で使用可能なすべてのデータデバイスが表示されます。
- 「データデバイス」 - 使用可能なすべてのデータサービス、または選択したデバイスサイズの使用可能な数を表示します。

デフォルトで割り当てられるディスクの数は、次に基づいて変化します。

- 使用可能な最大数 - 接続されるストレージに同じサイズと回転数のデバイスのみが含まれている場合、または複数のサイズから 1 つのサイズが選択されている場合
- 「なし」 - 接続されるストレージに混在した回転数が含まれている場合。

注: 一貫したパフォーマンス特性を提供するため、プールには同じサイズと回転数のデバイスのみを含めることを強くお勧めします。

## SAS-2 システムでのストレージの割り当て

シャーシ全体のドライブを個別に割り当てることができますが、JBOD からディスクを割り当てるときは、最適なプール構成になるように注意してください。管理が簡素化され、使用可能な全体の容量の割合が高まるため、一般に、プール数を少なくし、プールあたりのディスク数を多くすることが推奨されます。

システムは必要に応じて任意の増分でストレージを割り当てることができますが、各割り当てにはすべての JBOD に分散するようにして少なくとも 8 台のディスクを含めること、理想的にはもっと多くを含めることをお勧めします。

## データプロファイル構成

検証が完了したら、次の手順では、設定での RAS とパフォーマンスの目標が反映されたストレージプロファイルを選択します。提供される一連の可能性のあるプロファイルは、使用可能なストレージによって異なります。次の表は、可能性のあるすべてのプロファイルとその説明を示しています。

表 5-1 データプロファイル構成

データプロファイル	説明
デュアルパリティオプション	
トリプルミラー化	<p>データが 3 つにミラー化され、非常に信頼性の高い高速なシステム (たとえば重要なデータベース用のストレージ) がもたらされます。この構成は、最大のパフォーマンスと可用性の両方が必要な状況に向いています。双方向ミラー化と比較すると、3 方向ミラー化では、格納ブロックあたりの IOPS の増加と障害に対する保護レベルの向上を実現します。注: 拡張ストレージのないコントローラは、トリプルミラー化で構成すべきではありません。</p>
ダブルパリティ RAID	
ミラー化	<p>各ストライプに 2 つのパリティディスクが含まれている RAID。トリプルミラー化に比べて、高可用性がもたらされ、2 つのディスクで障害が発生してもデータは使用可能のままです。ダブルパリティ RAID は、ミラー化オプションよりも容量が増加するオプションであり、高スループットで順次アクセスのワークロード (バックアップなど) の場合やランダム読み取り性の低いコンポーネントに大量のデータを格納する場合に向いています。</p>
シングルパリティオプション	
ミラー化	<p>データがミラー化されるため、容量は半分減少しますが、高信頼性で高パフォーマンスのシステムがもたらされます。領域には十分な余裕があると考えられるがパフォーマンスに余裕がない場合 (データベースストレージなど) に推奨されます。</p>
シングルパリティ RAID、ナローストライプ	
ミラー化	<p>各ストライプに 3 つのデータディスクと 1 つのパリティディスクが含まれている RAID。シングルパリティによる保護で十分な場合は、シングルパリティ RAID のほうが単純なミラー化よりも得られる容量が多くなるオプションです。ミラー化オプションよりも低くなるランダム読み取り性能と容量増加とのバランスを取る必要があります。シングルパリティ RAID は、適度なランダム読み取りコンポーネントを使用する重要なアプリケーションに向いていると考えることができます。純粋なストリーミングワークロードの場合は、容量とスループットが高くなるダブルパリティ RAID オプションを選択してください。</p>
その他	
ストライプ化	<p>データは、ディスクにまたがってストライプ化され、冗長性はありません。パフォーマンスと容量が最大化されますが、1 つのディスクに障害が発生するとデータが失われます。この構成は推奨されません。純粋な</p>

データプロファイル	説明
トリプルパリティ RAID、ワイドストライプ	<p>ストリーミングワークロードの場合は、ダブルパリティ RAID の使用を検討してください。</p> <p>各ストライプに 3 つのパリティ用ディスクが含まれている RAID。このオプションは、ストライプデータを別にするともっとも容量が高くなります。ワイドストライプと低いランダム入出力パフォーマンスのため、1 つ以上のドライブ障害のあとでデータを再同期化する場合は、かかる時間が大幅に長くなることがあります。その他の RAID 構成と同様に、キャッシュが存在するため、読み取りパフォーマンスの影響を軽減できます。この構成は一般には推奨されません。</p>

拡張可能なシステムでは、一部のプロファイルを「NSPF」オプションで使用できます。これは「No Single Point of Failure」の略でシングルポイント障害がないことを表し、深刻な JBOD 障害によってデータ損失が発生しないように、データがミラーまたは RAID ストライプで構造化されていることを示します。システムではすでに、ほぼすべてのコンポーネントにわたって冗長性が構成されていることに注意してください。各 JBOD は、冗長パス、冗長コントローラ、冗長電源装置およびファンを備えています。NSPF による保護の対象となる唯一の障害は、ディスクバックプレーン障害 (ほぼ受動的なコンポーネント)、または全体的な管理ミス (両方のパスの 1 つの JBOD への切り離し) です。一般に、NSPF を採用すると、ストライプの幅に関する要件がより厳格になるため、容量は少なくなります。

ログデバイスは、ストライプ化またはミラー化されたプロファイルのみを使用して構成できません。ログデバイスはノード障害が発生して、ミラー化されていないログによってデータが失われる場合にのみ使用されるため、デバイスに障害が発生することと、そのあとにノードがただちにリブートすることの両方が必要です。これはほとんどありそうにない状況ですが、ログデバイスをミラー化すると、2 つの同時デバイス障害とノード障害が非常に短い時間帯に発生することが必要になるため、これを事実上不可能にできます。

注意: 異なるサイズのログデバイスが別のシャーシにある場合は、ストライプ化されたログプロファイルのみを作成できます。

ホットスペアは合計プールサイズの割合として割り当てられ、選択されたプロファイルには関係ありません (ホットスペアをサポートしていないストライプ化を除く)。ホットスペアはストレージの構成手順ごとに割り当てられるため、小さい単位でストレージを追加するより、ストレージを全体として構成する方がはるかに効率的です。

クラスタでは、キャッシュデバイスは、ストレージプールがインポートされているノードのみから使用できます。クラスタでは、両方のノードでキャッシュデバイスを構成して、同じプールの一部にできます。これを行うには、パッシブノード上のプールを引き継いだあと、ストレージを追加し、キャッシュデバイスを選択します。これには、常に、構成されているグローバル

キャッシュデバイスの半分を保持するという効果があります。キャッシュデバイス上のデータはフェイルオーバーで失われますが、新しいノード上では新しいキャッシュデバイスを使用できます。

注: 以前のソフトウェアバージョンでは、ワイドストライプによるダブルパリティがサポートされていました。このオプションは、信頼性を大幅に向上させる、ワイドストライプによるトリプルパリティに置き換えられました。以前のソフトウェアバージョンの下でワイドストライプによるダブルパリティとして構成されたプールも引き続きサポートされますが、新しく構成または再構成されるプールではそのオプションを選択できません。

## 既存のストレージプールのインポート

この操作によって、既存のストレージプールのほか、誤って構成解除された任意のプールをインポートできます。この操作は、出荷時設定へのリセットまたはサービス操作のあとにユーザーデータを回復するために使用できます。プールをインポートするには、接続されているすべてのストレージデバイスを繰り返し処理し、既存の状態をすべて検出することが必要です。これには膨大な時間がかかる場合があり、この間、ほかのストレージ構成操作は実行できません。BUI でプールをインポートするには、ストレージ構成画面にある「インポート」ボタンをクリックします。CLI でプールをインポートするには、「import」コマンドを使用します。

検出フェーズが完了すると、いくつかの識別特性を含む、使用可能なプールのリストが表示されます。ストレージが破棄されているか、または不完全な場合は、そのプールをインポートできません。ストレージ構成とは異なり、プール名は開始時ではなく、プールを選択するときに指定されます。デフォルトでは、以前のプール名が使用されますが、BUI で名前をクリックしたり、CLI で「name」プロパティを設定したりすることによってプール名を変更できます。

## ストレージの追加

このアクションは、既存のプールにストレージを追加するために使用します。この検証手順は、初期構成中の検証手順と同じです。ストレージは、最初にプールを構成するために使用したのと同じプロファイルを使用して追加する必要があります。現在のプロファイルでシステムを構成するためにストレージが不足している場合は、一部の属性が犠牲になることがあります。たとえば、ダブルパリティ RAID-Z NSPF 構成に 1 つの JBOD を追加すると、NSPF 特性を保持することが不可能になります。ただし、引き続き JBOD を追加し、JBOD 内に RAID ストライプを作成することにより、その過程で NSPF を犠牲にできます。

## ストレージの構成解除

この操作では、アクティブなファイルシステムと LUN をすべて削除し、ストレージプールを構成解除することにより、raw ストレージを将来のストレージ構成に使用できるようにします。この処理は、それ以降に raw ストレージがアクティブなストレージプールの一部として使用されていないかぎり、構成解除されたストレージプールをインポートすることによって取り消すことができます。


## ストレージプールのスクラブ

この操作によって、すべてのコンテンツでエラーがないかどうか検証する、ストレージプールのスクラブプロセスが開始されます。何らかの回復不可能なエラーが見つかった場合は、スクラブ操作または通常動作のどちらかを使用して、影響を受けているファイルが BUI に表示されます。また、必要に応じてスクラブを停止することもできます。

## BUI を使用したストレージの構成

### ▼ ストレージプールの構成

このタスクに到達する場合としては、アプライアンスの初期構成中か、または「構成」->「ストレージ」画面での構成時の 2 つがあります。

1. ストレージプールの一覧の上にある  ボタンをクリックします
2. ストレージプールの名前を入力します
3. 「ストレージの割り当ておよび検証」画面で、ストレージプールの JBOD 割り当てを構成します。JBOD の割り当ては、なし、半分、または全体のいずれかです。JBOD が検出されない場合は、JBOD の配線や電源を確認してください。
4. 「コミット」をクリックします。
5. 「追加されたストレージを構成」画面で、目的のデータプロファイルを選択します。ビジネスニーズにとって最適な構成が見つかるように、各プロファイルが可用性、パフォーマンス、および容量の点から評価されます。
6. 「コミット」をクリックします。



## ▼ 既存のプールへのキャッシュデバイスの追加

1. 使用可能な最初のスロットに新しい Readzilla または Logzilla デバイスをインストールします。スロットの位置については、『[Oracle ZFS Storage Appliance インストールガイド](#)』の「[概要](#)」を参照してください。
2. BUI で「構成」>「ストレージ」と移動します。
3. 「使用可能なプール」リストから、デバイスの追加先のプールを選択します。プールがオンラインであることを確認します。
4. 「追加」ボタンをクリックして、デバイスをプールに追加します。
5. プールに追加するデバイスを選択して、「コミット」をクリックします。
6. ログプロファイル (該当する場合) を選択して、「コミット」をクリックします。

## CLI を使用したストレージの構成

### ▼ 既存のプールへのキャッシュデバイスの追加

1. 使用可能な最初のスロットに新しい Readzilla または Logzilla デバイスをインストールします。スロットの位置については、『[Oracle ZFS Storage Appliance インストールガイド](#)』の「[概要](#)」を参照してください。
2. コマンド行で、次のように入力します。
3. 

```
: poc:> configuration storage
```
4. デバイスの追加先のプールを指定します。
5. 

```
: poc:configuration storage (pool_2)> set pool=pool_2
```
6. 

```
: pool = pool_2
```
7. 

```
: poc:configuration storage (pool_2)> add
```
8. :デバイスが正しくインストールされていることを確認するよう促すメッセージが表示されます。デバイスのタイプおよび速度は混在させないことを強くお勧めします。

9. プールのデバイス情報を表示します。
10. : poc:configuration storage (pool\_2) verify> show
11. : ID STATUS ALLOCATION DATA LOG CACHE RPM
12. : 0 ok custom 0 0 0/4 1.86T
13. : 1 ok custom 0 0/2 34G 0 15000
14. : 2 ok custom 0 0/2 34G 0 15000
15. 使用するディスクシェルフと、Logzilla または Readzilla の数を指定します。次の例では、1-log=1 により 1 番目のディスクシェルフから 1 つの Logzilla を割り当てています。
16. : poc:configuration storage (pool\_2) verify> set 1-log=1
17. : 1-log = 1
18. :注: 「1-log=2」という値では、1 番目のディスクシェルフから 2 つの Logzilla が割り当てられます。
19. :この例では、1 番目のディスクシェルフから 1 つの Readzilla を割り当てています。
20. : poc:configuration storage (pool\_2) verify> set 1-cache=1
21. : 1-cache = 1
22. 「done」と入力します。
23. : poc:configuration storage (pool\_2) verify> done
24. : 注: 奇数番号の Logzilla デバイスをプールに追加する場合、またはプールにまだプロファイルがない場合は、「set log\_profile=log\_mirror」を入力してログプロファイルを設定してください。
25. 「show」と入力してプロファイルを表示します。
26. : poc:configuration storage (pool\_2) config> show
27. :
28. : PROFILE CAPCTY NSPF DESCRIPTION

29. : `log_profile = log_stripe 17G no Striped log`
30. 「done」と入力してタスクを完了します。
31. : `poc:configuration storage (pool_2) config> done`
32. : `poc:configuration storage (pool_2)>`



## Storage Area Network の構成

---

SAN 構成ページを使用すると、アプライアンスを SAN (Storage Area Network) に接続できます。SAN は、次の 3 つの基本コンポーネントで構成されています。

- ネットワーク上のストレージにアクセスするクライアント
- ネットワーク上のストレージを提供するストレージアプライアンス
- クライアントをストレージにリンクするためのネットワーク

これらの 3 つのコンポーネントは、ネットワーク上で使用されているプロトコルには関係なく同じです。場合によっては、ネットワークがイニシエータとターゲットの間のケーブルであることもあります。ほとんどの場合は、何らかの種類の切り換えが使用されます。

### SAN のターゲットとイニシエータ

ターゲットとイニシエータは、プロトコルによって構成されます。詳細は、特定のプロトコル ([114 ページの「SAN ファイバチャネル」](#)、[208 ページの「iSCSI」](#)、または [132 ページの「SRP」](#)) に関するドキュメントを参照してください。

### SAN のターゲットグループとイニシエータグループ

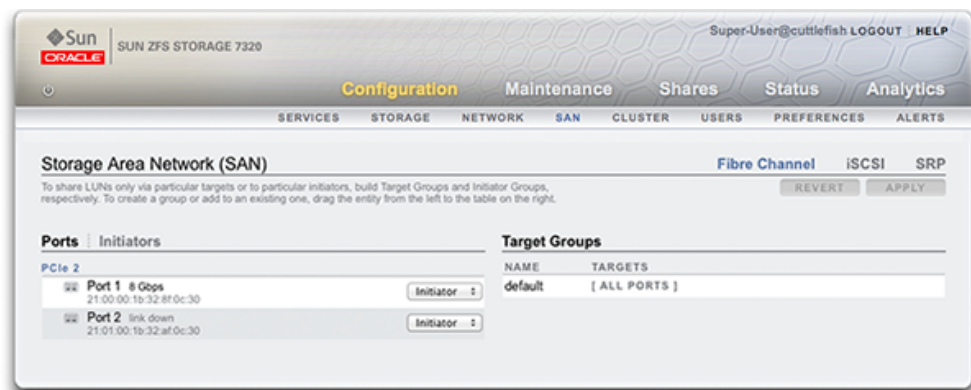
ターゲットグループとイニシエータグループによって、LUN に関連付けることができるターゲットとイニシエータのセットが定義されます。ターゲットグループに関連付けられた LUN は、そのグループ内のターゲットからのみ表示できます。LUN がターゲットグループに明示的に関連付けられていない場合は、デフォルトのターゲットグループ内にあり、プロトコルには関係なく、すべてのターゲットからアクセスできます。同様に、LUN を表示できるのは、それが所属するグループ内のイニシエータだけです。LUN がイニシエータグループに明示的に関連付けられていない場合は、デフォルトのイニシエータグループ内にあり、すべてのイニシエータからアクセスできます。デフォルトのイニシエータグループの使用は評価の目的には有効な場合がありますが、不要なイニシエータまたは競合するイニシエータに LUN が公開される可能性があるため、その使用はお勧めできません。

イニシエータが複数のグループに所属する場合に起こり得る LUN の競合を回避するため、グループを LUN に関連付ける前に、すべてのグループ内のイニシエータを構成してください。

## BUI を使用した SAN の構成

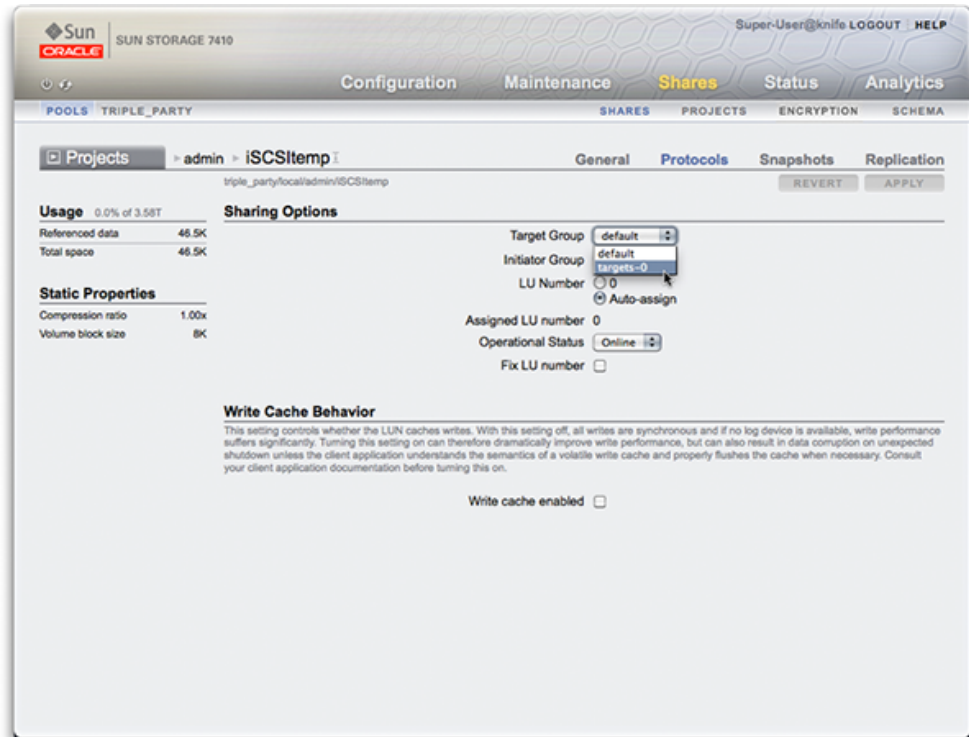
ターゲットを構成するには、「構成」>「SAN BUI」ページに移動し、「ファイバチャネル」、「iSCSI」、および「SRP」を使用してナビゲートし、「ポート」、「イニシエータ」、および「ターゲットグループ」コントロールを構成します。

図 6-1 「SAN BUI」ページ



LUN を関連付けるには、「シェア」>「シェア」>「プロトコル」ページに移動し、「ターゲットグループ」および「イニシエータグループ」コントロールを構成します。

図 6-2 LUN を関連付ける



## CLI を使用した SAN の構成

CLI の `configuration san` コンテキストを使用して、プロトコルタイプごとにターゲットとイニシエータに対する操作を行います。次に、`shares` CLI コンテキストを使用して LUN を作成し、それらをターゲットグループとイニシエータグループに関連付けます。

## SAN の用語

アプライアンスを SAN 上で動作するように構成するには、次のいくつかの基本的な SAN の用語を理解しておく必要があります。

表 6-1 SAN の用語

用語	説明
SCSI ターゲット	SCSI ターゲットは、イニシエータからの SCSI コマンドと入出力リクエストを処理するためのサービスを提供するストレージシステムエンドポイントです。SCSI ターゲットはストレージシステムの管理者によって作成され、一意のアドレス指定方法によって識別されます。構成済みの SCSI ターゲットは、0 個以上の論理ユニットで構成されています。
SCSI イニシエータ	SCSI イニシエータは、SCSI コマンドと入出力リクエストを送信することによって SCSI セッションを開始できるアプリケーションまたは運用システムエンドポイントです。SCSI イニシエータもまた、一意のアドレス指定方法によって識別されます (「SCSI ターゲット」を参照)。
論理ユニット	論理ユニットは、ストレージシステム内のコンポーネントを説明するために使用される用語です。一意に番号付けされ、LUN (Logical Unit Number) と呼ばれる番号が作成されます。高度な構成が可能なストレージシステムには、多数の LUN を含めることができます。これらの LUN を 1 つ以上の SCSI ターゲットに関連付けることにより、1 つ以上の SCSI イニシエータからアクセスできるデバイスである一意の SCSI デバイスが形成されます。
iSCSI	インターネット SCSI。IP ネットワークを経由して SCSI ベースのストレージをシェアするためのプロトコル。
iSER	RDMA のための iSCSI 拡張機能。RDMA サービスを提供するネットワーク (つまり、InfiniBand) を経由して iSCSI プロトコルをマップするプロトコル。iSER プロトコルは、正しく構成された IB ハードウェアの存在に基づいて、iSCSI サブシステムによって透過的に選択されます。CLI および BUI では、iSER に対応したすべてのコンポーネント (ターゲットとイニシエータ) が iSCSI コンポーネントとして管理されます。
FC	ファイバチャネル。光ファイバケーブル、FC スイッチ、および HBA から成る SAN (Storage Area Network) を経由して SCSI ベースのストレージをシェアするためのプロトコル。
SRP	SCSI RDMA プロトコル。RDMA サービスを提供するネットワーク (つまり、InfiniBand) を経由して SCSI ベースのストレージをシェアするためのプロトコル。
IQN	iSCSI 修飾名。iSCSI ネットワーク内のデバイスの一意の識別子。iSCSI では、IQN に対して iqn.date.authority:uniqueid の形式が使用されます。たと



用語	説明
	<p>例えば、アプライアンスは、iSCSI ターゲットの 1 つを識別するために IQN: iqn.1986-03.com.sun:02:c7824a5b-f3ea-6038-c79d-ca443337d92c を使用する可能性があります。この名前は、これが 1986 年 3 月に登録された、ある企業によって構築された iSCSI デバイスであることを示します。名前の authority は、単にその企業の DNS 名を逆にしただけのもので (この場合は、「com.sun」)。それ以降のすべては、Sun がそのターゲットを識別するために使用している一意の ID です。</p>
ターゲットポータル	<p>iSCSI プロトコルを使用している場合、ターゲットポータルは、イニシエータがターゲットに接続するときに使用できる IP アドレスと TCP ポート番号の一意的組み合わせを指します。</p>
ターゲットポータルグループ	<p>iSCSI プロトコルを使用している場合、ターゲットポータルグループは、ターゲットポータルのコレクションです。ターゲットポータルグループは透過的に管理されません。各ネットワークインタフェースには、そのインタフェースのアクティブなアドレスを含む、対応するターゲットポータルグループがあります。ターゲットをインタフェースにバインドすると、その iSCSI ターゲットが、そのインタフェースに関連付けられたポータルグループを使用して通知されます。</p>
CHAP	<p>チャレンジハンドシェイク認証プロトコル。イニシエータに対するターゲット、ターゲットに対するイニシエータ、またはその両方を認証できるセキュリティプロトコル。</p>
RADIUS	<p>ストレージノードに代わって、集中管理されたサーバーを使用して CHAP 認証を実行するためのシステム。</p>
ターゲットグループ	<p>ターゲットのセット。LUN は、1 つの特定のターゲットグループ内のすべてのターゲットにエクスポートされません。</p>
イニシエータグループ	<p>イニシエータのセット。イニシエータグループが LUN に関連付けられている場合は、そのグループからのイニシエータだけがその LUN にアクセスできます。</p>
ターゲット	<p>イニシエータからの SCSI コマンドと入出力リクエストを処理するためのサービスを提供するストレージシステムエンドポイント。ターゲットはストレージシステム管理者によって作成され、一意のアドレス指定方法によって識別されます。構成済みのターゲットは、0 個以上の論理ユニットで構成されています。</p>
イニシエータ	<p>SCSI コマンドと入出力リクエストを送信することによって SCSI セッションを開始できるアプリケーション</p>

用語	説明
	または運用システムエンドポイント。イニシエータもまた、一意のアドレス指定方法によって識別されます。

各 LUN には、ボリュームのエクスポート方法を制御するためのいくつかのプロパティがあります。詳細は、[329 ページの「プロトコル」](#)のセクションを参照してください。

## SAN ファイバチャネル

ファイバチャネル (FC) は、ほぼ SCSI のためのトランスポートとしてのみ使用されるギガビット速度のネットワークテクノロジーです。FC は、アプライアンスでサポートされるいくつかのブロックプロトコルのうちの 1 つです。FC を経由して LUN をシェアするには、アプライアンスにオプションの FC カードが 1 枚以上装着されている必要があります。

### FC ポートターゲットの構成

デフォルトでは、すべての FC ポートがターゲットモードになるように構成されます。アプライアンスを使用してバックアップ用のテープ SAN に接続する場合は、1 つ以上のポートをイニシエータモードで構成する必要があります。ポートをイニシエータモードで構成するには、アプライアンスをリセットする必要があります。複数のポートを、同時にイニシエータモードで構成できます。

各 FC ポートには WWN (World Wide Name) が割り当てられます。また、ほかのブロックプロトコルと同様に、FC ターゲットを [109 ページの「SAN のターゲットグループとイニシエータグループ」](#)にグループ化し、ポートの帯域幅を特定の LUN または LUN のグループ専用で割り当てることができます。FC ポートがターゲットとして構成されたあとは、リモートで検出されたポートを検証したり、確認したりできます。

Sun ZFS Storage Appliance を使用した FC SAN ブートソリューションの詳細は、<http://www.oracle.com/technetwork/articles/servers-storage-admin/fbsanboot-365291.html> (<http://www.oracle.com/technetwork/articles/servers-storage-admin/fbsanboot-365291.html>) にある *Oracle Sun ZFS Storage Appliance を使用した FC SAN ブートの実装に関するホワイトペーパー*を参照してください。

### クラスタ化に関する考慮事項

クラスタでは、イニシエータには各 LUN への 2 つのパス (またはパスのセット) があります。1 つのパス (またはパスのセット) は、その LUN に関連付けられたストレージをイン

ポートしたヘッドになり、もう一方のパス (またはパスのセット) は、そのヘッドのクラスタ化されたピアになります。最初のパス (またはパスのセット) はアクティブであり、2 番目のパス (またはパスのセット) はスタンバイです。テイクオーバーが発生した場合は、アクティブパスが使用不可になり、スタンバイパスが (短時間のあとに) アクティブに移行されたあと、入出力が続行されます。このマルチパスへのアプローチは、非対称論理ユニットアクセス (ALUA) と呼ばれます。これを ALUA 対応イニシエータと組み合わせると、クラスタテイクオーバーをより高レベルのアプリケーションに対して透過的にすることができます。

## FC イニシエータの構成

イニシエータは、それぞれの WWN で識別されます。また、ほかのブロックプロトコルと同様に、イニシエータに対して別名を作成できます。FC イニシエータに対する別名を作成しやすくするために、検出されたポートの WWN から WWN を選択できます。さらに、ほかのブロックプロトコルと同様に、イニシエータをグループに収集できます。LUN が特定のイニシエータグループに関連付けられている場合、その LUN は、そのグループ内のイニシエータからのみ表示できます。ほとんどの FC SAN では、LUN は常に、その LUN が作成されたシステムに対応するイニシエータグループに関連付けられています。

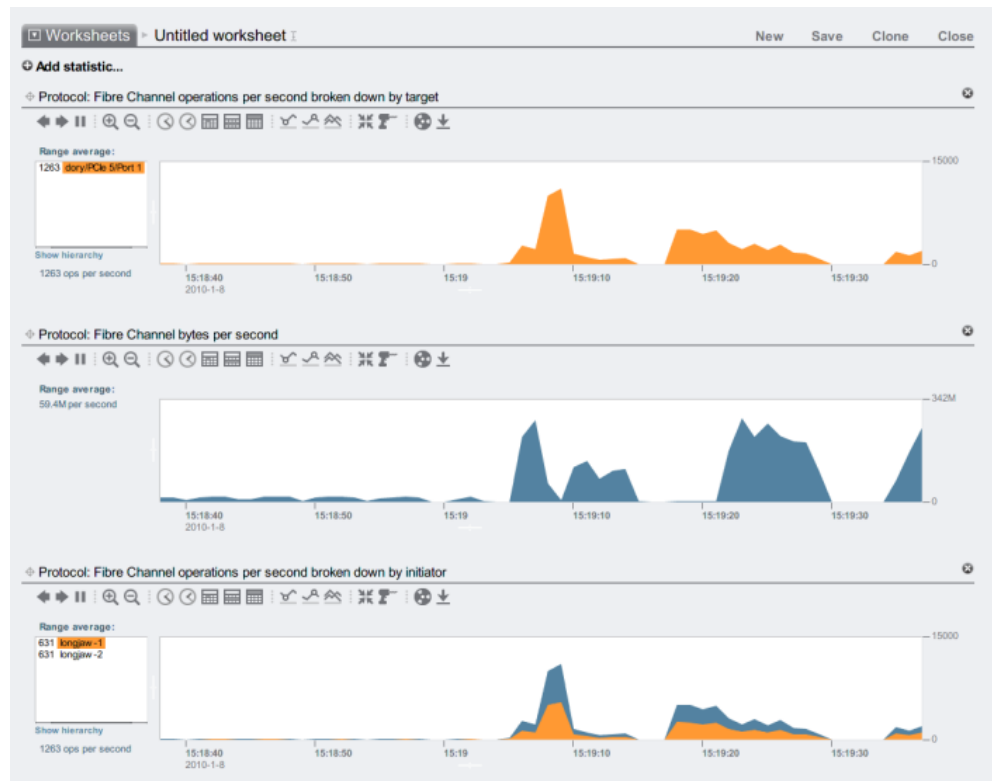
## クラスタ化に関する考慮事項

このアプライアンスは、ALUA に準拠したアレイです。ALUA 環境で FC イニシエータを正しく構成するには ALUA 対応ドライバが必要であり、さらにイニシエータ固有のチューニングが必要になる可能性があります。詳細は、Oracle ZFS Storage Appliance: クライアントマルチパスを設定する方法 (Doc ID 1628999.1) を参照してください。

## パフォーマンスの考慮事項

FC のパフォーマンスは、『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」を通して監視できます。その場合、操作またはスループットをイニシエータ、ターゲット、または LUN ごとに分解できます。

図 6-3 FC のパフォーマンス



また、操作の場合は、オフセット、待機時間、サイズ、および SCSI コマンドごとにも分解できます。これにより、FC 操作の内容だけでなく、その方法や理由についても理解できます。

## FC のトラブルシューティング

### FC の待ち行列のオーバーラン

このアプライアンスは、各ヘッド上の LUN を提供するためにリソースのグローバルなセットを利用するように設計されています。アプライアンス内の FC ポートは多数の並行要求を処理できるため、一般には、クライアント上の待ち行列の深さを制限することは必要ありません。そうであったとしても、これらの待ち行列がオーバーランして、SCSI トランスポートエラーが発生する可能性がわずかに存在します。このような待ち行列のオーバーランは多くの場合、次の状態の 1 つ以上に関連しています。

- フロントエンド上の多重定義されたポート。1 つの FC ポートに関連付けられたホストが多すぎるか、または 1 つの FC ポートを経由してアクセスされる LUN が多すぎる場合
- 縮退したアプライアンス動作モード。アクティブ/アクティブのクラスタ構成になるように設計されている環境でのクラスタテイクオーバーなど

待ち行列のオーバーランの可能性はわずかですが、待ち行列の深さをクライアント単位に積極的に制限すれば、この可能性を完全に解消できます。待ち行列の適切な深さの制限を決定するには、ターゲットポートの数を求め、それにポートあたりの並行コマンドの最大数 (2048) を掛け、さらにその積をプロビジョニングされる LUN の数で割るべきです。縮退した動作モードに対応するには、クラスタピアにまたがる LUN の数を合計して LUN の数を決定するべきですが、ターゲットポートの数には 2 つのクラスタピアのうちの少ない方を取ります。たとえば、アクティブ/アクティブ構成の 7420 デュアルヘッドクラスタで、1 つのヘッドが 2 つの FC ポートと 100 個の LUN を備え、もう一方のヘッドが 4 つの FC ポートと 28 個の LUN を備えている場合は、最悪の事態を考慮した待ち行列の最大の深さを 2 つのポートに 2048 コマンドを掛けて  $100 + 28$  の LUN で割った値、つまり LUN あたり 32 コマンドにするべきです。

待ち行列の最大の深さのチューニングはイニシエータに固有の作業ですが、Solaris では、大域変数 `ssd_max_throttle` を調整することによってこれが実現されます。

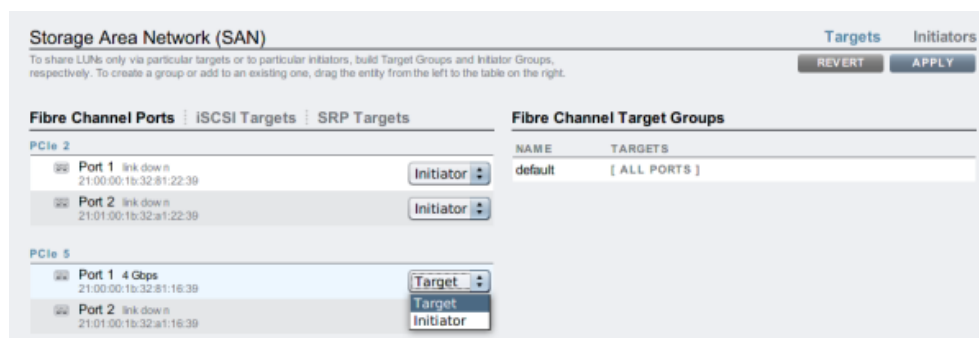
## FC のリンクレベルの問題

光学部品の破損や変形、不安定なケーブル配線などのリンクレベルの問題をトラブルシューティングするには、FC ポートごとのエラー統計情報を調べます。いずれかの数値が 0 から大幅に外れているか、または増加している場合は、リンクレベルの問題が発生していて、リンクレベルの診断の実行が必要なことを示している可能性があります。

## BUI を使用した FC の構成

### FC ポートのモードの変更

FC ポートを使用するには、BUI の「構成」>「SAN」画面で、下のスクリーンショットに示されているドロップダウンメニューを使用して FC ポートを「ターゲット」モードに構成します。このアクションを実行するには、root アクセス権が必要です。クラスタ構成では、各ヘッドノードで個別にポートを「ターゲット」モードに設定することに注意してください。



目的のポートを「ターゲット」に設定したら、「適用」ボタンをクリックします。アプライアンスがただちにリポートすることを通知する確認メッセージが表示されます。リポートすることを確認します。

アプライアンスがブートすると、アクティブな FC ターゲットが アイコンとともに表示され、マウスをその上に移動すると、移動 アイコンが表示されます。


## 検出された FC ポートの表示

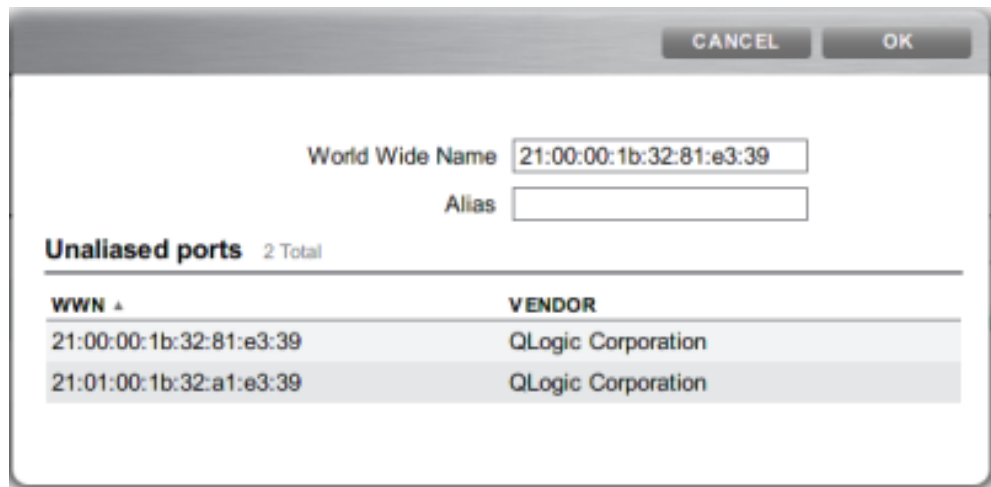
リンクエラーをトラブルシューティングできる「検出されたポート」ダイアログを表示するには、情報 アイコンをクリックします。「検出されたポート」ダイアログで、リスト内の WWN をクリックして関連するリンクエラーを表示します。

図 6-4 検出された FC ポート



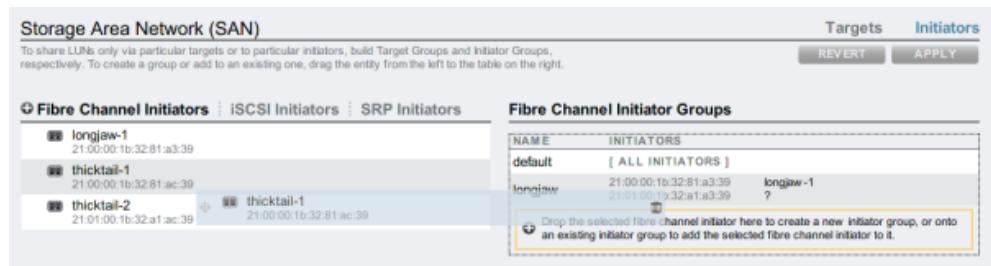
## FC イニシエータグループの作成

イニシエータグループの作成と管理は、「イニシエータ」画面で行います。別名化されていないポートを表示するには、追加  アイコンをクリックします。「別名」フィールドで意味のある別名を追加するには、リスト内の WWN をクリックします。



「イニシエータ」ページで、イニシエータを「FC イニシエータグループ」リストにドラッグして新しいグループを作成するか、または既存のグループに追加します。

図 6-5 「FC イニシエータグループ」リスト



「適用」ボタンをクリックして、新しいイニシエータグループをコミットします。これで、クライアントのイニシエータグループへの排他的アクセス権を持つ LUN を作成できます。

## LUN と FC イニシエータグループとの関連付け


LUN を作成するには、イニシエータグループをロールオーバーし、LUN の追加  アイコンをクリックします。「LUN を作成」ダイアログが表示され、関連付けられたイニシエータグループが選択された状態になります。名前とサイズを設定し、「適用」をクリックして LUN をストレージプールに追加します。



図 6-6 LUN と FC イニシエータグループとの関連付け

## CLI を使用した FC の構成

### FC ポートのモードの変更

```
dory:configuration san fc targets> set targets="wn.2101001B32A11639"
      targets = wwn.2101001B32A11639 (uncommitted)
dory:configuration san fc targets> commit
```

### 検出された FC ポートの表示

```
dory:configuration san fc targets> show
Properties:
      targets = wwn.2100001B32811639,wwn.2101001B32A12239
Targets:
NAME      MODE      WWN              PORT              SPEED
target-000 target    wwn.2100001B32811639  PCIe 5: Port 1    4 Gbit/s
target-001 initiator wwn.2101001B32A11639  PCIe 5: Port 2    0 Gbit/s
target-002 initiator wwn.2100001B32812239  PCIe 2: Port 1    0 Gbit/s
target-003 target    wwn.2101001B32A12239  PCIe 2: Port 2    0 Gbit/s
```

```
dory:configuration san fc targets> select target-000
dory:configuration san fc targets target-000> show
Properties:
    wwn = wwn.2100001B32811639
    port = PCIe 5: Port 1
    mode = target
    speed = 4 Gbit/s
    discovered_ports = 6
    link_failure_count = 0
    loss_of_sync_count = 0
    loss_of_signal_count = 0
    protocol_error_count = 0
    invalid_tx_word_count = 0
    invalid_crc_count = 0
Ports:
PORT      WWN                ALIAS                MANUFACTURER
port-000  wwn.2100001B3281A339  longjaw-1            QLogic Corporation
port-001  wwn.2101001B32A1A339  longjaw-2            QLogic Corporation
port-002  wwn.2100001B3281AC39  thicktail-1          QLogic Corporation
port-003  wwn.2101001B32A1AC39  thicktail-2          QLogic Corporation
port-004  wwn.2100001B3281E339  <none>                QLogic Corporation
port-005  wwn.2101001B32A1E339  <none>                QLogic Corporation
```

## FC イニシエータグループの作成

```
dory:configuration san fc initiators> create
dory:configuration san fc initiators (uncommitted)> set name=lefteye
dory:configuration san fc initiators (uncommitted)>
    set initiators=wwn.2101001B32A1AC39,wwn.2100001B3281AC39
dory:configuration san fc initiators (uncommitted)> commit
dory:configuration san fc initiators> list
GROUP      NAME
group-001  lefteye
|
+--> INITIATORS
    wwn.2101001B32A1AC39
    wwn.2100001B3281AC39
```

## LUN と FC イニシエータグループとの関連付け

次の例は、lefty という名前の LUN を作成し、それを fera イニシエータグループに関連付ける方法を示しています。

```
dory:shares default> lun lefty
dory:shares default/lefty (uncommitted)> set volsize=10
    volsize = 10 (uncommitted)
dory:shares default/lefty (uncommitted)> set initiatorgroup=fera
    initiatorgroup = default (uncommitted)
dory:shares default/lefty (uncommitted)> commit
```

## イニシエータとイニシエータグループの別名のスクリプト作成

次のスクリプト例を変更する方法および使用方法については、[34 ページの「CLI の使用法」](#)および[34 ページの「単純な CLI スクリプトとコマンドのバッチ処理」](#)のセクションを参照してください。

```
script
/*
 * This script creates both aliases for initiators and initiator
 * groups, as specified by the below data structure. In this
 * particular example, there are five initiator groups, each of
 * which is associated with a single host (thicktail, longjaw, etc.),
 * and each initiator group consists of two initiators, each of which
 * is associated with one of the two ports on the FC HBA. (Note that
 * there is nothing in the code that uses this data structure that
 * assumes the number of initiators per group.)
 */
groups = {
  thicktail: {
    'thicktail-1': 'wnn.2100001b3281ac39',
    'thicktail-2': 'wnn.2101001b32a1ac39'
  },
  longjaw: {
    'longjaw-1': 'wnn.2100001b3281a339',
    'longjaw-2': 'wnn.2101001b32a1a339'
  },
  tecopa: {
    'tecopa-1': 'wnn.2100001b3281e339',
    'tecopa-2': 'wnn.2101001b32a1e339'
  },
  spinedace: {
    'spinedace-1': 'wnn.2100001b3281df39',
    'spinedace-2': 'wnn.2101001b32a1df39'
  },
  fera: {
    'fera-1': 'wnn.2100001b32817939',
    'fera-2': 'wnn.2101001b32a17939'
  }
};
for (group in groups) {
  initiators = [];
  for (initiator in groups[group]) {
    printf('Adding %s for %s ... ',
           groups[group][initiator], initiator);
    try {
      run('select alias=' + initiator);
      printf('(already exists)\n');
      run('cd ..');
    } catch (err) {
      if (err.code != EAKSH_ENTITY_BADSELECT)
        throw err;
      run('create');
      set('alias', initiator);
      set('initiator', groups[group][initiator]);
      run('commit');
      printf('done\n');
    }
    run('select alias=' + initiator);
    initiators.push(get('initiator'));
    run('cd ..');
  }
}
```

```
    }  
    printf('Creating group for %s ... ', group);  
    run('groups');  
    try {  
        run('select name=' + group);  
        printf('(already exists)\n');  
        run('cd ..');  
    } catch (err) {  
        if (err.code != EAKSH_ENTITY_BADSELECT)  
            throw err;  
        run('create');  
        set('name', group);  
        run('set initiators=' + initiators);  
        run('commit');  
        printf('done\n');  
    }  
    run('cd ..');  
}
```

## iSCSI

インターネット SCSI は、SCSI ベースのストレージをシェアするためにアプライアンスでサポートされるいくつかのブロックプロトコルのうちの 1 つです。

### ターゲットの構成

iSCSI プロトコルを使用している場合、ターゲットポータルは、イニシエータがターゲットに接続するときを使用できる IP アドレスと TCP ポート番号の一意の組み合わせを指します。

iSCSI プロトコルを使用している場合、ターゲットポータルグループは、ターゲットポータルのコレクションです。ターゲットポータルグループは透過的に管理されます。各ネットワークインタフェースには、そのインタフェースのアクティブなアドレスを含む、対応するターゲットポータルグループがあります。ターゲットをインタフェースにバインドすると、その iSCSI ターゲットが、そのインタフェースに関連付けられたポータルグループを使用して通知されます。

注: セッションあたりの複数接続はサポートされていません。



IQN (iSCSI 修飾名) は、iSCSI ネットワーク内のデバイスの一意の識別子です。iSCSI では、IQN に対して `iqn.date.authority:uniqueid` の形式が使用されます。たとえば、アプライアンスは、iSCSI ターゲットの 1 つを識別するために `IQN:iqn.1986-03.com.sun:02:c7824a5b-f3ea-6038-c79d-ca443337d92c` を使用する可能性があります。この名前は、これが 1986 年 3 月に登録された、ある企業によって構築された iSCSI デバイスであることを示します。名前の authority は、単にその企業の DNS 名を逆にしただけのものです (この場合は、「com.sun」)。それ以降のすべては、Oracle がそのターゲットを識別するために使用している一意の ID です。

表 6-2 iSCSI ターゲットのプロパティ

ターゲットのプロパティ	説明
ターゲット IQN	このターゲットの IQN。IQN は手動で指定することも、自動生成することもできます。
別名	このターゲットの人間が読める形式のニックネーム。
認証モード	「なし」、「CHAP」、または「RADIUS」のいずれか。
CHAP 名	CHAP 認証が使用されている場合は、CHAP ユーザー名。
CHAP シークレット	CHAP 認証が使用されている場合は、CHAP シークレット。
ネットワークインタフェース	ターゲットポータルがこのターゲットをエクスポートするために使用されるインタフェース。

これらのプロパティに加えて、BUI では、ターゲットがオンラインまたはオフラインのどちらであるかが示されます。

表 6-3 ターゲットのステータスアイコン

アイコン	説明
	ターゲットはオンライン
	ターゲットはオフライン

## クラスタ化に関する考慮事項

クラスタ化されたプラットフォームでは、そのクラスタノード上に少なくとも 1 つのアクティブインタフェースが存在するターゲットはオンラインになります。ターゲットにインタフェースを割り当てる場合は注意してください。ターゲットは、不連続なヘッドノード上のポータルグループを使用するように構成されている可能性があります。その状況では、ターゲットは両方のヘッド上でオンラインになりますが、各ヘッドノードによって所有されているストレージに応じて異なる LUN をエクスポートします。ネットワークインタフェースは、テイクオーバーやフェイルバックまたは所有者変更の一部としてクラスタヘッド間で移行するため、iSCSI ターゲットは、対応するネットワークインタフェースがインポートおよびエクスポートされるに従ってオンラインおよびオフラインに移動します。

IPMP インタフェースにバインドされたターゲットは、その IPMP グループのアドレスを介してのみ通知されます。そのターゲットに、そのグループのテストアドレスを介して到達することはできません。LACP アグリゲーションの上に構築されたインタフェースにバインドされたターゲットは、そのアグリゲーションのアドレスを使用します。LACP アグリゲーション

が IPMP グループに追加された場合は、そのアドレスが IPMP テストアドレスになるため、ターゲットはそのアグリゲーションのインタフェースを使用できなくなります。

## イニシエータの構成

iSCSI イニシエータには、次の構成可能なプロパティがあります。

表 6-4 iSCSI イニシエータのプロパティ

プロパティ	説明
イニシエータ IQN	このイニシエータの IQN。
別名	このイニシエータの人間が読める形式のニックネーム。
CHAP を使用	CHAP 認証を有効または無効にします
CHAP 名	CHAP 認証が使用されている場合は、CHAP ユーザー名。
CHAP シークレット	CHAP 認証が使用されている場合は、CHAP シークレット。

## クライアント構成の計画

iSCSI クライアント構成を計画する場合は、次の情報が必要になります。

- どのイニシエータ (およびその IQN) が SAN にアクセスするか。
- CHAP 認証の使用を計画している場合、各イニシエータはどの CHAP 資格を使用するか。
- iSCSI ディスク (LUN) はいくつ必要で、どれだけの大きさにすべきか。
- LUN を複数のイニシエータ間でシェアする必要があるか。

アプライアンスが RADIUS を使用して CHAP 認証を実行できるようにするには、次の情報が一致する必要があります。

- アプライアンスは、RADIUS サーバーのアドレスと、この RADIUS サーバーと通信するときに使用するシークレットを指定する必要があります
- RADIUS サーバー (たとえば、クライアントファイル内) には、このアプライアンスのアドレスを提供し、かつ上と同じシークレットを指定するエントリが含まれている必要があります
- RADIUS サーバー (たとえば、ユーザーファイル内) には、CHAP 名を提供し、かつ各イニシエータの CHAP シークレットに一致するエントリが含まれている必要があります

- イニシエータが CHAP 名として自身の IQN 名を使用する場合 (推奨される構成) は、アプライアンスに、イニシエータボックスごとに個別のイニシエータエントリは必要ありません。RADIUS サーバーは、すべての認証手順を実行できます。
- イニシエータが個別の CHAP 名を使用する場合は、アプライアンスに、IQN 名から CHAP 名へのマッピングを指定する、そのイニシエータのためのイニシエータエントリが存在する必要があります。このイニシエータエントリで、そのイニシエータの CHAP シークレットを指定する必要はありません。

## iSCSI のトラブルシューティング

iSCSI の一般的な構成の誤りのトラブルシューティングに関するヒントについては、[208 ページの「iSCSI」](#)のセクションを参照してください。



## iSCSI のパフォーマンスのモニタリング

iSCSI のパフォーマンスは、『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」を通して監視できます。その場合、操作またはスループットをイニシエータ、ターゲット、または LUN ごとに分解できます。

## BUI を使用した iSCSI の構成

### ▼ 分析ワークシートの作成

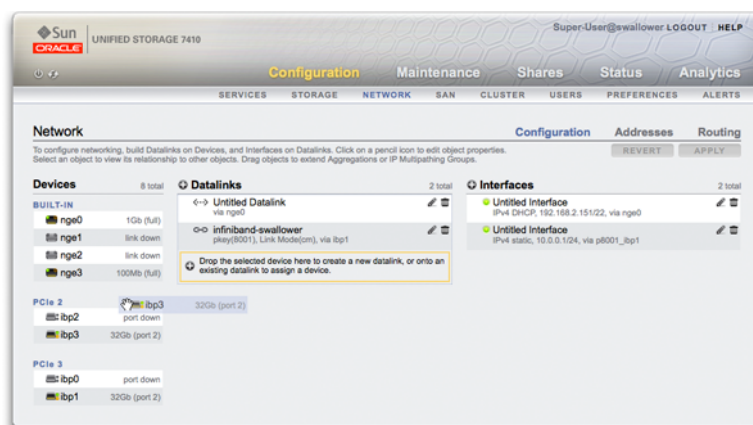
イニシエータによる動作を監視するための分析ワークシートを作成するには、次の手順を実行します。


1. 「分析」画面に移動します。
2. 「統計を追加」用の  追加アイコンをクリックします。すべての統計情報のメニューが表示されます。
3. メニューの「プロトコル」セクションの下にある「iSCSI 操作」>「イニシエータ別」を選択します。イニシエータによる現在の動作のグラフが表示されます。
4. より詳細な分析を監視するには、グラフの左側にあるフィールドからイニシエータを選択し、 アイコンをクリックします。詳細な分析のメニューが表示されます。

## ▼ iSER ターゲットの構成

BUI では、iSER ターゲットは「構成」>「SAN」画面で iSCSI ターゲットとして管理されます。


1. ibp(x) インタフェースを構成するには、目的の ibp(x) インタフェース (または ipmp) を選択して「データリンク」リストにドラッグし、「構成」>「ネットワーク」画面でデータリンクを作成します。
2. そのデータリンクを「インタフェース」リストにドラッグして、新しいインタフェースを作成します。




3. iSER ターゲットを作成するには、「構成」>「SAN」ページで「iSCSI ターゲット」リンクをクリックします。
4. 新しい iSER ターゲットを別名で追加するには、 追加アイコンをクリックします。
5. ターゲットグループを作成するには、今作成したターゲットを「iSCSI ターゲットグループ」リストにドラッグします。

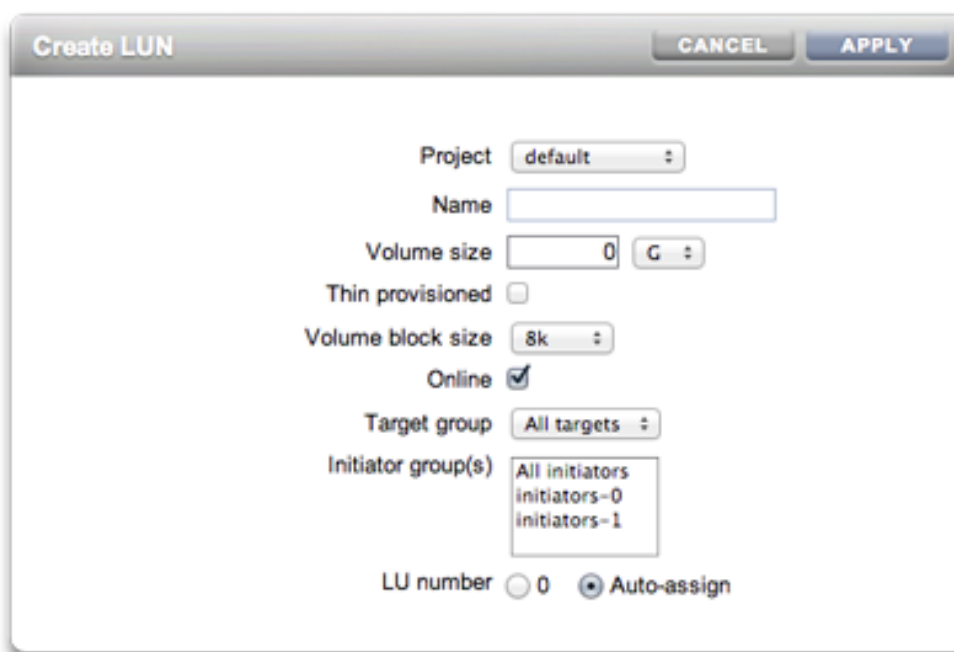




6. イニシエータを作成するにはには、「イニシエータ」リンクをクリックしてから、「iSCSI イニシエータ」リンクをクリックします。
7. 新しいイニシエータを追加するには、 追加アイコンをクリックします。
8. イニシエータ IQN と別名を入力し、「OK」をクリックします。イニシエータグループの作成はオプションですが、グループを作成しないと、ターゲットに関連付けられた LUN がすべてのイニシエータから使用可能になります。
9. グループを作成するには、イニシエータを「iSCSI イニシエータグループ」リストにドラッグします。



10. LUN を作成するには、「シェア」ページで「LUN」をクリックします。
11.  追加アイコンをクリックし、「ターゲットグループ」および「イニシエータグループ」メニューを使用して、新しい LUN をすでに作成したターゲットまたはイニシエータグループに関連付けます。



## CLI を使用した iSCSI の構成

### 自動生成された IQN による iSCSI ターゲットの追加

```

ahi:configuration san iscsi targets> create
ahi:configuration san iscsi targets target (uncommitted)> set alias="Target 0"
ahi:configuration san iscsi targets target (uncommitted)> set auth=none
ahi:configuration san iscsi targets target (uncommitted)> set interfaces=igb1
ahi:configuration san iscsi targets target (uncommitted)> commit
ahi:configuration san iscsi targets> list
TARGET    ALIAS
target-000 Target 0
|
+--> IQN
      iqn.1986-03.com.sun:02:daf0161f-9f5d-e01a-b5c5-e1efa9578416

```

### 特定の IQN と RADIUS 認証による iSCSI ターゲットの追加

```

ahi:configuration san iscsi targets> create
ahi:configuration san iscsi targets target (uncommitted)> set alias="Target 1"
ahi:configuration san iscsi targets target (uncommitted)>
  set iqn=iqn.2001-02.com.acme:12345
ahi:configuration san iscsi targets target (uncommitted)> set auth=radius

```

```

ahi:configuration san iscsi targets target (uncommitted)> set interfaces=igb1
ahi:configuration san iscsi targets target (uncommitted)> commit
ahi:configuration san iscsi targets> list
TARGET      ALIAS
target-000  Target 0
            |
            +--> IQN
                    iqn.1986-03.com.sun:02:daf0161f-9f5d-e01a-b5c5-e1efa9578416
target-001  Target 1
            |
            +--> IQN
                    iqn.2001-02.com.acme:12345

```

## CHAP 認証を使用する iSCSI イニシエータの追加

```

ahi:configuration san iscsi initiators> create
ahi:configuration san iscsi initiators initiator (uncommitted)>
    set initiator=iqn.2001-02.com.acme:initiator12345
ahi:configuration san iscsi initiators initiator (uncommitted)> set alias="Init 0"
ahi:configuration san iscsi initiators initiator (uncommitted)>
    set chapuser=thisismychapuser
ahi:configuration san iscsi initiators initiator (uncommitted)>
    set chapsecret=123456789012abc
ahi:configuration san iscsi initiators initiator (uncommitted)> commit
ahi:configuration san iscsi initiators> list
NAME      ALIAS
initiator-000  Init 0
            |
            +--> INITIATOR
                    iqn.2001-02.com.acme:initiator12345

```

## iSCSI ターゲットグループの追加

```

ahi:configuration san iscsi targets groups> create
ahi:configuration san iscsi targets group (uncommitted)> set name=tg0
ahi:configuration san iscsi targets group (uncommitted)>
    set targets=iqn.2001-02.com.acme:12345,
                    iqn.1986-03.com.sun:02:daf0161f-9f5d-e01a-b5c5-e1efa9578416
ahi:configuration san iscsi targets group (uncommitted)> commit
ahi:configuration san iscsi targets groups> list
GROUP     NAME
group-000 tg0
            |
            +--> TARGETS
                    iqn.2001-02.com.acme:12345
                    iqn.1986-03.com.sun:02:daf0161f-9f5d-e01a-b5c5-e1efa9578416

```

## iSCSI イニシエータグループの追加

```

ahi:configuration san iscsi initiators groups> create
ahi:configuration san iscsi initiators group (uncommitted)> set name=ig0
ahi:configuration san iscsi initiators group (uncommitted)>

```

```

set initiators=iqn.2001-02.com.acme:initiator12345
ahi:configuration san iscsi initiators group (uncommitted)> commit
ahi:configuration san iscsi initiators groups> list
GROUP      NAME
group-000  ig0
           |
           +-> INITIATORS
                iqn.2001-02.com.acme:initiator12345

```

## SRP

SCSI RDMA プロトコルは、RDMA サービスを提供するネットワーク (つまり、InfiniBand) を経由して SCSI ベースのストレージをシェアするためにアプライアンスでサポートされるプロトコルです。

## SRP ターゲットの構成



SRP ポートは、IPoIB や RDMA などのほかの IB ポートサービスとシェアされます。SRP サービスは、ターゲットモードでのみ動作できます。SRP ターゲットには、次の構成可能なプロパティがあります。

表 6-5 SRP ターゲットのプロパティ

プロパティ	説明
ターゲット EUI	このターゲットの EUI (Extended Unique Identifier)。EUI はシステムによって自動的に割り当てられ、SRP ポートサービスが実行されている HCA GUID と同じです。
別名	このターゲットの人間が読める形式のニックネーム。

これらのプロパティに加えて、BUI では、ターゲットがオンラインまたはオフラインのどちらであるかが示されます。

表 6-6 SRP ターゲットのステータスアイコン

アイコン	説明
	ターゲットはオンライン
	ターゲットはオフライン

## クラスタ化に関する考慮事項

クラスタ化されたプラットフォームでは、可用性の高い (マルチパス) 構成にするために、ピアターゲットは同じターゲットグループに構成する必要があります。SRP のマルチパス I/O は、イニシエータ側の構成オプションです。

## イニシエータの構成

SRP イニシエータには、次の構成可能なプロパティがあります。

表 6-7 SRP イニシエータのプロパティ

プロパティ	説明
イニシエータ EUI	このイニシエータの EUI。
別名	このイニシエータの人間が読める形式のニックネーム。

## SRP のパフォーマンスの監視




SRP のパフォーマンスは、『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」を通して監視できます。その場合、操作またはスループットをイニシエータまたはターゲットごとに分解できます。{{Server}}/wiki/images/cfg\_san\_srp.png

## BUI を使用した SRP ターゲットの構成

### ▼ SRP ターゲットの構成

ここでは、SRP ターゲットを構成するための手順について説明します。

1. HCA ポートを IB インタフェースに接続します。
2. アプライアンスによってターゲットが自動的に検出されます。
3. ターゲットグループを作成するには、「構成」>「SAN」画面に移動します。
4. 「ターゲット」リンクをクリックしてから、SRP ターゲットをクリックします。
5. 「SRP ターゲット」ページが表示されます。

6. ターゲットグループを作成するには、 移動アイコンを使用してターゲットを「ターゲットグループ」リストにドラッグします。
7. 「適用」をクリックします。
8. (オプション)「イニシエータ」画面でイニシエータとイニシエータグループを作成するには、 アイコンをクリックし、イニシエータから GUID を収集し、それに名前を割り当ててから、イニシエータグループにドラッグします。
9. LUN を作成し、それを前の手順で作成した SRP ターゲットとイニシエータに関連付けるには、「シェア」画面に移動します。
10. 「LUN」リンクをクリックしてから、LUN  アイコンをクリックします。「LUN を作成」ダイアログにある「ターゲットグループ」および「イニシエータグループ」メニューを使用して、LUN に関連付ける SRP グループを選択します。

## CLI を使用した SRP ターゲットの構成

次の例では、CLI の `configuration san targets srp groups` コンテキストを使用して、`targetSRPgroup` という SRP ターゲットグループを作成する方法を示します。

```
swallower:configuration san targets srp groups> create
swallower:configuration san targets srp group (uncommitted)> set name=targetSRPgroup
      name = targetSRPgroup (uncommitted)
swallower:configuration san targets srp group (uncommitted)>
set targets=eui.0002C903000489A4
      targets = eui.0002C903000489A4 (uncommitted)
swallower:configuration san targets srp group (uncommitted)> commit
swallower:configuration san targets srp groups> list
GROUP      NAME
group-000  targetSRPgroup
|
+--> TARGETS
      eui.0002C903000489A4
```

次の例では、CLI `shares` CLI コンテキストを使用し、LUN を作成して `targetSRPgroup` に関連付ける方法を示します。

```
swallower:shares default> lun mylun
swallower:shares default/mylun (uncommitted)> set targetgroup=targetSRPgroup
      targetgroup = targetSRPgroup (uncommitted)
swallower:shares default/mylun (uncommitted)> set volsize=10
      volsize = 10 (uncommitted)
swallower:shares default/mylun (uncommitted)> commit
swallower:shares default> list
Filesystems:
NAME      SIZE      MOUNTPOINT
test      38K      /export/test
```

---

LUNs:		
NAME	SIZE	GUID
myLun	10G	600144F0E9D19FFB00004B82DF490001





# ◆◆◆ 第 7 章

## ユーザー構成

---

このセクションでは、アプライアンスを管理できるユーザー、ユーザーに付与された承認を管理するロール、および BUI または CLI を使用してこれらをシステムに追加する方法について説明します。

ユーザーは次のいずれかになります。

- ローカルユーザー - すべてのアカウント情報がアプライアンスに保存されます。
- ディレクトリユーザー - 既存の [248 ページの「NIS」](#) または [250 ページの「LDAP」](#) アカウントを使用して、補足の承認設定をアプライアンスに保存します。これにより、既存の NIS または LDAP ユーザーにアプライアンスにログインして管理する権限を付与できます。

ローカルユーザーはデータサービスをサポートしていますが、注意が必要な点がいくつかあります。

- ローカルユーザーは UID を制御できません。これは、その他を使用する NFSv3 と、AUTH\_SYS を使用する NFSv4 の問題です。
- ローカルグループはサポートされません。
- データ目的のためにローカルユーザーを定義すると、ローカルユーザーが管理インタフェースにログインできることになります。

ユーザーに権限を付与するには、カスタムロールを割り当てます。

## ユーザーロール

ロールとは、ユーザーに割り当てることができる権限のコレクションです。さまざまな承認レベルを持つ管理者およびオペレータロールを作成することが望ましい場合があります。スタッフメンバーには、不要な権限を割り当てずに、必要に応じて適切なロールを割り当てることができます。

シェアの管理者パスワードを使用する (たとえば、*root* パスワードをすべてのユーザーに割り当てる) よりも、ロールを使用した方が大幅にセキュアであると考えられています。ロールは、

必要な承認だけにユーザーを制限し、『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「ログ」ログではユーザーのアクションが個々のユーザー名に帰属すると見なします。

デフォルトでは、非常に基本的な承認が含まれる「基本管理」というロールが存在します。

## ユーザー承認

承認を使用すると、ユーザーはシェアの作成、アプライアンスのリポート、システムソフトウェアの更新などの特定のタスクを実行できます。承認はスコープに分類され、各スコープには承認のスコープを狭くするオプションのフィルタセットがある場合があります。たとえば、すべてのサービスを再起動する承認の代わりにフィルタを使用すると、この承認で HTTP サービスだけを再起動できます。

次の表は、使用可能なスコープを示しています。

表 7-1 ユーザーの使用可能なスコープ

BUI のスコープ	CLI のスコープ	承認の例	フィルタの例
Active Directory	ad	Active Directory ドメインへの参加	ドメイン名
警告	alert	警告フィルタとしきい値の構成	.
分析	stat	このドリルダウンが存在する統計の読み取り	ドリルダウン
クラスタ化	cluster	クラスタピアへのリソースのフェイルバック	.
データセット	dataset	分析データセットのさまざまな面の管理	構成
ハードウェア	hardware	オンラインおよびオフラインのディスク	.
キーストア	keystore	キーストアの構成。	.
ネットワーク	net	ネットワークデバイス、データリンク、およびインタフェースの構成	.
プロジェクトおよびシェア	nas	プロジェクトおよびシェアの一般プロパティの変更	ルール、プロジェクト、シェア
ロール	role	ロールへの承認の構成	ロール名

BUI のスコープ	CLI のスコープ	承認の例	フィルタの例
SAN	stmf	SAN への承認の構成	
サービス	svc	サービスの再起動	サービス名
プロパティスキーマのシェア	schema	プロパティスキーマの変更	
システム更新	appliance	アプライアンスのリポート	アプライアンス名
	update	システムソフトウェアの更新	
ユーザー	user	パスワードの変更	ユーザー名
ワークフロー	workflow	ワークフローの変更	ワークフロー名
ワークシート	worksheet	ワークシートの変更	ワークシート名

BUI でスコープを参照して、その他の承認が存在することを確認します。現在、50 種類以上の承認が使用可能で、今後のアプライアンスソフトウェアの更新では承認が追加される可能性もあります。

## ユーザープロパティの管理

ユーザーおよびロールを管理する際は、次のプロパティを設定できます。

### ユーザープロパティ

ユーザーの追加時には次のプロパティのすべてを設定でき、ユーザーの編集時にはこれらのサブセットを設定できます。

表 7-2 ユーザープロパティ

プロパティ	説明
タイプ	ディレクトリ (NIS または LDAP から資格にアクセス)、またはローカル (このアプライアンス上にユーザーを保存)
ユーザー名	ユーザーの一意の名前
フルネーム	ユーザーの説明
パスワード/確認	ローカルユーザーの場合は、これらのフィールドの両方に初期パスワードを入力します

プロパティ	説明
セッションの注釈が必要です	有効にした場合、ユーザーがアプライアンスにログインするときに、ログインの目的について説明するテキストを入力する必要があります。この注釈は、チケットシステムでのリクエストで実行される作業を追跡する際に使用される場合があります。セッション注釈としてチケット ID を使用できます。セッションの注釈は、『 <a href="#">Oracle ZFS Storage Appliance 顧客サービスマニュアル</a> 』の「ログ」ログに表示されます。
キオスクユーザー	有効にした場合、ユーザーは「キオスク画面」設定の画面のみを表示できます。たとえば、これを使用すると、 <a href="#">46 ページの「ダッシュボード」</a> のみを表示するようにユーザーを制限できます。キオスクユーザーは、CLI 経由でアプライアンスにアクセスできなくなります。
キオスク画面	「キオスクユーザー」を有効にした場合に、このキオスクユーザーに許可される画面
ロール	このユーザーによって所有されるロール
例外	これらの承認は、選択したロールによって、通常は使用可能な承認から除外されます。

## ロールのプロパティ

ロールの管理時には、次のプロパティを設定できます。






表 7-3      ロールのプロパティ

プロパティ	説明
名前	一覧に表示されるロールの名前
説明	ロールの詳細な説明 (必要な場合)
承認	このロールへの承認

## BUI の「ユーザー」ページ


「BUI ユーザー」ページには、ユーザーとグループの両方が管理用のボタンとともに一覧表示されます。エントリの上にマウスを置くと、クローン、編集、および破棄ボタンが表示されます。エントリをダブルクリックすると、編集画面が表示されます。ボタンは次のとおりです。

表 7-4 BUI の「ユーザー」ページのアイコン

アイコン	説明
	新規ユーザー/ロールを追加します。これにより、必要なプロパティを入力できる新しいダイアログが表示されます。
	検索ボックスを表示します。検索文字列を入力して Enter ボタンを押すと、ユーザー/ロール一覧で該当するテキストが検索され、一致するエントリのみが表示されます。このアイコンを再度クリックするか、「すべて表示」をクリックすると、完全な一覧に戻ります。
	ユーザー/ロールのクローンを作成します。このエントリの値に基づいたフィールドで始まる新規ユーザー/ロールを追加します
	ユーザー/ロールを編集します
	ユーザー/ロール/承認を削除します

## BUI を使用したユーザーの構成

### ▼ 管理者の追加

1. 「ロール」一覧に、適切な管理者ロールが表示されているかチェックします。表示されていない場合は、ロールを追加します (個々のタスクを参照)。
2. 「ユーザー」の横にある  追加アイコンをクリックします。
3. ユーザープロパティを設定します。
4. 管理者ロールのチェックボックスにチェックマークを付けます。
5. ダイアログの最上部にある「追加」ボタンをクリックします。「ユーザー」一覧に新規ユーザーが表示されます。

### ▼ ロールの追加



1. 「ロール」の横にある  追加アイコンをクリックします。

2. ロールの名前および説明を設定します。
3. ロールに承認を追加します (個々のタスクを参照)。
4. ダイアログの最上部にある「追加」ボタンをクリックします。「ロール」一覧に新規ロールが表示されます。

## ▼ ロールへの承認の追加

1. 「スコープ」を選択します。このスコープでフィルタが使用可能な場合は、「スコープ」セレクトの下にフィルタが表示されます。
2. 適宜、フィルタを選択します。
3. 追加するすべての承認のチェックボックスにチェックマークを付けます。
4. 「承認」セクションの「追加」ボタンをクリックします。ダイアログ下部の一覧に承認が追加されます。

## ▼ ロールからの承認の削除

1. 「ロール」一覧のロール上にマウスを置いて、 編集アイコンをクリックします。
2. 最下部の一覧の承認上にマウスを置いて、右側にある  ごみ箱アイコンをクリックします。
3. ダイアログの最上部にある「適用」ボタンをクリックします。

## ▼ ダッシュボードのみを表示できるユーザーの追加

1. ディレクトリユーザーまたはローカルユーザーを追加します (個々のタスクを参照)。
2. キオスクモードを true に設定し、「キオスク画面」が「ステータス/ダッシュボード」に設定されていることを確認します。
3. これでユーザーはログインできますが、ダッシュボードしか表示できません。

## CLI を使用したユーザーの構成

BUI で使用可能なアクションは、CLI でも使用可能です。ユーザー、ロール、および承認の管理をナビゲートするときに「help」と入力すると、使用可能なコマンドの一覧が表示されます。

### CLI のユーザー構成の例

CLI ユーザーおよびロールのインタフェースを実演するために、次の例では「brendan」という NIS ユーザーをシステムに追加し、HTTP サービスを再起動するための承認を付与します。これには、この承認へのロールの作成も含まれます。

「webadmin」というロールの作成から開始します。

```
caji:> configuration roles
caji:configuration roles> role webadmin
caji:configuration roles webadmin (uncommitted)> set
  description="web server administrator"
  description = web server administrator (uncommitted)
caji:configuration roles webadmin (uncommitted)> commit
caji:configuration roles> show
Roles:

NAME            DESCRIPTION
basic           Basic administration
webadmin        web server administrator
```

webadmin ロールが作成されたので、HTTP サービスを再起動するための承認を追加します。この例では、有効な入力を一覧表示するタブ補完の出力も表示します。これは、有効なスコープおよびフィルタオプションを決定する際に役立ちます。

```
caji:configuration roles> select webadmin
caji:configuration roles webadmin> authorizations
caji:configuration roles webadmin authorizations> create
caji:configuration roles webadmin auth (uncommitted)> set scope=tab
ad          cluster    net          schema      update
alert      hardware  replication  stat        user
appliance  nas       role        svc         worksheet
caji:configuration roles webadmin auth (uncommitted)> set scope=svc
  scope = svc
caji:configuration roles webadmin auth (uncommitted)> show
Properties:
  scope = svc
  service = *
  allow_administer = false
  allow_configure = false
  allow_restart = false

caji:configuration roles webadmin auth (uncommitted)> set service=tab
*          ftp          ipmp         nis          ssh
ad         http         iscsi       ntp          tags
```

```
smb          identity      ldap          routing      vscan
datalink:igb0 idmap          ndmp          scrk
dns          interface:igb0 nfs           snmp
caji:configuration roles webadmin auth (uncommitted)> set service=http
                service = http (uncommitted)
caji:configuration roles webadmin auth (uncommitted)> set allow_restart=true
                allow_restart = true (uncommitted)
caji:configuration roles webadmin auth (uncommitted)> commit
caji:configuration roles webadmin authorizations> list
NAME      OBJECT          PERMISSIONS
auth-000  svc.http        restart
```

ロールが作成されたので、ユーザーセクションに移動してユーザー「brendan」を作成し、ロール「webadmin」を割り当てることができます。

```
caji:configuration roles webadmin authorizations> cd ../../..
caji:configuration> users
caji:configuration users> netuser brendan
caji:configuration users> show
Users:

NAME          USERNAME      UID      TYPE
Brendan Gregg brendan      130948   Dir
Super-User    root          0        Loc

caji:configuration users> select brendan
caji:configuration users brendan> show
Properties:
    logname = brendan
    fullname = Brendan Gregg
    initial_password = *****
    require_annotation = false
    roles = basic
    kiosk_mode = false
    kiosk_screen = status/dashboard

Children:
    exceptions => Configure this user's exceptions
    preferences => Configure user preferences
caji:configuration users brendan> set roles=basic,webadmin
                roles = basic,webadmin (uncommitted)
caji:configuration users brendan> commit
```

この時点で、ユーザー brendan は NIS パスワードを使用してログインし、アプライアンスで HTTP サービスを再起動できます。

## ▼ 管理者の追加

1. **configuration roles** に移動します。
2. 「show」と入力します。各ロールで **select, authorizations show** の順に実行して、適切な管理承認を持つロールを検索します。適切なロールが存在しない場合は、ロールの作成から開始します (個々のタスクを参照)。



3. `configuration users` に移動します。
4. ディレクトリユーザー (NIS、LDAP) の場合、`netuser` に続いて追加する既存のユーザー名を入力します。ローカルユーザーの場合は、「`user`」に続いて追加するユーザー名を入力してから、「`show`」と入力して、設定する必要があるプロパティを確認します。「`set`」と入力してから、「`commit`」と入力します。
5. この時点で、ユーザーは作成されていますが、すべてのプロパティがカスタマイズされているとは限りません。「`select`」に続いてユーザー名を入力します。
6. ここで、「`show`」と入力すると、設定の完全な一覧が表示されます。この時点で、[第8章「ZFSSA の設定」](#)のほかにも、ロールおよび承認の例外が追加される場合もあります。

## ▼ ロールの追加

1. `configuration roles` に移動します。
2. `role` に続いて、作成するロール名を入力します。
3. 説明を設定してから、「`commit`」と入力してロールをコミットします。
4. ロールに承認を追加します (個々のタスクを参照)。

## ▼ ロールへの承認の追加

1. `configuration roles` に移動します。
2. 「`select`」に続いてロール名を入力します。
3. 「`authorizations`」と入力します。
4. 「`create`」と入力して承認を追加します
5. 「`set scope=`」に続いてスコープ名を入力します。タブ補完を使用して、一覧を表示します。
6. 「`show`」と入力して、使用可能なフィルタと承認の両方を表示します。
7. 「`set`」と入力して目的の承認を `true` に設定して、フィルタを設定します (使用可能な場合)。タブ補完は、有効なフィルタ設定を表示する際に役立ちます。

8. 「commit」と入力します。これで、承認が追加されました。

## ▼ ロールからの承認の削除

1. `configuration roles` に移動します。
2. 「select」に続いてロール名を入力します。
3. 「authorizations」と入力します。
4. 「show」と入力して、承認を一覧表示します。
5. 「destroy」に続いて承認名 (例: "auth-001") を入力します。これで、承認が削除されました。

◆◆◆ 第 8 章



## ZFSSA の設定

---

このセクションでは、ロケール、セッションプロパティ、および SSH 鍵の設定について説明します。

### 設定のプロパティ

BUI にログインすると、自分のアカウントに次の設定を行うことができますが、ほかのユーザーのアカウント設定はできません。

表 8-1 設定

プロパティ	説明
初期ログイン画面	正常なログイン後に BUI でロードされる最初のページ。デフォルトでは、 <a href="#">46 ページの「ステータスダッシュボード」</a> です。
ロケール	デフォルトでは C です。C および POSIX ロケールでは、ASCII 文字または標準テキストのみがサポートされています。ISO 8859-1 では、アフリカンス語、バスク語、カタロニア語、デンマーク語、オランダ語、英語、フェロー語、フィンランド語、フランス語、ガリシア語、ドイツ語、アイスランド語、アイルランド語、イタリア語、ノルウェー語、ポルトガル語、スペイン語、およびスウェーデン語の各言語がサポートされています。
セッションタイムアウト	BUI から離れたあとに、ブラウザが自動的にセッションをログアウトする時間
現在のセッションの注釈	監査ログに追加される注釈テキスト
高度な分析統計	これにより、『 <a href="#">Oracle ZFS Storage Appliance Analytics ガイド</a> 』の「 <a href="#">Analytics</a> 」で追加の統計が使用可能になります。
SSH 公開鍵	RSA/DSA 公開鍵。テキストコメントを鍵に関連付けると、鍵が追加された理由を管理者が追跡できます。BUI では、これらの鍵は現在のユーザーにのみ適

プロパティ	説明
	用されるため、ほかのユーザーの鍵を追加する場合は CLI を使用します。

## CLI を使用した設定

設定は CLI の `configuration users` で行うことができます。次の例では、「brendan」ユーザーアカウントで高度な分析を有効にする方法を示します。

```
caji:> configuration users
caji:configuration users> select brendan
caji:configuration users brendan> preferences
caji:configuration users brendan preferences> show
Properties:
    locale = C
    login_screen = status/dashboard
    session_timeout = 15
    advanced_analytics = false

Children:
    keys => Manage SSH public keys

caji:configuration users brendan preferences> set advanced_analytics=true
    advanced_analytics = true (uncommitted)
caji:configuration users brendan preferences> commit
```

自分の構成を CLI の `configuration preferences`で行います。次の例では、自分のアカウントにセッション注釈を設定する方法を示します。

```
twofish:> configuration preferences
twofish:configuration preferences> show
Properties:
    locale = C
    login_screen = status/dashboard
    session_timeout = 15
    session_annotation =
    advanced_analytics = false

Children:
    keys => Manage SSH public keys

twofish:configuration preferences> set session_annotation="Editing my user preferences"
    session_annotation = Editing my user preferences (uncommitted)
twofish:configuration preferences> commit
```

## CLI を使用した SSH 公開鍵の設定

ほかのホストからの CLI スクリプトの実行を自動化するときに、SSH 公開鍵が必要になる場合があります。次の例では、CLI から SSH 鍵を追加する方法を示します。

```
caji:> configuration preferences keys
caji:configuration preferences keys> create
caji:configuration preferences key (uncommitted)> set type=DSA
caji:configuration preferences key (uncommitted)> set key="...DSA key text..."
      key = ...DSA key text...== (uncommitted)
caji:configuration preferences key (uncommitted)> set comment="fw-log1"
      comment = fw-log1 (uncommitted)
caji:configuration preferences key (uncommitted)> commit
caji:configuration preferences keys> show
Keys:
```

NAME	MODIFIED	TYPE	COMMENT
key-000	10/12/2009 10:54:58	DSA	fw-log1

鍵テキストは、空白のない鍵テキスト自体 (通常は数百文字) です。



## 警告の構成

---

このセクションでは、システムの警告、警告のカスタマイズ方法、および警告ログの検索場所について説明します。『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」から統計をモニターするには、カスタムしきい値警告を作成します。特定タイプの警告に応答するようにシステムを構成するには、警告アクションを使用します。

### 警告のカテゴリ

重要なアプライアンスイベントでは、ハードウェアおよびソフトウェアの障害などの警告がトリガーされます。これらの警告は、『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「ログ」に表示され、任意の警告アクションが実行されるように構成できます。

警告は次のカテゴリに分類されます。

表 9-1 警告のカテゴリ

カテゴリ	説明
クラスタ	クラスタイベント (リンクの障害やピアのエラーなど)
カスタム	カスタム警告の構成から生成されたイベント
ハードウェアイベント	アプライアンスのブートおよびハードウェア構成の変更
ハードウェア障害	任意のハードウェア障害
NDMP 操作	バックアップと復元、起動と完了のイベント。バックアップイベントのみまたは復元イベントのみの場合、このグループは「NDMP: バックアップのみ」および「NDMP: 復元のみ」として使用できます
ネットワーク	ネットワークポート、データリンク、および IP インタフェースのイベントと障害
フォンホーム	バンドルのアップロードイベントをサポートします
リモートレプリケーション	イベントおよび障害を送受信します。ソースイベントのみまたはターゲットイベントのみの場合、このグループは「リモートレプリケーション: ソースのみ」および「リモ-

カテゴリ	説明
	トレプリケーション: ターゲットのみ」として使用できません
サービス障害	ソフトウェア第11章「ZFSSA サービス」障害のイベント
しきい値	『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」の統計に基づくカスタム警告
ZFS プール	ストレージプールのイベント (スクラブやホットスペースの起動など)

## サポートされている警告アクション

次のアクションがサポートされています。

### 電子メールの送信

警告の詳細を示す電子メールを送信できます。構成には、電子メールアドレスおよび電子メール件名行が必要です。次のサンプルは、しきい値警告に基づいて送信された電子メールです。

```
From aknobody@caji.com Mon Oct 13 15:24:47 2009
Date: Mon, 13 Oct 2009 15:24:21 +0000 (GMT)
From: Appliance on caji <noreply@caji.com>
Subject: High CPU on caji
To: admin@hostname.com
```

```
SUNW-MSG-ID: AK-8000-TT, TYPE: Alert, VER: 1, SEVERITY: Minor
EVENT-TIME: Mon Oct 13 15:24:12 2009
PLATFORM: i86pc, CSN: 0809QAU005, HOSTNAME: caji
SOURCE: svc:/appliance/kit/akd:default, REV: 1.0
EVENT-ID: 15a53214-c4e7-eae4-dae6-a652a51ea29b
DESC: cpu.utilization threshold of 90 is violated.
AUTO-RESPONSE: None.
IMPACT: The impact depends on what statistic is being monitored.
REC-ACTION: The suggested action depends on what statistic is being monitored.
```

```
SEE: https://192.168.2.80:215/#maintenance/alert=15a53214-c4e7-eae4-dae6-a652a51ea29b
```

アプライアンスで電子メールを送信する方法についての詳細は、[280 ページの「SMTP」](#)サービス画面で構成できます。



## SNMP トラップの送信

SNMP トラップの宛先が [281 ページの「SNMP」](#) サービスに構成され、そのサービスがオンラインになっている場合、警告の詳細を示す SNMP トラップを送信できます。次の例は、Net-SNMP ツール `snmptrapd -P` によって表示される SNMP トラップです。

```
# /usr/sfw/sbin/snmptrapd -P
2009-10-13 15:31:15 NET-SNMP version 5.0.9 Started.
2009-10-13 15:31:34 caji.com [192.168.2.80]:
    iso.3.6.1.2.1.1.3.0 = Timeticks: (2132104431) 246 days, 18:30:44.31
    iso.3.6.1.6.3.1.1.4.1.0 = OID: iso.3.6.1.4.1.42.2.225.1.3.0.1
    iso.3.6.1.4.1.42.2.225.1.2.1.2.36.55.99.102.48.97.99.100.52.45.51.48.
99.49.45.52.99.49.57.45.101.57.99.98.45.97.99.50.55.102.55.49.50.54.
98.55.57 = STRING: "7cf0acd4-30c1-4c19-e9cb-ac27f7126b79"
    iso.3.6.1.4.1.42.2.225.1.2.1.3.36.55.99.102.48.97.99.100.52.45.51.48.
99.49.45.52.99.49.57.45.101.57.99.98.45.97.99.50.55.102.55.49.50.54.
98.55.57 = STRING: "alert.ak.xmlrpc.threshold.violated"
    iso.3.6.1.4.1.42.2.225.1.2.1.4.36.55.99.102.48.97.99.100.52.45.51.
48.99.49.45.52.99.49.57.45.101.57.99.98.45.97.99.50.55.102.55.49.50.
54.98.55.57 = STRING: "cpu.utilization threshold of 90 is violated."
```

## Syslog メッセージの送信

Syslog サービスが有効になっている場合は、警告の詳細を示す syslog メッセージを 1 つ以上のリモートシステムに送信できます。syslog ベイロードの例およびほかのオペレーティングシステムで syslog レシーバを構成する方法についての詳細は、[285 ページの「Syslog リレーサービス」](#)に関するドキュメントを参照してください。

## データセットの再開/一時停止

Analytics の『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[データセット](#)」は、再開や一時停止が可能です。これは、散発的なパフォーマンス課題を追跡するとき、およびこれらのデータセットを 24 時間毎日有効にすることが望ましくないときに特に役立ちます。

たとえば、CPU アクティビティのスパイクが 1 週間に 1 回または 2 回通知され、その他の分析で関連するドロップが NFS パフォーマンスに表示されたとします。一部の追加データセットを有効にしても、問題点を検証するには情報がまったく足りません。ホスト名およびファイル名のデータセットで NFS を有効にした場合は、原因をより深く理解できます。ただし、このような特定のデータセットは負荷が高くなる可能性があります。24 時間毎日有効のままにすると、すべてのユーザーのパフォーマンスが低下します。これは、データセットの再開/一時停止アクションが役立つ可能性のある状況です。しきい値警告は、CPU アクティビティのスパイクが検出されたときにのみ、ホスト名とファイル名のデータセットで一時停止中された NFS を再開するように構成できます。2 番目の警告は、短い間隔のデータが収集されたあとに、これらのデータセットを一時停止するように構成できます。最終結果とし

て、課題発生時にのみ必要なデータが収集され、このデータ収集によるパフォーマンス影響は最小化されます。

## ワークシートの再開/一時停止

これらのアクションでは、大量のデータセットが含まれる可能性のある Analytics の『Oracle ZFS Storage Appliance Analytics ガイド』の「ワークシートを開く」全体を再開または一時停止します。これを行う理由は、データセットを再開および一時停止する理由と同様です。

## ワークフローの実行

ワークフローは、オプションで警告アクションとして実行できます。ワークフローを警告アクションとして使用できるようにするには、その alert アクションが true に設定されている必要があります。詳細は、435 ページの「警告アクションとしてのワークフロー」を参照してください。

## しきい値警告

これらは、『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」からの統計に基づく警告です。しきい値警告の作成時のプロパティは次のとおりです。

表 9-2 しきい値警告のプロパティ

プロパティ	説明
しきい値	しきい値統計は、『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」で生成され、自己記述性があります (例:「プロトコル: 秒あたりの NFSv4 操作数」)
超える/下回る	しきい値と現在の統計とを比較する方法を定義します。
タイミング: 最低	現在の統計値がしきい値を上回る、または下回る必要がある期間
間のみ/途中のみ	これらのプロパティは、特定の時間帯 (勤務時間など) にのみしきい値が送信されるように設定できます。
この状況が続く場合に警告を再送信する間隔。	有効にすると、しきい値違反が存在するときに、設定した間隔ごとに警告アクション (電子メールの送信など) が再実行されます。

プロパティ	説明
この状況がクリアされた場合にも警告を送信する最低の間隔...	少なくとも設定した間隔でしきい値違反がクリアされた場合に、フォローアップ警告を送信します。

「しきい値警告を追加」ダイアログは、警告について説明したパラグラフとして参照できるように構成されています。デフォルトでは次のようになります。

しきい値 CPU: 使用率が 95 % を超える

タイミング最低 5 分、0:00 と 0:00 の間のみ、平日の途中のみ

この状況が続く場合に警告を再送信する間隔 5 分間。

少なくとも 5 分間、この条件がクリアされた場合にも警告を再送信します。

## BUI を使用した警告の構成

「構成」->「警告」ページの最上部には、「警告アクション」および「しきい値警告」のタブがあります。BUI を使用して構成する手順については、「タスク」を参照してください。

### ▼ しきい値警告の追加

1. 「しきい値警告」の横にある追加アイコンをクリックします。
2. モニターする統計を選択します。『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「統計」を使用すると、適切かどうかをチェックする統計を表示できます。
3. 「超える/下回る」、および目的の値を選択します。
4. 「タイミング」の詳細を入力します。デフォルトでは、しきい値を 5 分以上の間違反した場合にのみ警告が送信され、5 分ごとに再送信され、しきい値が 5 分間クリアされたあとに送信されます。
5. ドロップダウンメニューから警告アクションを選択し、右側にある必須フィールドに入力します。
6. 必要に応じて、「警告アクション」の横にある追加アイコンをクリックして、警告アクションの追加を続行します。
7. ダイアログの最上部にある「適用」をクリックします。

## ▼ 警告アクションの追加

1. 「警告アクション」の横にある追加アイコンをクリックします。
2. カテゴリを選択するか、すべて選択する場合は「すべてのイベント」を選択します。
3. すべてのイベントまたはイベントのサブセットを選択します。サブセットを選択した場合は、目的の警告イベントと一致するようにチェックボックスリストをカスタマイズします。
4. 「警告アクション」のドロップダウンメニューを使用して、警告タイプを選択します。
5. 警告アクションの詳細を入力します。「テスト」ボタンをクリックすると、テスト警告を作成し、この警告アクションを実行できます (電子メールや SNMP が正しく構成されているかどうかのチェックに役立ちます)。
6. 「警告アクション」の横にある追加アイコンをクリックすると、複数の警告アクションを追加できます。
7. 右上にある「追加」をクリックします。

## CLI を使用した警告の構成

警告は configuration alerts コンテキストを使用して、CLI から構成できます。CLI を使用して構成する手順については、「タスク」を参照してください。

## ▼ しきい値警告の追加

1. configuration alerts thresholds コンテキストを入力して、create コマンドを入力します。
2. 「set statname=[name はモニターする目的の統計]」を入力します。CLI 名を決定するには、「set statname=」と入力して Tab キーを押します。各統計の詳細は、『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「統計」を参照し、統計の名前をクリックしてください。
3. 「set limit=[number は目的のしきい値]」を入力します。
4. 「commit」と入力します。このしきい値警告の警告アクションをあとから追加する場合は、「監視」識別子であるしきい値 ID をメモします。

5. 「list」と入力して、新しいしきい値警告の名前や番号を特定します。設定したばかりの制限と統計名を持つしきい値を探します。
6. 「select threshold-[number は前のステップで特定した番号と同じもの」を入力します。
7. 「list」と入力します。必要に応じて、ここで引数を修正します。デフォルトでは、最小の送信、頻度の引数、およびクリアの最小値の引数は、5分に設定されています。これは、しきい値を5分以上の間違反した場合にのみ警告が送信され、5分ごとに再送信され、しきい値が5分間クリアされたあとに送信されることを意味します。
8. 「done」と入力し、続いてもう一度「done」と入力します。

## ▼ 警告アクションの追加

1. configuration alerts actions コンテキストを入力して、create コマンドを入力します。
2. 「get category = (unset)」を入力して「category」プロパティに移動します。
3. set category=thresholds」を入力します。
4. 「set thresholdid=[id は自動的に作成されたしきい値警告の識別子である ID」を入力します。
5. 「commit」と入力します。
6. 「list」と入力して、新しい警告アクションの名前や番号を特定します。アクションおよびハンドラが割り当てられていないしきい値を探します。
7. 「select actions-[number は前のステップで特定した番号と同じもの」を入力します。
8. 「action」、続いて「get」を入力します。
9. デフォルトでは、警告タイプは電子メールです。その選択内容でよければ、スキップして次のステップに進みます。選択を変更する場合は、「set handler=[type は snmptrap、syslog、resumdataset、suspenddataset、resumeworksheet、suspendworksheet、またはexecuteworkflow」を入力します。続いて「get」と入力して、必要な引数を表示します。引数を持たないのは、「snmptrap」および「syslog」のみです。
10. 必要なそれぞれの引数を設定します。たとえば、電子メール警告の件名を設定するには、「set subject=[subject は目的の電子メール件名」を入力します。

11. `show` コマンドを使用して、すべての引数が入力されていることを確認します。
12. 「`commit`」、続いて「`list`」と入力します。必要に応じて、ここで引数を修正します。
13. 「`done`」と入力し、続いてもう一度「`done`」と入力します。

# ◆◆◆ 第 10 章

## クラスタ構成

---

Oracle ZFS Storage Appliance では、アプライアンスの協調的なクラスタ化がサポートされています。この方針は、クライアント側の負荷分散、適切なサイト計画、プロアクティブおよびリアクティブな保守と修復、すべての Oracle ZFS Storage Appliance に組み込まれた単一アプライアンスのハードウェア冗長性などを含む可用性拡張への統合的なアプローチの一部である場合もあります。

クラスタ化機能は、ストレージリソースへの共有アクセスに依存します。クラスタ化を構成するには、両方のヘッドを同じモデルにする必要があります。7420 (2GHz または 2.40GHz CPU) は、同じプラットフォームをベースにしているため、7420 (1.86GHz または 2.00GHz CPU) とクラスタ化できます。

### クラスタの機能と利点

重要なことは、Oracle ZFS Storage Appliance のクラスタ化実装の範囲を理解することです。「クラスタ」とは、さまざまな目的を持った数多くの異なるテクノロジーを呼ぶときに業界で使用される用語です。ここでは、2 つのアプライアンスヘッドおよびシェアストレージで構成されるメタシステムを意味し、ヘッドのどちらかで特定のハードウェアまたはソフトウェアの障害が発生した場合に、改善された可用性を提供するために使用されます。クラスタには、正確に 2 つのアプライアンスまたはストレージコントローラが含まれます。このドキュメント全体では、簡潔にヘッドと呼びます。各ヘッドには、クラスタで使用可能なセットからストレージ、ネットワーク、およびその他のリソースのコレクションを割り当てることができます。これにより、2 つの主なトポロジのいずれかを構成できます。アクティブ/アクティブとは、クラスタに 2 つ (以上) のストレージプールがあり、そのうちの 1 つのプールに各ヘッドと、そのプールに格納されたデータにクライアントがアクセスするときに使用されるネットワークリソースが割り当てられている状態を表す用語です。アクティブ/パッシブとは、単一のストレージプールに、アクティブと指定されたヘッドと、関連付けられたネットワークインタフェースが割り当てられている状態を表します。Oracle ZFS Storage Appliance では、どちらのトポロジもサポートされています。両者の区別は人為的です。つまり、両者の間にはソフトウェアまたはハードウェアの違いはなく、単にストレージプールを追加または削除することによって任意に切り替えることができます。どちらの場合も、ヘッドに障害が発生すると、他方 (そのど

ア) がすべての既知のリソースを制御し、これらのリソースに関連付けられたサービスを提供します。

ヘッドの修復中に数時間または数日の停止時間が発生する代わりに、クラスタ化を使用するとピアアプライアンスは修復または交換の実行中でもサービスを提供できます。さらに、クラスタではソフトウェアの順次アップグレードもサポートされています。これにより、新しいソフトウェアへの移行に関連する業務の混乱を減らすことができます。一部のクラスタ化テクノロジーには、可用性拡張機能を超える特定の追加機能が備わっています。Oracle ZFS Storage Appliance のクラスタ化サブシステムは、このような機能を提供するように設計されていません。特に、複数ヘッド間における負荷分散の提供、ストレージ障害における可用性の改善、クライアントへの複数アプライアンス間における統合されたファイルシステム名前空間の提供、または障害回復目的での広範囲な領域におけるサービス責任の分割は行われません。これらの機能も同様に、このドキュメントの範囲外です。ただし、Oracle ZFS Storage Appliance およびそれが提供するデータプロトコルでは、次のように可用性を改善できる数多くのその他の機能と方針がサポートされています。

- データの[第13章「レプリケーション」](#)、1 つ以上の地理的に離れたサイトでの障害回復で使用可能、
- クライアント側でのデータのミラー化、任意に配置された複数のストレージサーバーで提供される冗長 [208 ページの「iSCSI」](#) LUN を使用して実行可能、
- 負荷分散、[202 ページの「NFS」](#) プロトコルに組み込まれ、外部ハードウェアまたはソフトウェアによってほかの一部のプロトコルに提供可能 (読み取り専用データに適用)、
- 冗長ハードウェアコンポーネント (電源装置を含む)、ネットワークデバイス、およびストレージコントローラ、
- 『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「[問題](#)」ソフトウェア、障害が発生したコンポーネントの特定、サービスからの削除、および正常なハードウェアの修復または交換を行う技術者への指針の提供が可能、
- ネットワークファブリック冗長性 (LACP および [270 ページの「IPMP」](#) 機能で提供)
- 冗長ストレージデバイス (RAID)。

ほかの可用性機能についての追加情報は、このドキュメントの適切なセクションを参照してください。

## クラスタのデメリット

クラスタ化とスタンドアロンのどちらの Oracle ZFS Storage Appliance 構成を選択するのかを決定する際は、クラスタ化操作のコストと利点を比較検討することが重要になりま



す。クラスタ化を自動的な構造決定であると見なすことが IT 業界全体の常識ですが、この認識はこの業界の一部のベンダーによって広められた、クラスタ化のリスクと利益の理想像を反映しています。クラスタ化では、2 番目のヘッドに伴う明らかに高い先行投資および現在進行中のハードウェアおよびサポートのコストに加えて、技術上および運用上のリスクも課されます。このようなリスクの一部は、すべての担当者にクラスタ化操作の訓練を徹底的に行うことによって軽減できます。その他のリスクは、クラスタ化操作の概念に固有のもので、リスクには次のものがあります。

- テイクオーバー中のプロトコル依存の動作で、アプリケーションの耐性がなくなる可能性がある
- クラスタソフトウェア自体で障害が発生したり、スタンドアロン操作では発生しないような障害が別のサブシステムで発生したりする可能性がある
- 管理の複雑さが増し、管理タスクの実行時にオペレータエラーが発生する可能性が高い
- 多重障害や重大なオペレータエラーにより、スタンドアロン構成では発生しないようなデータ損失や破損が発生する可能性がある
- 予期しないソフトウェアやハードウェアの状態から回復することがより困難になる。

このようなコストやリスクは基本的なものであり、さまざまな方法で市場のすべてのクラスタ化製品またはクラスタ対応製品 (Oracle ZFS Storage Appliance を含む) に適用されるため、完全に除去したり軽減したりできません。ストレージ設計者は、クラスタ化の主な利点 (まれに発生する壊滅的なハードウェアまたはソフトウェアの障害時に使用できなくなる期間を数時間から数分以下に削減できること) について比較検討する必要があります。コストと利点の分析において、Oracle ZFS Storage Appliance の配備でクラスタ化を使用することが優先されるかどうかは、SLA 条件などのローカル要素、対応可能なサポート担当者とその資格、予算の制約、さまざまな障害が発生する可能性、および可用性を拡張するための代替方針の適切性によって異なります。これらの要素はサイト、アプリケーション、および業務に大きく依存するため、個別に評価する必要があります。このセクションの残りの資料について理解すれば、統合されたストレージインフラストラクチャーの設計および導入時に適切な選択を行うことができます。

## クラスタの用語

ここで定義する用語は、ドキュメント全体で使用されます。ほとんどの場合は、従来よりも広いコンテキストで詳細に説明されており、幅広い概念が含まれています。クラスタ状態およびリソースタイプについては、次のセクションで説明します。必要に応じて、このセクションに戻って参照してください。

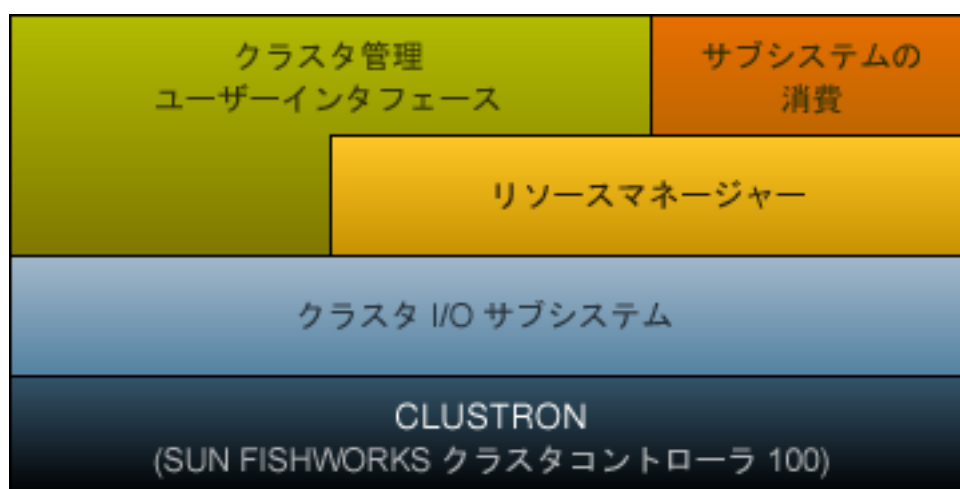
- エクスポート: 特定のヘッド上でリソースを非アクティブにするプロセス

- フェイルバック: AKCS\_OWNER 状態から AKCS\_CLUSTERED に移行するプロセス (すべての外部リソース (ピアに割り当てられたリソース) はエクスポートされてから、そのピアによってインポートされる)
- インポート: 特定のヘッド上でリソースをアクティブにするプロセス
- ピア: クラスタ内のその他のアプライアンス
- 再度参加: リソースマップをピアから取得して再同期すること
- リソース: 1 つまたは両方のヘッドに存在する物理または仮想オブジェクト (アクティブの可能性あり)
- テイクオーバー: AKCS\_CLUSTERED または AKCS\_STRIPPED 状態から AKCS\_OWNER に移行するプロセス (すべてのリソースがインポートされる)

## クラスタ化の理解

シリーズに組み込まれたクラスタ化サブシステムは、3 つの主要な構成単位で構成されています (図 1 を参照)。クラスタ I/O サブシステムおよびハードウェアデバイスは、クラスタ内でのヘッド間通信用のトランスポートを提供し、ピアの状態のモニタリングを受け持ちます。このトランスポートは、リソースマネージャーによって使用されます。これにより、データサービスプロバイダおよびほかの管理サブシステムがクラスタ化システムと接続できます。最後に、クラスタ管理ユーザーインターフェースは、設定タスク、リソースの割り当て、モニタリング、およびテイクオーバーとフェイルバックの操作を提供します。これらの各構成単位については、以降のセクションで詳細に説明します。

図 10-1 クラスタ化サブシステム



## クラスタ相互接続 I/O

すべてのヘッド間通信は、CLUSTERON ハードウェアで提供される 3 つのクラスタ I/O リンクのいずれかで転送される 1 つ以上のメッセージで構成されます (次の図を参照)。このデバイスは、2 つの低速シリアルリンクと 1 つの Ethernet リンクを提供します。シリアルリンクを使用すると、信頼性が高まります。極端に負荷が高い状態のシステムでは、Ethernet リンクが十分な速さでサービスを提供できない可能性があります。不正な障害検出および不要なテイクオーバーは、クラスタ化されたシステムが負荷に対応するための最悪の手段です。テイクオーバー中に、リクエストは処理されず、代わりにクライアントによってキューに入れられます。これにより、すでに高負荷な状態に加えて、テイクオーバー後に遅延したリクエストが大量に発生します。Oracle ZFS Storage Appliance で使用されるシリアルリンクは、このような障害モードの影響を受けにくくなっています。Ethernet リンクは、再度参加時の同期などのハートビート以外のメッセージに対して、より高パフォーマンスのトランスポートを提供し、バックアップハートビートを提供します。

3 つのリンクはすべて、通常の EIA/TIA-568B (8 線式、ギガビット Ethernet) ストレートケーブルを使用して形成されます。2 つの同一コントローラ間でストレートケーブルを使用するには、下記の配線に関するセクションで示すように、ケーブルを使用して 2 つのコネクタの反対のソケットに接続する必要があります。

図 10-2 ZS3-2 コントローラのクラスタ I/O ポート



表 10-1 ZS3-2 コントローラのクラスタ I/O ポート

図の説明			
1 シリアル 0	2 シリアルアクティビティ-LED	3 シリアルステータス LED	4 Ethernet

図の説明		
5 シリアル 1	6 Ethernet ステータス LED	7 Ethernet アクティビティ LED

図 10-3 ZS3-4 および 7x20 コントローラのクラスタ I/O ポート

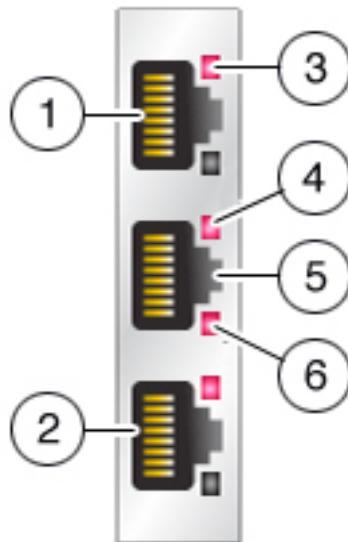


図 2。ZS3-4 および 7x20 コントローラのクラスタ I/O ポート

表 10-2 ZS3-4 および 7x20 コントローラのクラスタ I/O ポート

図の説明			
1 シリアル 1	2 シリアル 0	3 シリアルステータス LED	4 Ethernet ステータス LED
5 Ethernet	6 Ethernet アクティビティ LED		

クラスタ化されたヘッドが相互に通信を行うのは、クラスタ相互接続により確立されたセキュアなプライベートネットワーク経由だけで、サービスや管理向けのネットワークインタフェース経由では決して相互に通信を行いません。メッセージは、リモートヘッドの障害を検出するときに使用される通常のハートビートと、リソースマネージャーおよびクラスタ管理サブシステムに関連する、より高いレベルのトラフィックの 2 つの一般的なカテゴリに分類されます。ハートビートは 3 つのすべてのリンク上で送信 (および予期) されます。ハートビート

は一定の間隔で継続的に転送され、すべてのハートビートが同一であり、一意の情報が含まれないときは肯定応答も再転送も行われません。その他のトラフィックは、任意のリンク (通常は転送時に最速のリンク) 上に送信できます。このトラフィックでは、より高レベルのソフトウェアで信頼できる転送を保持するために、必要に応じて肯定応答、検証、および再転送が行われます。

メッセージのタイプまたは送信元に関係なく、すべてのメッセージは単一の 128 バイトパケットとして送信され、1 - 68 バイトのデータペイロード、およびデータ整合性を確認するための 20 バイトの検証ハッシュが含まれます。シリアルリンクは 115200 ビット/秒で動作し、9 データビットおよび単一のスタートビットとストップビットが含まれます。Ethernet は 1G ビット/秒で動作します。したがって、シリアルリンクにおける効率的なメッセージ待機時間は約 12.2 ミリ秒です。Ethernet の待機時間は大幅に変動します。一般的な待機時間は数マイクロ秒ですが、アプライアンス管理ソフトウェアにとって効率的な待機時間は、システム負荷のためにより大きくなる可能性があります。

通常、ハートビートメッセージは、3 つのすべてのクラスタ I/O リンクで各ヘッドによって 50 ミリ秒間隔で送信されます。メッセージ受信の失敗は、200 ミリ秒 (シリアルリンク) または 500 ミリ秒 (Ethernet リンク) 後にリンク障害と見なされます。3 つのすべてのリンクで障害が発生した場合、ピアで障害が発生したと想定され、テイクオーバーアービトラクションが実行されます。パニックの場合は、パニックが発生したヘッドが単一の通知メッセージを各シリアルリンク上に転送します。ほかのリンクの状態に関係なく、そのピアではすぐにテイクオーバーが開始されます。このような特徴を考慮すると、通常、クラスタ化サブシステムは次の時間内にそのピアに障害が発生したことを検出できます。

- 550 ミリ秒 (ピアが応答を停止した場合、または電源を喪失した場合)、または
- 30 ミリ秒 (ピアでオペレーティングシステムのパニックを引き起こす致命的なソフトウェアエラーが発生した場合)。

このセクションで説明する値はすべて固定値です。Oracle ZFS Storage Appliance はアプライアンスとして、これらのパラメータを調整する機能を提供しません (必要ありません)。これらの値は、実装の詳細と見なされており、ここでは情報目的でのみ提供されます。いつでも通知なしで変更されることがあります。

---

**注記** - クラスタの物理的な再配置のあとのデータ破損を回避するために、すべてのクラスタ配線が新しい場所に正しく設置されていることを確認してください。詳細は、[178 ページの「スプリットブレイン」状態の回避](#)を参照してください。

---

## クラスタリソース管理の理解

リソースマネージャーは、適切なネットワークインタフェースセットが plumb され、適切なストレージプールがアクティブであり、数多くの構成パラメータがクラスタ化された 2 つのヘッド

間で同期化された状態であることを確認する役割があります。このサブシステムのアクティビティの大部分は、管理者には表示されません。ただし、重要な側面が 1 つ表示されます。リソースは、インポート (アクティブ化) されるタイミングおよびインポート (アクティブ化) するかどうかによって複数のタイプに分類されます。アクティブの定義は、リソースクラスによって異なることに注意してください。たとえば、ネットワークインタフェースはネットクラスに属し、インタフェースが起動したときにアクティブになります。もっとも重要な 3 つのリソースタイプは、シングルトン、プライベート、およびレプリカです。

レプリカはもっとも単純です。管理者には表示されず、クラスタ構成画面にも表示されません (図 4 を参照)。レプリカは常に存在し、両方のヘッドで常にアクティブです。一般に、これらのリソースは単純に、2 つのヘッド間で同期化する必要があるサービスプロパティ用のコンテナとして動作します。

レプリカと同様に、シングルトンリソースでは状態が同期化されます。ただし、シングルトンは正確に 1 つのヘッドで常にアクティブになっています。管理者は、各シングルトンが通常アクティブになっているヘッドを選択できます。該当ヘッドに障害が発生すると、そのピアでシングルトンがインポートされます。シングルトンは、クラスタ化の可用性特性にとって重要です。シングルトンは、障害が発生したヘッドから動作しているピアに移動すると一般に考えられているリソースであり、ネットワークインタフェースおよびストレージプールが含まれます。ネットワークインタフェースはクライアントが既知のストレージサービスセットを検索するときに使用する IP アドレスのコレクションであるため、該当するインタフェースのアドレスへのアクセス時にクライアントに表示されるであろうストレージプールと同じヘッドに、各インタフェースが割り当てられることが重要となります。図 4 では、PrimaryA インタフェースに関連付けられたすべてのアドレスは、常にインポート済みのプール 0 を含むヘッドで提供されます。一方、PrimaryB インタフェースに関連付けられたすべてのアドレスは、常にプール 1 と同じヘッドで提供されます。

プライベートリソースは、割り当てられたヘッドにのみ認識され、障害発生時にテイクオーバーされません。一般に、これはネットワークインタフェースでのみ役立ちます。特定のユースケースについては、以降の説明を参照してください。

図 10-4 ZS3-2 のクラスタ化の例



リソースタイプは、ほかにも複数存在します。これらは、管理者には表示されない実装の詳細です。このようなタイプの 1 つがシンビオートです。このリソースでは、インポートおよびエクスポートされたときに別のリソースに従うことができます。このリソースタイプのもっとも重要な使用法は、ストレージプールのディスクおよびフラッシュデバイスを表現するときです。このようなリソースはディスクセットと呼ばれ、リソースが含まれる ZFS プールよりも前に常にインポートされる必要があります。各ディスクセットは、外部ストレージ格納装置内のディスクの半分で構成されます。クラスタ化されたストレージシステムには、任意の数のディスクセットを接続でき（ハードウェアサポートによって異なる）、各 ZFS プールは 1 つ以上のディスクセット内のストレージデバイスで形成されます。ディスクセットには ATA デバイスが含まれる場合があるため、マルチパス環境で使用される ATA デバイスに固有のアフィリケーション関連の動作を回避するには、明示的にインポートおよびエクスポートする必要があります。ディスクをリソースとして表現することは、これらのアクティビティを適切なときに実行するための簡単な方法です。管理者がストレージプールの所有権を設定または変更すると、同時にこれに関連付けられたディスクセットの所有権の割り当ても透過的に変更されます。すべてのシンビオートと同様に、ディスクセットリソースはクラスタ構成ユーザインタフェースには表示されません。

表 10-3 クラスタリソース管理

リソース	アイコン	偏在	障害時のテイクオーバー
シングルトン		いいえ	はい
レプリカ	なし	はい	該当なし
プライベート		いいえ	いいえ
シンビオート	なし	親タイプと同じ	親タイプと同じ

新規リソースが作成されると、まずリソースが作成されるヘッドに割り当てられます。ヘッドが AKCS\_OWNER 状態でないかぎり、この所有権は変更できません。したがって、通常はリソースを所有するヘッド上にリソースを作成するか、またはリソースの所有権を変更する前にテイクオーバーする必要があります。一般に、どちらか一方のヘッドからリソースを削除できますが、エクスポートされたストレージプールを削除することはできません。通常、割り当てられた所有者がどちらのヘッドであるかに関係なく、現在管理されているヘッド上のリソースを削除すると最適な結果が得られます。

大部分の構成設定 (サービスプロパティ、ユーザー、ロール、アイデンティティマッピング規則、SMB 自動ホーム、iSCSI イニシエータ定義など) は、両方のヘッドで自動的にレプリケートされます。したがって、クラスタの状態に関係なく、これらの設定を両方のヘッドで構成する必要はありません。構成が変更されたときに一方のアプライアンスが停止した場合は、サービスを提供する前に、次のブートでクラスタに再度参加するときに他方のアプライアンスにレプリケートされます。次のような例外が多少あります。

- シェアおよび LUN の定義およびオプションは、そのプールが通常割り当てられているヘッドに関係なく、基になるプールを制御するヘッドにのみ設定できます。
- 「アイデンティティ」サービスの構成 (アプライアンス名や場所など) はレプリケートされません。
- シャーシに指定された名前は、割り当てられたヘッドにのみ表示されます。
- 各ネットワークルートは、特定のインタフェースにバインドされます。各ヘッドに特定のサブネット内のアドレスを持つインタフェースが割り当てられ、そのサブネットにアプライアンスがトラフィックを制御するためのルーターが存在する場合は、同じゲートウェイアドレスが使用されている場合でも、インタフェースごとにルートを作成する必要があります。これにより、基になるネットワークリソースが 2 つのヘッド間をシフトするときに、各ルートが個別にアクティブになることが可能になります。詳細は、ネットワークの考慮点を参照してください。
- SSH ホスト鍵はレプリケートされず、シェアも行われません。したがって、プライベート管理インタフェースが構成されていない場合、障害が発生したノードに割り当てられたアドレスを使用して CLI にログインしようとすると、鍵が一致しない可能性があります。同じ制限が、BUI へのアクセスに使用される SSL 証明書に適用されます。





そのあと、共通構成が透過的にレプリケートされ、管理者が各アプライアンスヘッドにリソースのコレクションを割り当てるのが基本モデルとなります。次に、これらのリソース割り当てによって、クライアントに表示されるストレージリソースにネットワークアドレスがバインドされます。どちらのアプライアンスがリソースのコレクションを制御するのかに関係なく、クライアントは予期されるネットワークの場所で必要なストレージにアクセスできます。



## クラスタのテイクオーバーとフェイルバック

クラスタ化されたヘッドノードは、常に次の状態のいずれかになります。

表 10-4 クラスタの状態

状態	アイコン	CLI/BUI の表示	説明
UNCONFIGURED		クラスタ化が構成されていません	まったくクラスタ化されていないシステムは、この状態になります。システムが設定中であるか、またはクラスタ設定タスクがまだ完了していません。
OWNER		アクティブ (テイクオーバーが完了しました)	クラスタ化が構成され、このノードでクラスタ内のすべてのシェアリソースが制御されます。クラスタ設定がユーザーインターフェースから完了した直後、およびそのピアに障害が発生したことが検出されたとき (テイクオーバー後など) に、システムはこの状態に移行します。管理者が手動でフェイルバック操作を実行するまで、この状態のままです。
STRIPPED		準備完了 (フェイルバックを待機中)	クラスタ化が構成され、このノードではシェアリソースが制御されません。クラスタ設定がほかのノードのユーザーインターフェースから完了した直後、またはリブート、電源切断、その他の障害が発生したあとに、システムはSTRIPPED になります。管理者が手動でフェイルバック操作を実行するまで、ノードはこの状態のままです。
CLUSTERED		アクティブ	クラスタ化が構成され、両方のノードがリソース割り当てに応じてシェアリソースを所有しています。各ノードが ZFS プールを所有し、CLUSTERED

状態	アイコン	CLI/BUI の表示	説明
-		クラスタに再度参加しています...	状態である場合、2 つのノードは「アクティブ/アクティブ」クラスタと一般に呼ばれる状態になります。
-		不明 (切断または再起動中)	ピアアプライアンスの電源が切られているか、ブート中であるか、クラスタ相互接続リンクがすべて停止しているか、またはクラスタ化がまだ構成されていません。

これらの状態間の遷移は、2 つの操作 (テイクオーバーとフェイルバック) の一部として発生します。

テイクオーバーは常時発生する可能性があります。前述のとおり、テイクオーバーはピアの障害が検出されるたびに試みられます。クラスタ構成 CLI または BUI を使用して、手動でトリガーすることもできます。これは、テスト目的でも、順次ソフトウェアアップグレードの実行 (1 番目のヘッドはほかのヘッドが古いソフトウェアを実行するサービスを提供している間にアップグレードされ、2 番目のヘッドは新しいソフトウェアが検証されたあとにアップグレードされる) でも役立ちます。最後に、テイクオーバーはヘッドがブートし、そのピアが存在しないことを検出すると発生します。これにより、通常は片方のヘッドで永続的に障害が発生したり、両方のヘッドが一時的に電源を喪失したりするときにサービスを再開できます。

フェイルバックは自動的には発生しません。障害が発生したヘッドが修復されブートされると、そのヘッドはクラスタに再度参加し (すべてのリソースのビュー、プロパティ、および所有権の再同期化)、管理者がフェイルバック操作を実行するまで待機を継続します。そのときまで、元の動作しているヘッドはすべてのサービスの提供を継続します。これにより、ヘッドが本稼動サービスに戻る前に、最初にテイクオーバーをトリガーした問題の詳細な調査、新規ソフトウェアリビジョンの検証、またはその他の管理タスクを実行できます。フェイルバックはクライアントに大きな影響を与えるため、業務固有の要件およびプロセスに応じてスケジュールするようにしてください。例外が 1 つあります。ヘッド A に障害が発生して、ヘッド B がテイクオーバーしたと想定します。ヘッド A がクラスタに再度参加すると、ヘッド B が存在しないか、障害が発生したことが検出された場合にテイクオーバーの対象になり

ます。原則として、元の問題を調査する機会がない場合でも、サービスを提供しないよりは提供の方が適切です。したがって、以前に障害が発生したヘッドへのフェイルバックは自動的に発生しませんが、テイクオーバーはいつでも実行できます。

クラスタを設定すると、初期状態は設定を OWNER 状態で開始したノードと、STRIPPED 状態のその他のノードで構成されます。初期フェイルバック操作を実行して、STRIPPED ノードにシェアリソースのその部分を渡すと、両方のノードが CLUSTERED 状態になります。両方のクラスタノードに障害が発生するか、電源が切られると、同時起動時にノードが調停され、一方が OWNER 状態になり、他方が STRIPPED 状態になります。

フェイルバック中に、すべての外部リソース (ピアに割り当てられたリソース) がエクスポートされ、そのピアによってインポートされます。障害が発生したためにインポートできないプールでは、STRIPPED ノードのリポートがトリガーされます。障害が発生したプールでフェイルバックを試みると、インポート失敗の結果として STRIPPED ノードがリポートする可能性があります。

## クラスタ化された環境での構成変更

大部分のアプライアンス構成は、サービスプロパティとシェア/LUN プロパティのどちらかで表現されます。シェアおよび LUN プロパティはストレージプール自体のユーザーデータとともに格納されます (したがって、そのストレージリソースの現在の所有者に常にアクセス可能です) が、サービス構成は各ヘッド内に格納されます。両方のヘッドが一貫性のあるサービスを確実に提供するためには、変更が発生したとき、または以前に停止したヘッドがそのピアに再度参加するときに、すべてのサービスプロパティが同期化される必要があります。すべてのサービスはレプリカリソースで表現されるため、どちらかのヘッドでプロパティが変更されるたびに、この同期化はアプライアンスソフトウェアによって自動的に実行されます。

したがって、管理者が構成の変更をレプリケートする必要はありません (実際に冗長になります)。標準の操作手順では、この属性が反映され、初期クラスタ構成が完了したら 2 つのヘッドのどちらかにのみ変更を加えることが求められます。初期クラスタ構成のプロセスでは、既存のすべての構成が新規に構成されたピアにレプリケートされることにも注意してください。一般に、クラスタ化された構成変更では、次の 2 つのベストプラクティスがあります。

- 基になるストレージまたはネットワークインタフェースリソースを現在制御している (新規リソースの作成中の場合は、制御する予定の) ヘッド上で、すべてのストレージおよびネットワーク関連の構成変更を行います。
- 両方のヘッドではなく、どちらかのヘッド上で、その他のすべての変更を行います。サイトポリシーでは、この目的でマスターと見なされるヘッドを指定すべきです。また、ど

ちらのヘッドが機能しているか、および構成されているストレージプールの数に依存するべきです。アプライアンスソフトウェアでは、この区別が行われないことに注意してください。

アムネジア (ピアが機能していない間に個々の構成変更が行われた結果、各ヘッドでの変更が失われる) の問題は、大幅に誇張して説明されています。これは、各ヘッド上のシステム構成に個々の変更を行うメカニズムが存在しない Oracle ZFS Storage Appliance で特に当てはまります。このような単純化によって、集中管理された構成リポジトリの必要性と、単純なアプローチに対する議論が大幅に緩和されます。現在どちらのヘッドが動作していても、適切な構成であると想定され、そのピアがブート時に同期化されます。今後の製品拡張では、構成の不一致を解決するための代替ポリシーを選択できるようになりますが、この基本アプローチが簡単で、理解のしやすさも備えています。つまり、2 番目のヘッドでは、既存の本稼動システムですでに使用されている構成パラメータセットが採用されます (したがって、適切である可能性が高くなります)。確実に適切な状態を維持するために、管理者はクラスタが修復されたらすぐに、障害が発生したヘッドがクラスタに再度参加することを確認してください。

## クラスタ化におけるストレージの考慮点

Oracle ZFS Storage Appliance をクラスタで使用するようにサイズ変更する際は、さらに 2 つの考慮点が重要となります。おそらく、もっとも重要な決定は、すべてのストレージプールの所有権を同じヘッドに割り当てるか、ストレージプール間で分割するかです。ここでは、次の表に示すようなトレードオフがあります。一般に、公称動作中のスループットを最適化する場合や、フェイルオーバー後のパフォーマンスに問題がない場合を除いて、プールは単一のヘッドに構成する必要があります。フェイルオーバーした状態時のパフォーマンス特性の正確な変更は、ワークロードの性質やサイズに大きく依存します。一般に、ヘッドが特定の軸で最大パフォーマンスに近づくほど、そのヘッドのピアでワークロードをテイクオーバーしたときの、その軸におけるパフォーマンス低下が大きくなります。当然、複数のプールがある場合は、この低下が両方のワークロードに適用されます。

どちらかの構成でも、任意の ReadZilla デバイスを使用できるのは、そのデバイスが割り当てられたプールが、そのプールの所有権が割り当てられたヘッド上にインポートされている場合だけであることに注意してください。つまり、ヘッドの障害のためにプールでテイクオーバーが発生すると、インポート済みのヘッドに未使用の ReadZilla デバイスがインストールされている場合でも、そのプールでは読み取りキャッシュができなくなります。このため、「アクティブ/パッシブ」クラスタでの ReadZilla は、[第5章「ストレージ構成」](#)のドキュメントの説明どおりに構成する必要があります。これは、LogZilla デバイスには適用されません。このデバイスはストレージファブリックに配置され、どちらのヘッドがプールをインポートしても常にアクセス可能です。

表 10-5 クラスタ化におけるストレージの考慮点

変数	単一ノードの所有権	異なるヘッドで所有される複数のプール
合計スループット (通常の操作)	いつでも最大で合計 CPU リソースの 50%、DRAM の 50%、および合計ネットワーク接続の 50% をサービスの提供に使用できます。この方法は単純です。単一のヘッドのみがクライアントリクエストを処理するため、他方のヘッドはアイドル状態です。	いつでも全 CPU および DRAM リソースをサービスの提供に使用できます。いつでも最大で全ネットワーク接続の 50% を使用できます (フェイルオーバーをサポートするには、各ヘッドにダークネットワークデバイスが必要です)。
合計スループット (フェイルオーバー)	公称動作に関連するスループットの変更はありません。	動作しているヘッドのリソースの 100% がサービスの提供に使用されます。公称動作に関連する合計スループットは、公称動作中の使用率に応じて、およそ 40% から 100% の範囲です。
I/O 待機時間 (フェイルオーバー)	フェイルオーバー操作中は、Read Zilla を使用できません。これにより、使用可能な読み取りキャッシュに適合する読み取りの多いワークロードの待機時間が大幅に増加する可能性があります。書き込み操作の待機時間は影響を受けません。	フェイルオーバー操作中は、Read Zilla を使用できません。これにより、使用可能な読み取りキャッシュに適合する読み取りの多いワークロードの待機時間が大幅に増加する可能性があります。ヘッドリソースの競合が大きくなると、読み取り操作と書き込み操作の両方の待機時間が増加する可能性があります。これは、通常のヘッドではなく、動作しているヘッドで 2 つのワークロードが実行されると発生します。各ヘッドでの公称ワークロードがヘッドの最大性能に達すると、フェイルオーバー状態の待機時間が非常に長くなる可能性があります。
ストレージ柔軟性	使用可能な物理ストレージはすべて、シェアおよび LUN で使用できます。	特定のプールに割り当てられたストレージのみが、そのプールのシェアおよび LUN で使用できます。ストレージはプール間でシェアされないため、他方のプールには空き領域があるときに 1 つのプールが満杯になると、一部のストレージが無駄になる可能性があります。
ネットワーク接続	各ヘッド上のすべてのネットワークデバイスを、そのヘッドがサービスを提供している間に使用できます。	各ヘッド上の全ネットワークデバイスの半分のみを、そのヘッドがサービスを提供している間に使用できます。したがって、物理的に別々の

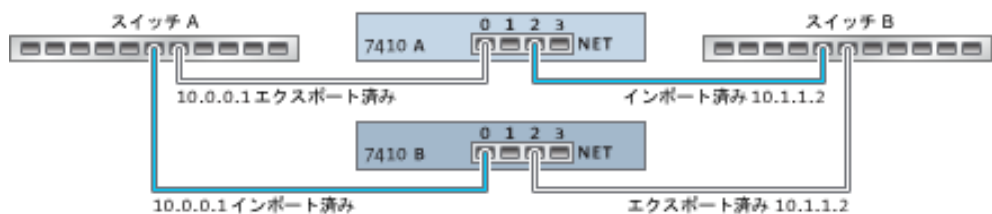
変数	単一ノードの所有権	異なるヘッドで所有される複数のプール
		ネットワークの半数にしか各プールを接続できません。

2 番目に重要なストレージの考慮点は、単一障害点なし (NSPF) でプール構成を使用することです。クラスタ化を使用することはアプリケーションが可用性を非常に重視することを意味するため、単一 JBOD の障害で可用性が失われるような方法でストレージプールを構成する正当な理由はほとんどありません。このアプローチの弱点は、NSPF 構成では、単一障害点ありの構成を行う場合よりも多数の JBOD が必要になる点です。必要な機能が非常に少ない場合は、目的の RAID レベルで NSPF を提供するのに十分な JBOD をインストールすることが経済的でない可能性があります。

## クラスタ化におけるネットワークの考慮点

ネットワークデバイス、データリンク、およびインタフェースに障害が発生しても、クラスタ化サブシステムではヘッドに障害が発生しません。アプライアンスの内部または外部で発生したネットワーク障害から保護するには、IPMP または LACP あるいはその両方を使用する必要があります。可用性に対する包括的なアプローチには、ネットワークの正しい構成、およびネットワーク全体の冗長化計画が必要です。

図 10-5 ネットワークのクラスタ化



ネットワークインタフェースに静的な IP 構成が含まれる場合は、シングルトンリソースとプライベートリソースのどちらかとして構成できます。DHCP を使用して構成したインタフェースはプライベートにする必要があり、クラスタで DHCP を使用することは推奨されていません。シングルトンリソースとして構成された場合、インタフェースの構築に使用されるすべてのデータリンクおよびデバイスは、一度に 1 つのヘッドでのみアクティブにできます。同様に、フェイルオーバー状態でサービスを提供するには、各ヘッド上の対応するデバイスを同じネットワークに接続する必要があります。この例は前の図で示しています。

ネットワークインタフェースをデバイスおよびデータリンクから構成する場合は、クラスタが正しく動作するように、各シングルトンインタフェースに同じ識別子を使用するデバイス、および両方のヘッドで使用可能な機能が含まれることが重要になります。デバイス識別子はデバイスタイプおよび最初にアプライアンスで検出される順序によって異なるため、クラスタ化されたヘッドに同じハードウェアをインストールする必要があります。両方のヘッドの各スロットには同じハードウェアを装着する必要があり、両方のヘッドにはスロットを同じ順序で装着する必要があります。公認の Oracle 再販業者またはサービス担当者は、これらの要件を満たすハードウェアアップグレードの計画を支援できます。

常に、ルートは明示的に単一のネットワークインタフェースにバインドされます。ルートはリソースマネージャ内でシンビオットとして表現され、バインド先のインタフェースが運用可能なときにのみアクティブにできます。したがって、現在スタンバイモードのインタフェースにバインドされたルート (エクスポート済み) は、そのインタフェースがテイクオーバープロセス中に有効化されるまで無効です。これは、2 つのプールが構成され、共通のサブネットで使用可能にされる場合に重要です。サブネットが、ほかの 1 つ以上のネットワークに到達するためにアプライアンスで使用されるルーターのホームであれば、個別のルート (たとえば、2 番目のデフォルトルート) を構成して、そのサブネットに接続するアクティブおよびスタンバイの各インタフェースにバインドする必要があります。

例:

- インタフェース e1000g3 は 'alice' に割り当てられ、e1000g4 は 'bob' に割り当てられています。
- 各インタフェースは 172.16.27.0/24 ネットワーク内のアドレスを持ち、172.16.27.1 経由で到達可能な 172.16.64.0/22 ネットワーク内のクライアントにサービスを提供するために使用できます。
- 172.16.27.1 経由で 172.16.64.0/22 に到達するルートを 2 つ作成する必要があります。1 つは e1000g3 にバインドし、もう 1 つは 1000g4 にバインドします。

クラスタ化された各ヘッドに、管理でのみ使用される IP アドレス (ほとんどの場合は専用の管理ネットワーク上にある) を割り当て、インタフェースをプライベートリソースとして指定することはよい方法です。これにより、AKCS\_STRIPPED 状態で、フェイルバックの待機中である場合でも、管理ネットワークから動作中のヘッドに到達できるようになります。これが重要なのは、ヘッドはサービスを提供していない場合でも、サービス (LDAP や Active Directory など) が使用中で、ほかのネットワークリソースへのアクセスが必要な場合です。これが現実的でない場合は、システムコンソールを使用してヘッドを管理できるように、信用できるネットワークまたはシリアル端末あるいはその両方にサービスプロセッサを接続する必要があります。

これらのアクションのどちらも実行しない場合は、フェイルバックが完了するまで、新規にブートしたヘッドの管理またはモニターができません。特定のストレージプール用のサービスを提供しているヘッドをモニターまたは管理することが必要な場合があります。これが役

立つ可能性が高いのは、ストレージ自体の一部を変更 (シェアプロパティの変更や新規 LUN の作成など) する必要がある場合です。これを行うには、管理タスクを実行するサービスインタフェースのいずれかを使用するか、一致するプールを管理するためにのみ使用される個別のシングルトンインタフェースを割り当てます。どちらの場合でも、管理に使用されるプールと同じヘッドにインタフェースを割り当てるようにしてください。

## プライベートのローカル IP インタフェース

プライベートのローカル IP インタフェースを作成する際は、次のガイドラインを使用してください。

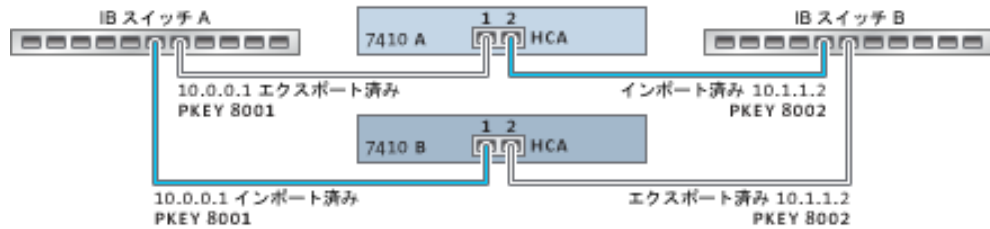
- クラスタピアのプライベート IP インタフェースと同じ名前で作成すると、プライベート IP インタフェースがローカルで作成されます。
- ピアのプライベートインタフェースが使用しているデータリンクは削除できず、削除ボタンはグレー表示されます。
- IPMP グループに属する IP インタフェースは、すべて同じタイプで、同じヘッドに属する必要があります。IPMP グループを作成する場合は、すべてシングルトンインタフェースを使用するか、すべてプライベート IP インタフェースを使用する必要があり、またクラスタノードがこれらのインタフェースの所有者である必要があります。
- IPMP グループのタイプは作成時にのみ設定され、ベースとなるリンクのタイプによって決まります。
- IP インタフェースの所有権は IPMP グループの所有権から独立して変更できないため、IPMP グループに属する IP インタフェースは「クラスタ:リソース」ページに表示されません。
- プライベート IPMP グループのタイプまたは所有権は変更できないため、このグループは「クラスタ:リソース」ページに表示されません。

## クラスタ化における Infiniband の考慮点

Ethernet デバイス上に組み込まれたネットワークと同様に、アプライアンスの内部および外部でのネットワーク障害から保護するには、Infiniband ネットワークを冗長ファブリックトポロジの一部にする必要があります。ネットワークトポロジには、HCA、スイッチ、およびサブネットマネージャーの広範囲な冗長計画とともに、リンクレベルでのネットワーク障害から保護する IPMP を追加するようにしてください。



図 10-6 クラスタ化における Infiniband の考慮点

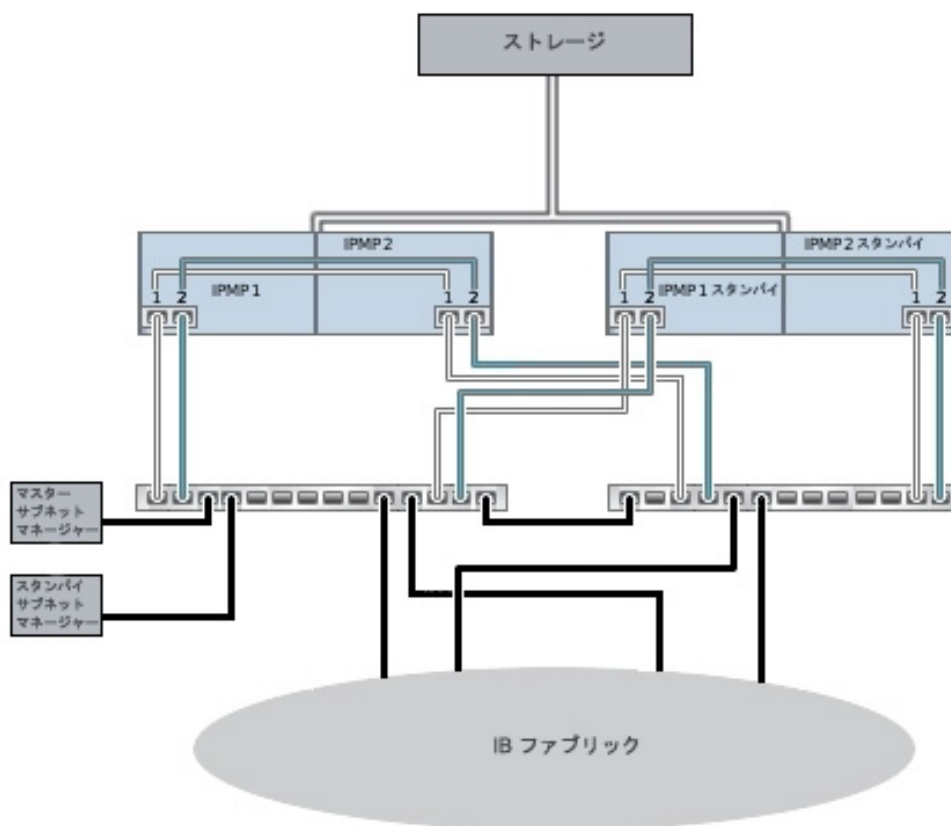


適切なクラスタ構成を確保するには、各ヘッドに同一スロットの同一 HCA を装着する必要があります。さらに、対応する各 HCA ポートを、同一メンバーシップ権限を持つサブネットワークマネージャー上の同じパーティション (pkey) に構成し、同じネットワークに接続する必要があります。複雑さを減らして適切な冗長性を確保するには、各ポートが Infiniband サブネットワーク内の 1 つのパーティションにのみ属することをお勧めします。ネットワークインタフェースに静的な IP 構成が含まれる場合は、シングルトランジソースとプライベートリソースのどちらかとして構成できます。シングルトランジソースとして構成された場合、インタフェースの構築に使用されるすべての IB パーティションデータリンクおよびデバイスは、いつでも 1 つのヘッドでのみアクティブにできます。この具体的な例が上の図に示されています。対応するポートのパーティションメンバーシップの変更は、上記のクラスタ化規則と整合性がある方法で同時に発生する必要があります。公認の Oracle 再販業者またはサービス担当者は、これらの要件を満たすハードウェアアップグレードの計画を支援できます。

## クラスタ化における冗長パスのシナリオ

次の図は、サブネットワークマネージャー冗長性のためのクラスタ構成を示しています。2 つのデュアルポート HCA をサーバスイッチの冗長ペアに接続すると、冗長性が高くなります。

図 10-7 サブネットマネージャー冗長性のためのクラスタ構成



## 「スプリットブレイン」状態の回避

クラスタ化されたシステムの一般的な障害モードは、スプリットブレインと呼ばれます。この状況では、クラスタ化された各ヘッドは、そのピアに障害が発生したと見なされ、テイクオーバーが試みられます。追加ロジックが存在しない状況では、診断や修正が困難な予期しない壊滅的な動作が広範囲にわたって発生する可能性があります。この状況は、一般的にヘッドでシェアされた通信媒体に障害が発生することが要因となります。Oracle ZFS Storage Appliance の場合は、クラスタ I/O リンクに障害が発生すると生じます。アプリケーションソフトウェアでは、組み込みトリプルリンク冗長に加えて (テイクオーバーの発生を回避するには、単一リンクのみが必要ですが)、ヘッドがテイクオーバーを継続する必要があるかどうかを決定するアービトレーション手順も実行されます。

同様の製品では、数多くのアービトレーションメカニズムが採用されています。一般には、定足数ディスク (SCSI 予約を使用) または定足数ゲーターを使用する必要があります。追加ハードウェアを必要としない ATA ディスクの使用をサポートするために、Oracle ZFS Storage Appliance は、ストレージファブリック自体に依存する別のアプローチを使用して、必要な相互排他性を提供します。アービトレーションプロセスは、ストレージファブリックに定義済みの順序で表示される各 SAS エクспанダ上での SAS ZONE LOCK コマンドの実行で構成されます。どちらかのアプライアンスで、このようなすべてのロック取得の試みに成功すると、テイクオーバーが続行されます。他方のアプライアンスは自分自身をリセットします。ブートして、そのピアが到達不能であることを検出したクラスタ化されたアプライアンスは、テイクオーバーを試みて、同じアービトレーションプロセスに移行するため、1 つ以上のクラスタ I/O リンクが復元されるまで、繰り返しリセットされます。これにより、他方のヘッドであとに発生する障害によって、機能停止が延長されることはありません。これらの SAS ゾーンのロックは、フェイルバックが実行されるか、AKCS\_OWNER 状態のヘッドがストレージファブリックへの自分のアクセスを最後に更新してから約 10 秒経過すると解除されます。

このアービトレーションメカニズムは単純で、費用がかからず、追加のハードウェアも必要ありませんが、ストレージファブリック内の 1 つ以上の共通 SAS エクспанダにアクセスしている両方のクラスタ化されたアプライアンスに依存します。通常の状態では、各アプライアンスがすべてのエクспанダにアクセスし、アービトレーションは 2 つ以上の SAS ゾーンのロックで構成されます。ただし、アプライアンスが共通のエクспанダにアクセスしない多重障害シナリオが発生する可能性があります。たとえば、2 本の SAS ケーブルが取り除かれるか、JBOD の電源が切られると、各アプライアンスはエクспанダの個々のサブネットにアクセスします。この場合、各アプライアンスは正常に到達可能なすべてのエクспанダをロックし、そのピアに障害が発生したと判断して、テイクオーバーの続行を試みます。これにより、ディスクアフィリエーションの競合や重大なデータ破損のために、回復不能なハンゲアップに至る可能性があります。

この状況の結果は重大ですが、これは多重障害の場合にしか発生しないことに注意してください (4 つ以上の障害の場合にのみ、しばしば発生します)。Oracle ZFS Storage Appliance に組み込まれたクラスタ化解決方法は、単一障害点なしを実現し、不要なコストや複雑さをシステムに追加せずに、発生しやすい障害からデータと可用性の両方を保護するように設計されています。大規模な多重障害が発生すると、サービスまたはデータあるいはその両方が失われる可能性があります。これは RAID なしのレイアウトでは無制限のディスク障害から保護できないのと同様です。

図 10-8 スプリットブレインの回避



さいわいにも、このような障害シナリオの大部分はヒューマンエラーから発生し、ハードウェアを正しくインストールして、クラスタ設定および管理のベストプラクティスの訓練をスタッフに実施すれば完全に回避できます。管理者は 3 つのすべてのクラスタ I/O リンクが接続され機能していること (図を参照)、およびすべてのストレージ配線がアプライアンスに付属するセットアップポスターで示すとおり接続されていることを常に確認する必要があります。クラスタを本稼動に移行する前およびそのあとはいつでも、各 JBOD (図を参照) への 2 つのパスが検出されることが特に重要になります。容量の増加や障害のあるコンポーネントの交換をサポートするために一時的に配線を変更する場合は、当然例外です。管理者は警告を使用して、クラスタ相互接続リンクおよび JBOD パスの状態をモニターし、障害を迅速に修正するようにしてください。適切な接続を保持すると、ハードウェアまたはソフトウェアコンポーネントに障害が発生した場合でも、可用性とデータ整合性の両方が保護されます。

図 10-9 クラスタの 2 つのパス

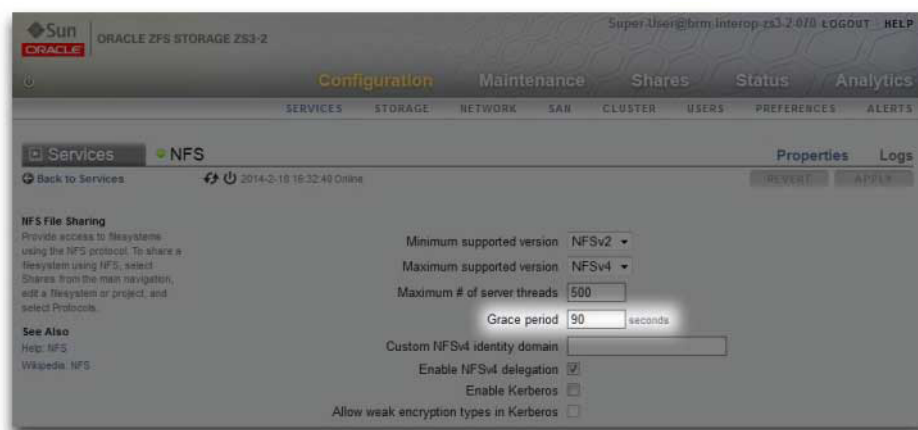


## テイクオーバーの影響の見積もりと削減

テイクオーバーおよびフェイルバックの最中には間隔があり、この間はストレージへのアクセスをクライアントに提供できません。この間隔の長さは構成によって異なり、クライアントに与える正確な影響は、データアクセスに使用しているプロトコルに依存します。これらの影響を理解して緩和することにより、正常なクラスタ配備と最悪のタイミングでのコストのかかる障害とを区別できるようになります。

一般に、NFS (すべてのバージョン) クライアントは機能停止をアプリケーションソフトウェアから非表示にします。これにより、サーバーが使用不可の間は I/O 操作が遅延します。NFSv2 および NFSv3 は、サービス復元のほぼ直後に回復するステートレスプロトコルです。NFSv4 は起動時にクライアント猶予期間を組み込み、一般にこの間は I/O を実行できません。Oracle ZFS Storage Appliance では、この猶予期間を調整できます (図を参照)。猶予期間を短縮すると、テイクオーバーまたはフェイルバックあるいはその両方の明らかな影響が少なくなります。計画的な停止の場合、Oracle ZFS Storage Appliance は、NFSv4 クライアントに対して猶予なしの回復を提供するため、猶予期間の遅延が回避されます。猶予なしの回復の詳細は、NFS 202 ページの「プロパティ」の猶予期間プロパティを参照してください。

図 10-10 クラスタの猶予期間



サービス中断中の iSCSI 動作はイニシエータに依存しますが、一般に、サービスがクライアント固有のタイムアウト期間内に復元されると、イニシエータは回復します。詳細は、イニシエータのドキュメントを確認してください。一般に、iSCSI ターゲットはテイクオーバーの完了直後に、追加の遅延なしでサービスを提供できます。

SMB、FTP、および HTTP/WebDAV はコネクション型のプロトコルです。これらのサービスに関連付けられたセッション状態は、基になるストレージおよびネットワーク接続とともに転送できないため、これらのプロトコルのいずれかを使用するすべてのクライアントはテイクオーバーまたはフェイルバック中に切断され、操作の完了後に再接続する必要があります。

複数の要素でテイクオーバー時間（およびこれに密接に関連するフェイルバック時間）が影響を受けますが、大部分の構成で、これらの時間はディスクセットリソースのインポートに必要な時間で占められます。ディスクセットごとの一般的なインポート時間は、15 秒から 20 秒の範囲です（ディスクセット数にリニアに比例します）。ディスクセットは 1 つの JBOD の半分で構成される（その半分の JBOD のディスクベイが装着され、ストレージプールに割り当てられている場合）ことを思い出してください。未割り当てのディスクおよび空のディスクベイは、テイクオーバー時間に影響を与えません。ディスクセットリソースのインポートにかかる時間は、管理者によって調整または変更可能なパラメータの影響を受けません。したがって、クラスタ化された配備を計画している管理者は、次のいずれかを行うようにしてください。

- 関連するテイクオーバー時間をクライアントが許容できるように、インストール済みのストレージを制限する

- 予測される最大テイクオーバー時間を超えるようにクライアント側タイムアウト値を調整する。

通常はディスクセットのインポートは大部分がテイクオーバー時間で構成されますが、唯一の要素ではないことに注意してください。プールのインポートプロセス中に、インテントログの記録を再実行する必要があり、各シェアおよび LUN は適切なサービスを使用してシェアする必要があります。単一のシェアまたは LUN におけるこれらのアクティビティの実行に必要な時間の合計は、非常に短い時間 (数十ミリ秒程度) ですが、シェア数が増えると、テイクオーバー時間に大きく影響する可能性があります。したがって、シェア数を比較的少なく (数千個以下に) 保持すると、これらの時間を大幅に短縮できます。

通常、どのような構成でも、フェイルバック時間はテイクオーバー時間よりも長くなります。これは、フェイルバックが次のような 2 段階の操作であるためです。まず、ソースアプライアンスが、割り当てられた所有者以外のすべてのリソースをエクスポートします。次にターゲットアプライアンスが、独自に割り当てられたリソースでのみ標準テイクオーバー手順を実行します。したがって、障害発生時には、常にヘッド A からヘッド B へのフェイルバックの方がヘッド B からヘッド A へのテイクオーバーよりも時間がかかります。この追加のフェイルバック時間は、テイクオーバー時間に比べるとエクスポートされるディスクセットの数に依存しません。したがって、シェアおよび LUN の数を少なく維持すると、フェイルバックへの影響はテイクオーバーよりも大きくなる可能性があります。フェイルバックは常に管理者によって開始される点に留意することも重要です。したがって、フェイルバックにより発生するサービス中断が長くなる場合は、業務の混乱がもっとも少ない時間にスケジュールできます。

注: このセクションに記載した推定時間は、ソフトウェア/ファームウェアバージョン 2009.04.10.1-0 の場合のもので、その他のバージョンでの実行は異なり、実際のパフォーマンスは異なる可能性があります。クラスタ化されたアプライアンスを本番環境に配備する前に、テイクオーバーおよびクライアントアプリケーションへの正確な影響をテストすることが重要です。

## BUI を使用したクラスタ構成

クラスタの構成または構成解除を行うには、次の手順を使用します。

クラスタ化の構成解除は、クラスタ化されたストレージコントローラのいずれかを出荷時のデフォルト構成に戻して、すべてのリソースの所有権を動作しているピアに再度割り当てる破壊的な操作です。クラスタ化の構成を解除する理由は 2 つあります。クラスタ化の使用をやめて、代わりに独立した 2 つのストレージアプライアンスを構成する。障害が発生したストレージコントローラを新しいハードウェアに交換するか、またはストレージコントローラを出荷時のアプライアンスソフトウェアに交換する (一般に、この交換はサービスプロバイダが行います)。

## ▼ クラスタ化の構成

1. 電源および 1 つ以上の Ethernet ケーブルを各アプライアンスに接続します。
2. 下記の「ノード配線」の説明どおりに、クラスタ相互接続コントローラをケーブルで接続します。また、クラスタ設定を続行し、設定プロセス中にこれらのケーブルを動的に追加することもできます。
3. アプライアンスに付属するセットアップポスターの JBOD 配線図で示すとおり、HBA をシェア JBOD にケーブルで接続します。
4. 両方のアプライアンスの電源を入れます。ただし、構成は開始しないでください。構成を実行する 2 つのアプライアンスのうち 1 つだけを選択します。どちらを選択してもかまいません。構成目的で、これをプライマリアプライアンスと呼びます。このアプライアンスのシリアルコンソールに接続してアクセスし、スタンドアロンアプライアンスを構成するときと同じ方法で、tty ベースの初期構成を実行します。注: セカンダリアプライアンスでは、tty ベースの初期構成を実行しないでください。クラスタ設定中に自動的に構成されます。
5. プライマリアプライアンスで、BUI または CLI に移動してクラスタ設定を開始します。クラスタ相互接続コントローラがインストールされている場合は、クラスタ設定を初期設定の一部として選択できます。または、この時点でスタンドアロン構成を実行して、クラスタ設定はあとにすることもできます。後者の場合は、「構成」->「クラスタ」で「セットアップ」ボタンをクリックして、クラスタ構成タスクを実行できます。
6. クラスタ設定の最初の手順では、アクティブなクラスタリンクの図が表示されます。画面上に青い 3 本の実線 (接続ごとに 1 本) が表示されます。表示されない場合は、ここで足りないケーブルを追加します。3 本の線がすべて表示されたら、「コミット」ボタンをクリックして続行できます。
7. 2 番目のアプライアンスのアプライアンス名と初期 root パスワードを入力します (これは新しいアプライアンスでシリアルコンソールの初期設定を行うのと同様です)。「コミット」ボタンをクリックすると、2 番目のアプライアンスが構成されるときに進行状況バーが表示されます。
8. プライマリアプライアンスの初期設定の一部としてクラスタ化を設定する場合は、単一アプライアンスの場合と同様に、初期構成を実行するように求めるプロンプトが表示されます。構成の変更はすべて、他方のアプライアンスに自動的に反映されます。次の制限および警告を考慮しながら、初期構成を続行します。DHCP 経由で構成されたネットワークインタフェースは、ヘッド間でフェイルオーバーできないため、クライアントによるストレージへのアクセスには使用できません。したがって、クライアントがストレージにアクセスするときに使用されるネットワークインタフェースには、静的な IP アドレスを割り当ててください。tty



ベースの初期構成中に DHCP が構成されたネットワークインタフェースを選択し、クライアントアクセスにそのインタフェースを使用する場合は、アドレスタイプを「静的」に変更してから続行する必要があります。ベストプラクティスは、管理用のプライベートネットワークインタフェースを構成して、各ヘッドに割り当てる方法です。これにより、クラスタの状態に関係なく、ネットワーク上のどちらかのヘッド経由 (BUI または CLI) で管理が可能です。ルートが必要な場合は、各ヘッドに割り当てられるインタフェース上にルートを作成します。具体的な例については、前のセクションを参照してください。

9. ストレージプールの手順に達するまで、初期構成を続行します。テイクオーバーが発生すると、クライアントがそのストレージプールに到達するときに使用するネットワークインタフェースとともに、各ストレージプールがクラスタピアに引き継がれます。2 つのストレージプールを作成する場合、通常、各ヘッドではクライアントは割り当てられたプールにアクセスできます。ヘッドのどちらかで障害が発生しても、他方からクライアントは両方のプールにアクセスできます。単一のプールを作成する場合は、プールが割り当てられていないヘッドは、そのピアに障害が発生したときのみサービスをクライアントに提供します。ストレージプールは、作成時にヘッドに割り当てられます。ストレージ構成ダイアログには、各ヘッドに個別に割り当てられたプールを作成するオプションがあります。プールに割り当てることができるストレージの最小単位は、1 つのディスクです。複数のプールを作成する場合、それらのプールが同じサイズである必要はありません。管理が簡素化され、使用可能な全体の容量の割合が高まるため、プール数を少なくし、プールあたりのディスク数を多くすることが推奨されます。すべての JBOD を通して、各プールには少なくとも 8 つ (理想的にはさらに多く) のディスクを含めることが推奨されます。
10. 基本構成が完了したら、各ヘッドにリソースを割り当てることができます。一般に、ネットワークインタフェースのみを割り当てる必要があります。ストレージプールは、ストレージ構成段階で自動的に割り当てられています。
11. 下記のとおり、リソースの割り当てをコミットして、クラスタユーザーインタフェースから初期フェイルバックを実行します。プライマリプライアンスの初期設定をまだ実行している場合は、この画面が設定シーケンスの最後として表示されます。初期構成後に手動でクラスタ構成を実行する場合は、「構成/クラスタ」画面に移動して、これらのタスクを実行します。詳細は、下記のクラスタユーザーインタフェースを参照してください。

## ▼ クラスタ化の構成解除

1. 出荷時の構成にリセットするストレージコントローラを選択します。サイトで障害が発生したストレージコントローラを運用に戻さない場合は、障害が発生したストレージコントローラを交換するときに手順 3 はスキップできます。

2. 出荷時の構成にリセットするストレージコントローラのシステムコンソールから、出荷時のリセットを実行します。
3. ストレージコントローラがリセットされ、そのピアが正常にテイクオーバーを開始します。注: 出荷時の状態にリセットされたストレージコントローラがブートを開始する前に (たとえば、ブートメニューよりも先に進む前に)、電源を切って、そのピアがテイクオーバーを完了するまで待機します。
4. クラスタ相互接続ケーブルを切り離し (上記を参照)、電源を切ったストレージコントローラをクラスタの外部ストレージ格納装置から切り離します。
5. 残りのストレージコントローラで、「構成」->「クラスタ化」画面の「構成解除」ボタンをクリックします。すべてのリソースがストレージコントローラに割り当てられ、ストレージコントローラはクラスタのメンバーではなくなります。
6. 切り離されたストレージコントローラ (存在する場合) は、正常に自身のストレージに接続し、電源を入れ、構成できるようになります。障害が発生したストレージコントローラを交換する場合は、交換品を残りのストレージコントローラおよびストレージに取り付けて、上記のクラスタ設定タスクを開始します。

---

注記 - クラスタに 2 つ以上のプールがある場合は、構成解除後にすべてのプールの所有権が残りのストレージコントローラに割り当てられます。2010.Q1.0.0 よりも前のソフトウェアバージョンでは、この構成はサポートされていませんでした。古いソフトウェアバージョンが実行されている場合は、次のいずれかを実行する必要があります。片方または両方のプールを削除する、交換用ストレージコントローラを取り付ける、上記のクラスタ設定タスクを実行する、いずれかのプールの所有権を交換用ストレージコントローラに再度割り当てる、またはストレージコントローラごとに複数のプールをサポートする 2010.Q1.0.0 以降のソフトウェアリリースにアップグレードする。

---

## CLI を使用したクラスタ化の構成

### ▼ クラスタ構成をシャットダウンする

1. 次の CLI コマンドを使用して、クラスタ状態を検証します。

```
nas-7420-1a:> configuration cluster
nas-7420-1a:configuration cluster> show
```

2. 次は、クラスタプロパティの例です。state は、コマンドを実行するヘッドのステータスを示します。peer\_state は、ほかのヘッドのステータスを示します。

```

state = AKCS_OWNER
description = Active (takeover completed)
peer_asn = 365ed33c-3b9d-c533-9349-8014e9da0408
peer_hostname = nas-7420-1b
peer_state = AKCS_STRIPPED
peer_description = Ready (waiting for failback)

```

3. 次の表を使用して、ノードのステータスを検証します。

このノード	ほかのノード	状態
AKCS_CLUSTERED	AKCS_CLUSTERED	両方のノードが通常の状態で行 中です。
AKCS_OWNER	AKCS_STRIPPED	このノードは、すべてのリソースを含 み、アクティブなノードです。ほかの ノードはスタンバイ状態にあり、リ ソースがありません。
AKCS_OWNER	rebooting	別のノードがリブート中で、このノ ードにすべてのリソースが含まれま す。
AKCS_OWNER	unknown	このノードはパートナーを認識してい ません。

注記 - ヘッドのステータスが一致しない場合、クラスタに問題が発生している可能性があります。続行する前に、Oracle サポートに連絡してください。

## ▼ スタンバイヘッドをシャットダウンする

1. CLI を使用して次のコマンドを実行して、スタンバイヘッドをシャットダウンします。

```

nas-7420-1b:configuration cluster> cd /
nas-7420-1b:> maintenance system poweroff
This will turn off power to the appliance. Are you sure? (Y/N)

```

2. ほかのヘッドをシャットダウンすることを確認するために、Y を入力します。

注記 - 両方のヘッドのステータスが AKCS\_CLUSTERED である場合、動作しているヘッドのテイクオーバーが自動的に開始されます。

3. スタンバイヘッドの電源が切断されており、クラスタ状態が OWNER/unknown であることを確認します。
4. CLI を使用して次のコマンドを実行して、アクティブヘッドをシャットダウンします。

```
nas-7420-1a:configuration cluster> cd /
nas-7420-1a:> maintenance system poweroff
This will turn off power to the appliance. Are you sure? (Y/N)
```

5. アクティブヘッドをシャットダウンすることを確認するために、Y を入力します。
6. 両方のヘッドの電源が切断されていることを確認します。ILOM プロンプトから次を実行します。

```
-> show /SYS power_state
```

7. ディスクシェルフの電源を切断します。

## ▼ クラスタ化の構成解除

- CLI でクラスタ化の構成を解除した場合の動作は、「BUI の構成解除」ボタンの動作と同じです。クラスタが正しい状態でないときに、ユーザーがクラスタの構成解除を試みると、エラーが表示されます。

```
configuration cluster> help
Subcommands that are valid in this context:

resources          => Configure resources

help [topic]       => Get context-sensitive help. If [topic] is specified,
                    it must be one of "builtins", "commands", "general",
                    "help", "script" or "properties".

show               => Show information pertinent to the current context

done               => Finish operating on "cluster"

get [prop]         => Get value for property [prop]. ("help properties"
                    for valid properties.) If [prop] is not specified,
                    returns values for all properties.

setup              => Run through initial cluster setup

failback           => Fail back all resources assigned to the cluster peer

takeover           => Take over all resources assigned to the cluster peer

unconfig           => Unconfigure the cluster

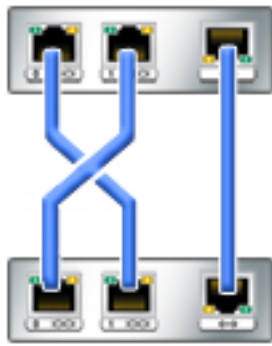
links              => Report the state of the cluster links
```

## クラスタノードの配線

クラスタ化されたヘッドノードは、コントローラの背面にあるクラスタ相互接続ポートを使用して接続する必要があります。

### ZS3-2 クラスタの配線

図 10-11 ZS3-2 クラスタの配線



ZS3-2 コントローラには、ヘッドが通信できる 3 つの冗長リンク (2 つのシリアルリンク (1 番目の 2 つのコネクタ) と 1 つの Ethernet リンク (3 番目のコネクタ)) が備わっています。

Cat 5 以上のストレート Ethernet ケーブル (このクラスタ構成では 3 本の 1m ケーブル) を使用して、左側の図に従ってヘッドノードを接続します。

クラスタ配線は、ヘッドノードの電源を入れる前に実施することも、クラスタ設定ガイド付きタスクを実行しながら実施することもできます。このページの後半で示すように、ユーザーインターフェースに各リンクのステータスが表示されます。クラスタ構成に進む前に、3 本のリンクをすべて構築しておく必要があります。

## ZS3-4 および 7x20 クラスタの配線

図 10-12 ZS3-4 および 7x20 クラスタの配線



ZS3-4 および 7x20 コントローラには、ヘッドが通信できる 3 つの冗長リンク (2 つのシリアルリンク (外側の 2 つのコネクタ) と 1 つの Ethernet リンク (中央のコネクタ)) が備わっています。

Cat 5 以上のストレート Ethernet ケーブル (このクラスタ構成では 3 本の 1m ケーブル) を使用して、左側の図に従ってヘッドノードを接続します。

クラスタ配線は、ヘッドノードの電源を入れる前に実施することも、クラスタ設定ガイド付きタスクを実行しながら実施することもできます。このページの後半で示すように、ユーザーインタフェースに各リンクのステータスが表示されます。クラスタ構成に進む前に、3 本のリンクをすべて構築しておく必要があります。

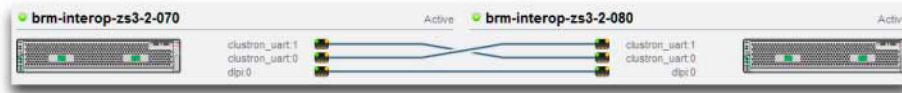
## ストレージシェルフの配線

クラスタ構成を開始する前に、ストレージシェルフを両方のアプライアンスに接続する必要があります。『Oracle ZFS Storage Appliance インストールガイド』の「設置」を参照するか、システムに付属するクイックセットアップのポスターに従ってください。

## クラスタ構成の BUI ページ

「構成」->「クラスタ」ビューには、クラスタカードのステータス、クラスタヘッドノードの状態、およびすべてのリソースの概要がグラフィカルに表示されます。

図 10-13 クラスタの構成ビュー



このインタフェースには次のオブジェクトがあります。

- 各システムのサムネイルピクチャー。管理インタフェースがアクセスされているシステムが左側に表されます。各サムネイルには、標準的なアプライアンス名および現在のクラスタ状態 (上記のアイコンと、説明のラベル) を示すラベルが付けられます。
- ハードウェアで動的に更新する各クラスタカード接続のサムネイル。そのリンクが接続されアクティブなどときは実線でリンクを接続し、その接続が切断された場合や、ほかのシステムが再起動/リポートしている間は線が消えます。
- 各システムに現在割り当てられているプライベートリソースとシングルトンリソースの一覧 (前述の「概要」を参照)。各クラスタノードのサムネイルの下に、さまざまなリソース属性とともに表示されます。
- リソースごとに、そのリソースが割り当てられるアプライアンス (つまり、両方が CLUSTERED 状態のときにリソースを提供するアプライアンス)。現在のアプライアンスが OWNER 状態のときは、所有者フィールドがポップアップメニューとして表示されます。これは編集可能で、「適用」をクリックするとコミットできます。
- リソースごとに、リソースがプライベートであるかどうかを示すロックアイコン。現在のアプライアンスが OWNER または CLUSTERED 状態のときは、ロックアイコンをクリックしてから「適用」をクリックすると、リソースをロック (プライベートにする) またはロック解除 (シングルトンにする) できます。リモートピアに属するプライベートリソースは、どちらかのリソース一覧には表示されないことに注意してください。

この BUI には次のボタンがあります。

表 10-6 シェルフ配線インタフェースボタン

ボタン	説明
セットアップ	クラスタがまだ構成されていない場合は、クラスタ設定ガイド付きタスクを実行してから、現在の画面に戻ります。このタスクの詳細は、前述の説明を参照してください。
構成解除	クラスタの構成を解除して、ノードをスタンドアロン操作にアップグレードします。このタスクの詳細は、後述の説明を参照してください。

ボタン	説明
適用	リソース変更が保留中 (黄色で強調表示された行) の場合、その変更をクラスタにコミットします。
戻す	リソース変更が保留中 (黄色で強調表示された行) の場合、その変更を元に戻し、現在のクラスタ構成を表示します。
フェイルバック	現在のアプライアンス (左側) が OWNER の場合、他方のアプライアンスが所有するリソースをここにフェイルバックします。両方のノードは、CLUSTERED 状態 (アクティブ/アクティブ) のままです。
テイクオーバー	現在のアプライアンス (左側) が CLUSTERED または STRIPPED の場合、他方のアプライアンスを強制的にリポートし、そのリソースをテイクオーバーして、現在のアプライアンスを OWNER にします



# ◆◆◆ 第 11 章

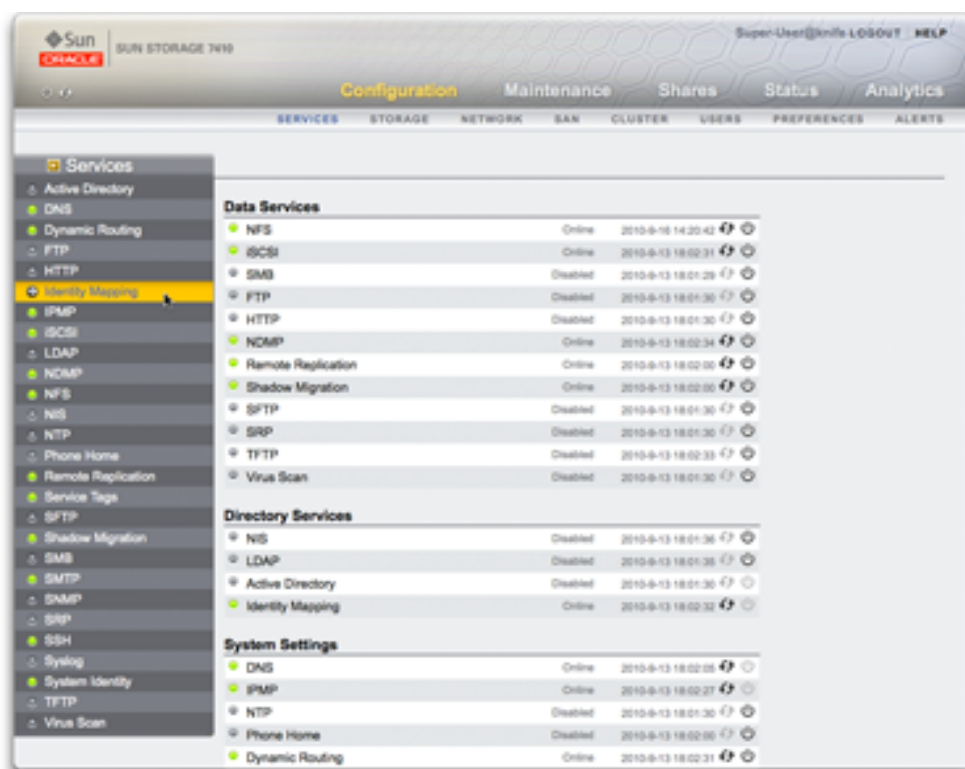
## ZFSSA サービス

「Services」画面は、サービス間で迅速なナビゲートを行うためのサイドパネルを特徴としています。

### 使用可能なサービス

次の ZFSSA サービスを構成できます。

図 11-1 「サービスの構成」BUI ページ



## データサービス

表 11-1 使用可能なデータサービス

サービス	説明	使用されるポート
<a href="#">202 ページの「NFS」</a>	NFSv3 および NFSv4 プロトコル経由でのファイルシステムアクセス	111 および 2049
<a href="#">208 ページの「iSCSI」</a>	iSCSI プロトコル経由での LUN アクセス	3260 および 3205
<a href="#">210 ページの「SMB」</a>	SMB プロトコル経由でのファイルシステムアクセス	SMB-over-NetBIOS 139
	SMB-over-TCP 445	
	NetBIOS Datagram 138	
	NetBIOS Name Service 137	
<a href="#">226 ページの「FTP」</a>	FTP プロトコル経由でのファイルシステムアクセス	21
<a href="#">228 ページの「HTTP」</a>	HTTP プロトコル経由でのファイルシステムアクセス	80
<a href="#">230 ページの「NDMP」</a>	NDMP ホストサービス	10000
<a href="#">239 ページの「リモートレプリケーション」</a>	リモートレプリケーション	216
<a href="#">240 ページの「シャドウ移行」</a>	シャドウデータ移行	
<a href="#">240 ページの「SFTP」</a>	SFTP プロトコル経由でのファイルシステムアクセス	218
<a href="#">243 ページの「SRP」</a>	SRP プロトコル経由でのブロックアクセス	
<a href="#">244 ページの「TFTP」</a>	TFTP プロトコル経由でのファイルシステムアクセス	
<a href="#">245 ページの「ウイルススキャン」</a>	ファイルシステムのウイルススキャン	

## ディレクトリサービス

注: 0 - 99 の UID および GID は、将来のアプリケーションで使用するためにオペレーティングシステムベンダーによって予約されています。階層化された製品のエンドシステムユーザーまたはベンダーによるそのような UID や GID の使用はサポートされておらず、使用し

た場合は将来のアプリケーションでセキュリティ関連の問題が発生する可能性があります。

表 11-2 使用可能なディレクトリサービス

サービス	説明	使用されるポート
<a href="#">248 ページの「NIS」</a>	NIS サービスからのユーザーおよびグループ認証	
<a href="#">250 ページの「LDAP」</a>	LDAP ディレクトリからのユーザーおよびグループ認証	389
<a href="#">254 ページの「Active Directory」</a>	Microsoft Active Directory サーバーによるユーザー認証	
<a href="#">260 ページの「アイデンティティマッピング」</a>	Windows エンティティと UNIX ID との対応付け	

## サービス設定

表 11-3 サービス設定

サービス	説明	使用されるポート
<a href="#">267 ページの「DNS」</a>	ドメインネームサービスクライアント	53
<a href="#">270 ページの「動的ルーティング」</a>	RIP および RIPng 動的ルーティングプロトコル	
<a href="#">270 ページの「IPMP」</a>	IP フェイルオーバー用の IP マルチパス	
<a href="#">271 ページの「NTP」</a>	ネットワークタイムプロトコルクライアント	
<a href="#">275 ページの「フォンホーム」</a>	製品の登録およびサポート構成	443
<a href="#">279 ページの「サービスタグ」</a>	製品のインベントリサポート	443
<a href="#">280 ページの「SMTP」</a>	送信メールサーバーの構成	
<a href="#">281 ページの「SNMP」</a>	警告に関するトラップを送信したり、アプライアンスのステータス情報を提供したりするための SNMP	
<a href="#">285 ページの「Syslog」</a>	警告に関する syslog メッセージを送信したり、サービスの syslog メッセージを転送したりするための syslog リレー	
<a href="#">291 ページの「システム ID」</a>	システムの名前および場所	

## リモートアクセスサービス

表 11-4 使用可能なリモートアクセスサービス

サービス	説明	使用されるポート
<a href="#">291 ページの「SSH」</a>	CLI アクセス用の SSH	22
<a href="#">279 ページの「REST」</a>	RESTful API	

## セキュリティーサービス

表 11-5 使用可能なセキュリティーサービス

サービス	説明	使用されるポート
Kerberos	Kerberos V の認証	88
Kerberos V のパスワードの変更 および設定 (SET_CHANGE)	464	
Kerberos V のパスワードの変更 および設定 (RPCSEC_GSS)	749	

## 最小限必要なポート

ネットワーク上のセキュリティーを提供するため、ネットワークアーキテクチャー内部にファイアウォールを配備できます。ポート番号は、ファイアウォール規則の作成に使われ、ホストとサービスを指定して、ネットワーク上でトランザクションを一意に識別します。

次のリストに、アプライアンスの全機能を使用可能にするファイアウォール規則を作成するために最小限必要なポートを示します。

### インバウンドポート

- icmp/0-65535 (PING)
- tcp/1920 (EM)
- tcp/215 (BUI)
- tcp/22 (SSH)
- udp/161 (SNMP)

### アウトバウンドポート

- tcp/80 (WEB)
- tcp/443 (SSL WEB)

注記 - tcp/443 のアウトバウンドポートは、フォンホームメッセージの送信、サポートバンドルのアップロード、および更新通知に使用されます。レプリケーションの場合、可能なかぎり GRE (Generic Routing Encapsulation) トンネルを使用します。これにより、トラフィックがバックエンドインタフェースで実行し、トラフィックを遅くする可能性のあるファイアウォールを回避できます。NFS コアで GRE トンネルを使用できない場合、フロントエンドインタフェース経由でレプリケーションを実行する必要があります。この場合、ポート 216 も開いている必要があります。

## BUI を使用したサービスの構成

BUI サービスの画面を使用して、上記表内のサービスや設定を表示および変更します。サービス行をダブルクリックすると、そのサービスの定義画面が表示されます。次の表に、サービス画面のアイコンおよびボタンを示します。

表 11-6 サービスの BUI ページのアイコンとボタン

アイコン	説明
	プロパティの構成やログの表示を行うサービス画面に移動します。このボタンはマウスオーバーで表示されます
	サービスが有効で正常に機能しています。
	サービスがオフラインまたは無効です。
	サービスに問題があり、オペレータの注意が必要です。
	サービスを有効または無効にします
	サービスを再起動します
	このサービスの有効/無効が使用できません
	現在使用不可能なサービスを再起動します。最初にサービスを有効にしてください

### ▼ 特定のサービス画面の表示


1. 特定のサービスのプロパティを表示または編集するには、サービス名の左側にあるサービスのステータスアイコンにマウスを合わせます。

2. ステータスアイコンが矢印のアイコンに変わり、クリックすると、選択したサービスのプロパティ画面が表示されます。



## ▼ 特定のサービス画面の表示

- どのサービス画面でも、サービスのタイトル (各画面の左上近く) の左にある小さい矢印のアイコンをクリックすると、すべてのサービスのサイドパネルを表示できます。このアイコンを再度クリックすると、リストが非表示になります。

## ▼ サービスの有効化

- サービスがオンラインでない場合は、電源アイコン  をクリックすると、サービスがオンライン  になります

## ▼ サービスの無効化

- サービスがオンラインのときに無効にする場合は、電源アイコン  をクリックすると、サービスがオフライン  になります

## ▼ プロパティの定義

1. サービスのプロパティを定義するには、サービスをダブルクリックします。
2. プロパティを変更して、「適用」をクリックします。
3. プロパティをリセットするには、「戻す」をクリックします。

## ▼ サービスのログの表示

1. 一部のサービスでは、サービスの問題を診断するのに役立つ情報がサービスのログに記録されます。サービス画面の右上に「ログ」ボタンがある場合、そのサービスにはログが用意されています。ログには次の情報が含まれていることがあります。
  - サービスの状態が変更された時間

■ サービスからのエラーメッセージ

2. ログの内容は個々のサービスに固有であり、アプライアンスソフトウェアへの将来の更新で変わることがあります。このバージョンのアプライアンスでよく使用されるメッセージの例を次に示します。

ログメッセージの例	説明
起動メソッドの実行中	サービスが起動しています
「起動」メソッドがステータス 0 で終了しました	サービスが正常に起動したことを報告しました (0 == 成功)
「リフレッシュ」メソッドがステータス 0 で終了しました	サービスの設定がそのサービス構成に基づいて正常にリフレッシュされました
停止メソッドの実行中	サービスがシャットダウンしています
有効	(システムのブート中などに) サービスを起動するべきかどうかを確認するためにサービス状態がチェックされ、サービスが有効な状態であることがわかりました
無効	(システムのブート中などに) サービスを起動するべきかどうかを確認するためにサービス状態がチェックされ、サービスが無効な状態であることがわかりました

## CLI を使用したサービスの構成

CLI サービスのセクションは、`configuration services` の下にあります。`show` コマンドを使用すると、すべてのサービスの現在の状態が一覧表示されます。

次の例は [271 ページの「NTP」](#) サービスからのものです。

```
[ Oct 11 21:05:31 Enabled. ]
[ Oct 11 21:07:37 Executing start method (...). ]
[ Oct 11 21:13:38 Method "start" exited with status 0. ]
```

例の最初のログイベントは、システムが 21:05 にブートされたことを示しています。21:07:37 の 2 番目のエントリでこのサービスの起動が開始されたことが記録され、21:13:38 に起動が完了しました。NTP の性質とシステムクロックの調整のため、ログに示されるように、このサービスの起動が完了するのに数分かかることがあります。

```
caji:> configuration services
caji:configuration services> show
Services:
          ad => disabled
          smb => disabled
```

```

    dns => online
dynrouting => online
    ftp => disabled
    http => disabled
identity => online
    idmap => online
    ipmp => online
iscsi => online
    ldap => disabled
    ndmp => online
    nfs => online
    nis => disabled
    ntp => disabled
replication => online
    scrk => disabled
    sftp => disabled
shadow => online
    smtp => online
    snmp => disabled
    ssh => online
syslog => disabled
    tags => online
    tftp => disabled
vscan => disabled

```

Children:

```

    ad => Configure Active Directory
    smb => Configure SMB
    dns => Configure DNS
dynrouting => Configure Dynamic Routing
    ftp => Configure FTP
    http => Configure HTTP
identity => Configure System Identity
    idmap => Configure Identity Mapping
    ipmp => Configure IPMP
iscsi => Configure iSCSI
    ldap => Configure LDAP
    ndmp => Configure NDMP
    nfs => Configure NFS
    nis => Configure NIS
    ntp => Configure NTP
replication => Configure Remote Replication
    scrk => Configure Phone Home
    sftp => Configure SFTP
shadow => Configure Shadow Migration
    smtp => Configure SMTP
    snmp => Configure SNMP
    srp => Configure SRP
    ssh => Configure SSH
syslog => Configure Syslog
    tags => Configure Service Tags
    tftp => Configure TFTP
    vscan => Configure Virus Scan
routing => Configure Routing Table

```



## ▼ サービスの選択

1. サービスを選択すると、その状態の表示、有効化や無効化、およびプロパティの設定を行うことができます。
2. サービスを選択するには、その名前を入力します。たとえば、`nis` を選択するには、次のように入力します。

```
caji:configuration services> nis
caji:configuration services nis>
```

## ▼ サービスの状態の表示

- `show` コマンドを使用すると、サービスの状態を表示できます。

```
caji:configuration services nis> show
Properties:
    <status> = online
    domain = fishworks
    broadcast = true
    ypservers =
```

## ▼ サービスの有効化

- サービスを有効にするには、`enable` コマンドを使用します。

```
caji:configuration services nis> enable
```

## ▼ サービスの無効化

- サービスを無効にするには、`disable` コマンドを使用します。

```
caji:configuration services nis> disable
```

## ▼ プロパティの設定

1. 選択したサービスのプロパティを設定するには、`set` コマンドを使用します。

2. プロパティを設定したあと、`commit` コマンドを使用して新しい構成を保存しアクティブ化します。

```
caji:configuration services nis> set domain="mydomain"
      domain = mydomain (uncommitted)
caji:configuration services nis> commit
caji:configuration services nis> show
Properties:
      <status> = online
      domain = mydomain
      broadcast = true
      ypservers =
```

3. プロパティ名は BUI で表示される名前と似ていますが、CLI での名前は通常はそれよりも短く、短縮されている場合もあります。

## ▼ サービスのヘルプの表示

- `help` を入力すると、サービス用のコマンドがすべて表示されます。

```
caji:configuration services nis> help
Subcommands that are valid in this context:

  help [topic]      => Get context-sensitive help. If [topic] is specified,
                    it must be one of "builtins", "commands", "general",
                    "help", "script" or "properties".

  show              => Show information pertinent to the current context

  commit            => Commit current state, including any changes

  done              => Finish operating on "nis"

  enable            => Enable the nis service

  disable           => Disable the nis service

  get [prop]        => Get value for property [prop]. ("help properties"
                    for valid properties.) If [prop] is not specified,
                    returns values for all properties.


  set [prop]        => Set property [prop] to [value]. ("help properties"
                    for valid properties.) For properties taking list
                    values, [value] should be a comma-separated list of
                    values.
```

## NFS

ネットワークファイルシステム (NFS) とは、ネットワーク上でファイルをシェアするための業界標準のプロトコルです。Sun ZFS Storage Appliance では、NFS バージョン

2、3、および 4 がサポートされています。ファイルシステムの名前空間の構築方法の詳細は、[306 ページの「ファイルシステムの名前空間」](#)のセクションを参照してください。ローカルユーザーでの NFS については、[第7章「ユーザー構成」](#)を参照してください。

## プロパティ

- サポートされる最小バージョン - このドロップダウンリストを使用して、アプライアンスでサポートされる NFS のバージョンを制御します。
- サポートされる最大バージョン - このドロップダウンリストを使用して、アプライアンスでサポートされる NFS のバージョンを制御します。
- サーバースレッドの最大数 - 同時に実行される NFS リクエストの最大数を定義します (20 - 1000)。この値は、最低でも予測される並行 NFS クライアント数に及ぶようにしてください。
- 猶予時間 - 計画外の停止によるアプライアンスのリブート後に、すべてのクライアントがロック状態を回復する必要がある秒数を定義します (15 - 600 秒)。このプロパティは NFS v4 クライアントのみに影響します (NFS v3 はステートレスであるため、再生する状態はありません)。この期間中、NFS サービスは古いロック状態の再生のみを処理します。猶予期間が終了するまで、ほかのサービスリクエストは処理されません。デフォルトの猶予期間は 90 秒です。猶予期間を短くすると、サーバーのリブート後に NFS クライアントはより迅速に処理を再開できますが、そのすべてのロック状態を回復できない可能性が高くなります。計画的な停止期間中、Oracle ZFS Storage Appliance は、NFSv4 クライアントに対して猶予なしのロック状態の回復を提供します。計画的な停止は『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「[更新](#)」などのイベント中に発生し、アプライアンスは CLI `maintenance system reboot` コマンド、または BUI 電源アイコン  を使用してリブートします。計画的な停止の場合、NFS サービスは、猶予期間の遅延を生じさせることなく、すべてのサービスリクエストを処理します。
- カスタム NFSv4 識別情報ドメイン - このプロパティを使用して、NFSv4 ユーザーおよびグループ識別情報のマップ用のドメインを定義します。このプロパティを設定しない場合、アプライアンスでは DNS を使用して識別情報のドメインを取得するために、まず `_nfsv4idmapdomain` DNS リソースレコードを確認し、次に DNS ドメイン自体にフォールバックします。
- NFSv4 委譲を有効にする - このプロパティを選択すると、クライアントはファイルをローカルにキャッシュし、サーバーに接続しないで変更を行えます。このオプションはデフォルトで有効になっており、通常はパフォーマンスが向上しますが、ほんの一部の環境では問題が発生することがあります。この設定の無効化は、特定のワークロードのパフォーマンスを慎重に測定し、その設定によって測定可能なパフォーマンスの利

点を得られることが検証された場合にのみ行うようにしてください。このオプションは NFSv4 のマウントにのみ影響します。

- マウント表示 - このプロパティを使用すると、シェアアクセスリストや、NFS クライアントからのリモートマウントに関する情報の可用性を制限できます。「フル」はフルアクセスを許可します。「制限付き」は、クライアントがアクセスを許可されているシェアしか表示できないようにアクセスの制限を行います。クライアントでは、サーバーで定義されたシェアのアクセスリストや、ほかのクライアントが行なったサーバーからのリモートマウントは表示できません。このプロパティはデフォルトで「フル」に設定されています。
- Kerberos を有効化 - Kerberos サービスを有効または無効にします。
- \* Kerberos での脆弱な暗号化タイプを許可 - DES (des-cbc-crc, des-cbc-md5) およびエクスポート可能な ArcFour with HMAC/md5 (arcfour-hmac-exp) のサポートを有効または無効にします。このプロパティはデフォルトで無効になっています。
- \* Kerberos レルム - レルムとはドメインのようなもので、同じマスター KDC の下にあるシステムをグループとして定義する論理ネットワークです。レルム名は、任意の ASCII 文字列で構成できます。通常、レルム名は DNS ドメイン名と同じですが、このレルム名は大文字で指定します。この表記規則を使用すると、ありふれた名前を使用し続けているときに、Kerberos サービスの問題と DNS 名前空間の問題を区別することができます。
- \* Kerberos マスター KDC - 各レルムには、主体データベースのマスターコピーを保持するサーバーが含まれている必要があります。マスター KDC とスレーブ KDC のもっとも大きな違いは、マスター KDC だけがデータベース管理リクエストを処理することです。たとえば、パスワードの変更や新しい主体の追加はマスター KDC で行う必要があります。
- \* Kerberos スレーブ KDC - スレーブには、主体データベースの重複コピーが含まれています。マスター KDC サーバーとスレーブ KDC サーバーはどちらも、認証の確立に使用されるチケットを作成します。
- \* Kerberos admin 主体 - このプロパティは管理者を識別します。主体名は慣例により、プライマリ、インスタンス、レルムという 3 つのコンポーネントに分けられます。主体は、joe、joe/admin、または joe/admin@ENG.EXAMPLE.COM として指定できます。このプロパティはシステムの Kerberos サービス主体の設定にのみ使用され、保持されません。
- \* Kerberos admin パスワード - 管理者のパスワードを定義します。このプロパティはシステムの Kerberos サービス主体の設定にのみ使用され、保持されません。
- Oracle Intelligent Storage Protocol - NFSv4 サービスには、Oracle Database NFSv4 クライアントが ZFS Storage Appliance NFSv4 サーバーに最適化情報を渡すことができるようにする Oracle Intelligent Storage Protocol のサ

ポートが含まれています。詳細は、[488 ページの「Oracle Intelligent Storage Protocol」](#)を参照してください。

サービスのプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)に記載されています。

NFS の最小バージョンと最大バージョンを同じ値に設定すると、アプライアンスはそのバージョンだけを使用してクライアントと通信します。これは、ある NFS バージョンまたはその他 (ワークロードを含むその NFS バージョンのパフォーマンス特性など) に問題が見つかり、もっとも有効に機能するバージョンだけをクライアントに使用させたい場合に役立つことがあります。

## Kerberos レルム

Kerberos レルムを構成すると、いくつかのサービス主体が作成され、システムのローカル keytab に必要な鍵が追加されます。[271 ページの「NTP サービス」](#)の構成は、Kerberos 化された NFS を構成する前に行う必要があります。次のサービス主体が作成され、Kerberos 化された NFS をサポートするように更新されます。

```
host/node1.example.com@EXAMPLE.COM  
nfs/node1.example.com@EXAMPLE.COM
```

アプライアンスがクラスタ化されている場合は、クラスタノードごとに主体と鍵が生成されます。

```
host/node1.example.com@EXAMPLE.COM  
nfs/node1.example.com@EXAMPLE.COM  
host/node2.example.com@EXAMPLE.COM  
nfs/node2.example.com@EXAMPLE.COM
```

これらの主体がすでに作成されている場合は、レルムを構成すると、各主体のパスワードがリセットされます。アプライアンスが Active Directory ドメインに参加するように構成されている場合は、そのアプライアンスを Kerberos レルムの一部として構成することはできません。

KDC と Kerberos 化されたクライアントの設定については、[http://docs.oracle.com/cd/E26502\\_01/html/E29015/index.html](http://docs.oracle.com/cd/E26502_01/html/E29015/index.html) ([http://docs.oracle.com/cd/E26502\\_01/html/E29015/index.html](http://docs.oracle.com/cd/E26502_01/html/E29015/index.html)) を参照してください。NFS プロパティを Kerberos 用に設定後、「シェア」->「ファイルシステム」->「プロトコル」画面でセキュリティーモードを Kerberos を使用するモードに変更します。

Kerberos 用に次のポートがアプライアンスで使用されます。

- Kerberos V の認証: 88
- Kerberos V のパスワード SET\_CHANGE の変更および設定: 464
- Kerberos V のパスワード RPCSEC\_GSS の変更および設定: 749

注: Kerberos 化された NFS クライアントは、これらの主体の FQDN に解決される IP アドレスを使用してアプライアンスにアクセスする必要があります。たとえば、あるアプライアンスに複数の IP アドレスが構成されている場合、Kerberos 化された NFS クライアントで使用できるのは、そのアプライアンスの FQDN に解決される IP アドレスだけです。

## サービスのログ

NFS サービスで利用できるログは次のとおりです。

表 11-7 NFS で利用できるログ

ログ	説明
network-nfs-server:default	マスター NFS サーバーログです
appliance-kit-nfsconf:default	アプライアンスの NFS 構成イベントのログです
network-nfs-cbd:default	NFSv4 コールバックデーモンのログです
network-nfs-mapid:default	NFSv4 ユーザーおよびグループの資格をマップする NFSv4 mapid デーモンのログです
network-nfs-status:default	NFS ロックのクラッシュおよび回復機能を支援する NFS statd デーモンのログです
network-nfs-nlockmgr:default	ファイルのレコードロック処理をサポートする NFS lockd デーモンのログです

## NFS の分析

NFS アクティビティは、『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」セクションでモニターできます。これには、次のものが含まれます。

- 1 秒あたりの NFS 処理数
- ... 処理の種類別 (読み取り/書き込み/...)
- ... シェア名別
- ... クライアントのホスト名別
- ... アクセスされたファイル名別
- ... アクセス待機時間別

注: NFS サーバーがリブートまたはフェイルオーバーした場合、ファイル名はクライアントから新たに開くまでサーバーで不明となります。Analytics ワークシートでは、ファイルは *unknown* として表示されます。

## NFS の BUI および CLI のプロパティー

次の表は、CLI のプロパティーと上記の BUI のプロパティーの対応付けを示しています。

表 11-8 NFS の BUI および CLI のプロパティー

CLI のプロパティー	BUI のプロパティー
version_min	サポートされる最小バージョン
version_max	サポートされる最大バージョン
nfsd_servers	サーバースレッドの最大数
grace_period	猶予期間
mapid_domain	カスタム NFSv4 識別情報ドメイン
enable_delegation	NFSv4 委譲を有効にする
mount_visibility	クライアントのシェア情報の制限レベル
krb5_allow_weak_crypto	Kerberos での脆弱な暗号化タイプ (arcfour-hmac-md5-exp, des-cbc-md5, および des-cbc-crc) を許可
krb5_realm	Kerberos レルム
krb5_kdc	Kerberos マスター KDC
krb5_kdc2	Kerberos スレーブ KDC
krb5_admin	Kerberos admin 主体

## ▼ NFS を介したファイルシステムのシェア

1. 「構成」->「サービス」画面に移動します。
2. NFS サービスが有効でオンラインになっていることを確認します。有効でない場合は、サービスを有効にします。
3. [第12章「シェア、プロジェクト、およびスキーマ」](#)画面を表示して、既存のシェアを編集するか、新しいシェアを作成します。

4. 編集しているシェアの「プロトコル」タブをクリックし、NFS シェアが有効になっていることを確認します。この画面では、NFS シェアモード (読み取り、読み取り/書き込み) の構成も行えます。

## iSCSI サービス

アプライアンスで LUN を構成すると、iSCSI (Internet Small Computer System Interface) ターゲットを介してそのボリュームをエクスポートできます。iSCSI サービスでは、iSCSI イニシエータは iSCSI プロトコルを使用してターゲットにアクセスできます。

このサービスは、iSNS プロトコルを使用した検出、管理、および構成をサポートします。iSCSI サービスは、CHAP を使用して単方向 (ターゲットがイニシエータを認証する) および双方向 (ターゲットとイニシエータが相互に認証する) の両方の認証をサポートします。また、RADIUS データベースでの CHAP 認証データ管理もサポートします。

システムでは、2 つの独立したステップで、最初に認証を実行し、2 番目に承認を実行します。

---

注記 - iSCSI イニシエータおよびターゲットの構成の例については、[第6章「Storage Area Network の構成」](#)のセクションを参照してください。

---

## iSCSI サービスのプロパティ

表 11-9 iSCSI サービスのプロパティ

プロパティ	説明
iSNS を使用	iSNS 検出が有効かどうかを指定します
iSNS サーバー	iSNS サーバーです
RADIUS を使用	RADIUS が有効かどうかを指定します
RADIUS サーバー	RADIUS サーバーです
RADIUS サーバーシークレット	RADIUS サーバーのシークレットです

サービスのプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)に記載されています。CLI のプロパティ名は、上記のプロパティ名の短縮版です。



## iSCSI サービスの認証

ローカルイニシエータに CHAP 名と CHAP シークレットが指定されている場合は、システムによって認証が行われます。ローカルイニシエータに CHAP プロパティが指定されていない場合は、認証が行われないため、すべてのイニシエータが承認の対象となります。

## iSCSI サービスの承認

iSCSI サービスでは、イニシエータグループ内で使用できるイニシエータのグローバルリストを指定できます。

## iSCSI サービスのターゲットとイニシエータ

iSCSI ターゲットおよびイニシエータの詳細は、[第6章「Storage Area Network の構成」](#)を参照してください。

## iSCSI のトラブルシューティング

イニシエータがターゲットに接続できない場合:

- イニシエータの IQN がイニシエータリストで識別される IQN と一致していることを確認します。
- iSNS サーバーの IP アドレスが正しいことを確認し、さらにその iSNS サーバーが構成されていることも確認します。
- ターゲットの IP アドレスがイニシエータ側で正しいことを確認します。
- イニシエータの CHAP 名およびシークレットが両方で一致していることを確認します。
- ターゲットの CHAP 名およびシークレットがどのイニシエータの CHAP 名およびシークレットとも一致しないことを確認します。
- RADIUS サーバーの IP アドレスとシークレットが正しいことを確認し、さらにその RADIUS サーバーが構成されていることも確認します。
- LUN にアクセスするイニシエータがその LUN のイニシエータグループのメンバーであることを確認します。
- その LUN をエクスポートするターゲットがオンラインになっていることを確認します。
- LUN の動作ステータスがオンラインになっていることを確認します。
- 各 LUN の論理ユニット番号を確認します。

フェイルオーバーまたはフェイルバック時に、Red Hat クライアントからの iSER 縮小コピー I/O が残っていない場合:

- `/etc/iscsi/iscsid.conf` ファイル内の `node.session.timeo.replacement_timeout` パラメータを 300 秒に変更します。

## SMB サービス

SMB サービスでは、SMB プロトコルを使用してファイルシステムにアクセスできます。サポートされている SMB のバージョンは SMB1、SMB2.0 です。ファイルシステムは、[第12章「シェア、プロジェクト、およびスキーマ」](#)構成の SMB を使用してシェアするように構成する必要があります。

## SMB サービスのプロパティ

- LAN Manager 互換性レベル - サポートされている認証モード (LM、NTLM、LMv2、NTLMv2)。それぞれの互換性レベルでサポートされる認証モードの詳細は、Oracle Solaris Information Library の *smb* を参照してください。NTLMv2 は、広く知られているセキュリティの脆弱性を回避するための推奨の最小セキュリティレベルです。
- 優先ドメインコントローラ - [254 ページの「Active Directory」](#)ドメインへの参加時に使用する優先ドメインコントローラ。このコントローラが使用できない場合、Active Directory は DNS SRV レコードや Active Directory サイトを利用して適切なドメインコントローラを見つけます。
- Active Directory サイト - Active Directory ドメインへの参加時に使用するサイト。サイトとは、すべてが高帯域で低遅延のネットワークリンクを使用して接続されるマシンの論理的な集まりです。このプロパティを構成し、優先ドメインコントローラを指定しない場合、Active Directory ドメインに参加すると、このサイトにあるドメインコントローラが外部ドメインコントローラよりも優先されます。
- サーバースレッドの最大数 - 同時に実行されるサーバースレッド (ワーカー) の最大数。デフォルトは 1024 です。
- 動的 DNS を有効化 - アプライアンスが Active Directory ドメインの DNS レコードを更新するために動的 DNS を使用するかどうかを選択します。デフォルトはオフです。
- Oplocks を有効化 - アプライアンスが Opportunistic Locks を SMB クライアントに許可するかどうかを選択します。これにより、ほとんどのクライアントのパフォーマンスが向上します。デフォルトはオンです。SMB サーバーは oplock をクライアントプロセスに許可して、ロックが行われている間にクライアントがデータをキャッシュできるようにします。サーバーが oplock を取り消すと、クライアントはキャッシュされたデータをサーバーにフラッシュします。

- リストを共有する匿名アクセスを制限 - このオプションを有効にした場合、クライアントはシェアリストを受信する前に SMB サービスを認証する必要があります。無効にした場合、匿名クライアントはシェアリストにアクセスできます。
- システムコメント - 意味のあるテキスト文字列です。
- アイドルセッションタイムアウト - セッションの非アクティブのタイムアウト設定です。
- プライマリ WINS サーバー - TCP/IP の設定で構成されたプライマリ WINS アドレスです。
- セカンダリ WINS サーバー - TCP/IP の設定で構成されたセカンダリ WINS アドレスです。
- WINS から除外された IP アドレス - WINS への登録から除外された IP アドレスです。
- SMB 署名が有効 - このオプションにより、SMB 署名機能を使用した SMB クライアントとの相互運用性が有効になります。パケットが署名されている場合は、署名が検証されます。署名されていないパケットは、署名の検証なしに受け入れられます (SMB 署名が必要ない場合 - 下記を参照)。
- SMB 署名が必要 - SMB 署名が必要な場合は、すべての SMB パケットに署名が必要で、署名がないと拒否されます。また、署名をサポートしないクライアントはサーバーに接続できません。
- ゼロの VC を無視 - SMB クライアントで新しい接続を確立するときに、ゼロの Virtual Circuit (VC) 番号を指定して、アプライアンスが以前の接続やファイルロックをこのクライアントからすべて除去するようリクエストすることがあります。ただし、このプロトコルアーティファクトでは、クライアントのネットワークアドレス変換 (NAT) や同じホストに割り当てられた複数の DNS エントリを考慮しません。これと組み合わせると、マスクされたネットワークまたは冗長ネットワークの場所同士でゼロの VC リクエストがやり取りされると、関係のないアクティブな接続がリセットされる可能性があります。デフォルトでは、古いファイルロックを防ぐためにゼロの VC リクエストが受け付けられません。ただし、SMB セッションがエラーで切断されている場合は、ゼロの VC リクエストを無視すると、問題が解決することがあります。

サービスのプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)に記載されています。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## SMB のシェアプロパティ

SMB を介してシェアをエクスポートする際に、いくつかの[295 ページの「シェアのプロパティ」](#)を特定の方法で設定する必要があります。

表 11-10 SMB のシェアプロパティ

プロパティ	説明
<a href="#">第12章「シェア、プロジェクト、およびスキーマ」</a>	SMB クライアントは大文字小文字を区別しない動作を想定しているため、このプロパティを「混在」または「区別しない」に設定する必要があります。
<a href="#">第12章「シェア、プロジェクト、およびスキーマ」</a>	ファイルシステムで UTF-8 以外のファイル名が許可されている場合、SMB クライアントは正しく機能しないことがあります。
ブロック不可の必須ロック	このプロパティを有効にして、バイト範囲ロックが正常に機能できるようにする必要があります。
<a href="#">329 ページの「シェアのプロトコル」</a>	クライアントがシェアを参照するとき使用する名前。この名前が <a href="#">第12章「シェア、プロジェクト、およびスキーマ」</a> から継承される方法については、 <a href="#">329 ページの「シェアのプロトコル」</a> のドキュメントを参照してください。
<a href="#">329 ページの「シェアのプロトコル」</a>	ファイルシステムに格納されている ACL の範囲を超えた別のレイヤーのアクセス制御を追加する ACL。このプロパティの詳細は、 <a href="#">329 ページの「シェアのプロトコル」</a> のドキュメントを参照してください。

[第12章「シェア、プロジェクト、およびスキーマ」](#)および[第12章「シェア、プロジェクト、およびスキーマ」](#)プロパティは、シェアの作成時にのみ設定できます。

## NFS と SMB の相互運用性

アプライアンスでは、[202 ページの「NFS」](#) および SMB クライアントが同時に同じシェアにアクセスできます。アプライアンスを NFS と SMB の相互運用向けに正しく構成するには、次のコンポーネントを構成する必要があります。

- [254 ページの「Active Directory」](#) サービスを構成します。
- [260 ページの「アイデンティティマッピングサービス」](#)の方針を決めて、サービスを構成します。
- SMB を構成します。
- シェアでのアクセス制御、ACL エントリ、および ACL 継承を構成します。

SMB および NFSv3 は同じアクセス制御モデルを使用しません。最良の結果を得るためには、SMB アクセス制御モデルがより冗長なモデルであるように、SMB クライアントからルートディレクトリに ACL を構成します。継承可能で簡単な ACL エントリについては、[338 ページの「「シェア」>「シェア」>「アクセス」](#)のドキュメントを参照してください。

## SMB の DFS 名前空間

分散ファイルシステム (Distributed File System、DFS) とは、SMB および MSRPC プロトコルを介して提供される仮想化技術です。DFS を使用すると、管理者は、異なるサーバー上にあるシェアフォルダを 1 つ以上の DFS 名前空間に透過的に関連付けることで、それらをグループ化できます。DFS 名前空間は、組織内にあるシェアフォルダの仮想ビューです。管理者は、名前空間に入れるシェアフォルダを選択したり、それらのフォルダが表示される際の階層を設計したり、シェアフォルダが名前空間で示す名前を決めたりできます。ユーザーが名前空間を表示すると、フォルダは単一の大容量ファイルシステムに常駐しているように見えます。ユーザーは、データを収容しているサーバー名やシェアフォルダを認識しなくても、名前空間のフォルダをナビゲートできます。

1 つのシステムにつき 1 つのシェアだけをスタンドアロンの DFS 名前空間としてプロビジョニングできます。ドメインベースの DFS 名前空間はサポートされません。各クラスタノードに別個のストレージプールが備わっている場合でも、1 つのクラスタにつき 1 つの DFS 名前空間しかプロビジョニングできません。SMB シェアを DFS 名前空間としてプロビジョニングするには、DFS 管理 MMC スナップインを使用してスタンドアロンの名前空間を作成します。

アプライアンスが [254 ページの「Active Directory」](#) ドメインに参加しない場合、ワークグループのユーザーが DFS 名前空間を変更できるようにするには追加の構成が必要です。SMB ローカルユーザーが DFS 名前空間を作成または削除できるようにするには、そのユーザーにサーバーで作成された別個のローカルアカウントが必要です。次の例では、SMB ローカルユーザー `dfsadmin` が DFS 名前空間を操作できるようにする手順を示します。

## SMB の Microsoft スタンドアロン DFS 名前空間管理ツールのサポートマトリックス

次の表に、Windows の各種オペレーティングシステムバージョンの Microsoft DFS ツールの操作 (サブコマンドまたはオプション) を示します。アプライアンスでスタンドアロンの DFS 名前空間を管理するために、これらのうちのどれがアプライアンスの DFS サービスでサポートされているかを識別します。

Microsoft Windows systems	XP 2003 2003 Vista 2008 2008 Win7							
			R2			R2		
	SP3	SP2	SP2	SP2	SP2	SP1	SP1	
dfscmd CLI:								
/map [comment] [/restore]	y	y	y	y	y	y	y	y
/unmap	y	y	y	y	y	y	y	y
/add [/restore]	y	y	y	y	y	y	y	y
/remove	y	y	y	y	y	y	y	y
/view [/partial   /full]	y	y	y	y	y	y	y	y

dfsutil CLI (old format):							
/addstdroot [/comment]	y	y	y	n	n	y	y
/remstdroot	y	y	y	n	n	y	y
/root:<DfsName> /view	n	n	n	y	y	y	y
/addlink [/comment]	NA	NA	NA	y	y	y	y
/remove link	NA	NA	NA	y	y	y	y
/state /display	NA	NA	NA	y	y	y	y
/state /enable	NA	NA	NA	y	y	y	y
/state /disable	NA	NA	NA	y	y	y	y
/ttl /display	NA	NA	NA	y	y	y	y
/ttl /set	NA	NA	NA	y	y	y	y
/server:<MachineName> /view	y	y	y	y	y	y	y
dfsutil CLI (new format):							
root addstd [comment]	NA	NA	NA	n	n	y	y
root remove	NA	NA	NA	n	n	y	y
root (view namespace)	NA	NA	NA	y	y	y	y
link add [comment]	NA	NA	NA	y	y	y	y
link remove	NA	NA	NA	y	y	y	y
link (view)	NA	NA	NA	y	y	y	y
target add	NA	NA	NA	y	y	y	y
target remove	NA	NA	NA	y	y	y	y
target (view)	NA	NA	NA	y	y	y	y
property comment (view)	NA	NA	NA	y	y	y	y
property comment set	NA	NA	NA	y	y	y	y
property ttl (view)	NA	NA	NA	y	y	y	y
property ttl set	NA	NA	NA	y	y	y	y
property state (view)	NA	NA	NA	y	y	y	y
property state offline	NA	NA	NA	y	y	y	y
property state online	NA	NA	NA	y	y	y	y
DFS GUI:							
add standalone root	y	y	y	n	n	n	n
remove standalone root	y	y	y	n	n	n	n
change root comment	y	y	y	n	n	n	n
change root timeout	y	y	y	n	n	n	n
add link	y	y	y	n	n	n	n
remove link	y	y	y	n	n	n	n
change link comment	y	y	y	n	n	n	n
change link timeout	y	y	y	n	n	n	n
add link's target	y	y	y	n	n	n	n
remove link's target	y	y	y	n	n	n	n
enable link's referral (target)	y	y	y	n	n	n	n
disable link's referral (target)	y	y	y	n	n	n	n
hide root	y	y	y	y	y	y	y
show root	y	y	y	y	y	y	y
display links	y	y	y	n	n	n	n
display targets	y	y	y	n	n	n	n
				XP 2003 2003 Vista 2008 2008 Win7			
					R2		R2
				SP3	SP2	SP2	SP1

注: y - サポートされている、n - サポートされていない、NA - 該当なし

■ Solaris では DFS リンクターゲットを検証しません。

- コメントおよびタイムアウト (TTL) を変更および表示するための CLI コマンドはルートとリンクの両方に適用されます。
- 状態を表示するための CLI コマンドは、ルート、ルートのターゲット、リンク、およびリンクのターゲットに適用できます。
- 状態を変更するための CLI コマンドは、リンクおよびリンクのターゲットにのみ適用できます。

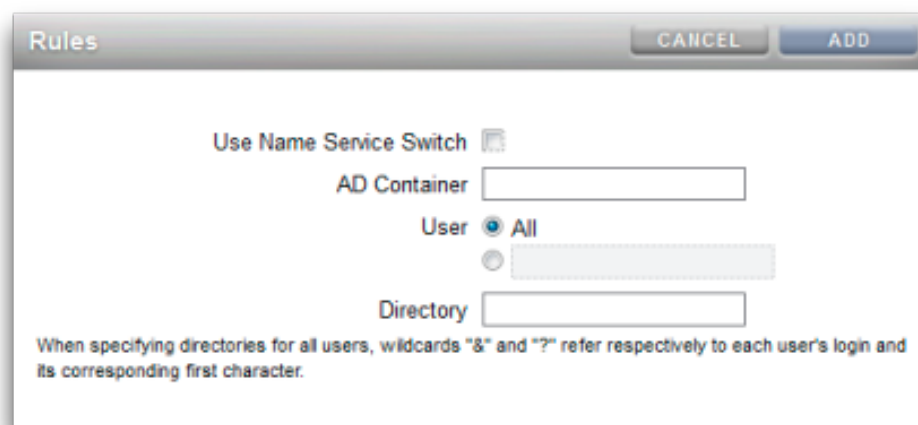
## ▼ 例: DFS ネームスペースの操作

1. ユーザー `dfsadmin` のローカルユーザーアカウントをサーバーで作成します。Windows マシンでローカルユーザーを最初に作成したときと同じパスワードを必ず使用してください。
2. `dfsadmin` をローカル SMB グループの管理者に追加します。
3. DFS 名前空間を変更する Windows マシンで `dfsadmin` としてログインします。

## SMB の自動ホームサービス

Windows ファイル共有では、自動ホームは SMB プロトコルを使用してファイルシステムにアクセスできます。自動ホームは、SMB を介してシステムにアクセスするユーザーのホームディレクトリシェアを定義して保持します。自動ホーム規則では、SMB クライアントをホームディレクトリに対応付けます。

図 11-2 自動ホーム規則の設定



- ネームサービススイッチを使用 - ネームサービススイッチ (NSS) をオンまたはオフに切り替えます。NSS 規則とすべてのユーザー用の規則を同時に作成することはできません。
- AD コンテナ - Active Directory のコンテナを設定します (例: dc=com,dc=fishworks,ou=Engineering,CN=myhome)。
- ユーザー - すべてのユーザーまたは指定したユーザー用の自動ホーム規則を設定します。ユーザーを指定する際の「&」および「?」のワイルドカードは、ユーザーのログインとそれに対応する最初の文字を意味します。
- ディレクトリ - その規則のディレクトリを設定します (例: /export/wdp)。

## ▼ SMB の自動ホーム規則の追加

1. 自動ホーム規則を追加するには `create` コマンドを使用し、既存の規則を一覧表示するには `list` コマンドを使用します。この例では、ユーザー「Bill」に対する規則を追加してから、規則を一覧表示します。

```
twofish:> configuration services smb
twofish:configuration services smb> create
twofish:configuration services rule (uncommitted)> set use_nss=false
twofish:configuration services rule (uncommitted)> set user=Bill
twofish:configuration services rule (uncommitted)> set directory=/export/wdp
twofish:configuration services rule (uncommitted)> set container="dc=com,dc=fishworks,
ou=Engineering,CN=myhome"
twofish:configuration services rule (uncommitted)> commit
twofish:configuration services smb> list
```



```

RULE      NSS      USER      DIRECTORY      CONTAINER
rule-000  false   Bill      /export/wdp    dc=com,dc=fishworks,
           ou=Engineering,CN=myhome

```

2. ワイルドカード文字を使用すると、自動ホーム規則を作成できます。& 文字はユーザーのユーザー名に相当し、? 文字はユーザーのユーザー名の最初の文字に相当します。次の例では、ワイルドカードを使ってすべてのユーザーを適合させます。

```

twofish:configuration services smb> create
twofish:configuration services rule (uncommitted)> set use_nss=false
twofish:configuration services rule (uncommitted)> set user=*
twofish:configuration services rule (uncommitted)> set directory=/export/?/&
twofish:configuration services rule (uncommitted)> set container="dc=com,dc=fishworks,
           ou=Engineering,CN=myhome"
twofish:configuration services rule (uncommitted)> commit
twofish:configuration services smb> list
RULE      NSS      USER      DIRECTORY      CONTAINER
rule-000  false   Bill      /export/wdp    dc=com,dc=fishworks,
           ou=Engineering,CN=myhome

```

3. ネームサービススイッチを使用して自動ホーム規則を作成することもできます。

```

twofish:configuration services smb> create
twofish:configuration services rule (uncommitted)> set use_nss=true
twofish:configuration services rule (uncommitted)> set container="dc=com,dc=fishworks,
           ou=Engineering,CN=myhome"
twofish:configuration services rule (uncommitted)> commit
twofish:configuration services smb> list
RULE      NSS      USER      DIRECTORY      CONTAINER
rule-000  true    Bill      /export/wdp    dc=com,dc=fishworks,
           ou=Engineering,CN=myhome

```

## SMB のローカルグループ

ローカルグループとは、追加の権限が付与されるドメインユーザーのグループです。

表 11-11 SMB のローカルグループ

グループ	説明
管理者	管理者は、ファイルの所有権を変更するためのファイルアクセス権を必要としません。
バックアップオペレータ	バックアップオペレータは、ファイルのバックアップと復元のためのファイルアクセス制御を必要としません。

### ▼ SMB ローカルグループへのユーザーの追加

- ユーザーを追加するには、次を実行します。

```
twofish:configuration services smb> groups
twofish:configuration services smb groups> create
twofish:configuration services smb member (uncommitted)> set user=Bill
twofish:configuration services smb member (uncommitted)> set group="Backup Operators"
twofish:configuration services smb member (uncommitted)> commit
twofish:configuration services smb groups> list
MEMBER      USER              GROUP
member-000  WINDOMAIN\Bill    Backup Operators
```

## SMB のローカルアカウント

ローカルアカウントおよびユーザー ID は Windows のユーザー ID に対応付けられません。*guest* アカウントは特別な読み取り専用のアカウントであり、アプライアンスで読み取り/書き込み用に構成することはできません。

## SMB の MMC の統合

Microsoft 管理コンソール (MMC) とは、スナップインと呼ばれる、登録されているコンポーネントの拡張フレームワークであり、ネットワーク上のローカルシステムとリモートシステムの両方に包括的な管理機能を提供します。コンピュータの管理は、Microsoft 管理コンソールツールの集まりであり、ローカルとリモートのサービスやリソースの構成、モニター、および管理に使用できます。

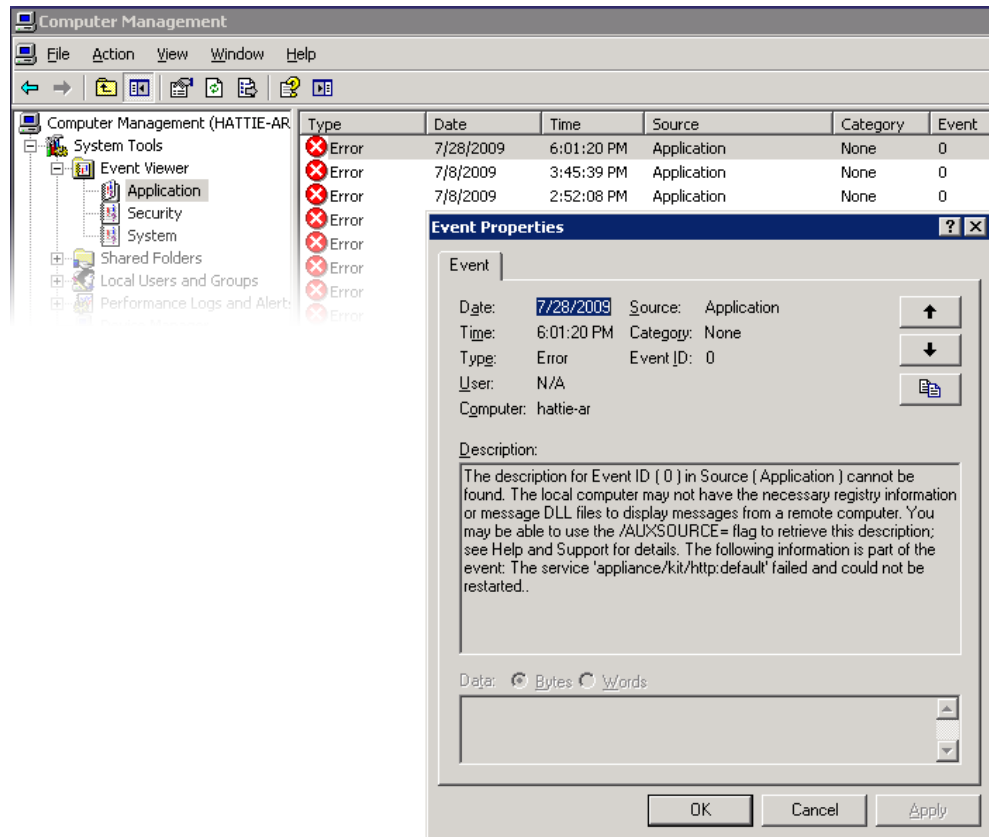
Sun ZFS Storage 7000 アプライアンスのワークグループモードで MMC 機能を使用するためには、管理コンソールを使用する Windows 管理者をアプライアンスの [210 ページの「ローカルグループ」](#) に必ず追加してください。追加しない場合は、MMC を使用してアプライアンスへの接続を試みたときに、*Access is denied* またはそれに類似したエラーが管理クライアントで表示されることがあります。

Sun ZFS Storage 7000 アプライアンスでは、次のコンピュータの管理機能をサポートしています。

## SMB のイベントビューア

イベントビューア MMC スナップインを使用して、アプリケーションログ、セキュリティーログ、およびシステムログの表示がサポートされます。これらのログには、Sun ZFS Storage 7000 システムの警告ログ、監査ログ、およびシステムログの内容が表示されます。次のスクリーンショットは、エラーイベントのアプリケーションログとプロパティダイアログを示しています。

図 11-3 SMB のイベントビューア



## SMB のシェア管理

シェア管理でサポートされている機能は次のとおりです。

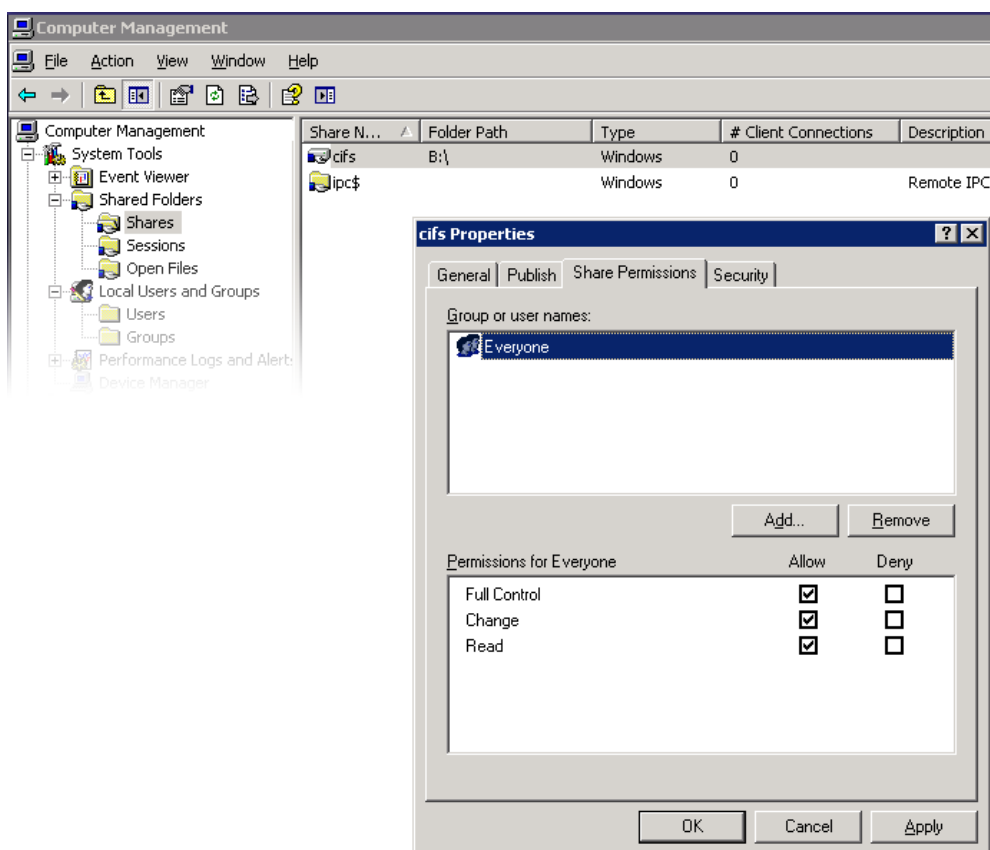
- シェアの表示
- シェアでの ACL の設定
- シェアアクセス権の変更
- シェアの記述の設定

MMC によって現在サポートされていない機能は次のとおりです。

- シェアの追加または削除
- クライアント側のキャッシュプロパティの設定

- ユーザープロパティの最大許容数の設定

図 11-4 「SMB シェアのアクセス権」プロパティ



## SMB のユーザー、グループ、および接続

次の機能がサポートされています。

- ローカル SMB ユーザーおよびグループの表示
- ユーザーの接続のリスト (1 つの接続で開いているファイルの数のリストなど)
- ユーザーの接続のクローズ
- 開いているファイルのリスト (ファイルモードやファイルオープンモードでのロックの数のリストなど)
- 開いているファイルのクローズ

図 11-5 1つの接続で開いているファイル

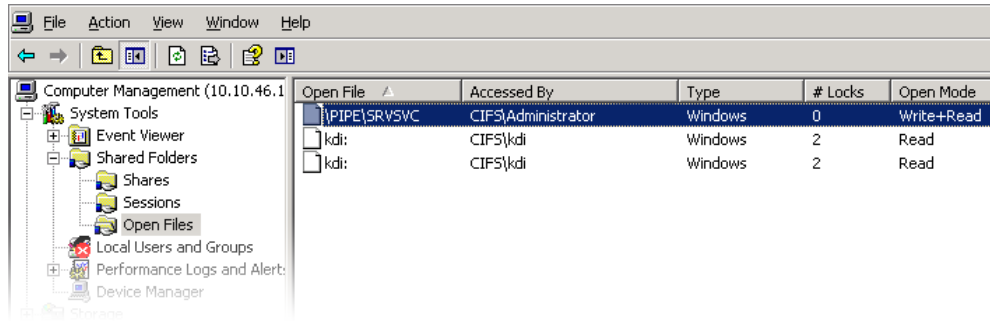
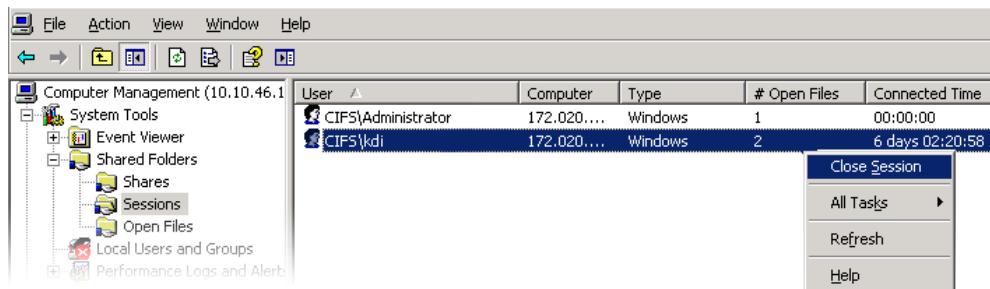


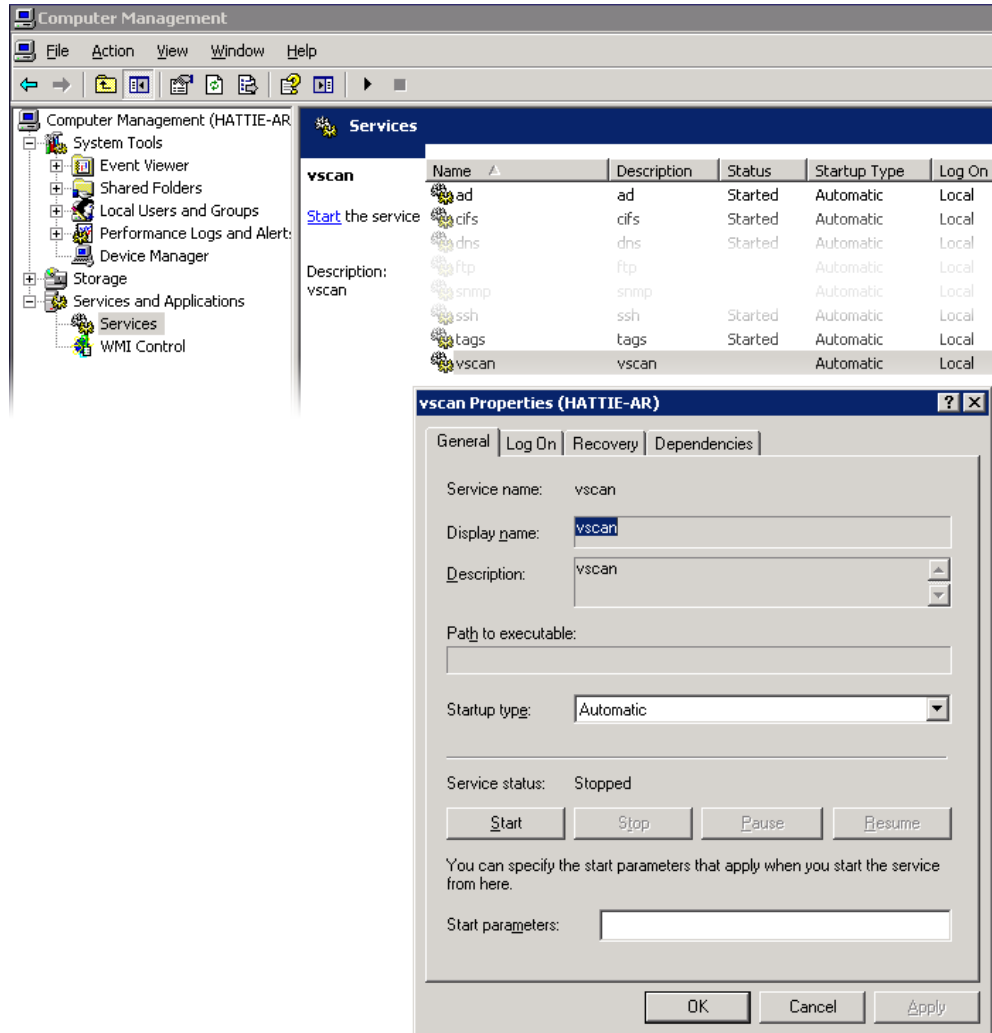
図 11-6 開いているセッション



## SMB サービスの一覧表示

ZFSSA サービスの一覧表示がサポートされています。コンピュータの管理 MMC アプリケーションを使用してサービスを有効または無効にすることはできません。次のスクリーンショットは、vscan サービスの「一般」プロパティを示しています。

図 11-7 vscan のプロパティ



適切なユーザーだけが管理操作にアクセスできるようにするために、MMC を使用してリモートで実行される操作に対していくつかのアクセス制限が設けられています。

表 11-12 ユーザーおよび許可される操作



ユーザー	許可される操作
一般ユーザー	シェアの表示。
「管理者」または「パワーユーザー」グループのメンバー	シェアの管理、ユーザーの接続の表示。


ユーザー	許容される操作
「管理者」グループのメンバー	開いているファイルのリストとファイルのクローズ、ユーザーの接続の切斷、サービスとイベントログの表示。

## BUI を使用した SMB の構成

### ▼ 初期構成

アプライアンスの初期構成は BUI または CLI を使って行うことができ、20 分もかかりません。初期設定は、あとで BUI の「保守」>「システム」コンテキストまたは CLI を使用して再度行うこともできます。通常、初期構成は次の BUI 手順を使って行われます。

1. ネットワークデバイス、データリンク、およびインタフェースを構成します。
2. データリンクの追加またはインタフェースの  アイコンを使用するか、データリンクまたはインタフェースリストへのデバイスのドラッグ&ドロップを使用して、インタフェースを作成します。
3. 必要なプロパティを設定し、「適用」ボタンをクリックしてそれらをリストに追加します。
4. 各インタフェースをそれぞれアクティブまたはスタンバイに設定します。
5. ページのいちばん上にある「適用」ボタンをクリックして、変更を確定します。
6. DNS を構成します。
7. ベースドメイン名を指定します。
8. ドメイン名前空間の Active Directory 部分でホスト名とサーバーのレコードを解決できる 1 つ以上のサーバーの IP アドレスを指定します。
9. クロック同期が確保されるように NTP 認証鍵を構成します。
10.  アイコンをクリックして、新しい鍵を追加します。
11. 新しい鍵の番号、種類、および非公開の値を指定して、変更を適用します。その鍵が、指定した各 NTP サーバーの横にオプションとして表示されます。




12. その鍵を適切な NTP サーバーに関連付けて、変更を適用します。クロック同期を確保するには、同じ NTP サーバーを使用するようにアプライアンスと SMB クライアントを構成します。
13. Active Directory をユーザーとグループのディレクトリサービスとして指定します。
14. ディレクトリドメインを設定します。
15. 「適用」ボタンをクリックして変更を確定します。
16. ストレージプールを構成します。
17.  アイコンをクリックして、新しいプールを追加します。
18. プール名を設定します。
19. 「ストレージの割り当ておよび検証」画面で、ストレージプールの JBOD 割り当てを構成します。JBOD の割り当ては、なし、半分、または全体のいずれかです。JBOD が検出されない場合は、JBOD の配線や電源を確認してください。
20. 「コミット」ボタンをクリックして、次の画面に進みます。
21. 「追加されたストレージを構成」画面で、目的のデータプロファイルを選択します。それぞれのデータプロファイルは、利用度、パフォーマンス、および容量の点で格付けされます。これらの格付けを使用して、ビジネスニーズにもっとも適した構成を判断します。
22. 「コミット」ボタンをクリックして構成を有効にします。
23. リモートサポートを構成します。
24. アプライアンスが直接インターネットに接続されていない場合は、リモートサポートサービスが Oracle との通信に使用する HTTP プロキシを構成します。
25. オンラインアカウントのユーザー名とパスワードを入力します。確認のためにプライバシーに関する説明が表示されます。
26. 登録先のインベントリチームを選択します。各アカウントのデフォルトチームは、アカウントのユーザー名に接頭辞「\$」を付けたものと同じです。
27. 初期構成の変更を確定します。





## ▼ Active Directory の構成

1. Active Directory ドメインにアプライアンスのアカウントを作成します。詳細な手順については、Active Directory のドキュメントを参照してください。
2. 「構成」>「サービス」>「Active Directory」画面で、「ドメインに参加」ボタンをクリックします。
3. Active Directory ドメイン、管理ユーザー、管理パスワードを指定し、「適用」ボタンをクリックして変更を確定します。

## ▼ プロジェクトとシェアの構成

1. プロジェクトを作成します。
2. 「シェア」画面で、 アイコンをクリックして「プロジェクト」パネルを展開します。
3. 「追加...」リンクをクリックして新しいプロジェクトを追加します。
4. プロジェクト名を指定して、変更を適用します。
5. 「プロジェクト」パネルから新しいプロジェクトを選択します。
6.  アイコンをクリックして、ファイルシステムを追加します。
7. そのファイルシステムに対して  アイコンをクリックします。
8. 「一般」リンクをクリックし、「プロジェクトから継承」チェックボックスのチェックマークを外します。
9. /export 下のマウントポイントを選択します (SMB シェアがリソース名でアクセスされる場合も含む)。
10. プロジェクトの「プロトコル」画面で、リソース名をオンに設定します。
11. プロジェクトの sharesmb および「シェアレベル ACL」を有効にします。
12. 「適用」ボタンをクリックして構成を有効にします。

## ▼ SMB データサービスの構成

1. 「構成」>「サービス」>「SMB」画面で、 アイコンをクリックして、サービスを有効にします。
2. このページのプロパティのセクションに記載されている推奨事項に従って SMB プロパティを設定し、「適用」ボタンをクリックして構成を有効にします。
3. 「構成」>「サービス」>「SMB」画面の「自動ホーム」リンクをクリックし、上記の自動ホーム規則のセクションに記載されている説明に従って、SMB クライアントがホームディレクトリに対応付けられるように自動ホーム規則を設定します。次に、「適用」ボタンをクリックして構成を有効にします。
4. 「構成」>「サービス」>「SMB」画面の「ローカルグループ」リンクをクリックし、 アイコンをクリックして、上記のローカルグループのセクションに記載されている説明に従って、管理者またはバックアップオペレータユーザーをローカルグループに追加します。次に、「適用」ボタンをクリックして構成を有効にします。

## FTP サービス

FTP (ファイル転送プロトコル) サービスでは、FTP クライアントからファイルシステムにアクセスできます。匿名ログインは許可されません。ユーザーは「サービス」でどのネームサービスが構成されていても認証を行う必要があります。

## FTP のプロパティ

### FTP の一般設定

表 11-13 FTP の一般設定

プロパティ	説明
ポート (受信接続用)	FTP が待機するポート。デフォルトは 21 です
最大接続数 (無限の場合「0」)	同時に実行される FTP 接続の最大数です。この値は、予測される並行ユーザーの数に及ぶように設定します。デフォルトは 30 です。各接続でシステムプロセスが作成され、あまりにも多くの (何千もの) 接続を許可すると、DoS 攻撃が発生する可能性があるからです

プロパティ	説明
タイミング攻撃を防ぐために遅延エンジンをオンにする	認証中に小さな遅延を挿入して、タイミング測定によるユーザー名推測の試みを欺きます。これをオンにすると、セキュリティが向上します
デフォルトログイン root	FTP ログインの場所です。デフォルトは「/」で、シェア階層の最上位を指します。すべてのユーザーは、FTP サービスの認証に成功したあとでこの場所にログインします
ログインレベル	proftpd ログの詳細レベルです。
新規作成されたファイルとディレクトリからマスクするアクセス権	ファイルが作成されるときに削除するファイルアクセス権です。最新のアップロードがだれでも書き込み可能にならないように、group と world の書き込みがデフォルトで非表示になります

## FTP のセキュリティ設定

表 11-14 FTP のセキュリティ設定

プロパティ	説明
SSL/TLS を有効化	SSL/TLS で暗号化された FTP 接続を許可します。これによって FTP トランザクションが確実に暗号化されます。デフォルトは無効です。
受信 SSL/TLS 接続用ポート	SSL/TLS で暗号化された FTP サービスが待機するポート。デフォルトは 21 です。
root ログインを許可	root ユーザーの FTP ログインを許可します。これはデフォルトでオフになっています。FTP 認証は平文であり、ネットワーク盗聴攻撃のセキュリティリスクをもたらすからです
許容可能なログイン試行の最大数	失敗したログイン試行の回数。その回数を超えると、FTP 接続が切断されるため、ユーザーは再接続して再度試す必要があります。デフォルトは 3 です
外部データ接続アドレスを許可	外部 FTP 接続を許可して、FTP サーバー間の直接ファイル転送を可能にします。このプロパティはデフォルトでオフになっています。

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## FTP のログ

表 11-15 FTP のログ

ログ	説明
proftpd	FTP イベント (成功したログインおよび失敗したログイン試行を含む) を記録します
proftpd_xfer	ファイル転送のログです
proftpd_tls	SSL/TLS 暗号化に関連する FTP イベントを記録します

## BUI を使用した FTP の構成

### ▼ シェアへの FTP アクセスの許可

1. 「構成」->「サービス」に移動します
2. FTP サービスが有効でオンラインになっていることを確認します。有効でない場合は、サービスを有効にします。
3. 「シェア」画面でシェアを選択または追加します。
4. 「プロトコル」セクションに移動し、FTP アクセスが有効になっていることを確認します。この画面では、アクセスモード (読み取り、読み取り/書き込み) の設定も行えます。

## HTTP サービス

HTTP サービスでは、HTTP プロトコル、HTTPS プロトコル、および HTTP の拡張 WebDAV (Web ベースの分散オーサリングおよびバージョン管理) を使用してファイルシステムにアクセスできます。このサービスにより、クライアントは Web ブラウザを介してシェアファイルシステムにアクセスすることも、ローカルファイルシステムとして (クライアントソフトウェアでサポートされている場合) シェアファイルシステムにアクセスすることもできます。これらの HTTP および HTTPS シェアにアクセスするための URL の形式は、それぞれ次のようになっています。

`http://hostname/shares/mountpoint/share_name`

`https://hostname/shares/mountpoint/share_name`

HTTPS サーバーは、自己署名付きセキュリティ証明書を使用します。

## HTTP のプロパティ

表 11-16 HTTP のプロパティ

プロパティ	説明
クライアントログインが必要	シェアアクセスが許可されるためにはクライアントの認証が必要です。クライアントで作成されるファイルにはその所有権が割り当てられます。これを設定しない場合、作成されるファイルは HTTP サービスがユーザー「nobody」を使って所有します。後述の認証に関するセクションを参照してください。
プロトコル	サポートするアクセス方法 (HTTP、HTTPS、または両方) を選択します。
HTTP ポート (受信接続用)	HTTP ポート。デフォルトは 80 です
HTTPS ポート (セキュアな受信接続用)	HTTPS ポート。デフォルトは 443 です

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## HTTP の認証とアクセス制御

「クライアントログインが必要」オプションを有効にした場合、アプライアンスはローカルユーザー、NIS ユーザー、または LDAP ユーザーに有効な認証資格を提供しないクライアントへのアクセスを拒否します。Active Directory 認証はサポートされていません。

基本的な HTTP 認証だけがサポートされています。HTTPS を使用していなければ、ユーザー名とパスワードは暗号化されていない状態で送信されます。その方法がすべての環境に適切であるとは限りません。

通常、認証されたユーザーには、NFS や FTP で割り当てられるのと同じアクセス権が HTTP で割り当てられます。ほかのプロトコルで説明したように、認証されたユーザーが作成したファイルやディレクトリはそのユーザーが所有します。特権ユーザー (99 以下の UID 番号を持つユーザー) は、アクセス制御の目的で「nobody」とみなされます。特権ユーザーが作成したファイルは「nobody」が所有します。

「クライアントログインが必要」オプションを無効にした場合、アプライアンスは、クライアントが資格を提供しても、その認証を試みません。新しく作成されたファイルは「nobody」が所有し、すべてのユーザーはアクセス制御の目的で「nobody」とみなされます。

認証の有無にかかわらず、作成されたファイルやディレクトリでアクセス権はマスクされません。作成されたファイルには UNIX アクセス権 666 (すべてのユーザーによる読み取りと書き込みが可能) が割り当てられ、作成されたディレクトリには UNIX アクセス権 777 (すべてのユーザーによる読み取り、書き込み、実行が可能) が割り当てられます。

## HTTP のログ

表 11-17 HTTP のログ

ログ	説明
network-http:apache22	HTTP サービスのログです

## HTTP の構成

### ▼ シェアへの HTTP アクセスの許可

1. 「構成」->「サービス」に移動します
2. HTTP サービスが有効でオンラインになっていることを確認します。有効でない場合は、サービスを有効にします。
3. 「シェア」画面でシェアを選択または追加します。
4. 「プロトコル」セクションに移動し、HTTP アクセスが有効になっていることを確認します。この画面では、アクセスモード (読み取り、読み取り/書き込み) の設定も行えます。

## NDMP サービス

NDMP (Network Data Management Protocol) サービスを使用すると、システムは、データ管理アプリケーション (DMA) と呼ばれるリモート NDMP クライアントで制御される NDMP ベースのバックアップおよび復元操作に参加できます。NDMP を使用すると、アプライアンスのユーザーデータ (アプライアンスで管理者が作成したシェアに格納されているデータ) をローカルに接続されたテープデバイスとリモートシステムの両方にバックアップ

および復元することができます。ローカルに接続されたテープデバイスは、リモートシステムへのバックアップや復元のために DMA に公開することもできます。

NDMP は、システム構成データのバックアップや復元には使用できません。代わりに、`[[Maintenance:System:ConfigurationBackup]`構成のバックアップおよび復元]]機能を使用してください。

## NDMP のローカル構成とリモート構成の違い

アプライアンスでは、ローカル構成 (テープドライブがアプライアンスに物理的に接続されている) およびリモート構成 (データが同じネットワーク上の別のシステムにストリーミングされる) の両方を使用したバックアップと復元をサポートしています。どちらの場合も、サポートされている DMA でバックアップを管理する必要があります。

ローカル構成では、サポートされているテープデバイス (ドライブとチェンジャー (ロボット) の両方を含む) は、サポートされている SCSI またはファイバチャネル (FC) カード (イニシエータモードで構成) を使ってシステムに物理的に接続されます。これらのデバイスは、58 ページの「NDMP のステータス」画面に表示することができます。DMA がデバイスを走査すると、NDMP サービスによってこれらのデバイスが DMA に提供されます。これらのデバイスを DMA で構成すると、アプライアンスまたは同じネットワーク上のほかのシステムのバックアップや復元にそれらを使用できるようになります。テープドライブまたはチェンジャーをシステムに追加したり、そのようなデバイスをシステムから削除したりしたあとで、変更が NDMP サービスで認識されるようにするには、リポートが必要な場合があります。その後、テープデバイス名が変わった可能性があるため、DMA の再構成が必要となる場合があります。

リモート構成では、テープデバイスはバックアップまたは復元の対象となるシステム (データサーバー) に物理的には接続されず、DMA が実行されているシステムまたは別個のシステム (テープサーバー) に接続されます。DMA がほかの 2 つのシステムを制御するため、これらは一般に「3 ウェイ構成」と呼ばれます。この構成では、データストリームは IP ネットワークを介してデータサーバーとテープサーバーの間で転送されます。

## NDMP のバックアップの形式とタイプ

NDMP プロトコルでは、バックアップデータの形式を指定しません。アプライアンスでは、それぞれの実装に対応する 3 つのバックアップタイプとオンテープ形式をサポートします。DMA では、NDMP 環境変数「TYPE」に次の値を指定してバックアップタイプを選択できます。

表 11-18 NDMP のバックアップの形式とタイプ

バックアップタイプ	詳細
dump	ファイルシステム専用のファイルベース。ファイル履歴とダイレクトアクセスリカバリ (DAR) をサポートしません。
tar	ファイルシステム専用のファイルベース。ファイル履歴とダイレクトアクセスリカバリ (DAR) をサポートしません。
zfs	ファイルシステムとボリューム用のシェアベース。ファイル履歴やダイレクトアクセスリカバリ (DAR) をサポートしませんが、一部のデータセットではバックアップ速度が向上することがあります。NDMPv4 でのみサポートされます。

標準の NDMP データストリーム形式は存在しないため、アプライアンスで生成されるバックアップストリームは、互換性のあるソフトウェアが実行されている 7000 シリーズのアプライアンスでのみ復元できます。アプライアンスソフトウェアの将来のバージョンでは通常、そのソフトウェアの古いバージョンからバックアップされたストリームを復元できますが、その反対が必ずしも当てはまるとは限りません。たとえば、「zfs」バックアップタイプが 2010.Q3 で新しく追加された場合、2010.Q1 以前が実行されているシステムで、2010.Q3 の「zfs」タイプを使って作成されたバックアップストリームを復元することはできません。

## NDMP の「dump」および「tar」を使用したバックアップ

「dump」および「tar」バックアップタイプを使用してバックアップを行う場合、管理者はバックアップパスと呼ばれるファイルシステムパスによってバックアップするデータを指定します。たとえば、管理者が `/export/home` のバックアップを構成した場合、そのパスにマウントされたシェアがバックアップされます。同様に、バックアップストリームが `/export/code` に復元される場合、ファイルは、別のパスからバックアップされていてもそのパスに復元されません。

バックアップに指定できるのは、既存のシェアのマウントポイントであるパスか、または既存のシェアの中に含まれているパスだけです。バックアップパスがシェアのマウントポイントと一致する場合は、そのシェアだけがバックアップされます。一致しなければ、パスはシェアの中に含まれている必要があり、その場合は、そのパスの下にあるそのシェアの部分だけがバックアップされます。どちらの場合も、バックアップパス下の指定されたシェア内にマウントされたほかのシェアはバックアップされません。これらのシェアはバックアップ対象として個別に指定する必要があります。



スナップショット - バックアップパスがライブファイルシステム (`/export/code` など) またはライブファイルシステムの中に含まれるパス (`/export/code/src` など) を示す場合、アプライアンスはただちに新しいスナップショットを取得し、そのスナップショットから指定のパスをバックアップします。バックアップが完了すると、スナップショットは破棄されます。バックアップパスがスナップショット (`/export/code/.zfs/snapshot/mysnap` など) を示す場合、新しいスナップショットは作成されず、システムは指定のスナップショットからバックアップします。

シェアメタデータ - 複雑なシェア構成のバックアップと復元を簡略化するため、「dump」と「tar」のバックアップには、バックアップパスに関連付けられたプロジェクトおよびシェア用のシェアメタデータが含まれています。このメタデータには、プロトコルシェアプロパティ、割り当て制限プロパティ、「シェア」画面で構成されるその他のプロパティなど、アプライアンスのシェア構成が記述されています。これを、NDMP でもバックアップおよび復元される、ディレクトリ構造やファイルアクセス権のようなファイルシステムメタデータと混同しないようにしてください。

たとえば、`/export/proj` をバックアップする場合は、マウントポイントが `/export/proj` で始まるすべてのシェアのシェアメタデータと、その親プロジェクトのシェアメタデータがバックアップされます。同様に、`/export/someshare/somedir` をバックアップする場合は、シェアが `/export/someshare` にマウントされていれば、そのシェアとそのプロジェクトのシェアメタデータがバックアップされます。

復元時は、復元パスの宛先が既存のシェアの中に含まれていない場合、必要に応じてバックアップストリーム内のプロジェクトとシェアが、バックアップに格納されているそれぞれの元のプロパティを使って再作成されます。たとえば、プロジェクト `proj1` とシェア `share1` および `share2` が含まれる `/export/foo` をバックアップしたあとで、そのプロジェクトを破棄してバックアップから復元する場合、これらの 2 つのシェアとプロジェクトは、復元操作の一部としてバックアッププロパティを使用して再作成されます。

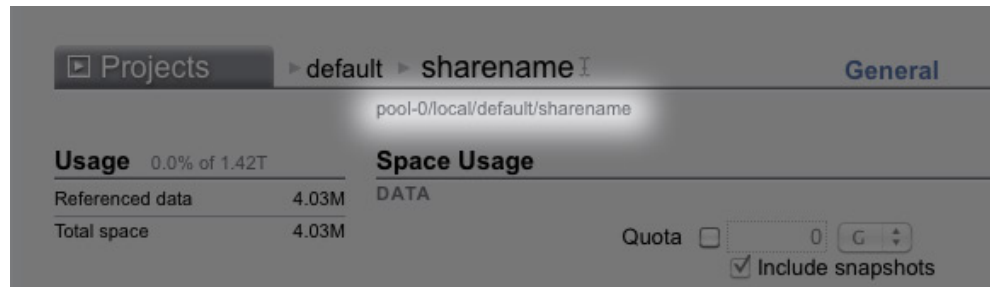
復元中は、自動的に再作成されるプロジェクトが存在すれば、その既存のプロジェクトが使用され、新しいプロジェクトが自動的に作成されることはありません。自動的に再作成されるシェアが存在し、そのマウントポイントが、アプライアンスで元のバックアップパスと復元の宛先に基づいて予測されるものと一致する場合は、その既存のシェアが使用され、新しいシェアが自動的に作成されることはありません。そうでない場合は、バックアップのメタデータから新しいシェアが自動的に作成されます。同じ名前のシェア (マウントポイントは異なる) がすでに存在する場合は、新しく作成されるシェアに「`ndmp-`」で始まる一意の名前と適切なマウントポイントが付けられます。

アプライアンスにもう存在しないデータセットを含むストリームを復元して、アプライアンスでそのバックアップストリームに指定されたとおりにデータセットを再作成できるようにするか、復元先のシェアを事前に作成しておくことをお勧めします。このどちらかを行うことで、前述の自動シェア作成に関連する予期しない結果を回避できます。

## NDMP の「zfs」を使用したバックアップ

「zfs」タイプを使用してバックアップを行う場合、管理者はアプライアンスでバックアップするデータを指定するときにその正規名を使用します。BUI では、正規名はシェアの名前の下に表示されます。

図 11-8 NDMP のシェアの名前



CLI では、正規名は `canonical_name` プロパティの値で確認できます。正規名は「/」で始まりませんが、バックアップパスの構成時は正規名に接頭辞「/」を付ける必要があります。

プロジェクトとシェアのどちらも「zfs」タイプを使ったバックアップに指定できます。正規名をそのまま指定する場合は、新しいスナップショットが作成されてバックアップに使用されます。接尾辞「@snapshot」を使用すると、特定のスナップショットをバックアップ用に指定できます。その場合は、新しいスナップショットは作成されず、指定したスナップショットがバックアップされます。次に例を示します。

表 11-19 正規名とバックアップされるシェア

正規名	バックアップされるシェア
pool-0/local/default	「default」というローカルプロジェクトとそのすべてのシェアの新しいスナップショット。
pool-0/local/default@yesterday	ローカルプロジェクト「default」の「yesterday」という名前の付いたスナップショットと、スナップショット「yesterday」を含むすべてのシェア。
pool-0/local/default/code	ローカルプロジェクト「default」に含まれるシェア「code」の新しいスナップショット。「code」にはファイルシステムまたはボリュームを指定できます。
pool-0/local/default/code@yesterday	ローカルプロジェクト「default」に含まれるシェア「code」の「yesterday」という名前の付いたスナップ

正規名	バックアップされるシェア
	ショット。「code」にはファイルシステムまたはボリュームを指定できます。

「zfs」バックアップタイプを使用したレベルベースの増分バックアップには以前の増分バックアップから作成されるベーススナップショットが必要なため、新しいスナップショットが作成されるレベルバックアップでは、デフォルトで新しいスナップショットがその後の増分バックアップに使用できるように保持されます。DMA で UPDATE=n の設定により、バックアップがその後の増分バックアップに使用されないことを指定した場合、新しく作成されたスナップショットはバックアップ後に破棄されます。既存のユーザースナップショットがバックアップ後に破棄されることはありません。詳細は、後述の「増分バックアップ」を参照してください。

シェアメタデータ - シェアメタデータ (シェアの構成) は、常に「zfs」バックアップに含まれています。「zfs」タイプを使用してフルバックアップを復元する場合、復元先のプロジェクトまたはシェアはもう存在していないはずですが、復元先のプロジェクトやシェアは、バックアップストリームのメタデータから再作成されます。「zfs」タイプを使用して増分バックアップを復元する場合、復元先のプロジェクトまたはシェアはすでに存在しているはずですが、プロパティが、バックアップストリームのメタデータから更新されます。詳細は、後述の「増分バックアップ」を参照してください。

## NDMP の増分バックアップ

アプライアンスでは、上記のすべてのバックアップタイプを対象にレベルベースの増分バックアップをサポートしています。レベルバックアップを指定するには、通常 DMA で次の 3 つの環境変数を指定します。

変数	詳細
LEVEL	バックアップレベルを識別する 0 - 9 の整数です。
DMP_NAME	特定の増分バックアップセットを指定します。DMP_NAME に異なる値を指定することで、複数のレベル増分バックアップセットを同時に使用できます。
UPDATE	このバックアップをその後の増分バックアップのベースとして使用できるかどうかを指定します

定義によると、レベル N のバックアップには、N よりも小さい LEVEL を使用した同じシェアの同じバックアップセット (「DMP\_NAME」で指定) の前回のバックアップ以降に変更されたすべてのファイルが含まれます。レベル 0 のバックアップには常にすべてのファイル

が含まれます。UPDATE の値が「y」(デフォルト) の場合は、現在のバックアップが記録され、N よりも大きいレベルの将来のバックアップではこのバックアップがベースとして使用されます。これらの変数は通常 DMA によって管理されるため、管理者が直接構成する必要はありません。

増分バックアップスケジュールの例を次に示します。

表 11-20 増分バックアップスケジュールの例

日	詳細
月初め	レベル 0 のバックアップ。バックアップには、シェア内のすべてのファイルが含まれます。
各月の 7 日、14 日、21 日	レベル 1 のバックアップ。バックアップには、最後のフル (月次) バックアップ以降に変更されたすべてのファイルが含まれます
毎日	レベル 2 のバックアップ。バックアップには、最後のレベル 1 のバックアップ以降に変更されたすべてのファイルが含まれます

その月の 24 日になったときにファイルシステムの状態を回復するには、通常、管理者はその月の 1 日に作成したレベル 0 のバックアップを新しいシェアに復元し、次にその月の 21 日に作成したレベル 1 のバックアップを復元し、最後にその月の 24 日に作成したレベル 2 のバックアップを復元します。

レベルベースの増分バックアップを実装するには、アプライアンスでシェアごとにレベルバックアップ履歴を追跡する必要があります。「tar」と「dump」のバックアップでは、レベルバックアップ履歴はシェアメタデータに保持されます。増分バックアップではファイルシステムがトラバースされ、前回のレベルバックアップ以降に変更されたファイルがバックアップに含まれます。復元時には、システムはバックアップストリーム内のすべてのファイルを復元するだけです。したがって、上記の例では 24 日のレベル 2 のバックアップを任意のファイルシステムに復元することが可能で、そのバックアップストリームに含まれるファイルは、復元先のファイルシステムがそのファイルのバックアップ元のファイルシステムと一致しなくても復元されます。ただし、元のファイルシステムの状態を回復するためには、空のツリーから始めて以前のレベルバックアップを復元する上記のような手順を使用することを最良事例としてお勧めします。

「zfs」タイプの効率的なレベルベースの増分バックアップを実施するには、別の方法が使用されます。増分セットに含まれているバックアップでは、バックアップに使用されたスナップショットを破棄するのではなくシステムに残します。その後の増分バックアップでは、このスナップショットをベースとして使用し、変更されたファイルシステムブロックをすばやく特定して、バックアップストリームを生成します。結果として、その後の増分バックアップを作成

する場合は、バックアップ後に NDMP サービスによって残されたスナップショットを破棄してはいけません。

この動作のもう 1 つの重要な結論は、増分ストリームを復元するためには、ファイルシステムの状態が、増分ストリームのベーススナップショットでの状態とぴったり一致する必要があります。つまり、レベル 2 のバックアップを復元するためには、ファイルシステムが前回のレベル 1 のバックアップが完了したときとまったく同じようになっている必要があります。上記の一般的な手順では、24 日のレベル 2 のバックアップストリームの復元時にシステムの状態は 21 日のレベル 1 のバックアップが完了したときとまったく同じであるため（そのバックアップが復元されたばかりなので）、このことが保証されます。

NDMP サービスでは、「zfs」の増分バックアップストリームを、最新のスナップショットが増分ストリームのベーススナップショットと一致しないファイルシステムに復元しようとした場合、またはファイルシステムがそのスナップショット以降に変更されている場合にエラーを報告します。復元が開始される直前にベーススナップショットにロールバックするよう NDMP サービスを構成できます。そのためには、NDMP 環境変数「ZFS\_FORCE」に値「y」を指定するか、NDMP サービスの「データセットをロールバック」プロパティを構成し（後述の「プロパティ」を参照）。

## NDMP のプロパティ

NDMP サービスの構成は、次のプロパティで構成されています。

表 11-21 NDMP のプロパティ

プロパティ	説明
バージョン	使用している DMA でサポートされている NDMP のバージョンです。
TCP ポート (v4 のみ)	NDMP のデフォルトの接続ポートは 10000 です。NDMPv3 では常にこのポートを使用します。NDMPv4 では必要に応じて別のポートも使用できます。
デフォルト復元プール	「tar」または「dump」を使用して完全な復元を実行する場合、システムは、ターゲットにシェアがマウントされていない場合はデータセットを再作成します。NDMP プロトコルではマウントポイントしか指定しないため、システムはプロジェクトやシェアを再作成するプールを選択します。複数のプールを備えたシステムでは、このプロパティを使用して 1 つ以上のプールを指定できます。複数のプールを指定する必要があるのは、各ヘッドにアクティブなプールを備えたクラスターだけです。このリストが、ストレージのすべての構成変更と同期された状態に維持されるようにする必要があります。

プロパティ	説明
	<p>す。プールが存在しないか、またはどのプールもオンラインになっていない場合、システムはランダムにデフォルトプールを選択します。</p>
<p>メタデータのみの変更を無視</p>	<p>内容が変更されたファイルのみをバックアップし、アクセス権や所有権などのメタデータしか変更されていないファイルは無視するようシステムに指示します。このオプションは「tar」と「dump」の増分バックアップにだけ適用され、デフォルトでは無効になっています。</p>
<p>トークンベースのバックアップを許可</p>	<p>ZFS バックアップ用のトークンベースの方法を有効または無効にします。このプロパティはデフォルトでオフになっています。</p>
<p>復元前の ZFS ロールバック (v4 のみ)</p>	<p>「zfs」タイプのバックアップだけに適用されます。増分バックアップの復元時にターゲットのプロジェクトおよびシェアを、増分復元のベースとして使用するスナップショットにロールバックするかどうかを決めます。プロジェクトおよびシェアがロールバックされる場合、そのスナップショット以降に行われた変更はすべて失われます。この設定は通常「ZFS_FORCE」環境変数 (前述の「増分バックアップ」を参照) を介して DMA によって制御されますが、このプロパティを使って DMA 設定をオーバーライドして、常にこれらのデータセットがロールバックされるようにしたり、ロールバックされないようにしたりできます。それらをロールバックしないと、手動でロールバックされていないかぎり、復元に失敗します。このプロパティは、管理者に ZFS_FORCE などのカスタム環境変数の構成を許可しない DMA で使用します。</p>
<p>直接アクセス復元を許可</p>	<p>復元操作時に、システムが順次検索でなく、位置でファイルを見つけられるようにします。このオプションを有効にすると、多数のテープから少数のファイルを回復するのにかかる時間を短縮できます。個々のファイルをあとで回復できるようにするためには、バックアップ時にこのオプションを指定する必要があります。</p>
<p>直接パスを復元 (v3 のみ)</p>	<p>ファイルの復元時に、そのファイルだけでなく、そのファイルへの完全な絶対パスも復元されることを指定します。デフォルトでは、このオプションは無効です。</p>
<p>DMA テープモード (ローカルに接続されたドライブ用)</p>	<p>DMA が System V または BSD セマンティクス のどちらを受け付けるかを指定します。デフォルトは System V で、ほとんどの DMA にはこれが推奨されます。このオプションは、NDMP 経由でエクスポートされるローカルに接続されたテープドライブにのみ適用できます。使用している DMA がどちらのモードを受け付けるかについては、DMA のドキュメントを参照してください。このオプションを変更すると、DMA</p>

プロパティ	説明
	がデバイスを走査するときにエクスポートされるデバイスが変更されるだけなので、この設定の変更後は DMA でそのテープデバイスを再構成する必要があります。
DMA ユーザー名およびパスワード	DMA の認証に使用されます。システムは MD5 をユーザー認証に使用します

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## NDMP のログ

表 11-22 NDMP のログ

ログ	説明
system-ndmpd:default	NDMP サービスのログです

## リモートレプリケーション

リモートレプリケーションサービスは、ほかの Oracle ZFS Storage Appliance との間のプロジェクトとシェアのレプリケーションを容易にします。この機能については、[第13章「レプリケーション」](#)のドキュメントで詳しく説明されています。

このサービスを有効にすると、アプライアンスはほかのアプライアンスからレプリケーション更新を受信し、その構成されたアクションに従ってローカルのプロジェクトおよびシェアに対してレプリケーション更新を送信します。このサービスを無効にすると、受信されるレプリケーション更新が失敗し、ローカルのプロジェクトおよびシェアはレプリケートされません。

このサービスにプロパティが指定されていなくても、管理者はこのサービスを使って、このアプライアンスへのレプリケートデータを持つアプライアンスを (ソースの下に) 表示したり、このアプライアンスからレプリケートできるアプライアンスを (ターゲットの下に) 構成したりできます。リモートレプリケーションの管理の詳細は、[第13章「レプリケーション」](#)のドキュメントを参照してください。

## シャドウ移行

シャドウ移行サービスは、外部または内部ソースからのデータの自動移行を可能にします。この機能については、[第14章「シャドウ移行」](#)で詳しく説明されています。このサービス自体は、自動バックグラウンド移行を制御するだけです。このサービスが有効か無効かに関係なく、データは帯域内リクエストに合わせて同期的に移行されます。

テストを目的とする場合、またはシャドウ移行のせいでシステムにかかる負荷が大きくなりすぎている場合だけは、このサービスを無効にするようにしてください。無効にすると、ファイルシステムが移行を終了することはありません。このサービスの主な目的は、バックグラウンド移行専用のスレッドの数を調整できるようにすることです。

## シャドウ移行のプロパティ

表 11-23 シャドウ移行のプロパティ

プロパティ	説明
スレッド数	データのバックグラウンド移行専用のスレッドの数。これらのスレッドはマシン全体に影響し、その数を増やすと、リソース（ネットワーク、I/O、および CPU）の消費量が増えるという問題がありますが、並行性および移行全体の速度が向上する可能性があります。

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## SFTP サービス

SFTP (SSH File Transfer Protocol) サービスでは、SFTP クライアントからファイルシステムにアクセスできます。匿名ログインは許可されません。ユーザーは「サービス」でどのネームサービスが構成されていても認証を行う必要があります。

## SFTP のプロパティ

- ポート (受信接続用) - SFTP が待機するポート。デフォルトは 218 です
- root ログインを許可 - root ユーザーの SFTP ログインを許可します。これはデフォルトでオフになっています。



- ログインレベル - SFTP ログメッセージの詳細レベルです
- SFTP 鍵 - SFTP 認証用の RSA/DSA 公開鍵です。テキストコメントを鍵に関連付けると、鍵が追加された理由を管理者が追跡できます。2011.1 ソフトウェアリリースの時点で、SFTP の鍵管理が変更され、セキュリティが強化されました。SFTP 鍵を作成するときに、有効なユーザー割り当てで「ユーザー」プロパティを含めることが求められます。SFTP 鍵はユーザー別にグループ化され、SFTP でユーザー名を使用して認証されます。「ユーザー」プロパティが含まれていない既存の SFTP 鍵は、これまでどおり認証されますが再作成することをお勧めします。

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## SFTP ポート

SFTP サービスでは、アプライアンスへの接続に非標準のポート番号を使用します。これは、ポート 22 への管理上の SSH 接続との競合を回避するためです。デフォルトでは、SFTP ポートは 218 であり、接続前に SFTP クライアントで指定する必要があります。たとえば、SFTP を使用する OpenSolaris クライアントは次のコマンドで接続を行います。

```
manta# sftp -o "Port 218" root@guppy
```

## SFTP のログ

表 11-24 SFTP のログ

ログ	説明
network-sftp:default	SFTP サービスのイベントを記録します

## SFTP の構成

### ▼ シェアへの SFTP アクセスの許可

1. 「構成」->「サービス」に移動します
2. SFTP サービスが有効でオンラインになっていることを確認します。有効でない場合は、サービスを有効にします。

3. 「シェア」画面でシェアを選択または追加します。
4. 「プロトコル」セクションに移動し、SFTP アクセスが有効になっていることを確認します。この画面では、アクセスモード (読み取り、読み取り/書き込み) の設定も行えます。

## ▼ リモートアクセス用の SFTP サービスの構成

1. 適切な管理者ロールを持つローカルユーザーまたはネットワークユーザー (LDAP または NIS) を作成します。(第7章「ユーザー構成」を参照してください)。
2. Solaris ホスト/クライアントで `ssh-keygen -t dsa` コマンドを入力して、SSH 認証鍵を生成します。
3. その鍵を格納するファイルの名前を入力します。
4. 必要に応じてパスフレーズを入力するか、このフィールドを空白のままにして、SFTP シェアに直接ログオンします。その鍵の場所が表示されます。次のような鍵が表示されます。
5. `: ssh-dss AAAAB3NzaC1kc3MAAACBAPMMs5h8UWk1NPf/VJDDEo00AwT  
+s6iZxkCmmrgAmLfTX9izWk+`
6. `: bsvNldOIXN/6EgkusLjo/+UaEt5+704vMHCIraq3AIVHLS5tVjeX3iCs  
+fDo0qwXZg3Brh8QBAaWk3`
7. `:ywr2osull1tHh4v/HwEAHZq5mVWXav0pO3bgmxl0/  
+VAAAAFQDIJxnm52DfyEdQQMTY+jRVvzGwMQA`
8. `: AAIAhTP6Ey  
+2gGFICKkvUofsco4d8pbqH8duE9P6Y88s0+opuj52GkAdRUt2fRrdM9Cf3h4lIOc8Bw9`
9. `:  
bZIBzrCKBNWBUDZG56tsfldilW6vS6gxKrmL2v7fSp9WYPsxZGhOLfU29zW4n2W  
VcVHbGyFEoVe+taq`
10. `: aq+AYJaWoHnjZL1/  
LpQAAAIAOLc8+uc3hDOcK3pAkYdg8b2rYIGOAZU4py0rq24DGPeVHd5h5jbe4p`
11. `:WDM70uYqGCOPYiOKeEoMnJpczRX5qjl  
+BfoUY4sH24WWwsKkT8XX9PUAa0WT+7axEqg2N6YelaTJ95J`
12. `:vMaj6E7HkAlra2Sj2H/LSDktL42UL+j1Wx5A== username sunray`

13. 「構成」>「サービス」>「SFTP」に移動します。「鍵」の下にあるプラス記号 (+) をクリックします。
14. 「新しい鍵」ウィンドウで、「DSA」を選択します。
15. 鍵の部分 (上記の例では AAAA で始まり、Wx5A== で終わる部分) のみをコピーし、「鍵」フィールドに貼り付けます。ユーザー名を入力し、リマインダとしてコメントを追加します。
16. 注: 鍵には空白を含めないでください。
17. 「シェア」>「シェア」に移動し、プラス記号 (+) をクリックしてファイルシステムを作成します。
18. 「ファイルシステムを作成」ウィンドウで、ファイルシステム名 (sftp など) を入力し、シェアのアクセス権を「読み取り/書き込み」に変更して、「適用」をクリックします。
19. 鉛筆アイコンをクリックして、シェアのプロパティーを設定します。(第12章「シェア、プロジェクト、およびスキーマ」を参照してください)。
20. シェアにアクセスするには、これらの例に示すように、sftp コマンドを使用します。
21. : sftp -o "port=218" <username> 10.x.x.151:/export/sftp
22. : Connecting to 10.x.xx.151...
23. : Changing to: /export/sftp
24. : sftp>
25. : Example with -v option:
26. : sftp -v -o "IdentityFile=/home/<username>/.ssh/id\_dsa" -o "port=218"
27. : root 10.x.xx.151:/export/sftp

## SRP サービス

アプライアンスで LUN を構成すると、SCSI リモートプロトコル (SRP) ターゲットを介してそのボリュームをエクスポートできます。SRP サービスでは、イニシエータは SRP プロトコルを使用してターゲットにアクセスできます。

SRP ターゲットおよびイニシエータについては、[第6章「Storage Area Network の構成」](#)を参照してください。

SRP ターゲットの管理の例については、[第6章「Storage Area Network の構成」](#)を参照してください。

## TFTP サービス

簡易ファイル転送プロトコル (TFTP) とは、ファイル転送のための簡易プロトコルです。TFTP は、軽量で簡単に実装できるように設計されているため、通常の FTP の機能の大部分が省略されています。TFTP は、リモートサーバーとの間でファイル (または電子メール) のみを読み書きします。ディレクトリを表示することはできず、現時点ではユーザー認証もありません。

## TFTP のプロパティ

表 11-25 TFTP のプロパティ

プロパティ	説明
デフォルトルートディレクトリ	TFTP ログインの場所です。デフォルトは「/export」で、シェア階層の最上位を指します。すべてのユーザーは、TFTP サービスの認証に成功したあとでこの場所にログインします

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## TFTP の構成

### ▼ シェアへの TFTP アクセスの許可

1. 「構成」->「サービス」に移動します
2. TFTP サービスが有効でオンラインになっていることを確認します。有効でない場合は、サービスを有効にします。

3. 「シェア」画面でシェアを選択または追加します。
4. 「プロトコル」セクションに移動し、TFTP アクセスが有効になっていることを確認します。この画面では、アクセスモード (読み取り、読み取り/書き込み) の設定も行えます。

## ウイルススキャンサービス

ウイルススキャンサービスでは、ファイルシステムレベルでウイルスをスキャンします。いずれかのプロトコルからファイルがアクセスされると、ウイルススキャンサービスは最初にそのファイルのスキャンし、ウイルスが見つかった場合はファイルのアクセス拒否と隔離を行います。最新のウイルス定義でスキャンされたファイルは、次の変更が行われるまで再スキャンされません。キャッシュされたファイルデータが含まれているか、NFSv4 サーバーによって読み取り権限が委任されている NFS クライアントからアクセスされたファイルは、すぐに隔離されない可能性があります。

## ウイルススキャンのプロパティ

表 11-26 ウイルススキャンのプロパティ

プロパティ	説明
スキャンする最大ファイルサイズ	このサイズよりも大きいファイルはスキャンされません。パフォーマンスの著しい低下を回避するためです。これらの大規模ファイルは実行可能ファイルになりそうもないため (データベースファイルなど)、それらが脆弱なクライアントにリスクをもたらす可能性は低くなります。デフォルト値は 1G バイトです。
最大ファイルサイズを超えるファイルへのアクセスを許可	デフォルトで有効になっており、最大スキャンサイズよりも大きいファイル (したがって、クライアントに返される前にスキャンされていない) にアクセスできます。セキュリティ要件の厳しいサイトにいる管理者は、このオプションを無効にし、最大ファイルサイズを大きくすることで、アクセス可能なすべてのファイルが確実にウイルススキャンの対象になるようにすることができます。

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## ウイルススキャンのファイル拡張子

このセクションでは、スキャンされるファイルの制御方法について説明します。デフォルト値「\*」を使用すると、すべてのファイルがスキャンされます。すべてのファイルのスキャンすると、パフォーマンスに影響が出る可能性があるため、スキャンするファイルのサブセットを指定できます。

たとえば、リスクの高いファイルのみ (zip ファイルを含む) をスキャンするが、パターン「data-archive\*.zip」に一致する名前のファイルはスキャンしないようにするには、構成を次のように設定できます。

表 11-27 ウイルススキャンのファイル拡張子

アクション	パターン
スキャン	exe
スキャン	com
スキャン	bat
スキャン	doc
スキャン	zip
スキャンしない	data-archive*.zip
スキャンしない	*

注: スキャンリストに明示的に含まれていないほかのすべてのファイルタイプを除外するには、「スキャンしない \*」を使用する必要があります。「file.name.exe.bat.jpg123」という名前のファイルは、名前の「jpg123」部分 (拡張子) のみが規則と比較されるため、スキャンされません。

含める設定の前に除外する設定を使用しないでください。たとえば、「スキャンしない \*」設定は、それ以降のすべてのファイルタイプを除外してしまうため、含める設定の前に使用しないでください。次の例は、ファイルを一切スキャンしません。

表 11-28 ウイルススキャンのアクション

アクション	パターン
スキャンしない	*
スキャン	exe
スキャン	com
スキャン	bat

アクション	パターン
スキャン	doc
スキャン	zip
スキャンしない	data-archive*.zip

## スキャンエンジン

このセクションでは、使用するスキャンエンジンを指定します。スキャンエンジンとは、ファイルをスキャンさせるためにアプライアンスが ICAP (Internet Content Adaptation Protocol, RFC 3507) を使用してアクセスする、サードパーティーの外部ウイルススキャンサーバーです。

表 11-29 スキャンエンジンのプロパティ

プロパティ	説明
有効	このスキャンエンジンを使用します
ホスト	スキャンエンジンサーバーのホスト名または IP アドレスです
最大接続	同時接続の最大数です。一部のスキャンエンジンは、接続数を 8 に制限するとより適切に動作します。
ポート	このスキャンエンジンのポートです

## ウイルススキャンのログ

表 11-30 ウイルススキャンのログ

ログ	説明
vscan	ウイルススキャンサービスのログです

## ウイルススキャンの構成

### ▼ シェアに対するウイルススキャンの構成

1. 「構成」->「サービス」->「ウイルススキャン」に移動します。
2. 必要なプロパティを設定します。

3. 構成を適用または確定します。
4. 「シェア」に移動します。
5. ファイルシステムまたはプロジェクトを編集します。
6. 「一般」タブを選択します。
7. 「ウイルススキャン」オプションを有効にします。

## NIS サービス

ネットワーク情報サービス (NIS) とは、集中管理用のネームサービスです。アプライアンスはユーザーとグループに対して NIS クライアントとして動作できるため、次の操作が可能になります。

- NIS ユーザーは[226 ページの「FTP サービス」](#)および[228 ページの「HTTP サービス」](#)にログインできます。
- アプライアンス管理用の権限を NIS ユーザーに付与できます。アプライアンスでは NIS 情報に独自の権限設定を付加します。

注: 0 - 99 の UID および GID は、将来のアプリケーションで使用するためにオペレーティングシステムベンダーによって予約されています。階層化された製品のエンドシステムユーザーまたはベンダーによるそのような UID や GID の使用はサポートされておらず、使用した場合は将来のアプリケーションでセキュリティ関連の問題が発生する可能性があります。

## NIS のプロパティ

表 11-31 NIS のプロパティ

プロパティ	説明
ドメイン	使用する NIS ドメインです
サーバー: ブロードキャストを使用して検索	アプライアンスは、NIS ブロードキャストを送信して、そのドメインの NIS サーバーを見つけます
サーバー: リスト表示されたサーバーを使用	NIS サーバーのホスト名または IP アドレスです

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。



アプライアンスは、リストに記載されているか、ブロードキャストを使って見つかった最初の NIS サーバーに接続し、応答が停止した場合は次の NIS サーバーに切り替えます。

## NIS のログ

表 11-32 NIS のログ

ログ	説明
network-nis-client:default	NIS クライアントサービスのログです
appliance-kit-nsswitch:default	アプライアンスのネームサービスのログ。そのログを介して NIS クエリーが行われます
system-identity:domain	アプライアンスのドメイン名構成プログラムのログです

## NIS の構成

### ▼ NIS からのアプライアンス管理者の追加

NIS の資格を使ってログインし、アプライアンスを管理する既存のユーザーが NIS に含まれている場合:

1. 「構成」->「サービス」->「NIS」に移動します
2. NIS ドメインとサーバーのプロパティを設定します。
3. 構成を適用または確定します。
4. 「構成」->「ユーザー」に移動します
5. タイプ「ディレクトリ」を使ってユーザーを追加します
6. ユーザー名を NIS ユーザー名に設定します
7. このユーザーに承認を追加するために、引き続き第7章「ユーザー構成」に記載されている手順を実行します。

## LDAP サービス

LDAP (Lightweight Directory Access Protocol) とは、ユーザー、グループ、ホスト名、およびその他のリソース (オブジェクトと呼ばれる) の管理を集中して行うためのディレクトリサービスです。このサービスはアプライアンス上で LDAP クライアントとして機能するため、次の操作が可能になります。

- LDAP ユーザーは [226 ページの「FTP サービス」](#) および [228 ページの「HTTP サービス」](#) にログインできます。
- シェアでルートディレクトリの ACL を構成する際に、(数値 ID ではなく) LDAP ユーザー名を使用できます。
- アプライアンス管理用の権限を LDAP ユーザーに付与できます。アプライアンスでは LDAP 情報に独自の権限設定を付加します。
- LDAP サーバーの証明書は自己署名付きでもかまいません。
- 信頼できる CA 証明書のリストを指定することはできません。各証明書は、アプライアンス管理者によって個別に受け入れられる必要があります。
- LDAP サーバーの証明書が期限切れの場合は、そのサーバーをリストから削除してから再度追加して、その新しい証明書を受け入れる必要があります。

注: 0 - 99 (両端を含む) の UID は、将来のアプリケーションで使用するためにオペレーティングシステムベンダーによって予約されています。階層化された製品のエンドシステムユーザーまたはベンダーによるそのような UID の使用はサポートされておらず、使用した場合はほかのアプリケーションでセキュリティの問題が発生する可能性があります。

## LDAP のプロパティ

使用している環境に適した設定については、LDAP サーバー管理者に問い合せてください。

- SSL/TLS で LDAP トラフィックを保護 - TLS (Transport Layer Security, SSL の後継) を切り替えて、LDAP サーバーへのセキュアな接続を確立します
- ベース検索 DN - ディレクトリ検索の開始位置になる、ベースオブジェクトの識別名を指定します。
- 検索範囲 - ベースオブジェクトを基準にした、検索対象となる LDAP ディレクトリ内のオブジェクトを定義します。検索結果は、ベース検索オブジェクトの直下にあるオブジェクト (1 レベル) だけに制限することも、ベース検索オブジェクトの下にあるすべてのオブジェクト (サブツリー) を含めることもできます。デフォルトは 1 レベルです。
- 認証方法 - LDAP サーバーにアプライアンスを認証させるために使用される方法。アプライアンスでは、Simple (RFC 4513)、SASL/DIGEST-MD5、および SASL/GSSAPI 認証をサポートしています。Simple 認証方法を使用する場合は、SSL/TLS

を有効にして、ユーザーの DN とパスワードが平文で送信されないようにしてください。SASL/GSSAPI 認証方法を使用する場合は、自己バインド資格レベルのみを利用できます。

- バインド資格レベル - LDAP サーバーにアプライアンスを認証させるために使用される資格。
- \*「匿名」では、アプライアンスはすべてのユーザーが利用できるデータにのみアクセスできます。
- \*「プロキシ」では、指定のアカウントでバインドするようサービスに指示します。
- \* プロキシ DN - プロキシ認証に使用されるアカウントの識別名です。
- \* プロキシパスワード - プロキシ認証に使用されるアカウントのパスワードです。
- \* 自己 -「自己」では、ユーザーの ID と資格を使用してアプライアンスを認証します。「自己」認証は、SASL/GSSAPI 認証方法でのみ使用できます。
- スキーマ定義 - アプライアンスで使用されるスキーマ。管理者は、このプロパティを使用して、ユーザー、グループ、およびネットグループのデフォルトの検索記述子、属性マッピング、およびオブジェクトクラスマッピングをオーバーライドできます。詳細は、[250 ページの「LDAP サービス」](#)を参照してください。
- サーバー - 使用する LDAP サーバーのリスト。サーバーを 1 つだけ指定した場合、アプライアンスはそのサーバーだけを使用し、そのサーバーに障害が発生すると、LDAP サービスは使用できなくなります。複数のサーバーを指定した場合、優先順位はなく、機能している任意のサーバーをいつでも使用できます。いずれかのサーバーに障害が発生すると、リスト内の別のサーバーが使用されます。指定されたすべてのサーバーに障害が発生しないかぎり、LDAP サービスは引き続き使用できます。

## LDAP のカスタムマッピング

LDAP ディレクトリでユーザーとグループを検索するには、アプライアンスは検索記述子を使用し、どのオブジェクトクラスがユーザーとグループに対応するか、およびどの属性が必要なプロパティに対応するかを認識する必要があります。デフォルトでは、アプライアンスは RFC 2307 によって指定されたオブジェクトクラス (*posixAccount* と *posixGroup*) および次のリストに示すデフォルトの検索記述子を使用しますが、これは環境ごとにカスタマイズできます。次の例で使用されるベース検索 DN は *dc=example,dc=com* です。

表 11-33 LDAP のカスタムマッピング

検索記述子	デフォルト値	例
users	ou=people, <i>base search DN</i>	ou=people,dc=example,dc=com
groups	ou=group, <i>base search DN</i>	ou=group,dc=example,dc=com

検索記述子	デフォルト値	例
netgroups	ou=netgroup,base search DN	ou=netgroup,dc=example,dc=com

使用される検索記述子、オブジェクトクラス、および属性のカスタマイズには、「スキーマ定義」プロパティを使用します。デフォルトの検索記述子をオーバーライドするには、使用する DN 全体を入力します。アプライアンスでは、この値を未変更のまま使用し、「ベース検索 DN」および「検索範囲」プロパティの値は無視します。ユーザーとグループの属性およびオブジェクトをオーバーライドするには、適切なタブ（「ユーザー」、「グループ」、または「ネットグループ」）を選択し、*default = new* 構文を使用してマッピングを指定します。この場合、*default* はデフォルト値であり、*new* は使用する値です。例:

- ユーザーのオブジェクトクラスとして *posixAccount* の代わりに *unixaccount* を使用するには、「ユーザー」タブの「オブジェクトクラスマッピング」に *posixAccount = unixaccount* と入力します。
- ユーザーオブジェクトの属性として *uid* の代わりに *employeenumber* を使用するには、「ユーザー」タブの「属性マッピング」に *uid = employeenumber* と入力します。
- グループのオブジェクトクラスとして *posixGroup* の代わりに *unixgroup* を使用するには、「グループ」タブの「オブジェクトクラスマッピング」に *posixGroup = unixgroup* と入力します。
- グループオブジェクトの属性として *cn* の代わりに *groupaccount* を使用するには、「グループ」タブの「属性マッピング」に *cn = groupaccount* と入力します。

次に、マッピングに適したオブジェクトクラスと属性のリストを示します。

- クラス:
- \* *posixAccount*
- \* *posixGroup*
- \* *shadowAccount*
- 属性 - ユーザー:
- \* *uid*
- \* *uidNumber*
- \* *gidNumber*
- \* *gecos*
- \* *homeDirectory*
- \* *loginShell*
- \* *userPassword*
- 属性 - グループ:

- \* uid
- \* memberUid
- \* cn
- \* userPassword
- \* gidNumber
- \* member
- \* uniqueMember
- \* memberOf
- \* isMemberOf

## LDAP のログ

ログの例を次に示します。


表 11-34 LDAP のログ

ログ	説明
appliance-kit-nsswitch:default	アプライアンスのネームサービスのログ。そのログを介して LDAP クエリーが行われます

## LDAP の構成

### ▼ アプライアンス管理者の追加

既存の LDAP ユーザーが LDAP 資格を使ってログインし、アプライアンスを管理できるようにするには、次の手順を使用します。

1. 「構成」=>「サービス」=>「LDAP」ページで、使用するプロパティを入力します。使用できるプロパティについては、[250 ページの「LDAP のプロパティ」](#)を参照してください。
2. 選択したプロパティを適用するには「適用」をクリックします。もう一度やり直すには「戻す」をクリックします。
3. LDAP サーバーを追加するには、「サーバー」セクションで  の追加アイコンをクリックします。サーバーについては、[250 ページの「LDAP のプロパティ」](#)の「サーバー」セクションを参照してください。

- LDAP サーバーを構成するには、「新しい LDAP サーバー」ボックスに LDAP サーバーのアドレスを入力し、使用する LDAP 証明書ソースを選択します。「証明書ソース」では、「サーバー」を選択すると、現在のサーバーが検索され、(セキュアな方法で) 証明書が取得されます。その証明書はあとで提示される証明書の検証に将来使用されます。
- 「構成」=>「ユーザー」ページで、必要に応じて LDAP ユーザー名を使ってユーザーを追加します。ユーザーの追加の詳細は、[第7章「ユーザー構成」](#)を参照してください。

## Active Directory

Active Directory サービスでは、Microsoft Active Directory データベースにアクセスできます。このデータベースには、ユーザー、グループ、シェア、およびその他のシェアオブジェクトに関する情報が格納されています。このサービスにはドメインモードとワークグループモードの 2 つのモードがあり、それによって [210 ページの「SMB」](#) ユーザーの認証方法が決まります。ドメインモードで動作している場合、[210 ページの「SMB」](#) クライアントは AD ドメインコントローラを介して認証されます。ワークグループモードでは、[210 ページの「SMB」](#) クライアントはローカルユーザーとしてローカルに認証されます。ローカルユーザーの詳細は、「[ユーザー](#)」を参照してください。

## Active Directory のプロパティー

### Active Directory のドメインへの参加

デフォルトで Active Directory にアカウントがまだ存在しない場合は、ドメインへの参加操作の一環としてシステムのマシン信頼アカウントがコンピュータアカウント (cn=Computers) のデフォルトコンテナ内に自動的に作成されます。次のユーザーがドメインへの参加を実行できます。

- ドメイン管理者。マシン信頼アカウントがいずれかのコンテナに格納されているドメインに任意の数のシステムに参加させることができます。
- 1 つ以上の組織単位に対する権限を持つ、委任された管理者。担当する組織単位にマシンアカウントの場所が指定されているドメインに任意の数のシステムに参加させることができます。
- 管理者によって事前にステージングされたマシンアカウントを持つ通常のユーザー。管理者による事前承認済みとして 1 つのシステムをドメインに参加させることができます。
- 通常のユーザー。通常、限定された数のシステムの参加が承認されます。

Active Directory ドメインに参加するための使用可能なプロパティーを次に示します。

- Active Directory ドメイン - Active Directory ドメインの完全修飾名または NetBIOS 名です
- ユーザー - Active Directory でコンピュータアカウントを作成するための資格を持つ AD ユーザーです
- パスワード - 管理ユーザーのパスワードです
- 追加の DNS 検索パス - このオプションプロパティを指定すると、DNS クエリが、プライマリ DNS ドメインと Active Directory ドメインのほかに、このドメインとも照らし合わせて解決されます
- 組織単位 - システムのマシン信頼アカウントが作成される代替の組織単位を指定します。組織単位は、ドメインを基準とした識別名 (DN) 形式を使用した、1 つ以上の名前と値のペアのコンマ区切りリスト (ou=innerOU,ou=outerOU など) として指定されます。
- 事前作成アカウントを使用 - システムのアカウントが存在するが、指定された組織単位にそのアカウントが含まれていない場合は、事前に作成されたアカウントが使用されます。

## Active Directory のワークグループへの参加

次のリストに、ワークグループに参加するための構成可能なプロパティを示します。

- Windows ワークグループ - ワークグループです

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## Active Directory のドメインとワークグループ

サービスを直接有効にしたり無効にしたりするのではなく、ドメインやワークグループに参加することによってサービスが変更されます。ドメインに参加するには、指定の Active Directory ドメインにアプライアンスのアカウントを作成する必要があります。アカウント名は、最大 15 文字を設定でき、Active Directory ドメインに登録されているほかの名前に対して一意である必要があります。さもないと、類似の名前のアプライアンスとの競合が発生し、機能に問題が起こる可能性があります。コンピュータアカウントが確立されると、アプライアンスはセキュアにデータベースのクエリを実行してユーザー、グループ、およびシェアに関する情報を検索できます。

暗黙のうちにワークグループに参加すると、Active Directory ドメインから切り離され、Active Directory データベースに格納されている [210 ページの「SMB」](#) クライアントはシェアに接続できなくなります。

Kerberos 化された NFS をサポートするように Kerberos レルムが構成されている場合は、Active Directory ドメインに参加するようにシステムを構成することはできません。

## Active Directory の LDAP 署名

LDAP 署名には構成オプションがありません。ドメインコントローラとの通信時にそのオプションが自動的にネゴシエーションされるからです。LDAP 署名がストレージプライアンスとドメインコントローラとの通信時に行われるのに対して、SMB 署名は SMB クライアントとストレージプライアンスの間の通信時に行われます。

## Active Directory の Windows Server 2012 でのサポート

Windows Server 2012 は、2011.1.5 以降のソフトウェアバージョンで完全にサポートされています。

## Active Directory の Windows Server 2008 でのサポート

表 11-35 Active Directory の Windows Server 2008 でのサポート

Windows バージョン	サポートされるソフトウェアバージョン	回避方法
Windows Server 2003	すべて	なし
Windows Server 2008 SP1	2009.Q2 3.1 以前	必要に応じて KB957441 のホットフィックスを適用します。セクション B を参照してください。
	2009.Q2 4.0 - 2011.1.1	KB951191 のホットフィックスを適用し、必要に応じて KB957441 のホットフィックスを適用する必要があります。セクション A および B を参照してください。
	2011.1.2 以降	KB951191 のホットフィックスを適用する必要があります。セクション A を参照してください。
Windows Server 2008 SP2	2009.Q2 4.0 - 2011.1.1	セクション C を参照してください。
	2011.1.2 以降	なし
Windows Server 2008 R2	2009.Q2 4.0 - 2011.1.1	セクション C を参照してください。



Windows バージョン	サポートされるソフトウェアバージョン	回避方法
	2011.1.2 以降	なし

## Active Directory の Windows Server 2008 でのサポートのセクション A: Kerberos の問題 (KB951191)

- 2009.Q2.4.0 以降にアップグレードし、Windows 2008 ドメインコントローラで Windows Server 2008 SP2 または R2 が実行されている場合は、何も行う必要はありません。
- 2009.Q2.4.0 以降にアップグレードし、Windows 2008 ドメインコントローラで Windows Server 2008 SP1 が実行されている場合は、KB951191 に記載されているホットフィックスを適用するか、Windows 2008 SP2 をインストールする必要があります。

## Active Directory の Windows Server 2008 でのサポートのセクション B: NTLMv2 の問題 (KB957441)

- 次は、アプライアンスで 2011.1.2 より前のソフトウェアバージョンが実行されている場合にのみ適用されます。
- ドメインコントローラで Windows Server 2008 SP1 が実行されている場合は、<http://support.microsoft.com/kb/957441/> (<http://support.microsoft.com/kb/957441/>) のホットフィックスも適用するようにしてください。これにより、アプライアンスがデフォルトの LMCompatibilityLevel 設定を使用してドメインに参加できないという NTLMv2 の問題が解決されます。
- Windows 2008 SP1 ドメインコントローラの LMCompatibilityLevel が 5 に設定されている場合は、このホットフィックスをインストールする必要があります。ホットフィックスの適用後は、KB957441 の説明に従って新しいレジストリキーを作成および設定する必要があります。
- 2011.1.2 以降にアップグレードした場合は、前述のホットフィックスは必要ありません。

## Active Directory の Windows Server 2008 でのサポートのセクション C: NTLMv2 に関する注意点

- 次は、アプライアンスで 2011.1.2 より前のソフトウェアバージョンが実行されている場合にのみ適用されます。ドメインコントローラで Windows Server 2008 SP2 または R2 が実行されている場合は、ホットフィックスを適用する必要はありませんが、KB957441 の説明に従ってレジストリ設定を適用する必要があります。

- 2011.1.2 以降にアップグレードした場合は、何も行う必要はありません。

## BUI を使用した Active Directory の構成

### ▼ ドメインへの参加

1. [210 ページの「SMB」](#) コンテキストで Active Directory サイトを構成します。(オプション)
2. [210 ページの「SMB」](#) コンテキストで優先ドメインコントローラを構成します。(オプション)
3. [271 ページの「NTP」](#) を有効にするか、またはアプライアンスとドメインコントローラのクロックが 5 分以内のところまで確実に同期されるようにします。
4. [267 ページの「DNS」](#) インフラストラクチャーで Active Directory ドメインへの委任が正しく行われるようにするか、または [267 ページの「DNS」](#) コンテキストでドメインコントローラの IP アドレスを追加のネームサーバーとして追加します。
5. Active Directory ドメイン、管理ユーザー、および管理パスワードを構成します。
6. 構成を適用または確定します。

### ▼ ワークグループへの参加

1. ワークグループ名を構成します。
2. 構成を適用または確定します。

## CLI を使用した Active Directory の構成

CLI インタフェースを示すために、次の例では、既存の構成を表示し、ワークグループに参加してからドメインに参加しています。

### ▼ 例 - CLI を使用した Active Directory の構成

1. 既存の構成を表示します。

```

twofish:> configuration services ad
twofish:configuration services ad> show
Properties:
    <status> = online
    mode = domain
    domain = eng.fishworks.com

Children:
    domain => Join an Active Directory domain
    workgroup => Join a Windows workgroup

```

2. アプライアンスが現在、ドメイン「eng.fishworks.com」で動作していることに注意してください。次の例では、そのドメインを切り離し、ワークグループに参加します。

```

twofish:configuration services ad> workgroup
twofish:configuration services ad workgroup> set workgroup=WORKGROUP
twofish:configuration services ad workgroup> commit
twofish:configuration services ad workgroup> done
twofish:configuration services ad> show
Properties:
    <status> = disabled
    mode = workgroup
    workgroup = WORKGROUP

```

3. 次の例では、別のドメインに参加する準備として、サイトおよび優先ドメインコントローラを構成します。

```

twofish:configuration services ad> done
twofish:> configuration services smb
twofish:configuration services smb> set ads_site=sf
twofish:configuration services smb> set pdc=192.168.3.21
twofish:configuration services smb> commit
twofish:configuration services smb> show
Properties:
    <status> = online
    lmauth_level = 4
    pdc = 192.168.3.21
    ads_site = sf
twofish:configuration services smb> done

```

4. 次の例では、プロパティが構成されたあと、新しいドメインに参加します。AD ドメインへの参加時は、ノードをコミットするたびにユーザーおよびパスワードを設定する必要があります。

```

twofish:> configuration services ad
twofish:configuration services ad> domain
twofish:configuration services ad domain> set domain=fishworks.com
twofish:configuration services ad domain> set user=Administrator
twofish:configuration services ad domain> set password=*****
twofish:configuration services ad domain> set searchdomain=it.fishworks.com
twofish:configuration services ad domain> commit
twofish:configuration services ad domain> done
twofish:configuration services ad> show

```

Properties:

```
<status> = online
mode = domain
domain = fishworks.com
```

## アイデンティティマッピングサービス

アイデンティティマッピングサービスでは、従来の UNIX UID (および GID) と Windows SID の両方を使用して Windows と UNIX のユーザー ID を同時に管理します。アイデンティティマッピングで BUI および CLI を使用方法については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。

## アイデンティティマッピングのプロパティ

アイデンティティマッピングサービスでは、SID、UID、および GID の間のマッピングのデータベースを作成および維持します。3 つの異なるマッピング方法を使用できます。マッピングをある特定のアイデンティティに使用できる場合は、一時的なマッピングが作成されます。次のマッピングモードを使用できます。

### アイデンティティマッピングの規則ベースのマッピング

規則ベースのマッピング方法では、ID を名前でマップするためのさまざまな規則を作成する必要があります。これらの規則は、Windows ID と UNIX ID の間の等価性を確立します。

### アイデンティティマッピングのディレクトリベースのマッピング

ディレクトリベースのマッピングでは、ID が相手方プラットフォームの同等の ID にどのようにマップされるかについての情報を [250 ページの「LDAP」](#) または [254 ページの「Active Directory」](#) オブジェクトの注釈として付ける必要があります。ディレクトリベースのマッピングを使用する場合は、次の属性を割り当てる必要があります。

- AD 属性 - Unix ユーザー名 - 同等の UNIX ユーザー名の AD データベース内の名前
- AD 属性 - Unix グループ名 - 同等の UNIX グループ名の AD データベース内の名前
- ネイティブ LDAP 属性 - Windows ユーザー名 - 同等の Windows ID の LDAP データベース内の名前

CLI のプロパティ名は、上記のプロパティ名の短縮版です。

[254 ページの「Active Directory」](#) または [250 ページの「LDAP」](#) スキーマの追加については、Solaris CIFS 管理に関するガイドのユーザーおよびグループ向けのディレクトリベースのアイデンティティマッピングの管理 (タスクマップ) のセクションを参照してください。

## アイデンティティマッピングの IDMU

Microsoft では、「UNIX 用 ID 管理」または IDMU と呼ばれる機能を提供しています。このソフトウェアは Windows Server 2003 で使用でき、Windows Server 2003 R2 以降にバンドルされています。この機能は、アンバンドル形式の「Services For Unix」と呼ばれていた機能の一部です。

IDMU の主な使用目的は、Windows を NIS/NFS サーバーとしてサポートすることです。IDMU は、「UNIX 属性」パネルを「Active Directory ユーザーおよびコンピュータ」ユーザーインターフェースに追加して、管理者がいくつかの UNIX 関連パラメータ (UID、GID、ログインシェル、ホームディレクトリ、さらにグループについても同様) を指定できるようにします。これらのパラメータは、AD で RFC2307 に類似した (同じではない) スキーマを介して使用できます。また、NIS サービスでも使用できます。

IDMU マッピングモードを選択すると、アイデンティティマッピングサービスは、これらの UNIX 属性を使用して Windows ID と UNIX ID の間のマッピングを確立します。この方法はディレクトリベースのマッピングに非常によく似ていますが、カスタムスキーマを許可する代わりに、IDMU ソフトウェアによって確立されたプロパティスキーマをクエリー検索するのはアイデンティティマッピングサービスだけです。この方法を使用した場合、ほかのディレクトリベースのマッピングは実行されない可能性があります。

## アイデンティティマッピング規則

このページでは、次のプロパティを使用してマッピングを作成できます。

- マッピングタイプ - 資格を許可または拒否します。詳細は、[260 ページの「拒否マッピング」](#)を参照してください。
- マッピング方向 - マッピングの方向。両方向、Windows から UNIX のみ、または UNIX から Windows のみで資格をマップできます。詳細は、[260 ページの「マッピング規則の方向を示す記号」](#)を参照してください。
- Windows ドメイン - Windows ID の Active Directory ドメインです。
- Windows ID - Windows ID の名前です。
- Unix ID - UNIX ID の名前です。






- Unix ID タイプ - UNIX ID のタイプ (ユーザーまたはグループ) です。





## 拒否マッピング

拒否マッピングの規則は、ユーザーがアイデンティティマッピングサービスから (一時的な ID を含む) どのマッピングも取得できないようにします。Windows ユーザーと UNIX ユーザーに対してドメイン全体またはユーザー固有の拒否マッピングを作成できます。たとえば、グループ「guest」のすべての UNIX ユーザーに対して [210 ページの「SMB」](#) シェアへのアクセスを拒否するマッピングを作成できます。ほかのマッピングと競合する拒否マッピングを作成することはできません。

## マッピング規則の方向を示す記号

名前ベースのマッピングの作成後は、次の記号によって各規則のセマンティクスが示されます。

- align="center"| - Windows ID を UNIX ID にマップします。また、UNIX ID を Windows ID にマップします
- align="center"| - Windows ID を UNIX ID にマップします
- align="center"| - UNIX ID を Windows ID にマップします
- align="center"| - Windows ID が資格を取得できないようにします
- align="center"| - UNIX ID が資格を取得できないようにします

アイコンが黒ではなくグレー (, , , , ) の場合、その規則は解決できない UNIX ID に一致します。

## アイデンティティマッピングのマッピング

「マッピング」ページには、現在の規則セットによる各種 ID のマッピング方法が表示されます。Windows エンティティまたは UNIX エンティティを指定すると、そのエンティティは相手方プラットフォームの対応する ID にマップされます。「ユーザープロパティ」および「グループプロパティ」セクションで結果として得られる情報には、マッピングのソースなど、アイデンティティマッピングに関する情報が表示されます。このページでは、「表示」および「フラッシュ」ボタンを使用して既存のマッピングを表示および削除できます。

## アイデンティティマッピングのログ

このページには、最新のアクティビティのログが表示されます。

## アイデンティティマッピングのベストプラクティス

- きめの細かいアイデンティティマッピング規則の構成は、[202 ページの「NFS」](#) および [210 ページの「SMB」](#) の両方のクライアントで同じユーザーを共通のファイルセットにアクセスさせる場合にのみ適用されます。[202 ページの「NFS」](#) および [210 ページの「SMB」](#) クライアントがばらばらのファイルシステムにアクセスしている場合は、アイデンティティマッピング規則を構成する必要はありません。
- アイデンティティマッピングサービスを再構成しても、アクティブな [210 ページの「SMB」](#) セッションには影響ありません。接続ユーザーが接続されたまま、10 分以内はその以前の名前マッピングが、追加のシェアへのアクセスを承認するために使用できます。承認されていないアクセスを防ぐには、シェアをエクスポートする前にマッピングを構成する必要があります。
- アイデンティティマッピングが提供するセキュリティは、ディレクトリサービスとの同期化と同程度にすぎません。たとえば、特定のユーザーへのアクセスを拒否する名前ベースのマッピングを作成し、そのユーザーの名前が変わった場合、マッピングはそのユーザーへのアクセスを拒否しなくなります。
- Windows ドメインのすべてのユーザーをすべての UNIX ID に対応付ける双方向マッピングは、Windows ドメインごとに 1 つしか作成できません。ドメイン全体の規則を複数作成する場合は、必ずそれらの規則が Windows から UNIX にのみマップするように指定してください。
- 可能なかぎりディレクトリベースのマッピングではなく IDMU マッピングモードを使用します。

## アイデンティティマッピングの概念

[210 ページの「SMB」](#) サービスでは、アイデンティティマッピングサービスを使用して Windows と UNIX の ID を関連付けます。[210 ページの「SMB」](#) サービスでは、ユーザーを認証するときに、アイデンティティマッピングサービスを使用してユーザーの Windows ID を適切な UNIX ID にマップします。Windows ユーザーに対する UNIX ID が存在しない場合は、一時的な UID と GID を使用して一時的な ID が生成されます。これらのマッピングにより、[210 ページの「SMB」](#) および [202 ページの「NFS」](#) クライアントは同時に同じシェアをエクスポートしたり、同じシェアにアクセスしたりできます。Windows ID と UNIX ID を関連付けることで、[202 ページの「NFS」](#) および [210 ページの「SMB」](#) クライアントは同じ ID をシェアできるため、同じファイルセットにアクセスできるようになります。

Windows オペレーティングシステムでは、アクセストークンの中にログインセッションに関するセキュリティ情報が含まれており、アクセストークンによってユーザー、ユーザーのグループ、およびユーザーの権限が識別されます。管理者は、Windows ユーザーおよびグループ

プをワークグループ内に定義するか、または [254 ページの「Active Directory」](#) ドメインコントローラで管理される SAM データベース内に定義します。各ユーザーおよびグループには SID が割り当てられます。SID は、ホストとローカルドメイン内、および考えられるすべての Windows ドメインにわたってユーザーまたはグループを一意に識別します。

UNIX では、ユーザー認証とファイルアクセス権に基づいてユーザー資格を作成します。管理者は、UNIX ユーザーおよびグループをローカルパスワードおよびグループファイル内に定義するか、または [248 ページの「NIS」](#) や [250 ページの「LDAP」](#) などのネームサービスまたはディレクトリサービス内に定義します。各 UNIX ユーザーおよびグループには UID と GID が割り当てられます。通常、UID または GID は 1 つの UNIX ドメイン内のユーザーまたはグループを一意に識別します。ただし、これらの値は複数のドメインにわたって一意ではありません。

## アイデンティティマッピングの大文字小文字の区別

Windows 名は大文字小文字を区別しませんが、UNIX 名は大文字小文字を区別します。ユーザー名 JSMITH、JSmith、および jsmith は、Windows では同じ名前ですが、UNIX では 3 つの異なる名前になります。大文字小文字の区別が名前マッピングに与える影響はマッピングの方向によって異なります。

- Windows から UNIX へのマッピングで一致とみなされる場合、Windows ユーザー名の大文字小文字が UNIX ユーザー名のそれに一致している必要があります。たとえば、Windows ユーザー名「jsmith」は UNIX ユーザー名「jsmith」のみに一致します。Windows ユーザー名「Jsmith」は一致しません。
- Windows から UNIX へのマッピングの大文字小文字の一致要件の例外が発生するのは、マッピングでワイルドカード文字「\*」を使用して複数のユーザー名をマップする場合です。アイデンティティマッピングサービスで、Windows ユーザー \*@some.domain を UNIX ユーザー「\*」に対応付けるマッピングが検出された場合、最初にそのままの Windows 名に一致する UNIX 名が検索されます。一致するものが見つからない場合、Windows 名全体が小文字に変換されて、再度一致する UNIX 名が検索されます。たとえば、Windows ユーザー名「JSmith@some.domain」は UNIX ユーザー名「jsmith」に対応付けられます。Windows ユーザー名を小文字にしたあとも一致するものが見つからない場合、ユーザーはマッピングを得られません。大文字小文字だけが異なる文字列を一致させる規則を作成できます。たとえば、Windows ユーザー「JSmith@sun.com」を UNIX ユーザー「jSmith」に対応付けるユーザー固有のマッピングを作成できます。それ以外では、サービスによって一時的な ID が Windows ユーザーに割り当てられます。
- UNIX から Windows へのマッピングで一致とみなされる場合、大文字小文字が一致している必要はありません。たとえば、UNIX ユーザー名「jsmith」は、大文字小文字に関係なく「JSMITH」という文字を含むすべての Windows ユーザー名に一致します。



## マッピングの永続性

アイデンティティマッピングサービスが名前マッピングを提供する場合、マッピングは 10 分間保存され、10 分たった時点で期限切れとなります。マッピングの 10 分間の有効期間内は、アイデンティティマッピングサービスの再起動後もマッピングは維持されます。マッピングが期限切れになったあとで [210 ページの「SMB」](#) サーバーがユーザーにそのマッピングをリクエストすると、サービスによってそのマッピングが再評価されます。

マッピングやネームサービスディレクトリを変更しても、マッピングの 10 分間の有効期間内は既存の接続に影響しません。サービスによってマッピングが評価されるのは、クライアントがシェアへの接続を試みる場合と期限切れでないマッピングが存在しない場合のみです。

## ドメイン全体のアイデンティティマッピング規則

ドメイン全体のマッピング規則では、Windows ドメインの名前の一部またはすべてを UNIX 名に一致させます。双方のユーザー名は正確に一致する必要があります。ただし、大文字小文字の区別による競合は例外で、これは前述の規則に従います。たとえば、「myDomain.com」のすべての Windows ユーザーを同じ名前の UNIX ユーザーに一致させ、すべての UNIX ユーザーを同じ名前の Windows ユーザーに一致させる双方向の規則を作成できます。別の例では、「myDomain.com」のグループ「Engineering」のすべての Windows ユーザーを同じ名前の UNIX ユーザーにマップする規則を作成できます。ほかのマッピングと競合するドメイン全体のマッピングを作成することはできません。

## 一時的なマッピング

特定のユーザーに適用される名前ベースのマッピング規則がない場合、拒否マッピングによってブロックされないかぎり、そのユーザーには一時的なマッピングを通じて一時的な資格が付与されます。一時的な UNIX 名を持つ Windows ユーザーがシステム上にファイルを作成すると、[210 ページの「SMB」](#) を使用してそのファイルにアクセスする Windows クライアントは、ファイルがその Windows ID によって所有されていると認識します。しかし、[202 ページの「NFS」](#) クライアントは「nobody」によって所有されていると認識します。

## アイデンティティマッピングの例

これは、CLI で 2 つの名前ベースの規則を追加する例です。最初の例では、Windows ユーザーと UNIX ユーザーの間の双方向の名前ベースのマッピングを作成します。

```
twofish:> configuration services idmap
twofish:configuration services idmap> create
```

```
twofish:configuration services idmap (uncommitted)> set
  windomain=eng.fishworks.com
twofish:configuration services idmap (uncommitted)> set winname=Bill
twofish:configuration services idmap (uncommitted)> set direction=bi
twofish:configuration services idmap (uncommitted)> set unixname=wdp
twofish:configuration services idmap (uncommitted)> set unixtype=user
twofish:configuration services idmap (uncommitted)> commit
twofish:configuration services idmap> list
MAPPING      WINDOWS ENTITY          DIRECTION    UNIX ENTITY
idmap-000    Bill@eng.fishworks.com   (U) ==      wdp (U)
```

次の例では、拒否マッピングを作成して、ドメイン内のすべての Windows ユーザーが資格を取得できないようにします。


```
twofish:configuration services idmap> create
twofish:configuration services idmap (uncommitted)> list
Properties:
      windomain = (unset)
      winname = (unset)
      direction = (unset)
      unixname = (unset)
      unixtype = (unset)

twofish:configuration services idmap (uncommitted)> set
  windomain=guest.fishworks.com
twofish:configuration services idmap (uncommitted)> set winname=*
twofish:configuration services idmap (uncommitted)> set direction=win2unix
twofish:configuration services idmap (uncommitted)> set unixname=
twofish:configuration services idmap (uncommitted)> set unixtype=user
twofish:configuration services idmap (uncommitted)> commit
twofish:configuration services idmap> list
MAPPING      WINDOWS ENTITY          DIRECTION    UNIX ENTITY
idmap-000    Bill@eng.fishworks.com   (U) ==      wdp (U)
idmap-001    *@guest.fishworks.com   (U) =>      "" (U)
```

## アイデンティティマッピングの構成

### ▼ アイデンティティマッピングの構成

1. 少なくとも 1 つの Active Directory ドメインに参加していることを確認します。Active Directory については、[254 ページの「Active Directory」](#)セクションを参照してください。
2. 「構成」=>「サービス」=>「アイデンティティマッピング」=>「プロパティ」ページで、使用するマッピングモードを選択します。マッピングモードについては、[260 ページの「プロパティ」](#)を参照してください。
3. 「ディレクトリベースのマッピング」を選択した場合は、追加のプロパティを構成する必要があります。これらのプロパティの詳細は、[260 ページの「ディレクトリベースのマッピング」](#)を参照してください。

4. 設定を保存するには「適用」をクリックします。もう一度やり直すには「戻す」をクリックします。
5. マッピングを作成するには、「規則」をクリックします。
6. 「規則」ページで、 の追加アイコンをクリックします。
7. 「マッピング規則を追加」ボックスで、必要な情報を入力します。詳細は、[260 ページの「規則」](#)を参照してください
8. 設定を保存するには「追加」をクリックします。そうでない場合は、「取り消し」をクリックします。マッピングを作成すると、「規則」リストにそれが表示されます。

#### ▼ マッピングの表示またはフラッシュ

1. 既存のマッピングを表示するには、「構成」=>「サービス」=>「アイデンティティーマッピング」=>「マッピング」ページで、必要な情報を入力します。マッピングについては、[260 ページの「マッピング」](#)を参照してください。
2. 「表示」をクリックします。指定したマッピングが表示されます。
3. マッピングを削除するには、「フラッシュ」をクリックします。マッピングが削除されます。

## DNS サービス

DNS (ドメインネームサービス) クライアントは、IP アドレスをホスト名に、およびその逆方向に解決する機能を提供し、アプライアンス上で常に有効になっています。オプションで、NIS または LDAP (あるいはその両方) によるセカンダリホスト名の解決が構成され有効になっている場合は、DNS で解決できないホスト名とアドレスのためにそれらをリクエストできます。ホスト名解決は、ユーザーが監査可能なアクションを実行した場所を示すための『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「ログ」ログ内、およびクライアントごとの統計を提供するための『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「Analytics」内などの、アプライアンスユーザーインタフェース全体で使用します。

DNS クライアントの構成可能なプロパティには、ベースドメイン名とサーバーリスト (IP アドレスで指定) が含まれています。1 つのドメイン名と 1 つ以上のサーバーアドレスを指定する必要があります。サーバーは、指定されたドメインに対して自身が権限を持っている必要はありませんが、そのドメインの NS (NameServer) レコードを返せる必要があります。

## DNS のプロパティ

表 11-36 DNS のプロパティ

プロパティ	説明
DNS ドメイン	ホスト名の部分検索の実行時に最初に検索されるドメイン名です
DNS サーバー	1 つ以上の DNS サーバー。IP アドレスを使用する必要があります。
IPv4 非 DNS 解決を許可	NIS または LDAP (あるいはその両方) が構成され有効になっている場合は、それを使って IPv4 アドレスをホスト名に、ホスト名を IPv4 アドレスに解決できます。
IPv6 非 DNS 解決を許可	NIS または LDAP (あるいはその両方) が構成され有効になっている場合は、それを使って IPv4 および IPv6 アドレスをホスト名に、ホスト名を IPv4 および IPv6 アドレスに解決できます。

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## DNS の構成

CLI には `nslookup` および `getent hosts` のビルトインが含まれています。これらを使用してホスト名の解決が機能していることをテストできます。

```
caji:> nslookup deimos
192.168.1.109 deimos.sf.fishworks.com
caji:> getent hosts deimos
192.168.1.109 deimos.sf.fishworks.com
```

## DNS のログ

表 11-37 DNS のログ

ログ	説明
network-dns-client:default	DNS サービスのイベントを記録します

## Active Directory と DNS

254 ページの「Active Directory」を使用する予定の場合は、サーバーがドメイン名前空間の Active Directory 部分でホスト名とサーバーのレコードを解決できる必要があります。たとえば、アプライアンスがドメイン example.com にあり、名前空間の Active Directory 部分が redmond.example.com である場合、ネームサーバーは example.com に対して権限のあるサーバーに到達できる必要があります。さらにドメイン redmond.example.com を 1 つ以上の Active Directory サーバーに委譲してそのドメインを処理する必要があります。これらの要件は、アプライアンス自体ではなく Active Directory で要求されるものです。要件が満たされない場合は、Active Directory ドメインに参加できません。

## DNS 以外での解決

DNS は、ホスト名と IP アドレスをマップするための標準的でエンタープライズグレードの高度にスケーラブルな信頼できるメカニズムです。機能している DNS サーバーを使用することはベストプラクティスの 1 つであり、通常は最良の結果をもたらします。環境によっては、一部のホストが NIS または LDAP マップでしか解決できない場合があります。使用している環境にこれが当てはまる場合は、DNS 以外でのホスト解決を有効にし、適切なディレクトリサービスを構成します。ホスト解決に LDAP を使用する場合は、ホストマップがデータベースの標準 DN である ou=Hosts,(ベース DN) にあり、標準スキーマを使用する必要があります。このモードがネットグループによって NFS シェアで使用されるときは、クライアントシステムではアプライアンスで構成されたのと同じホスト名解決メカニズムを使用することが必要な場合があります。そうしないと、NFS シェア例外が正しく機能しない可能性があります。

DNS 以外でのホスト解決が有効になっているときも、DNS は引き続き使用されます。DNS を使ってアドレスまたはホスト名を解決できない場合にのみ、最初に NIS (有効な場合)、次に LDAP (有効な場合) が名前またはアドレスの解決に使用されます。これは、紛らわしく一見矛盾したような結果を招く可能性があります。ホスト解決の結果は、前述の `getent` CLI コマンドを使用して検証できます。

これらのオプションは、使用しないことが強く推奨されています。

## DNS を使用しない操作

DNS を使用しない操作はアプライアンスでサポートされておらず、望ましくない結果を招く可能性があります。DNS を使用しないと、次に限らず、いくつかの機能が正しく動作しなくなります。

- 『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」を使用して、クライアントアドレスをホスト名に解決することはできません。
- 254 ページの「Active Directory」機能が動作しなくなる (ドメインに参加できなくなる)。
- SSL で保護された 250 ページの「LDAP」がホスト名を含む証明書で正しく機能しなくなる。
- 電子メールの送信を伴う警告やしきい値のアクションが、接続されたサブネット上のメールサーバーにしか送信できず、そのメールサーバーの IP アドレスを使用してすべてのアドレスを指定する必要がある。
- ホスト名の解決がタイムアウトとなるため、一部の処理に通常よりも長い時間がかかることがある。

## 動的ルーティングサービス

### RIP および RIPng 動的ルーティングプロトコル

RIP (ルーティング情報プロトコル) とは、ほかの RIP 対応のオンリンクホスト (通常はルーター) から受信されたメッセージに基づいて最適なルートを自動的に構成するためにアプライアンスが使用するディスタンスベクタ型の動的ルーティングプロトコルです。アプライアンスでは、IPv4 用に RIPv1 と RIPv2、IPv6 用に RIPng をサポートしています。これらのプロトコルを介して構成されるルートは、ルーティングテーブルで「dynamic」タイプとしてマーク付けされます。RIP および RIPng は、それぞれ UDP ポート 520 および 521 で待機します。

### 動的ルーティングのログ

表 11-38 動的ルーティング

ログ	説明
network-routing-route:default	RIP サービスのイベントを記録します
network-routing-ripng:quagga	RIPng サービスのイベントを記録します

## IPMP サービス

IPMP (Internet Protocol Network Multipathing) では、ネットワークの帯域幅と信頼性 (インタフェース冗長性) の両方を向上させるために、複数のネットワークインタフェース

を 1 つにグループ化できます。このセクションでは、いくつかのプロパティを構成できます。IPMP グループのネットワークインタフェースの構成については、[第4章「ネットワーク構成」](#)を参照してください。

## IPMP のプロパティ

表 11-39 IPMP のプロパティ

プロパティ	説明
失敗検出待機時間	IPMP でネットワークインタフェースが失敗したことを宣言し、その IP アドレスをフェイルオーバーするための時間です
フェイルバックを有効化	このサービスで、修復されたインタフェースへの接続を再開できるようにします

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## IPMP のログ

表 11-40 IPMP のログ

ログ	説明
network-initial:default	ネットワーク構成プロセスを記録します

## NTP サービス

時間情報プロトコル (NTP) サービスは、アプライアンスのクロックを正確に保つために使用できます。これは、ファイルシステムでの正確なタイムスタンプの記録とプロトコル認証のために重要です。アプライアンスでは UTC タイムゾーンを使用して時間を記録します。BUI に表示される時間は、使用しているブラウザのタイムゾーンオフセットを使用しません。

## NTP のプロパティ

表 11-41 NTP のプロパティ

プロパティ	説明	例
マルチキャストアドレス	自動的に検出される NTP サーバーのマルチキャストアドレスをここに入力します	224.0.1.1
NTP サーバー	アプライアンスが直接接続する 1 つ以上の NTP サーバー (対応する認証鍵があればそれも) を入力します	0.pool.ntp.org
NTP 認証鍵	NTP の有効期間を認証する際にアプライアンスで使用する 1 つ以上の NTP 認証鍵を入力します。後述の「認証」セクションを参照してください。	認証鍵: 10、タイプ: ASCII、非公開鍵: SUN7000

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

### NTP の検証

無効な構成が入力された場合は、警告メッセージが表示され、その構成は確定されません。これは次の場合に起こります。

- マルチキャストアドレスが使用されているが、NTP 応答が見つからない。
- NTP サーバーアドレスが使用されているが、そのサーバーが NTP に正しく応答しない。

### NTP の認証

認証されていないサーバーからの NTP なりすまし攻撃を防ぐために、NTP には非公開鍵暗号化スキームが備わっています。それによって、クライアントは NTP サーバーに関連付けられた非公開鍵を使ってその ID を検証できます。これらの鍵はトラフィックの暗号化やクライアントの認証には使用されず、NTP クライアント (つまりアプライアンス) で NTP サーバーを認証するためだけに使用されます。非公開鍵を NTP サーバーに関連付けるには、最初に非公開鍵を指定する必要があります。各非公開鍵には、それに関連付けられた一意の整数を種類および鍵とともに指定します。この種類には次のいずれかを指定する必要があります。



表 11-42 NTP の非公開鍵と整数

タイプ	説明	例
DES	DES 形式の 64 ビットの 16 進数 です	0101010101010101
NTP	NTP 形式の 64 ビットの 16 進数 です	8080808080808080
ASCII	1 - 8 文字の ASCII 文字列です	topsecret
MD5	MD5 認証スキームを使用する、1 - 8 文字の ASCII 文字列です。	md5secret

鍵を指定したあとは、NTP サーバーを特定の非公開鍵に関連付けることができます。指定した鍵に対して、認証される NTP サーバーのクライアントとサーバーの間で鍵番号、鍵の種類、および公開鍵の値がすべて一致する必要があります。

## NTP の BUI のクロック

BUI 画面の右側に、アプライアンスの時間 (サーバーの時間) とブラウザの時間 (クライアントの時間) の両方が表示されます。NTP サービスがオンラインでない場合は、「SYNC」ボタンをクリックして、アプライアンスの時間が使用しているクライアントブラウザの時間に一致するように設定できます。

## NTP のヒント

SMB を使用してファイルシステムをシェアしている場合、ユーザー認証エラーが発生しないように、クライアントのクロックをアプライアンスのクロックの 5 分以内のところまで同期させる必要があります。クロック同期を確保する方法の 1 つは、同じ NTP サーバーを使用するようにアプライアンスと SMB クライアントを構成することです。

表 11-43 NTP のクロック同期

ログ	説明
network-ntp:default	NTP サービスのログです

## BUI を使用した NTP の構成

BUI で NTP 認証鍵を追加するには、プラスのアイコンをクリックして、新しい鍵の鍵番号、種類、および非公開の値を指定します。鍵を追加したあとは、その鍵が、指定された各 NTP サーバーの横にオプションとして表示されます。

### ▼ BUI のクロック同期

アプライアンスの時間が使用しているブラウザの時間に一致するように設定します。

1. NTP サービスを無効にします。
2. 「同期」ボタンをクリックします。

## CLI を使用した NTP の構成

configuration services ntp の下で、authkey コマンドを使用して承認を編集します。

```
clownfish:configuration services ntp> authkey
clownfish:configuration services ntp authkey>
```

このコンテキストから、create コマンドを使用して新しい鍵を追加できます。

```
clownfish:configuration services ntp authkey> create
clownfish:configuration services ntp authkey-000 (uncommitted)> get
    keyno = (unset)
    type = (unset)
    key = (unset)
clownfish:configuration services ntp authkey-000 (uncommitted)> set keyno=1
    keyno = 1 (uncommitted)
clownfish:configuration services ntp authkey-000 (uncommitted)> set type=A
    type = A (uncommitted)
clownfish:configuration services ntp authkey-000 (uncommitted)> set key=coconuts
    key = ***** (uncommitted)
clownfish:configuration services ntp authkey-000 (uncommitted)> commit
clownfish:configuration services ntp authkey>
```

CLI によって認証鍵をサーバーに関連付けるには、serverkeys プロパティを、servers プロパティの対応するサーバーに関連付けられる鍵の値のリストに設定します。サーバーが認証を使用しない場合、対応するサーバー鍵は 0 に設定されます。たとえば、上記で作成された鍵を使用してサーバー「gefilte」と「carp」を認証するには、次のように入力します。

```
clownfish:configuration services ntp> set servers=gefilte,carp
servers = gefilte,carp (uncommitted)
```

```
clownfish:configuration services ntp> set serverkeys=1,1
      serverkeys = 1,1 (uncommitted)
clownfish:configuration services ntp> commit
clownfish:configuration services ntp>
```

鍵 1 を使ってサーバー「gefilte」を認証し、鍵 2 を使って「carp」を認証し、鍵 3 を使って「dory」を認証するには、次のように入力します。

```
clownfish:configuration services ntp> set servers=gefilte,carp,dory
      servers = gefilte,carp,dory (uncommitted)
clownfish:configuration services ntp> set serverkeys=1,2,3
      serverkeys = 1,2,3 (uncommitted)
clownfish:configuration services ntp> commit
clownfish:configuration services ntp>
```

鍵 1 を使ってサーバー「gefilte」と「carp」を認証し、さらに NTP サーバー「dory」の認証が行われないようにするには、次のように入力します。

```
clownfish:configuration services ntp> set servers=gefilte,carp,dory
      servers = gefilte,carp,dory (uncommitted)
clownfish:configuration services ntp> set serverkeys=1,1,0
      serverkeys = 1,1,0 (uncommitted)
clownfish:configuration services ntp> commit
clownfish:configuration services ntp>
```

## フォンホームサービス

フォンホームサービスの画面は、アプライアンス登録とフォンホームリモートサポートサービスの管理に使用します。

- 登録によって、使用しているアプライアンスが [Oracle Auto Service Request \(ASR\)](http://oracle.com/asr) (<http://oracle.com/asr>) 機能に結び付けられます。Oracle ASR は、アプライアンスによって報告された特定の問題のサービスリクエスト (SR) を自動的に開きます。登録によって、アプライアンスが My Oracle Support (MOS) にも結び付けられて、更新通知が検出されます。
- フォンホームサービスは、Oracle サポートと通信して、次の機能を提供します。
- 障害報告 - システムは自動サービス応答に関するアクティブな問題を Oracle に報告します。障害の性質によっては、サポートケースが開かれることがあります。これらのイベントの詳細は、『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「[問題](#)」で参照できます。
- ハートビート - システムが起動し動作中であることを示すために日単位のハートビートメッセージが Oracle に送信されます。Oracle サポートでは、アクティブになっているシステムが長期間にわたってハートビートの送信に失敗すると、アカウントの技術担当者に通知することがあります。

- システム構成 - 現在のソフトウェアとハードウェアのバージョンと構成、およびストレージ構成を説明する定期メッセージが Oracle に送信されます。このメッセージではユーザーデータやメタデータは送信されません。
- サポートバンドル - サポートバンドルを Oracle サポートにアップロードする前に、フォンホームサービスを有効にする必要があります。詳細は、『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「システム」を参照してください。
- 更新通知 - My Oracle Support (MOS) で新しいソフトウェア更新が入手可能になると、警告を作成します。詳細は、『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「ソフトウェアの更新の通知」を参照してください。

フォンホームサービスを使用するには登録が必要です。

## Oracle シングルサインオンアカウント

フォンホームサービスの障害報告およびハートビート機能を使用するには、有効な Oracle シングルサインオンアカウントのユーザー名とパスワードが必要です。<http://support.oracle.com> (<http://support.oracle.com>) に移動し、「登録」をクリックしてアカウントを作成します。

## フォンホームのプロパティ

サービスプロパティの変更については、[197 ページ](#)の「[BUI を使用したサービスの構成](#)」および [199 ページ](#)の「[CLI を使用したサービスの構成](#)」を参照してください。CLI 内部では、フォンホームサービスは `scrk` として知られています。

## フォンホームの Web プロキシ

アプライアンスが直接インターネットに接続されていない場合は、フォンホームサービスが Oracle との通信に使用する HTTP プロキシを構成する必要があることがあります。このプロキシ設定は、サポートバンドルのアップロードにも使用されます。サポートバンドルの詳細は、『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「システム」を参照してください。

表 11-44 フォンホームの Web プロキシ設定

プロパティ	説明
プロキシを使用	Web プロキシ経由で接続します
ホスト/ポート	Web プロキシのホスト名または IP アドレス、およびポートです

プロパティ	説明
ユーザー名	Web プロキシのユーザー名です
パスワード	Web プロキシのパスワードです

## アプライアンスの登録

アプライアンスをはじめて登録する場合は、Oracle シングルサインオンアカウントを入力する必要があります。[My Oracle Support \(http://support.oracle.com\)](http://support.oracle.com) に移動し、「登録」をクリックしてアカウントを作成します。

### ▼ BUI を使用したアプライアンスの登録

1. Oracle シングルサインオンアカウントのユーザー名とパスワードを入力します。プライバシーに関する説明が表示されます。この説明は BUI および CLI のどちらからでもいつでも確認できます。
2. 変更を確定します。
3. [My Oracle Support \(http://support.oracle.com/\)](http://support.oracle.com/) を使用して、[Auto Service Request \(ASR\) \(http://oracle.com/asr\)](http://oracle.com/asr) のアクティブ化を完了します。My Oracle Support で保留中の ASR アセットを管理および承認する方法 (Doc ID 1329200.1) を参照してください。

### ▼ CLI を使用したアプライアンスの登録

1. `soa_id` および `soa_password` をそれぞれ、Oracle シングルサインオンアカウントのユーザー名およびパスワードに設定します。
2. 変更を確定します。
3. [My Oracle Support \(http://support.oracle.com/\)](http://support.oracle.com/) を使用して、[Auto Service Request \(ASR\) \(http://oracle.com/asr\)](http://oracle.com/asr) のアクティブ化を完了します。My Oracle Support で保留中の ASR アセットを管理および承認する方法 (Doc ID 1329200.1) を参照してください。

#### 例 11-1 CLI の登録

```
dory:> configuration services scrk
```

```
dory:configuration services scrk>set soa_id=myuser
      soa_id = myuser(uncommitted)
dory:configuration services scrk> set soa_password=mypass
      soa_password = ***** (uncommitted)
dory:configuration services scrk> commit
```

## ▼ アカウント情報の変更

1. アプライアンスで使用される Oracle シングルサインオンアカウントを変更するには、「アカウントを変更しています...」をクリックします。
2. 変更を確定します。
3. My Oracle Support を使用して Auto Service Request (ASR) のアクティブ化を完了します。My Oracle Support で保留中の ASR アセットを管理および承認する方法 (Doc ID 1329200.1) を参照してください

## フォンホームのステータス

表 11-45 フォンホームのステータス

プロパティ	説明
最後にハートビートが送信された時間	最後のハートビートが Oracle サポートに送信された時間

## フォンホームの状態

有効な Oracle シングルサインオンアカウントを入力する前にフォンホームサービスが有効になっている場合は、保守状態でサービスが表示されます。フォンホームサービスを使用するには、有効な Oracle シングルサインオンアカウントを入力する必要があります。

## フォンホームのログ

フォンホームイベントのログは、『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「[ログ](#)」内にあります。

## REST

### RESTful API

ZFSSA RESTful API では、管理対象リソースの URL パスの代わりに、単純なリクエスト (GET、PUT、POST、DELETE HTTPlets など) を使って ZFSSA を管理できます。

ZFSSA RESTful ベースのアーキテクチャーは、階層化されたクライアントサーバーモデルとして定義されています。このモデルの利点は、クライアント構成を使わずに標準のハブ、ルーター、およびその他のネットワークシステム経由でサービスを透過的にリダイレクトできることを意味します。このアーキテクチャーは情報のキャッシングをサポートしており、多数のクライアントが同じ静的リソースをリクエストする際に役立ちます。

ZFSSA RESTful API の完全なドキュメントについては、Oracle ZFS Storage Appliance のドキュメントを参照してください。

## サービスタグ

サービスタグを使用すると、次のようなデータをアプライアンスに問い合わせることができるため、製品のインベントリ処理やサポートが容易になります。

- システムのシリアル番号
- システムタイプ
- ソフトウェアのバージョン番号

サービスタグは Oracle サポートに登録できます。これにより、Oracle 機器を簡単に追跡したり、保守呼び出しを円滑に行なったりすることができます。サービスタグはデフォルトで有効になっています。

### サービスタグのプロパティー

表 11-46 UDP/TCP ポートのプロパティー

プロパティー	説明
検出ポート	サービスタグの検出に使用される UDP ポート。デフォルトは 6481 です
リスナーポート	サービスタグデータの問い合わせに使用される TCP ポート。デフォルトは 6481 です

サービスプロパティの変更については、197 ページの「[BUI を使用したサービスの構成](#)」および 199 ページの「[CLI を使用したサービスの構成](#)」を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## SMTP サービス

SMTP サービスは、通常「[警告](#)」画面で構成した警告に対応して、アプライアンスで生成されたすべてのメールを送信します。SMTP サービスでは外部メールを受け入れません。アプライアンス自体で自動的に生成されたメールのみを送信します。

デフォルトでは、SMTP サービスは DNS (MX レコード) を使用してメールの送信先を判断します。DNS がアプライアンスのドメイン用に構成されていない場合、または送信メールの宛先ドメインに DNS MX レコードが正しく構成されていない場合は、一般にスマートホストと呼ばれる送信メールサーバーを介してすべてのメールを転送するようアプライアンスを構成できます。

## SMTP のプロパティ

表 11-47 SMTP のプロパティ

プロパティ	説明
スマートホスト経由でメールを送信	有効にした場合は、指定した送信メールサーバーを介してすべてのメールが送信されます。それ以外の場合は、DNS を使用して特定のドメインのメールの送信先が判断されます。
スマートホストホスト名	送信メールサーバーのホスト名です。
アドレスからカスタムを許可	有効にした場合は、電子メールの送信元アドレスが「アドレスからカスタム」プロパティに設定されます。デフォルトの送信元アドレスがスパムとわかっている場合など、これをカスタマイズすることが望ましい場合があります。
アドレスからカスタム	アウトバウンド電子メールに使用する送信元アドレスです。

サービスプロパティの変更については、197 ページの「[BUI を使用したサービスの構成](#)」および 199 ページの「[CLI を使用したサービスの構成](#)」を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

プロパティを変更するときは、「[警告](#)」を使用して、プロパティが正しいことを検証するためにテスト電子メールを送信できます。電子メールが配信されないもっとも一般的な理由



は、DNS の構成が間違っていることです。これにより、アプライアンスはどのメールサーバーにメールを配信したらよいか判断できません。前述のとおり、DNS を構成できない場合はスマートホストを使用できます。

## SMTP のログ

表 11-48 SMTP のログ

ログ	説明
network-smtp:sendmail	SMTP サービスのイベントを記録します
mail	SMTP アクティビティのログ (送信されたメールを含む) です

## SNMP サービス

SNMP (簡易ネットワーク管理プロトコル) サービスには、アプライアンスに関する 2 つの異なる機能があります。

- SNMP によってアプライアンスのステータス情報を配信できます。
- SNMP トラップを送信するように第9章「警告の構成」を構成できます。

このサービスが有効になっている場合は、SNMP バージョン 1、v2c、および v3 を使用できます。アプライアンスは、最大 50 個の物理および論理ネットワークインタフェースをサポートします。50 を超えるネットワークインタフェースでは、snmpwalk や snmpget などのコマンドのタイムアウトが発生する可能性があります。50 を超えるネットワークインタフェースが必要な場合は、Oracle サポートに連絡してください。

## SNMP のプロパティ

- バージョン: 「v1/2c」と「v3」を切り替えます。
- コミュニティ名: パブリックとユーザー入力を切り替えます。ユーザー入力を選択した場合は、コミュニティ名も入力する必要があります。「v3」を選択した場合は、このプロパティを使用できません。
- 承認ネットワーク/サブネット: 適切な IPv4 アドレスおよびサブネット (0 - 32 の整数) を入力します。「v3」を選択した場合は、このプロパティを使用できません。
- アプライアンス連絡先: 適切なアプライアンスの連絡先を入力します。

- ユーザー名/パスワード: 有効なユーザー名 (最大 501 文字) およびパスワード (8 - 501 文字) を入力します。「v1/2c」を選択した場合は、このプロパティを使用できません。
- 認証: 「MD5」と「SHA」の認証アルゴリズムを切り替えます。「v1/2c」を選択した場合は、このプロパティを使用できません。
- プライバシ: 「なし」と「DES」の暗号化アルゴリズムを切り替えます。「v1/2c」を選択した場合は、このプロパティを使用できません。
- エンジン ID: snmpd でハッシュされた EngineID 値。SNMP があらかじめ有効になっていなかった場合、そのラベルには「0x000」が表示されます。
- トラップ宛先: IPv4 アドレスを追加できます。「+」および「-」ボタンを使用すると、アドレスを追加または削除できます。

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

SNMP サービスには、MIB-II の場所を示す文字列もあります。このプロパティは、[291 ページの「システム ID」](#)の構成から取り込まれます。

## SNMP の MIB

SNMP サービスがオンラインの場合、承認ネットワークで次の MIB (管理情報ベース) にアクセスできます。

表 11-49 SNMP の MIB

MIB	目的
.1.3.6.1.2.1.1	MIB-II システム - ホスト名、連絡先、場所などの一般的なシステム情報
.1.3.6.1.2.1.2	MIB-II インタフェース - ネットワークインタフェースの統計
.1.3.6.1.2.1.4	MIB-II IP - IP アドレスやルーティングテーブルなどのインターネットプロトコル情報
.1.3.6.1.4.1.42	Sun Enterprise MIB (SUN-MIB.mib.txt)
.1.3.6.1.4.1.42.2.195	Sun FM - 障害管理の統計 (下記でリンクされる MIB ファイル)
.1.3.6.1.4.1.42.2.225	Sun AK - アプライアンスの情報と統計 (下記でリンクされる MIB ファイル)

注: Sun MIB ファイルは、[https://IP アドレスまたはホスト名:215/docs/snmp/](https://IPアドレスまたはホスト名:215/docs/snmp/) で入手できます。

## Sun FM MIB

Sun FM MIB (SUN-FM-MIB.mib) は、次のような SUN Fault Manager の情報にアクセスできるようにします。

- システム上のアクティブな問題
- Fault Manager のイベント
- Fault Manager の構成情報

読み取るべきメインテーブルが 4 つあります。

表 11-50 Sun FM MIB

OID	内容
.1.3.6.1.4.1.42.2.195.1.1	障害管理の問題
.1.3.6.1.4.1.42.2.195.1.2	障害管理のフォルトイベント
.1.3.6.1.4.1.42.2.195.1.3	障害管理のモジュール構成
.1.3.6.1.4.1.42.2.195.1.5	障害管理の障害のあるリソース

詳細は、上記でリンクされる MIB ファイルを参照してください。

## Sun AK MIB

Sun AK MIB (SUN-AK-MIB.mib) は、次の情報を提供します。

- 製品の説明の文字列およびパーツ番号
- アプライアンスソフトウェアのバージョン
- アプライアンスとシャーシのシリアル番号
- インストール、更新、およびブート時間
- クラスタの状態
- シェアのステータス - シェアの名前、サイズ、使用済みのバイト数、利用可能なバイト数

読み取るべきメインテーブルが 3 つあります。

表 11-51 Sun AK MIB

OID	内容
.1.3.6.1.4.1.42.2.225.1.4	一般的なアプライアンス情報
.1.3.6.1.4.1.42.2.225.1.5	クラスタのステータス
.1.3.6.1.4.1.42.2.225.1.6	シェアのステータス

詳細は、上記でリンクされる MIB ファイルを参照してください。

## SNMP の構成

### ▼ アプライアンスのステータスを配信するよう SNMP を構成

1. コミュニティー名、承認ネットワーク、および連絡先の文字列を設定します。
2. 必要に応じて、トラップの宛先をリモート SNMP ホストに設定します。それ以外は 127.0.0.1 に設定します。
3. 構成を適用または確定します。
4. サービスを再起動します。

### ▼ トラップを送信するよう SNMP を構成

1. コミュニティー名、連絡先の文字列、およびトラップの宛先を設定します。
2. 必要に応じて、SNMP クライアントを許可するよう承認ネットワークを設定します。それ以外は 127.0.0.1/8 に設定します。
3. 構成を適用または確定します。
4. サービスを再起動します。
5. 受信するトラップを送信するように警告を構成する必要があります。警告の詳細は、[第9章「警告の構成」](#)を参照してください。

## Syslog サービス

Syslog リレーサービスには、アプライアンスに関する 2 つの異なる機能があります。

- syslog メッセージを 1 つ以上のリモートシステムに送信するように第9章「警告の構成」を構成できます。
- アプライアンス上の syslog 対応のサービスではその syslog メッセージがリモートシステムに転送されます。

syslog メッセージとは、アプライアンスから 1 つ以上のリモートシステムに転送される小さなイベントメッセージです (つまり「大陸間の printf」と呼ぶのが望ましい)。このメッセージには、次の要素が含まれます。

- このメッセージを発行したシステムコンポーネントの種類を記述する facility
- このメッセージに関連付けられた状態の重要度を記述する severity
- 関連付けられたイベントの時間を UTC で記述する timestamp
- アプライアンスの正規名を記述する hostname
- このメッセージを発行したシステムコンポーネントの名前を記述する tag。メッセージの書式の詳細は、下記を参照してください。
- イベントそのものを記述する message。メッセージの書式の詳細は、下記を参照してください。

syslog レシーバーは、Solaris や Linux などのほとんどのオペレーティングシステムに備わっています。一部のサードパーティーやオープンソースの管理ソフトウェアパッケージでも Syslog をサポートしています。管理者は、syslog レシーバーを使用して、いくつかのシステムから発行されたメッセージを 1 つの管理システム上に集約し、1 組のログファイルに組み込むことができます。

Syslog リレーは、RFC 3164 で説明されている「classic」出力形式を使用するように構成することも、RFC 5424 で説明されている、より新しいバージョン管理された出力形式を使用するように構成することもできます。syslog メッセージは UDP データグラムで転送されます。そのため、ネットワークによってドロップされやすかったり、送信側のシステムのメモリーが少ない場合やネットワークが輻輳している場合にまったく送信されないことがあったりします。したがって、管理者はネットワーク内に複雑な不具合のあるシナリオでは一部のメッセージが欠けていたり、ドロップされていることを想定するようにしてください。

## Syslog のプロパティ

表 11-52 Syslog のプロパティ

プロパティ	説明
プロトコルバージョン	使用する Syslog プロトコルのバージョン (Classic または Modern) です
宛先	メッセージの中継先となる IPv4 および IPv6 の宛先アドレスのリストです。

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

### Classic Syslog: RFC 3164

Classic Syslog プロトコルには、優先度を表す単一の整数としてエンコードされる facility および level 値、timestamp、hostname、tag、およびメッセージ本文が含まれています。

tag は後述のタグのいずれかになります。

hostname は、[291 ページの「システム ID」](#)の構成で定義されているアプライアンスの正規名になります。

### 最新の Syslog: RFC 5424

Classic Syslog プロトコルには、優先度を表す単一の整数としてエンコードされる facility および level 値、バージョンフィールド (1)、timestamp、hostname、app-name、およびメッセージ本文が含まれています。Sun Storage システムによって中継される syslog メッセージでは、RFC 5424 の procid、msgid、および structured-data フィールドを nil 値 (-) に設定して、これらのフィールドにデータが含まれていないことを示します。

app-name は後述のタグのいずれかになります。

hostname は、[291 ページの「システム ID」](#)の構成で定義されているアプライアンスの正規名になります。

## syslog メッセージの書式

Syslog プロトコル自体はメッセージペイロードの形式を定義せず、どの種類の構造化データまたは構造化されていない人間が読める文字列をメッセージに含めるかは送信側に任せています。Sun Storage アプライアンスでは、次に示すように、syslog サブシステムタグ ak を使用して構造化された解析可能なメッセージペイロードを示します。ほかのサブシステムタグは人間が読める任意のテキストを示しますが、管理者はこれらの文字列の書式が不安定で、Sun Storage ソフトウェアの将来のリリースで予告なしに変更されたり、削除されたりする可能性があることを考慮してください。

表 11-53 syslog メッセージの書式

機能	タグ名	説明
デーモン	ak	アプライアンスサブシステムの汎用タグです。すべての警告には ak タグが付けられ、そのあとに SUNW-MSG-ID が続くことを示します。
デーモン	idmap	POSIX および Windows の ID 変換用の <a href="#">260 ページの「アイデンティティマッピング」</a> サービスです。
デーモン	smbd	シェアにアクセスするための <a href="#">210 ページの「SMB データプロトコル」</a> です。

## syslog 警告メッセージの書式

syslog メッセージ送信アクションで警告が構成されている場合は、次の標準フィールドから成るローカライズされたテキストを含む syslog メッセージペイロードが生成されます。各フィールドには、大文字のフィールド名とそのあとにコロンと空白文字が接頭辞として付けられます。

表 11-54 syslog 警告メッセージの書式

フィールド名	説明
SUNW-MSG-ID	警告に関連付けられた安定した Sun 障害メッセージ識別子です。管理者への警告を発生させる各システムの状態と障害の診断には、Sun の障害メッセージカタログで永続的な一意の識別子が割り当てられます。これらの識別子を簡単に電話で読み取るかノートブックで走り書きして、 <a href="http://sun.com/msg/">sun.com/msg/</a> にある対応するナレッジ記事にリンクできます。

フィールド名	説明
TYPE	状態の種類です。これは次のいずれかのラベルになります。ハードウェアコンポーネントやコネクタの障害を示す「失敗」、ソフトウェアの欠陥や構成の間違いを示す「欠陥」、バックアップアクティビティやリモートレプリケーションの完了など、障害や欠陥に関連しない状態を示す「警告」。
VER	このエンコード形式自体のバージョンです。この説明は、SUNW-MSG-ID 形式のバージョン「1」に相当します。VER フィールドが「1」の場合、解析コードは後続のすべてのフィールドが存在することを前提としている可能性があります。1 よりも大きい 10 進整数が指定されている場合は、追加フィールドが処理または無視されるように解析コードが書き込まれます。
SEVERITY	警告をトリガーした問題に関連付けられた状態の重要度です。重要度のリストについては後述します。
EVENT-TIME	このイベントに対応する時間です。時間の書式は、UTC による「Day Mon DD HH:MM:SS YYYY」となります。例: Fri Aug 14 21:34:22 2009。
PLATFORM	アプライアンスのプラットフォーム識別子です。このフィールドは Oracle Service でのみ使用されます。
CSN	アプライアンスのシャーシのシリアル番号です。
HOSTNAME	<a href="#">291 ページの「システム ID」</a> の構成で定義されているアプライアンスの正規名です。
SOURCE	イベントを発行したアプライアンスソフトウェア内のサブシステムです。このフィールドは Oracle Service でのみ使用されます。
REV	サブシステムの内部リビジョンです。このフィールドは Oracle Service でのみ使用されます。
EVENT-ID	このイベントに関連付けられた汎用一意識別子 (UUID) です。Oracle の障害管理システムでは、管理者が 1 つの状態に関連した複数のメッセージを集めて相互に関連付け、重複メッセージを検出できるように、UUID を各警告および障害診断に関連付けます。Oracle Service 要員は、EVENT-ID を使用して、問題に関連付けられた追加の事後分析情報を取得できます。この情報は Oracle が問題に対処する際に役立つ場合があります。
DESC	イベントに関連付けられた状態の説明です。



フィールド名	説明
AUTO-RESPONSE	システムに組み込まれている障害管理ソフトウェアによる問題 (存在する場合) への自動応答です。自動応答には、障害のあるディスク、DRAM メモリーチップ、およびプロセッサコアを事前にオフラインにするなどの機能があります。
REC-ACTION	推奨される保守アクションです。ここには推奨されるアクションのサマリーが記述されていますが、管理者は詳細な修復手順についてナレッジ記事やこのドキュメントを参照するようにしてください。

SEVERITY フィールドは次のいずれかの値に設定されます。

表 11-55 syslog の重大度フィールド

重大度	syslog レベル	説明
Minor	LOG_WARNING	現時点ではサービスを低下させない状態が発生しました。ただし、状態が深刻になる前に修正する必要があります。
Major	LOG_ERR	サービスを低下させる状態が発生しましたが、深刻ではありません。
Critical	LOG_CRIT	サービスを著しく低下させる状態が発生しました。ただちに修正する必要があります。

## レシーバーの構成例

ほとんどのオペレーティングシステムには syslog レシーバーが組み込まれていますが、レシーバーをオンにするためにはいくつかの構成手順が必要な場合があります。一般的なオペレーティングシステムのいくつかの例を次に示します。それぞれの syslog レシーバーの構成の詳細は、使用しているオペレーティングシステムまたは管理ソフトウェアのドキュメントを参照してください。

### Solaris レシーバーの構成

Solaris には syslog レシーバーとして動作できるバンドル版の `syslogd(1M)` が含まれていますが、リモート受信機能はデフォルトで無効になっています。Solaris で syslog トラフィックを受信できるようにするには、次のように `svccfg` と `svcadm` を使用して syslog 設定を変更します。

```
# svccfg -s system/system-log setprop config/log_from_remote = true
# svcadm refresh system/system-log
```

Solaris syslogd では Classic Syslog プロトコルのみを認識しています。受信されたメッセージのフィルタリングやロギングを構成する方法については、Solaris の syslog.conf(4) のマニュアルページを参照してください。

デフォルトでは、Solaris syslogd はメッセージを /var/adm/messages に記録し、テスト用の警告が次のように記録されます。

```
Aug 14 21:34:22 poptart.sf.fishpong.com poptart ak: SUNW-MSG-ID: AK-8000-LM, \
TYPE: alert, VER: 1, SEVERITY: Minor\nEVENT-TIME: Fri Aug 14 21:34:22 2009\n\
PLATFORM: i86pc, CSN: 12345678, HOSTNAME: poptart\n\
SOURCE: jsui.359, REV: 1.0\n\
EVENT-ID: 92dfef39-6e15-e2d5-a7d9-dc3e221becea\n\
DESC: A test alert has been posted.\n\
AUTO-RESPONSE: None.\nIMPACT: None.\nREC-ACTION: None.
```

## Linux レシーバーの構成

ほとんどの Linux ディストリビューションには syslog レシーバーとして動作できるバンドル版の sysklogd(8) デーモンが含まれていますが、リモート受信機能はデフォルトで無効になっています。Linux で syslog トラフィックを受信できるようにするには、`-r` オプションが含まれる (リモートロギングを可能にする) ように /etc/sysconfig/syslog 構成ファイルを編集します。

```
SYSLOGD_OPTIONS="-r -m 0"
```

その後、ロギングサービスを再起動します。

```
# /etc/init.d/syslog stop
# /etc/init.d/syslog start
```

一部の Linux ディストリビューションには、デフォルトで syslog UDP パケットを拒否する ipfilter パケットフィルタが含まれており、それらのパケットを許可するようにフィルタを変更する必要があります。これらのディストリビューションで、次のようなコマンドを使用して INPUT 規則を追加し、syslog UDP パケットが受け入れられるようにします。

```
# iptables -I INPUT 1 -p udp --sport 514 --dport 514 -j ACCEPT
```

デフォルトでは、Linux syslogd はメッセージを /var/log/messages に記録し、テスト用の警告が次のように記録されます。

```
Aug 12 22:03:15 192.168.1.105 poptart ak: SUNW-MSG-ID: AK-8000-LM, \
TYPE: alert, VER: 1, SEVERITY: Minor EVENT-TIME: Wed Aug 12 22:03:14 2009 \
PLATFORM: i86pc, CSN: 12345678, HOSTNAME: poptart SOURCE: jsui.3775, REV: 1.0 \
EVENT-ID: 9d40db07-8078-4b21-e64e-86e5cac90912 \
```

DESC: A test alert has been posted. AUTO-RESPONSE: None. IMPACT: None. \
 REC-ACTION: None.

## システム ID

このサービスでは、システムの名前と場所を構成できます。アプライアンスを別のネットワークの場所に移動したり、ほかの目的で使用したりする場合は、これらの変更が必要になることがあります。

## システム ID のプロパティ

表 11-56 システム ID のプロパティ

プロパティ	説明
システム名	ユーザーインターフェースに表示される、アプライアンスの単一の標準的な識別名。この名前は、システムへの接続に使用される (リモート DNS サーバーで構成される) DNS 名とは別です。この名前はいつでも変更できます
システムの場所	アプライアンスが物理的に置かれている場所を記述するテキスト文字列。 <a href="#">281 ページの「SNMP」</a> が有効になっている場合は、これが MIB-II の <i>syslocation</i> 文字列としてエクスポートされます

サービスプロパティの変更については、[197 ページの「BUI を使用したサービスの構成」](#)および [199 ページの「CLI を使用したサービスの構成」](#)を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## システム ID のログ

表 11-57 システム ID のログ

ログ	説明
system-identity:node	システム ID サービスのイベントとエラーを記録します

## SSH サービス

SSH (Secure Shell) サービスを使用すると、ユーザーはアプライアンスの CLI にログインして、BUI で実行できるのと同じ管理アクションのほとんどを実行できます。SSH サー

ビスはまた、日単位のログや『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」統計を取り出すためなど、リモートホストから自動化スクリプトを実行する手段としても使用できます。

## SSH のプロパティ

表 11-58 SSH のプロパティ

プロパティ	説明	例
サーバー鍵の長さ	一時的な鍵に含まれるビット数です。	768
鍵再生成間隔	一時的な鍵の再生成間隔 (秒) です。	3600
ログイン猶予期間	クライアントが認証に失敗した場合、指定された秒数を過ぎると、SSH 接続が切断されます。	120
root ログインを許可	root ユーザーが SSH を使用してログインできるようになります。	はい

サービスプロパティの変更については、[197 ページ](#)の「[BUI を使用したサービスの構成](#)」および [199 ページ](#)の「[CLI を使用したサービスの構成](#)」を参照してください。CLI のプロパティ名は、上記のプロパティ名の短縮版です。

## SSH のログ

表 11-59 SSH のログ

ログ	説明
network-ssh:default	SSH サービスのイベントとエラーのログです

## SSH の構成

### ▼ root による SSH アクセスの無効化

1. 「root ログインを許可」を false に設定します。
2. 構成を適用または確定します。

# ◆◆◆ 第 12 章 12

## シェア、プロジェクト、およびスキーマ

---

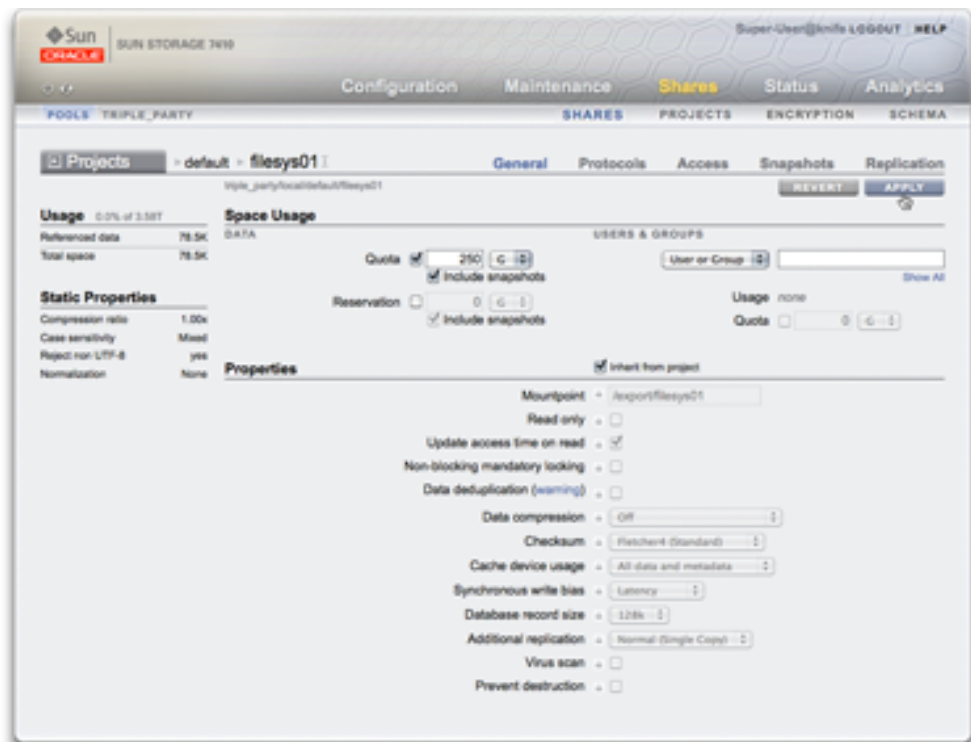
このセクションでは、ZFSSA のシェア、プロジェクト、およびスキーマについて説明します。

スペース管理や共通設定などの共通の管理目的で、シェアを ZFSSA プロジェクトにグループ化できます。標準の組み込みプロパティに加え、すべてのシェアとプロジェクトに使用できる追加のプロパティをいくつでも構成できます。これらのプロパティには、検証のための基本型が設定され、ほかのほとんどの標準プロパティと同様に継承されます。これらの値は、どのような方法でもソフトウェアで使用されることはなく、エンドユーザーだけが使用します。プロパティのスキーマはすべてのプールにわたってシステム全体に適用され、クラスタピア間で同期されます。

## シェアについて

### ストレージプール

図 12-1 同じようなシェアは 1 つのプロジェクトにまとめることができます。



ZFSSA は ZFS ファイルシステムに基づいています。ZFS はベースとなるストレージデバイスをプールにグループ化し、ファイルシステムと LUN は必要に応じてこのストレージから割り当てます。ファイルシステムまたは LUN を作成する前に、まず ZFSSA で [第5章「ストレージ構成」](#) する必要があります。ストレージプールが構成されると、静的にファイルシステムのサイズ変更を行う必要はありませんが、[298 ページの「シェアのスペース管理」](#) を使ってサイズを変更できます。

複数のストレージプールがサポートされていますが、[第5章「ストレージ構成」](#) で説明されているように、この種の構成には重大な欠点があるため、通常はお勧めできません。複数のプールは、データベースのミラー化されたプールやストリーミングワークロードの RAID-Z プールなど、2 つの異なるプロファイルのパフォーマンスや信頼性の特性が大幅に異なる場合にのみ使用するようしてください。

1 つのホストで複数のプールがアクティブになっている場合、BUI ではメニューバーにドロップダウンリストが表示され、それを使ってプールを切り替えることができます。CLI では、現在のプールの名前が括弧に囲まれて表示され、この名前は「pool」プロパティを設定することで変更できます。プールが 1 つしか構成されていない場合、これらのコントロールは表示されません。複数のプールを選択した場合、UI で選ばれるデフォルトプールは任意なので、シェアを操作する前に、スクリプト化された処理によってプール名が明示的に設定されるようにしてください。

## シェアの使用

シェアとは、サポートされているデータプロトコルを介して ZFSSA のクライアントにエクスポートされるファイルシステムや LUN です。ファイルシステムはファイルベースの階層をエクスポートし、[210 ページの「SMB」](#)、[202 ページの「NFS」](#)、[228 ページの「HTTP/WebDav」](#)、および [226 ページの「FTP」](#) を介してアクセスできます。LUN はブロックベースのボリュームをエクスポートし、[208 ページの「iSCSI」](#) またはファイバチャネルを介してアクセスできます。`project/share` タプルはプール内のシェアの一意識別子です。複数のプロジェクトに同じ名前のシェアを含めることはできますが、1 つのプロジェクトに同じ名前のシェアを含めることはできません。1 つのプロジェクトにファイルシステムと LUN の両方を含めることができ、それらは同じ名前空間をシェアします。

## シェアのプロパティ

すべてのプロジェクトおよびシェアには、関連するプロパティがいくつかあります。これらのプロパティは次のグループに分類されます。

表 12-1 プロジェクトとシェアのプロパティ

プロパティタイプ	説明
継承	これはもっとも一般的なプロパティタイプで、プロジェクトとシェアの構成可能なプロパティのほとんどを表します。プロジェクトに属しているシェアは、プロパティのローカル設定を持つことも、親プロジェクトからその設定を継承することもできます。デフォルトでは、シェアはプロジェクトからすべてのプロパティを継承します。プロジェクトでプロパティが変更されると、そのプロパティを継承するすべてのシェアが新しい値を反映して更新されます。継承されると、マウントポイントと SMB のプロパティを除くすべてのプロパティは親プロジェクトと同じ値になります。継承されると、これらのプロパティはプロジェクト設定をその独自のシェア名と連結させます。

プロパティタイプ	説明
読み取り専用	これらのプロパティはプロジェクトとシェアに関する統計を表すもので、変更できません。このタイプのもっとも一般的なプロパティは、領域使用状況の統計です。
スペース管理	これらのプロパティ (割り当て制限と予約) はシェアとプロジェクトの両方に適用されますが、継承されません。100G の割り当て制限があるプロジェクトはシェア全体に強制的に適用されますが、明示的に設定しないかぎり、個々のシェアには割り当て制限は適用されません。
作成時	これらのプロパティはファイルシステムまたは LUN の作成時に指定できますが、シェアが作成されたあとでは変更できません。これらのプロパティはディスク上のデータ構造を制御するもので、国際化設定、大文字と小文字の区別、およびボリュームのブロックサイズが含まれます。
プロジェクトのデフォルト	これらのプロパティはプロジェクトで設定されますが、プロジェクト自体には影響しません。これらのプロパティは、ファイルシステムまたは LUN の作成時に初期設定を生成するために使用され、共通の継承不可能なプロパティセットがシェアに含まれている場合に役立つことがあります。これらのプロパティを変更しても既存のシェアには影響しないため、シェアの作成前または作成後に変更できます。
ファイルシステムローカル	これらのプロパティはファイルシステムにのみ適用され、ファイルシステムのルートディレクトリの管理に便利なプロパティです。これらをプロジェクトで設定することはできません。これらのアクセス制御プロパティは帯域内プロトコル操作によっても設定できます。
LUN ローカル	これらのプロパティは LUN にのみ適用され、継承されません。これらをプロジェクトで設定することはできません。
カスタム	これらはユーザー定義プロパティです。詳細は、 <a href="#">368 ページの「スキーマ」</a> を参照してください。

## シェアのスナップショット

スナップショットとは、ファイルシステムまたは LUN のポイントインタイムコピーです。スナップショットは、手動で作成することも、自動スケジュールを設定して作成することもできます。スナップショットは、最初は追加領域を消費しませんが、アクティブなシェアの変更に



伴って、今まで参照されなかったブロックが最新のスナップショットの一部として保持されます。やがて、最新のスナップショットは、スナップショットが取られた時点のファイルシステムのサイズに等しい大きさの追加領域を最大で占有します。

ファイルシステムのスナップショットはファイルシステムのルートの `.zfs/snapshot` スナップショットにあり、標準のプロトコルを介してアクセスできます。このディレクトリはデフォルトでは表示されず、明示的に `.zfs` ディレクトリに変更することでのみアクセスできます。この動作は[346 ページの「スナップショット」](#)ビューで変更できますが、それによってバックアップソフトウェアがライブデータのほかにスナップショットもバックアップする可能性があります。LUN のスナップショットには直接アクセスできませんが、それらをロールバックターゲットまたはクローンのソースとして使用することはできます。プロジェクトのスナップショットはプロジェクト内のすべてのシェアのスナップショットを取ることに相当し、スナップショットは名前でも識別されます。大きなプロジェクトのスナップショットに含まれるシェアのスナップショットの名前を変更した場合、そのスナップショットは同じスナップショットの一部とはみなされなくなります。また、スナップショットの名前を親プロジェクト内のスナップショットと同じ名前になるように変更した場合、そのスナップショットはそのプロジェクトのスナップショットの一部として扱われます。

シェアは、以前のスナップショットにロールバックする機能をサポートしています。ロールバックが発生すると、新しい方のスナップショット (および新しい方のスナップショットのクローン) が破棄され、アクティブなデータはスナップショットが取られたときの状態に戻ります。スナップショットにはデータのみが含まれ、プロパティは含まれないため、スナップショットが取られたあとで変更されたプロパティ設定はすべて残ります。

## シェアのクローン

---

ライセンスについて: リモートレプリケーションとクローニングは無償で評価することが許可されていますが、本稼働環境で使用するには、各機能のライセンスを個別に購入する必要があります。評価期間を過ぎたら、これらの機能のライセンスを購入するか、機能を非アクティブ化する必要があります。オラクル社は、ライセンスが遵守されているかどうかをいつでも監査する権利を保持しています。詳細は、「オラクル社のソフトウェアライセンス契約書 (SLA)」および「ハードウェアシステムと組み込みのソフトウェアオプションの権利書」を参照してください。

---

クローンとはシェアのスナップショットの書き込み可能コピーであり、管理のために独立したシェアとして扱われます。スナップショットと同様に、クローンは最初は追加の領域を取りませんが、新しいデータがクローンに書き込まれるのに伴って、新しい変更に必要な領域がクローンに関連付けられます。プロジェクトのクローンはサポートされていません。スナップショットとクローンの間で領域がシェアされ、スナップショットには複数のクローンを含めることができるため、アクティブなクローンを破棄しないでスナップショットを破棄することはできません。

## シェアのスペース管理

7000 シリーズでは、物理ストレージの管理に関するファイルシステムと LUN の動作が、ほかの多くのシステムと異なっています。294 ページの「概念」のページで説明しているように、ZFSSA では、共通の領域をすべてのファイルシステムと LUN でシェアする、プールされたストレージモデルを利用します。ファイルシステムには明示的なサイズが割り当てられることはなく、ファイルシステムで必要とされる大きさの領域が使用されるだけです。LUN では、シンプロビジョニングされないうえ、デバイスの内容全体を書き込むのに十分な物理領域が予約されます。シンプロビジョニングされる場合、LUN はファイルシステムのように機能し、データによって物理的に消費される領域の量のみを使用します。

ユーザーが適切な操作を行なっていることがほぼ明らかの場合、このシステムでは環境内での管理の柔軟性と簡易性が最大限に向上します。ユーザーのデータ使用量がモニターまたは制限される厳格な環境では、より注意深い管理が必要です。このセクションでは、管理者が領域使用状況を制御および管理するために使用できる一部のツールについて説明します。

## シェアのスペースに関する用語

詳細に進む前に、ZFSSA での領域使用状況について説明する際に使われるいくつかの基本用語を理解することが重要です。

- **物理データ** - ディスク上に物理的に格納されるデータのサイズです。通常、対応するデータの論理サイズに等しくなりますが、圧縮またはほかの要因によってサイズが異なる場合があります。物理データには、アクティブなシェアやすべてのスナップショットの領域が含まれます。通常、領域の計算は物理領域に基づいて強制的に適用され、管理されます。
- **論理データ** - ファイルシステムによって論理的に消費される領域の量です。圧縮は考慮されず、ファイルシステムによって消費される領域の量の理論的な上限として表示できます。異なる圧縮アルゴリズムを使用してファイルシステムを別の ZFSSA にコピーすると、消費量がこの量よりも少なくなります。この統計は明示的にエクスポートされず、通常は、消費される物理領域の量を測定し、現在の圧縮率を掛け合わせる方法でのみ算出できます。
- **参照データ** - スナップショットとは関係なく、アクティブなシェアによって参照されている領域の合計量を表します。これは、すべてのスナップショットが破棄された場合にシェアによって消費される領域の量です。また、ユーザーがデータプロトコルを介して直接管理できるデータの量でもあります。
- **スナップショットデータ** - シェアのすべてのスナップショットによって現在保持されているデータの合計量を表します。これは、すべてのスナップショットが破棄された場合に解放される領域の量です。

- 割り当て制限 - 割り当て制限は、特定のエンティティーで消費できる領域の量に対する制限を表します。ファイルシステム、プロジェクト、ユーザー、またはグループに基づいて設定でき、現在の領域使用状況には依存しません。
- 予約 - 予約は、特定のプロジェクトまたはファイルシステム用の領域の確保を表しています。ファイルシステムによって消費される実際の領域を増やさずに、プールの残りから使用可能領域を確保します。この設定は、ユーザーやグループには適用できません。従来のような、静的なファイルシステムのサイズ変更は、割り当て制限と予約を同じ値に設定することで実現できます。

## スナップショットについて

スナップショットは、スペース管理にとって興味深い問題を抱えています。スナップショットは、ある特定の時点でシェアによって参照される物理ブロックのセットを表します。最初は、このスナップショットで追加の領域が消費されることはありません。しかし、新しいデータが新しいシェア内で上書きされると、アクティブなシェア内のブロックには新しいデータのみが含まれ、古いブロックは最新（および場合によっては古い方）のスナップショットによって「保持」されます。アクティブなシェア内で内容が分岐するに連れて、スナップショットが徐々に追加の領域を消費するようになる可能性があります。

ほかのシステムの中には、スナップショットが無償であるように見せかけたり、スナップショットデータを保持する専用の領域を「確保」したりすることで、スナップショットのコストを明示しないものがあります。そのようなシステムは、スナップショットに固有の基本的事実を無視しようとしています。所定のサイズのファイルシステムのスナップショットを取り、ファイルシステム内のデータを 100% 書き換える場合は、当然ながら、最初にファイルシステム内にあったデータの 2 倍のデータへの参照を維持する必要があります。スナップショットは空ではないため、ほかのシステムがこの事実を明示しないままにするには、領域がいつばいになったときにスナップショットを黙って破棄するしかありません。多くの場合、これは非常に大きな問題を引き起こします。データを書き換える際にプロセスが異常な状態となって以前のスナップショットがすべて破棄され、そのプロセス内での復元が不可能になるためです。

Sun Storage 7000 シリーズでは、スナップショットのコストを常に明示しており、所定の環境の管理モデルにもっとも適合する方法でこの領域を管理するためのツールを用意しています。各スナップショットには、一意の領域と参照領域という 2 つの関連する領域統計があります。参照領域の量は、スナップショットが取られたときにファイルシステムによって消費された合計領域です。これは、スナップショットがすべてのデータブロックへの唯一の参照であり続けた場合に、そのスナップショットの理論上の最大サイズを表します。一意の領域は、現在のスナップショットによってのみ参照される物理領域の量を示します。スナップショットが破棄されると、一意の領域はプールの残りで使用できるようになります。すべてのスナップショットで消費される領域の量は、すべてのスナップショットにまたがる一意

の領域の合計と同じではありません。1 つのシェアと 1 つのスナップショットがある場合、すべてのブロックは、そのスナップショット、シェア、またはその両方によって参照される必要があります。しかし、複数のスナップショットがある場合は、特定のスナップショットではなく、スナップショットの一部のサブセットによってブロックを参照できます。たとえば、ファイルを作成し、2 つのスナップショット X と Y を取り、そのファイルを削除して、別のスナップショット Z を取った場合、ファイル内のブロックは X と Y によって保持され、Z では保持されません。この場合、Z を破棄しても領域は解放されませんが、X と Y の両方を破棄すると解放されます。このことから、スナップショットを破棄すると、隣接したスナップショットによって参照される一意の領域が影響を受ける可能性があります。ただし、スナップショットで消費される領域の合計量は常に減少します。

使用状況の内訳も利用できますが、プロジェクトまたはシェアの合計サイズは、常にすべてのスナップショットによって消費される領域から成ります。割り当て制限と予約をプロジェクトレベルで設定すると、この合計領域全体に物理的な制約を強制できます。また、割り当て制限と予約をファイルシステムレベルで設定して、これらの設定を参照データのみまたは合計データに適用できます。割り当て制限と予約を参照データまたは合計物理データに適用するかどうかは、管理環境によって異なります。通常、ユーザーがスナップショットを管理していない（つまり、自動スナップショットスケジュールが設定されている）場合には、割り当て制限の計算にスナップショットは含まれません。それ以外の場合、領域不足になっているのにファイルを削除できず、ユーザーが困惑する可能性があります。スナップショットの知識とそれらのスナップショットを管理する手段がなければ、管理者の介入なしにそのような状況を修復できなくなる可能性があります。このシナリオでは、スナップショットは、バックアップ機能を提供するためにシステムの動作に含まれるオーバーヘッドコストを表しています。一方、物理領域の要件に従ってユーザーが課金される環境もあります。その場合、スナップショットは、データセットの解約率を考慮して、ユーザーが要件を満たす程度のバックアップを確保できる選択肢となります。このような環境では、スナップショットを含め、物理データの合計に基づいて割り当て制限を強制する方が理にかなっています。スナップショットのコストを理解すれば、ユーザーは（ZFSSA の専用のロールを使用して）スナップショットをアクティブに管理する手段を手に入れることができます。

## ファイルシステムとプロジェクトの設定

割り当て制限と予約を強制するもっとも簡単な方法は、プロジェクトまたはファイルシステムごとにそれを行うことです。割り当て制限と予約は LUN には適用されませんが、それらの使用状況はプロジェクトの割り当て制限または予約の合計の中で表示されます。

### データの割り当て制限

データの割り当て制限は、ファイルシステムまたはプロジェクトが使用できる領域の量に対して制限を強制します。デフォルトでは、割り当て制限にはファイルシステム内のデータおよ

びすべてのスナップショットが含まれます。新しいデータの書き込みを試みているクライアントでは、ファイルシステムがいっぱいになるとエラーが発生しますが、これは割り当て制限のためか、またはストレージプールの領域不足のためです。298 ページの「スナップショットのセクション」で説明しているように、この動作は、特にスナップショットを使用している状況では理解しにくいことがあります。データブロックがスナップショットによって参照される場合、ファイルを削除するとファイルシステムによって新しいデータが書き込まれる可能性があります。このため、領域使用量を減少させる唯一の方法が既存のスナップショットを破棄することになる場合があります。

「スナップショットを含める」プロパティを設定しない場合、割り当て制限はファイルシステムによって参照される即時データにのみ適用され、スナップショットには適用されません。スナップショットで使用される領域はプロジェクトレベルの割り当て制限によって強制的に設定されますが、それ以外の場合には設定されません。この状況では、スナップショットによって参照されるファイルを削除すると、システムが全体として使用している領域が増えても、ファイルシステムで参照されるデータは減少します。(ファイルシステムが事前設定の割り当て制限に到達した場合とは対照的に) ストレージプールがいっぱいになった場合、領域を解放する唯一の方法がスナップショットを破棄することになる場合があります。

データの割り当て制限は厳格に強制されます。つまり、書き込みが認識されるまで正確なデータの書き込み量がわからないため、領域使用量が限界に近づいたときに、書き込めるデータ量を調整する必要があります。割り当て制限に到達したか、またはそれに近づいた状態で操作すると、パフォーマンスに影響する可能性があります。このため、一般的には、通常の操作手順の実行中には割り当て制限を下回ったままになるようにすることをお勧めします。

BUI で割り当て制限を管理するには、「シェア」->「一般」->「領域の使用」->「データ」に移動します。CLI で割り当て制限を管理するには、`quota` および `quota_snap` プロパティを使用します。

## データの予約

データの予約は、システム内のほかのシェアがより多くの領域を使用しようとしている場合でも、ファイルシステムまたはプロジェクトに少なくとも一定量の使用可能な領域を確保するために使用します。この未使用の予約はファイルシステムの一部とみなされるため、プール (またはプロジェクト) の残りが容量に達した場合、ほかのシェアが領域不足になってもファイルシステムでは引き続き新しいデータを書き込むことができます。

デフォルトでは、予約にはファイルシステムのすべてのスナップショットが含まれます。「スナップショットを含める」プロパティを設定しない場合、予約はファイルシステムの即時データにのみ適用されます。298 ページの「スナップショットのセクション」で説明しているように、スナップショットの取得時には、この動作は理解しにくいことがあります。(スナップショットではなく) ファイルシステムデータに対する予約が有効な場合、スナップショット

を取得するときは必ず、たとえそうなることがなくても、そのスナップショットが完全に分岐するのに十分な領域をシステムで予約する必要があります。たとえば、スナップショットのない 50G のファイルシステムで 100G を予約している場合、最初のスナップショットの取得で 50G の追加領域が予約され、最終的にはこのファイルシステムで合計 150G の領域が予約されます。データの完全な分岐を保証できるだけの十分な領域がない場合、スナップショットの取得は失敗します。

BUI で予約を管理するには、「シェア」->「一般」->「領域の使用」->「データ」に移動します。CLI で予約を管理するには、reservation および reservation\_snap プロパティを使用します。

## LUN をレプリケートするためのスペース管理

LUN を作成すると、(シンプロビジョニングされないかぎり) その LUN 用に構成した十分な物理領域が予約され、ほかのファイルシステムでは使用できません。レプリケーションでは、任意のサイズの LUN のスナップショットを取る場合、使用された LUN 領域の大きさに応じて、最大で LUN の 2 倍のサイズの領域も予約されます。

次のリストに、LUN のレプリケート時に必要な最大オーバーヘッド領域を示します。

- 更新と更新の合間のソースでは最大 100%
- 更新中のソースでは最大 200%
- ターゲットでは最大 200%

## ユーザーとグループの設定

### 現在の使用状況の表示

ユーザーおよびグループの割り当て制限が使用中であるかどうかに関係なく、ファイルシステムやプロジェクトに対してユーザーまたはグループごとの現在の使用状況を問い合わせることができます。古いバージョンのソフトウェアで作成したストレージプールは、この機能を使用する前に『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「更新」を適用することが必要な場合があります。遅延更新の適用後は、ユーザーごとおよびグループごとの使用状況と割り当て制限をサポートするバージョンにすべてのファイルシステムをアップグレードするために、多少の時間がかかる場合があります。

#### ▼ BUI での現在の使用状況の表示

1. BUI で現在の使用状況を表示するには、「シェア」>「シェア」>「一般」に移動します。

2. 「領域の使用 - ユーザーとグループ」セクションで、「ユーザーまたはグループ」ドロップダウンをクリックして「ユーザー」または「グループ」を選択し、シェア内またはプロジェクト全体にわたって任意のユーザーまたはグループの現在の使用状況を問い合わせます。
3. 問い合わせるユーザーまたはグループの名前を入力します。入力すると、問い合わせが行われます。

検索が完了すると、現在の使用状況が表示されます。さらに、「すべて表示」リンクでは、すべてのユーザーまたはグループの現在の使用状況のリストを含むダイアログが表示されます。このダイアログでは、ユーザーまたはグループのどちらかのタイプしか問い合わせできず、同時に両方を問い合わせる機能はサポートされていません。このリストには、正規の UNIX 名と Windows 名 (マッピングが有効な場合)、および使用状況と割り当て制限 (ファイルシステムの場合) が表示されます。

## ▼ CLI での現在の使用状況の表示

1. CLI で、特定のプロジェクトまたはシェアのコンテキストから `users` および `groups` コマンドを使用します。
2. 現在の使用状況を表形式で表示するには、`show` コマンドを使用します。
3. 特定のユーザーまたはグループの使用状況を取得するには、対象となるユーザーまたはグループを選択して `get` コマンドを使用します。

```
clownfish:> shares select default
clownfish:shares default> users
clownfish:shares default users> list
USER      NAME                USAGE
user-000  root                325K
user-001  ahl                 9.94K
user-002  eschrock            20.0G
clownfish:shares default users> select name=eschrock
clownfish:shares default user-002> get
      name = eschrock
      unixname = eschrock
      unixid = 132651
      winname = (unset)
      winid = (unset)
      usage = 20.0G
```

## ユーザーまたはグループ割り当て制限の設定

割り当て制限は、ファイルシステムレベルでユーザーまたはグループに設定できます。このような割り当て制限により、ファイルやディレクトリの所有者またはグループの POSIX または Windows アイデンティティに基づいて、強制的に物理データが使用されます。ユーザーと

グループの割り当て制限、およびファイルシステムとプロジェクトデータの割り当て制限との間には、いくつかの大きな相違点があります。

- ユーザーとグループの割り当て制限はファイルシステムにのみ適用できます。
- ユーザーとグループの割り当て制限は遅延実施を使用して実装されます。これは、データがディスクに書き込まれる前に、ユーザーが短時間だけその割り当て制限を超過できることを意味します。データがディスクに送られると、ファイルシステムレベルの割り当て制限の場合と同様に、ユーザーは新しい書き込みに関するエラーを受け取ります。
- ユーザーとグループの割り当て制限は、常に参照データに対して強制されます。これは、スナップショットがどの割り当て制限にも影響を与えないことを意味し、基になるブロックがシェアされていても、スナップショットのクローンは同じ量の有効な割り当てを消費します。
- ユーザーとグループの予約はサポートされていません。
- データの割り当て制限とは異なり、ユーザーとグループの割り当て制限は通常のファイルシステムデータで格納されます。これは、ファイルシステムの領域が不足している場合に、ユーザーとグループの割り当て制限に変更を加えられないことを意味します。最初に追加領域を使用できるようにしてから、ユーザーとグループの割り当て制限を変更する必要があります。
- ユーザーとグループの割り当て制限は、リモートレプリケーションの一部として送信されます。管理者は、ネームサービス環境が送信元と送信先で確実に同じになるようにする必要があります。
- シェア全体の NDMP バックアップおよび復元には、ユーザーとグループの割り当て制限が含まれます。既存のシェアへの復元は、現在の割り当て制限には影響しません。

## ▼ BUI を使用したユーザーまたはグループ割り当て制限の設定

1. BUI で、「シェア」>「シェア」>「一般」に移動します。
2. 「領域の使用 - ユーザーとグループ」セクションで、「ユーザーまたはグループ」ドロップダウンをクリックして「ユーザー」または「グループ」を選択し、シェア内またはプロジェクト全体にわたって任意のユーザーまたはグループの現在の使用状況を問い合わせます。
3. ブラウザでユーザー割り当て制限を管理するには、[320 ページ](#)の「一般」タブの「領域の使用」->「ユーザーとグループ」に移動します。使用状況の表示と同様に、ユーザーまたはグループを入力すると、現在の使用状況が表示されます。ユーザー名またはグループ名の入力が終わると、現在の使用状況が表示されたら、「割り当て制限」の横のボックスにチェックマークを付け、サイズフィールドに値を入力することで割り当て制限を設定できます。割り当て制限を無効にするには、ボックスのチェックマークを外します。変更が適用されたら、「適用」ボタンをクリックして変更します。



4. ページ上のすべてのプロパティは同時に確定されますが、ユーザーとグループの割り当て制限はほかのプロパティとは別に検証されます。無効なユーザーとグループおよびほかの無効なプロパティが入力された場合は、検証エラーのいずれか 1 つしか表示されない場合があります。そのエラーが修正されたあとで、変更をもう一度適用しようとすると、ほかのエラーが表示されます。

## ▼ CLI を使用したユーザーまたはグループ割り当て制限の設定

- CLI でユーザー割り当て制限を管理するには、シェアコンテキストから「users」または「groups」コマンドを使用します。割り当て制限を設定するには、特定のユーザーまたはグループを選択し、「set quota」コマンドを使用します。ファイルシステムの領域を消費しておらず、割り当て制限が設定されていないユーザーは、アクティブユーザーのリストに表示されません。そのようなユーザーまたはグループの割り当て制限を設定するには、「quota」コマンドを使用すると、そのあとで名前と割り当て制限を設定できるようになります。割り当て制限をクリアするには、値を「0」に設定します。

```
clownfish:> shares select default select eschrock
clownfish:shares default/eschrock> users
clownfish:shares default/eschrock users> list
USER      NAME      USAGE  QUOTA
user-000  root      321K   -
user-001  ahl       9.94K  -
user-002  eschrock  20.0G  -
clownfish:shares default/eschrock users> select name=eschrock
clownfish:shares default/eschrock user-002> get
      name = eschrock
      unixname = eschrock
      unixid = 132651
      winname = (unset)
      winid = (unset)
      usage = 20.0G
      quota = (unset)
clownfish:shares default/eschrock user-002> set quota=100G
      quota = 100G (uncommitted)
clownfish:shares default/eschrock user-002> commit
clownfish:shares default/eschrock user-002> done
clownfish:shares default/eschrock users> quota
clownfish:shares default/eschrock users quota (uncommitted)> set name=bmc
      name = bmc (uncommitted)
clownfish:shares default/eschrock users quota (uncommitted)> set quota=200G
      quota = 200G (uncommitted)
clownfish:shares default/eschrock users quota (uncommitted)> commit
clownfish:shares default/eschrock users> list
USER      NAME      USAGE  QUOTA
user-000  root      321K   -
user-001  ahl       9.94K  -
user-002  eschrock  20.0G  100G
user-003  bmc      -      200G
```

## アイデンティティ管理

ユーザーとグループの割り当て制限には、ZFSSA の260 ページの「[アイデンティティマッピング](#)」サービスが使用されます。このサービスでは、環境に応じて、ユーザーとグループを UNIX または Windows のアイデンティティとして指定できます。ファイル所有権と同様に、これらのアイデンティティは次の方法で追跡されます。

- UNIX マッピングがない場合、Windows アイデンティティへの参照が格納される。
- UNIX マッピングがある場合、その UNIX アイデンティティが格納される。

これは、アイデンティティの正規の形式が UNIX アイデンティティであることを意味します。マッピングがあとで変更された場合は、新しいマッピングが新しい UNIX アイデンティティに基づいて強制されます。マッピングが存在しないときに Windows ユーザーによってファイルが作成され、あとでマッピングが作成された場合、新しいファイルは、アクセス制御と使用状況の形式のために別の所有者のものとして扱われます。これはまた、ユーザー ID が再利用された (つまり、新しいユーザー名の関連付けが作成された) 場合、既存のファイルまたは割り当て制限がその新しいユーザー名で所有されているように見えることを意味します。

ファイルシステムの積極的な活用を試みる前に、アイデンティティマッピングの規則を確立することをお勧めします。それ以外の場合、マッピングに何らかの変更を加えると予期しない結果が生じることがあります。

## ファイルシステムの名前空間

ZFSSA 上のすべてのファイルシステムには、ファイルシステムデータのアクセスポイントとして機能する一意のマウントポイントを指定する必要があります。プロジェクトにもマウントポイントを指定できますが、このマウントポイントは、継承されたプロパティを使用して名前空間を管理するためのツールとしてのみ機能します。プロジェクトはマウントされないため、プロトコルを介してデータがエクスポートされることはありません。

すべてのシェアは `/export` の下にマウントされる必要があります。ファイルシステムを作成して `/export` にマウントすることは可能ですが、必須ではありません。そのようなシェアが存在しない場合は、必要に応じて階層のこの部分の下にディレクトリが動的に作成されます。各マウントポイントはクラスタ内で一意である必要があります。

## 名前空間の入れ子のマウントポイント

マウントポイントがほかのファイルシステムのマウントポイントの下にあるファイルシステムを作成できます。このシナリオでは、親ファイルシステムは子ファイルシステムの前にマウン

トされ、またその逆も同様です。入れ子のマウントポイントを使用するときは、次の場合を考慮するようにしてください。

- マウントポイントが存在しない場合は、マウントポイントが作成され、root およびモード 0755 によって所有されます。状況によっては、ファイルシステムの名前変更、破棄、または移動が行われると、このマウントポイントが解除されることがあります。念のため、子ファイルシステムを作成する前に親のシェア内にマウントポイントを作成するようにしてください。
- 親ディレクトリが読み取り専用で、マウントポイントが存在しない場合、ファイルシステムのマウントは失敗します。これはファイルシステムの作成時に同期的に発生することがありますが、継承されたマウントポイントを持つファイルシステムの名前変更など、大規模な変更を行うときに非同期的に発生することもあります。
- ファイルシステムの名前を変更したり、そのマウントポイントを変更したりすると、現在のマウントポイントの下にあるすべての子マウントポイントと新しいマウントポイント (異なる場合) がアンマウントされ、変更の適用後に再マウントされます。これにより、現在そのシェアにアクセスしているデータサービスがすべて中断します。
- 入れ子のマウントポイントの自動トラバースをサポートするかどうかは、下記で説明するように、プロトコルによって決まります。

## 名前空間のマウントポイントへのプロトコルアクセス

プロトコルの設定に関係なく、すべてのファイルシステムにはマウントポイントが必要です。ただし、これらのマウントポイントがどのように使用されるかはプロトコルによって異なります。

### 名前空間の NFSv2 / NFSv3

NFS では、各ファイルシステムは MOUNT プロトコル経由で表示できる一意のエクスポートです。NFSv2 と NFSv3 には入れ子のファイルシステムをトラバースする手段がなく、各ファイルシステムにはそのフルパスによってアクセスする必要があります。入れ子のマウントポイントは引き続き機能していますが、入れ子のマウントポイントを越えようとする、空のディレクトリがクライアント上に作成されます。この問題は自動マウントによるマウントを使用することで軽減できますが、動的な環境で入れ子のマウントポイントを透過的にサポートするには NFSv4 が必要です。

### 名前空間の NFSv2

NFSv4 では、NFSv3 と比べてマウントポイントの処理にいくつかの改善が加えられています。まず、階層内のそのポイントで使用できるシェアがない場合でも、その親ディレクトリ

をマウントできます。たとえば、`/export/home` がシェアされていた場合、`/export` をクライアント上にマウントして、実際のエクスポートを透過的にトラバースできます。さらに重要なことに、一部の NFSv4 クライアント (Linux を含む) はクライアント側での自動マウントをサポートしており、これは「ミラーマウント」と呼ばれることがあります。このようなクライアントがあれば、ユーザーがマウントポイントをトラバースすると、子ファイルシステムは適切なローカルマウントポイントで自動的にマウントされ、ファイルシステムがそのクライアント上でアンマウントされると無効になります。サーバー側から見ると、これらは別々のマウントリクエストですが、1 つにまとめてクライアントに送られ、シームレスなファイルシステムの名前空間を形成します。

## 名前空間の SMB

SMB プロトコルでは、各シェアがリソース名によって使用できるようになるため、マウントポイントを使用しません。ただし、各ファイルシステムには引き続き一意のマウントポイントが必要です。入れ子のマウントポイント (1 つのリソース内に複数のファイルシステムがある) は現在サポートされていないため、マウントポイントのトラバースを試みると空のディレクトリが作成されます。

## 名前空間の FTP / FTPS / SFTP

ファイルシステムは、それぞれの標準のマウントポイントを使ってエクスポートされます。入れ子のマウントポイントは完全にサポートされており、ユーザー対して透過的です。ただし、親がシェアされているときに入れ子のファイルシステムをシェアしないようにはできません。親のマウントポイントがシェアされている場合は、子もすべて同様にシェアされます。

## 名前空間の HTTP / HTTPS

ファイルシステムは `/shares` ディレクトリの下にエクスポートされるため、`/export/home` にあるファイルシステムは HTTP または HTTPS を介して `/shares/export/home` に表示されます。入れ子のマウントポイントは完全にサポートされており、ユーザー対して透過的です。FTP プロトコルのセクションで説明されているシェアオプションの競合に関する動作と同じ動作が、HTTP にも当てはまります。

## 「シェア」>「シェア」

### BUI での「シェア」>「シェア」の操作

シェアの UI には、「シェア」>「シェア」からアクセスします。デフォルトのビューには、システム上のすべてプロジェクトのシェアが表示されます。

#### シェアリスト




デフォルトのビューは、システム上のすべてのシェアのリストです。このリストを使用すると、シェアの名前変更、プロジェクト間でのシェアの移動、および個々のシェアの編集を実行できます。シェアは「ファイルシステム」と「LUN」の 2 つのリストに分けられ、このビューにあるタブを切り替えることで選択できます。シェアごとに、次のフィールドが表示されます。

表 12-2 BUI のシェアリスト

フィールド	説明
名前	シェアの名前。すべてのプロジェクトが表示されている場合は、プロジェクト名もここに含まれます。シェア名は編集可能なテキストフィールドです。名前をクリックすると、新しい名前を入力できます。Return キーを押すか、名前からフォーカスを移動すると、変更が確定されます。シェアの名前を変更するには、アクティブクライアントの接続を切断する必要があるため、このアクションの確認を求められます。
サイズ	ファイルシステムの場合、これはファイルシステムの合計サイズになります。LUN の場合、これはボリュームのサイズになり、シンプロビジョニングの有無は問われません。詳細は、 <a href="#">310 ページの「使用状況統計」</a> を参照してください。
マウントポイント	ファイルシステムのマウントポイント。これは NFS を介して使用できるパスで、FTP および HTTP の場合は相対パスになります。SMB を介してエクスポートされるファイルシステムだけはそのリソース名を使用しますが、各ファイルシステムでは、システム上のどこかに一意のマウントポイントが引き続き必要です。
GUID	LUN 用の SCSI GUID。詳細は、 <a href="#">329 ページの「「シェア」&gt;「シェア」&gt;「プロトコル」 - BUI ページ」</a> を参照してください。

各シェアでは、次のツールを使用できます。

表 12-3 BUI の「シェア」&gt;「シェア」のアイコン

アイコン	説明
	シェアを別のプロジェクトに移動します。プロジェクトパネルが展開されていない場合は、シェアがプロジェクト上にドロップされるまで自動的にパネルが展開されます。
	個別のシェアを編集します (行をダブルクリックする方法でも利用可能)。
	シェアを破棄します。このアクションによってシェアのデータがすべて破棄され、元に戻すことはできないため、このアクションの確認を求められます。

## シェアの編集

シェアを編集するには、鉛筆アイコンをクリックするか、またはシェアリスト内の行をダブルクリックします。これでシェアが選択され、いくつかの異なるタブが表示されます。これらのタブから選択して、シェアのプロパティを編集できます。すべての機能の詳細は、各タブのセクションを参照してください。

- [320 ページの「一般」](#)
- [329 ページの「プロトコル」](#)
- [338 ページの「アクセス」](#)
- [346 ページの「スナップショット」](#)
- [第13章「レプリケーション」](#)

シェアの名前は、プロジェクトパネルの右側の左上隅に表示されます。シェア名の最初のコンポーネントはそのシェアが含まれているプロジェクトで、そのプロジェクト名をクリックすると、[[Shares:Projects|プロジェクト詳細]] に移動します。シェアの名前を変更するには、シェア名をクリックして新しいテキストを入力する方法もあります。シェアのアクティブクライアントの接続を切断する必要があるため、このアクションの確認を求められます。

## 使用状況統計

ビューの左側に (展開された場合はプロジェクトパネルの下に)、現在の領域の使用状況統計を示す表があります。これらの統計は、特定のシェア (シェアを編集している場合)、またはプール全体 (シェアリストを表示している場合) が対象です。値が 0 のプロパティがあれば、そのプロパティは表から除外されます。次の使用状況統計が表示されます。

- 「使用可能な容量」- この統計はタイトルの容量を容量の割合で表したもので、暗黙的に表示されます。使用可能な領域は、シェアまたはプロジェクト上のすべての割り

当て制限、またはプールの絶対容量を反映します。ここに表示される数値は、使用済みの合計領域に使用可能な領域を加えた値です。

- 「参照されるデータ」- データで参照されているデータの量。これには、必要なメタデータだけでなく、すべてのファイルシステムデータまたは LUN ブロックも含まれます。圧縮を使用すると、この値はシェア内に含まれるデータの論理サイズよりもかなり小さくなる可能性があります。シェアがスナップショットのクローンである場合、この値はシェア内に理論的に含めることができる物理ストレージよりも小さくなり、0 になる場合があります。
- 「スナップショットデータ」- プロジェクトのスナップショットを含む、シェアのすべてのスナップショットで使用される領域の量です。このサイズは、すべてのスナップショットで消費される一意の領域の合計と同じではありません。複数のスナップショットによって参照されるブロックは、スナップショットごとの使用状況統計には含まれませんが、シェアのスナップショットデータの合計に表示されます。
- 「未使用予約」- ファイルシステムに予約が設定されている場合、この値はファイルシステム用に予約されている残りの領域の量を示します。この値は LUN には設定されません。ZFSSA では、この領域がほかのシェアによって消費されないようにして、ファイルシステムに十分な領域を確保します。予約にスナップショットが含まれていない場合は、スナップショットの取得時にスナップショット全体を上書きするのに十分な領域が必要です。予約の詳細は、[320 ページの「一般プロパティ」](#)のセクションを参照してください。
- 「合計容量」- 参照データ、スナップショットデータ、および未使用予約の合計。

## 静的プロパティ

シェアビューの左側には、特定のシェアの編集集中に静的 (作成時の) プロパティも表示されます。これらのプロパティは作成時に設定されるもので、設定後は変更できません。次の静的プロパティが表示されます。

- 「圧縮率」- 圧縮が有効になっている場合、ここには、現在シェアに適用されている圧縮率が表示されます。圧縮率は乗数で表されます。たとえば、2x という圧縮は、圧縮されていない内容の半分の領域をデータが消費していることを意味します。圧縮と使用可能なアルゴリズムの詳細は、[320 ページの「一般プロパティ」](#)のセクションを参照してください。
- 「大文字小文字の区別」- ディレクトリ検索で大文字と小文字を区別するかどうかを制御します。次のオプションがサポートされています。

BUI の値	CLI の値	説明
混在	mixed	大文字と小文字の区別は、使用されるプロトコルによって決まりま

BUI の値	CLI の値	説明
		す。NFS、FTP、および HTTP の場合は、検索で大文字と小文字を区別します。SMB の場合は、検索で大文字と小文字を区別しません。これはデフォルトであり、プロトコル間の一貫性よりも各種プロトコルの適合性を優先しています。このモードの使用時は、大文字と小文字を区別するプロトコルを使用して別個のファイルを作成できますが、SMB 経由でアクセスするとそのファイルはクラッシュします。このような状況では、SMB サーバーは、ファイル名を一意に識別する「符号化された」競合を作成します。
区別しない	insensitive	すべての検索で大文字と小文字を区別せず、最初から大文字と小文字を区別する (NFS などの) プロトコルよりも優先されます。これにより、大文字と小文字を区別するプロトコルのクライアントで混乱が起きる可能性があります。符号化された名前が SMB 経由で使用される原因となるファイル名の重複がクライアントで発生しなくなります。この設定は、SMB がプライマリプロトコルでほかのプロトコルが二次的なものとみなされる場合、つまり想定される標準への適合性が問題とならない場合にのみ使用するようしてください。
区別する	sensitive	すべての検索で大文字と小文字を区別し、最初から検索で大文字小文字を区別しない SMB よりも優先されます。SMB サーバーは符号化された名前を使ってファイル名の重複に対処でき、また、この設定によって Windows アプリケーションが正常ではない動作をする可能性があるため、通常はこの設定を使用しないようしてください。

- 「非 UTF-8 を拒否」- この設定は、すべてのファイルおよびディレクトリに対して UTF-8 エンコーディングを強制します。これを設定した場合、無効な UTF-8 エンコー



ディングを使用してファイルまたはディレクトリを作成しようとしても失敗します。この設定は、エンコーディングが標準によって定義されていない NFSv3 にのみ影響します。NFSv4 では常に UTF-8 が使用され、SMB では適切なエンコーディングがネゴシエーションされます。通常はこの設定を「オン」にしてください。そうしない場合、(特に大文字と小文字を区別して比較を行うためにエンコーディングを認識する必要があります) SMB が、無効な UTF-8 エンコーディングで作成されるファイル名をデコードできません。この設定は、クライアントが別のエンコーディングを使用するように構成された既存の NFSv3 を配備する場合にのみ「オフ」にしてください。このプロパティが「オフ」に設定されているときに SMB または NFSv4 を有効にすると、NFSv3 クライアントが無効な UTF-8 エンコーディングのファイルまたはディレクトリを作成した場合に未定義の結果が発生する可能性があります。正規化プロパティが「なし」以外の値に設定されている場合は、このプロパティを「オン」に設定する必要があります。

- 「正規化」- この設定は、ファイルシステムやディレクトリに対して実行される Unicode 正規化がある場合に、それを制御します。Unicode は、同じ論理名を異なるエンコーディングで表す機能をサポートしています。正規化を使わないと、格納されているディスク上の名前が異なってしまう、いずれかの代替形式を使った検索がファイルの作成方法やアクセス方法によっては失敗します。このプロパティが「なし」(デフォルト) 以外の値に設定されている場合は、「非 UTF-8 を拒否」プロパティも「オン」に設定する必要があります。正規化の動作、および各種形式の動作の詳細は、Wikipedia の Unicode 正規化に関する項目を参照してください。



BUI の値	CLI の値	説明
なし	none	正規化は行われません。
形式 C	formC	<i>Normalization Form Canonical Composition (NFC)</i> - 文字は正準等価性によって分解され、再度合成されます。
形式 D	formD	<i>Normalization Form Canonical Decomposition (NFD)</i> - 文字は正準等価性によって分解されます。
形式 KC	formKC	<i>Normalization Form Compatibility Composition (NFKC)</i> - 文字は互換等価性によって分解され、正準等価性によって再度合成されます。
形式 KD	formKD	<i>Normalization Form Compatibility Decomposition (NFKD)</i> - 文字は互換等価性によって分解されます。

- 「ボリュームブロックサイズ」- LUN のネイティブブロックサイズです。512 バイトから 1M までの 2 の累乗を指定でき、デフォルトは 8K です。
- 「起点」- これがクローンの場合は、クローンの作成元のスナップショットの名前になります。
- 「データ移行ソース」- これを設定した場合、このファイルシステムはローカルで、または NFS を介して既存のファイルシステムのシャドウをアクティブに作成します。データ移行の詳細は、[第14章「シャドウ移行」](#)のセクションを参照してください。

## シェアのプロジェクトパネル

BUI では、使用可能なプロジェクトのセットが、このビューの左側のプロジェクトパネルから常に使用できます。このプロジェクトパネルを展開したり折りたたんだりするには、「プロジェクト」タイトルバーの横の三角形をクリックします。

表 12-4 プロジェクトパネルのアイコン

アイコン	説明
	プロジェクトパネルを展開します
	プロジェクトパネルを折りたたみます

このパネルからプロジェクトを選択すると、選択されたプロジェクトの [356 ページの「プロジェクト」](#)ビューに移動します。このプロジェクトパネルは、シェアリスト内の行で移動ツールをクリックしたときにも自動的に展開されます。その後、シェアをドラッグ&ドロップしてプロジェクト間で移動させることができます。プロジェクトパネルでは、ショートカットを使用して、新しいプロジェクトを作成したり、プロジェクト全体からシェアのリストに戻ったりすることもできます。「すべて」テキストをクリックすることは、ナビゲーションバーで「シェア」項目を選択するのと同じことです。

プロジェクトパネルは、プロジェクトの数が比較的少ないシステムにとって便利です。このパネルは、多数のプロジェクトを管理するためのプライマリインタフェースとなるように設計されたものではありません。このタスクについては、[356 ページの「プロジェクト」](#)ビューを参照してください。

### ▼ シェアの作成

1. 1 つのプロジェクト内またはすべてのプロジェクトにわたってシェアを表示するには、「シェア」>「シェア」に移動します。

2. 「ファイルシステム」または「LUN」を選択します。
3. 「ファイルシステム」または「LUN」の横にあるプラスアイコンをクリックします。  
「ファイルシステムを作成」または「LUN を作成」ダイアログボックスが表示されます。
4. 「ファイルシステムを作成」または「LUN を作成」ダイアログボックスで、使用するプロパティを選択または入力します。

シェアのタイプ別のプロパティは次の場所に定義されています。

ファイルシステムの場合

- [338 ページの「ユーザー」](#)
- [338 ページの「グループ」](#)
- [338 ページの「アクセス権」](#)
- [320 ページの「マウントポイント」](#)
- [第12章「シェア、プロジェクト、およびスキーマ」](#) (作成時のみ)
- [第12章「シェア、プロジェクト、およびスキーマ」](#) (作成時のみ)
- [第12章「シェア、プロジェクト、およびスキーマ」](#) (作成時のみ)

LUN の場合

- [320 ページの「ボリュームサイズ」](#)
- [320 ページの「シンプロビジョニング済み」](#)
- [第12章「シェア、プロジェクト、およびスキーマ」](#) (作成時のみ)

## CLI での「シェア」>「シェア」の操作

シェアの CLI は shares の下にあります

### ナビゲーション

最初にプロジェクト (デフォルトプロジェクトを含む) を選択してから、シェアを選択する必要があります。

```
clownfish:> shares
clownfish:shares> select default
clownfish:shares default> select foo
clownfish:shares default/foo> get
Properties:
    aclinherit = restricted (inherited)
    aclmode = discard (inherited)
    atime = true (inherited)
```

```

casesensitivity = mixed
  checksum = fletcher4 (inherited)
  compression = off (inherited)
  compressratio = 100
  copies = 1 (inherited)
  creation = Mon Oct 13 2009 05:21:33 GMT+0000 (UTC)
  mountpoint = /export/foo (inherited)
normalization = none
  quota = 0
  quota_snap = true
  readonly = false (inherited)
  recordsize = 128K (inherited)
  reservation = 0
reservation_snap = true
secondarycache = all (inherited)
  nbmand = false (inherited)
  sharesmb = off (inherited)
  sharenfs = on (inherited)
  snapdir = hidden (inherited)
  snaplabel = project1:share1
  utf8only = true
  vscan = false (inherited)
  sharedav = off (inherited)
  shareftp = off (inherited)
  space_data = 43.9K
space_unused_res = 0
space_snapshots = 0
space_available = 12.0T
space_total = 43.9K
  root_group = other
root_permissions = 700
  root_user = nobody

```

## シェアの操作

シェアを作成するには、プロジェクトを選択して、`filesystem` または `lun` コマンドを発行します。必要に応じてプロパティを変更してから、変更を確定します。

```

clownfish:shares default> filesystem foo
clownfish:shares default/foo (uncommitted)> get
  aclinherit = restricted (inherited)
  aclmode = discard (inherited)
  atime = true (inherited)
  checksum = fletcher4 (inherited)
  compression = off (inherited)
  copies = 1 (inherited)
  mountpoint = /export/foo (inherited)
  quota = 0 (inherited)
  readonly = false (inherited)
  recordsize = 128K (inherited)
  reservation = 0 (inherited)
secondarycache = all (inherited)
  nbmand = false (inherited)
  sharesmb = off (inherited)
  sharenfs = on (inherited)
  snapdir = hidden (inherited)
  snaplabel = project1:share1
  vscan = false (inherited)
  sharedav = off (inherited)

```

```

        shareftp = off (inherited)
        root_group = other (default)
        root_permissions = 700 (default)
        root_user = nobody (default)
        casesensitivity = (default)
        normalization = (default)
        utf8only = (default)
        quota_snap = (default)
        reservation_snap = (default)
        custom:int = (default)
        custom:string = (default)
        custom:email = (default)
clownfish:shares default/foo (uncommitted)> set sharenfs=off
        sharenfs = off (uncommitted)
clownfish:shares default/foo (uncommitted)> commit
clownfish:shares default>

```

シェアを破棄するには、シェアコンテキストから `destroy` コマンドを使用します。

```

clownfish:shares default/foo> destroy
This will destroy all data in "foo"! Are you sure? (Y/N)
clownfish:shares default>

```

シェアの名前を変更するには、プロジェクトコンテキストから `rename` コマンドを使用します。

```

clownfish:shares default> rename foo bar
clownfish:shares default>

```

プロジェクト間でシェアを移動させるには、プロジェクトコンテキストから `move` コマンドを使用します。

```

clownfish:shares default> move foo home
clownfish:shares default>

```

ユーザーおよびグループの使用状況と割り当て制限を管理するには、特定のプロジェクトまたはシェアを選択してから `users` または `groups` コマンドを使用します。ユーザーとグループの割り当て制限を管理する方法の詳細は、[298 ページの「スペース管理」](#)のセクションを参照してください。

## 「シェア」>「シェア」の CLI プロパティー

CLI で使用可能なプロパティーおよび相当する BUI プロパティーは次のとおりです。プロパティーは、標準の CLI コマンド `get` および `set` を使用して設定できます。また、`unset` コマンドを使用して、プロパティーを親プロジェクトから継承することもできます。

表 12-5 「シェア」>「シェア」の CLI プロパティー

CLI 名	<a href="#">294 ページのタイプ</a>	BUI 名	BUI の場所
<code>acinherit</code>	継承	<a href="#">338 ページの「ACL 継承動作」</a>	アクセス

CLI 名	294 ページの「タイプ」	BUI 名	BUI の場所
aclmode	継承	338 ページの「モード変更時の ACL 動作」	アクセス
atime	継承	320 ページの「読み取り時のアクセス時間の更新」	一般
casesensitivity	作成時	第12章「シェア、プロジェクト、およびスキーマ」	静的
checksum	継承	第12章「シェア、プロジェクト、およびスキーマ」	一般
compression	継承	320 ページの「データ圧縮」	一般
compresratio	読み取り専用	第12章「シェア、プロジェクト、およびスキーマ」	静的
copies	継承	320 ページの「追加レプリケーション」	一般
creation	読み取り専用	-	-
dedup	継承	320 ページの「データ複製解除」	一般
exported	継承、レプリケーションパッケージのみ	第13章「レプリケーション」	一般
fixednumber	LUN ローカル	329 ページの「イニシエータグループ」	プロトコル
initiatorgroup	LUN ローカル	329 ページの「イニシエータグループ」	プロトコル
logbias	継承	320 ページの「同期書き込みバイアス」	一般
lunumber	LUN ローカル	329 ページの「LU 番号」	プロトコル
lunguid	読み取り専用、LUN ローカル	329 ページの「GUID」	プロトコル
mountpoint	継承	320 ページの「マウントポイント」	一般
nbmand	継承	320 ページの「ブロック不可の必須ロック」	一般
nodestroy	継承	320 ページの「破棄の防止」	一般

CLI 名	294 ページの「タイプ」	BUI 名	BUI の場所
normalization	作成時	第12章「シェア、プロジェクト、およびスキーマ」	静的
origin	読み取り専用	第12章「シェア、プロジェクト、およびスキーマ」	静的
quota	スペース管理	298 ページの「割り当て制限」	一般
quota_snap	スペース管理	298 ページの「割り当て制限/スナップショットを含める」	一般
readonly	継承	320 ページの「読み取り専用」	一般
recordsize	継承	320 ページの「データベースレコードサイズ」	一般
reservation	スペース管理	298 ページの「予約」	一般
reservation_snap	スペース管理	298 ページの「予約/スナップショットを含める」	一般
root_group	ファイルシステムローカル	338 ページの「グループ」	アクセス
root_permissions	ファイルシステムローカル	338 ページの「アクセス権」	アクセス
root_user	ファイルシステムローカル	338 ページの「ユーザー」	アクセス
rstchown	継承	320 ページの「所有権の変更の制限」	一般
secondary cache	継承	320 ページの「キャッシュデバイスの使用状況」	一般
shadow	作成時	第14章「シャドウ移行」	静的
sharedav	継承	329 ページの「プロトコル/HTTP/シェアモード」	プロトコル
shareftp	継承	329 ページの「プロトコル/FTP/シェアモード」	プロトコル
sharenfs	継承	329 ページの「プロトコル/NFS/シェアモード」	プロトコル

CLI 名	294 ページの「タイプ」	BUI 名	BUI の場所
sharesmb	継承	329 ページの「プロトコ ル/SMB/リソース名」	プロトコル
snapdir	継承	346 ページの「.zfs/ snapshot 表示」	スナップショット
snaplabel	継承	346 ページの「定期ス ナップショットラベル」	スナップショット
space_available	読み取り専用	第12章「シェア、プロジェ クト、およびスキーマ」	使用状況
space_data	読み取り専用	第12章「シェア、プロジェ クト、およびスキーマ」	使用状況
space_snapshots	読み取り専用	第12章「シェア、プロジェ クト、およびスキーマ」	使用状況
space_total	読み取り専用	第12章「シェア、プロジェ クト、およびスキーマ」	使用状況
space_unused_res	読み取り専用	第12章「シェア、プロジェ クト、およびスキーマ」	使用状況
sparse	LUN ローカル	320 ページの「シンプ ロビジョニング済み」	一般
targetgroup	LUN ローカル	329 ページの「ターゲッ トグループ」	プロトコル
utf8only	作成時	「非 UTF-8 を拒否」	静的
volblocksize	作成時	第12章「シェア、プロジェ クト、およびスキーマ」	静的
vscan	継承	320 ページの「ウイル ススキャン」	一般

## 「シェア」>「シェア」>「一般」- BUI ページ

BUI のこのセクションでは、特定のプロトコルに依存しない、またアクセス制御やスナップショットに関連しない、シェア全体の設定を制御します。CLI ではすべてのプロパティが 1 つのリストにまとめられていますが、このセクションでは、両方のコンテキストでのプロパティの動作について説明します。

これらは、シェアにプロジェクトから継承するか明示的に設定することができる標準プロパティです。BUI では一度にすべてのプロパティを継承することしかできませんが、CLI では個別のプロパティを継承できます。



これらのプロパティと CLI がどのように対応しているかについては、[315 ページの「CLI での「シェア」>「シェア」の操作」](#)のセクションを参照してください。

## 領域の使用

ストレージプール内の領域は、すべてのシェアの間でシェアされます。ファイルシステムは必要に応じて動的に拡大または縮小できますが、シェア単位で領域制限を強制することもできます。ファイルシステム単位で割り当て制限と予約を強制できます。割り当て制限はユーザー単位およびグループ単位でも強制できます。ファイルシステムの領域の使用状況の管理 (割り当て制限と予約を含む) の詳細は、[298 ページの「スペース管理」](#)のセクションを参照してください。

## ボリュームサイズ

iSCSI でエクスポートされる LUN の論理サイズ。このプロパティは LUN に対してのみ有効です。このプロパティは LUN のサイズを制御します。デフォルトでは、LUN はボリュームを完全に満たすだけの十分な領域を予約します。詳細は、[320 ページの「シンプロビジョニング済み」](#)プロパティを参照してください。LUN がクライアントにアクティブにエクスポートされている間にサイズを変更すると、結果は不定になることがあります。その場合、クライアントが再度接続を行う必要が生じたり、LUN 上のファイルシステムのデータが破壊されたりする可能性があります。この操作を実行する前に、使用している特定の iSCSI クライアントのベストプラクティスを確認してください。

## シンプロビジョニング済み

ボリュームの領域を予約するかどうかを制御します。このプロパティは LUN に対してのみ有効です。デフォルトでは、LUN はボリュームを完全に満たすだけの十分な領域を予約します。これにより、不適切なときにクライアントで領域不足エラーが発生することはなくなります。このプロパティにより、利用可能な領域の量を超えるボリュームサイズが可能になります。設定されている場合、LUN は LUN に書き込まれている領域だけを消費します。これで LUN のシンプロビジョニングが可能になりますが、ほとんどのファイルシステムは基になるデバイスから「領域不足」を受け取ることは想定していないため、シェアで領域不足が発生すると、クライアントが不安定になったりクライアントのデータが破壊されたりする可能性があります。

設定されていない場合、ボリュームサイズの動作は、スナップショットを除いて予約と同様になります。したがって同じ欠点もあります。たとえば、利用可能な領域の量を超える点にスナップショットが理論上到達する場合は、スナップショットの作成に失敗します。詳細は、[363 ページの「プロジェクト - 予約」](#)を参照してください。

## マウントポイント

ファイルシステムがマウントされる場所。このプロパティはファイルシステムに対してのみ有効です。

「マウントポイント」プロパティには、次の制限事項があります。

- /export の下の場所を指定する必要があります。
- ほかのシェアと競合することはできません。
- フェイルオーバーを正しく行うには、クラスタピア上のほかのシェアと競合することはできません。

「マウントポイント」プロパティを継承するとき、現在のデータセットの名前がプロジェクトのマウントポイント設定に付加され、スラッシュ (/) で結合されます。たとえば、「home」プロジェクトのマウントポイント設定が /export/home の場合、「home/bob」が継承するマウントポイントは /export/home/bob になります。

SMB シェアはそのリソース名を介してエクスポートされ、マウントポイントはプロトコル上では見えません。ただし、SMB 専用のシェアにも、ZFS 上に有効な一意のマウントポイントが必要です。

いくつかの制限事項がありますが、マウントポイントはほかのシェアの下に入れ子にすることができます。詳細は、[306 ページ](#)の「[ファイルシステムの名前空間](#)」のセクションを参照してください。

## 読み取り専用

ファイルシステムの内容を読み取り専用にするかどうかを制御します。このプロパティはファイルシステムに対してのみ有効です。読み取り専用ファイルシステムの内容は、どのようなプロトコル設定であっても変更できません。ファイルシステムの名前の変更、破棄、およびプロパティの変更を行う機能は、この設定に影響されません。さらに、ファイルシステムが読み取り専用の場合、[338 ページ](#)の「[アクセス制御](#)」のプロパティを変更するには、ファイルシステムのルートディレクトリの属性を変更する必要があるため、アクセス制御のプロパティは変更できません。

## 読み取り時のアクセス時間の更新

ファイルのアクセス時間を読み取り時に更新するかどうかを制御します。このプロパティはファイルシステムに対してのみ有効です。POSIX 標準では、ファイルのアクセス時間は、ファイルが最後に読み取られた時間を適切に反映する必要があります。したがって、ほとんど読み取り専用のワークロードに使用される場合も、ベースとなるファイルシステムに

対して書き込みを発行する必要があります。主に多数のファイルの読み取りから成る作業セットの場合は、このプロパティをオフにすると、標準には準拠しなくなりますが、パフォーマンスが向上する可能性があります。このような更新は非同期に発生し、グループ化されているため、この効果が見えるのは重い負荷がかかっている場合だけです。

## ブロック不可の必須ロック

SMB ロックのセマンティクスを POSIX セマンティクスに強制するかどうかを制御します。このプロパティはファイルシステムに対してのみ有効です。デフォルトでは、ファイルシステムには POSIX 標準に従ってファイル動作が実装されます。これらの標準は、SMB プロトコルに必要な動作とは基本的に互換性を持ちません。シェアのプライマリプロトコルが SMB の場合は、このオプションを常に有効にしてください。このプロパティを変更した場合は、すべてのクライアントの接続を切断し、再度接続する必要があります。

## データ複製解除

データの重複コピーを削除するかどうかを制御します。複製解除は、プール全体に対してブロックベースで同期式に実行されます。プロジェクト単位またはシェア単位で有効化できません。有効にするには、プロジェクトまたはシェアの一般プロパティの画面で「データ複製解除」チェックボックスにチェックマークを付けます。ステータスダッシュボードの使用状況の領域に、複製解除の比率が表示されます。

複製解除が有効になっているときに書き込まれたデータは、複製解除テーブルに入力され、データのチェックサムによってインデックスが付けられます。複製解除では、暗号として強力な SHA-256 チェックサムの使用が強制されます。後続の書き込みでは複製データが識別され、既存のコピーだけがディスク上に保持されます。複製解除は、同じサイズのブロック間で、同じレコードサイズで書き込まれたデータに対してのみ実行できます。最適な結果を得るには、そのデータを使用しているアプリケーションと同じレコードサイズを設定してください。ストリーミングワークロードの場合は、大きいレコードサイズを使用してください。

データに複製が含まれていない場合に「データ複製解除」を有効にすると、オーバーヘッド (CPU 負荷のかかるチェックサム、ディスク上の複製解除テーブルのエントリ) が増加し、利点は何もありません。データに複製が含まれている場合は、「データ複製解除」を有効にすると、あるブロックのコピーは何度発生しても保存されるのは 1 つだけであるため、領域を節約できます。チェックサムの計算によって負荷が増加すること、および複製解除テーブルのメタデータに対するアクセスと管理が必要になることから、複製解除は必然的にパフォーマンスに影響を与えます。

複製解除は、シェアの計算されたサイズには影響しませんが、プールに使用される領域の量には影響を与えます。たとえば、2 つのシェアに同じ 1G バイトのファイルが含まれてい

る場合、それぞれ 1G バイトのサイズで表示されますが、プールの合計は 1G バイトだけで、複製解除の比率は 2x と報告されます。

パフォーマンスに関する警告: 複製解除では、その性質から、ブロックに対して書き込みまたは解放が行われたときに複製解除テーブルの変更が必要になります。複製解除テーブルが DRAM に取まらない場合は、書き込みや解放により、それまでは発生していなかったランダムな読み取りアクティビティが大量に発生する可能性があります。結果として、複製解除を有効にするとパフォーマンスに重大な影響を与えることがあります。さらに、特にシェアやスナッチショットを削除する場合などに、複製解除を有効にしたことによるパフォーマンス低下がプール全体に現れることがあります。一般に、シェアに高い比率で複製データが含まれていることと、複製データとそれを参照するためのテーブルが DRAM に十分に収まることがわかっている場合以外は、複製解除を有効にすることは推奨されません。複製解除がパフォーマンスに悪影響を与えているどうかを調べるには、[第8章「ZFSSA の設定」](#)を有効にしてから、『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」を使用して「DMU オブジェクトタイプ別の ZFS DMU 操作内訳」を測定し、維持された DDT 操作 (データ複製解除テーブル操作) が ZFS 操作よりも高い比率で発生しているかどうかを確認します。発生している場合は、ファイル入出力よりも複製解除テーブルのためにより多くの入出力が使用されています。

## データ圧縮

ディスクに書き込む前にデータを圧縮するかどうかを制御します。シェアは、オプションで、データを圧縮してからストレージプールに書き込むことができます。これにより、ストレージの利用率は大幅に向上しますが、CPU の使用率が高くなります。デフォルトでは、圧縮は行われません。圧縮によって領域がまったく節約されない場合は、データを再度読み取るときの不要な圧縮解除を避けるために、データがディスクにコミットされていません。圧縮アルゴリズムを選択する前に、必要なパフォーマンステストをすべて実行し、達成された圧縮率を測定することをお勧めします。

BUI の値	CLI の値	説明
オフ	オフ	圧縮は行われません
LZJB (最速)	lzjb	単純なランレングスエンコーディングで、かなり単純な入力にしか機能しませんが、CPU をあまり消費しません。
GZIP-2 (高速)	gzip-2	gzip 圧縮アルゴリズムの軽量版。
GZIP (デフォルト)	gzip	標準の gzip 圧縮アルゴリズム。
GZIP-9 (最大の圧縮)	gzip-9	gzip で達成できる最大限の圧縮。これは大量の CPU を消費し、わ

BUI の値	CLI の値	説明
		ずかな利点しか得られないことがよくあります。

## チェックサム

データブロックに使用するチェックサムを制御します。ZFSSA では、ディスク上のすべてのデータにチェックサムが付けられ、従来の欠点 (特に実体のない読み取り/書き込み) を回避しています。これにより、デバイスから無効なデータが返された場合にシステムで検出できます。通常動作にはデフォルトのチェックサム (fletcher4) で十分です。こだわりの強いユーザーはチェックサムの強度を上げることができますが、CPU 負荷は増加します。メタデータには常に同じアルゴリズムを使用してチェックサムが付けられるため、この影響を受けるのはユーザーのデータ (ファイルまたは LUN のブロック) だけです。

BUI の値	CLI の値	説明
Fletcher 2 (レガシー)	fletcher2	16 ビット Fletcher チェックサム
Fletcher 4 (標準)	fletcher4	32 ビット Fletcher チェックサム
SHA-256 (最強)	sha256	SHA-256 チェックサム

## キャッシュデバイスの使用状況

シェアにキャッシュデバイスを使用するかどうかを制御します。デフォルトでは、すべてのデータセットがシステム上の任意のキャッシュデバイスを使用します。キャッシュデバイスはストレージプールの一部として構成され、もう 1 段のキャッシュを提供して、より高速な階層型アクセスを実現します。キャッシュデバイスの詳細は、[第5章「ストレージ構成」](#)のセクションを参照してください。このプロパティは、ストレージプールに現在キャッシュデバイスが構成されているかどうかには依存しません。たとえば、キャッシュデバイスが存在しない場合でも、このプロパティを「all」に設定できます。あとでそのようなデバイスが追加されると、シェアは向上したパフォーマンスを自動的に利用します。プライマリ (DRAM) キャッシュの使用は、このプロパティに影響されません。

BUI の値	CLI の値	説明
すべてのデータとメタデータ	all	通常のファイルまたは LUN データ、およびメタデータをすべてキャッシュします。
メタデータのみ	metadata	メタデータだけをキャッシュデバイスに保持します。これを使用する

BUI の値	CLI の値	説明
		と、ディレクトリ構造をすばやくたどることができますが、ファイルの内容を取得するには、データデバイスからの読み取りが必要になることがあります。
キャッシュデバイスを使用しない	none	このシェアのデータはキャッシュデバイスにキャッシュされません。データはプライマリキャッシュにキャッシュされるか、データデバイスに保存されるだけです。

## 同期書き込みバイアス

この設定は、同期書き込みを処理するときの動作を制御します。デフォルトでは、システムは待機時間に関して同期書き込みを最適化します。つまり、ログデバイスを活用して応答時間を短くします。システムに個別のファイルシステムが複数存在する場合、ログデバイスに競合が発生し、すべてのコンシューマで待機時間が増加する可能性があります。複数のファイルシステムが同期セマンティクスをリクエストしている場合でも、一部のファイルシステムがほかのファイルシステムよりも待機時間の影響を受けやすくなる可能性があります。

よくあるのが、個別のログを持つデータベースのケースです。ログは待機時間の影響を特に受けやすく、データベース自体は、同様に同期セマンティクスを要求していても、待機時間ではなく帯域幅の影響を受けます。このような環境では、このプロパティをメインデータベースでは「throughput」に設定し、ログファイルシステムでは「latency」のままにしておく、パフォーマンスが大幅に向上する可能性があります。ログデバイスが存在しない場合でも、この設定によって動作が変更されますが、それほど大きな影響はない可能性があります。

Oracle Intelligent Storage Protocol では、「同期書き込みバイアス」設定を省略できます。Oracle Intelligent Storage Protocol では、ファイルシステムで定義されている書き込みバイアスを使用する代わりに、Oracle Database NFSv4 クライアントによって提供される書き込みバイアス値を使用できます。Oracle Database NFSv4 クライアントによって送信された書き込みバイアス値は、その書き込みリクエストのためにのみ使用されます。詳細は、[488 ページの「Oracle Intelligent Storage Protocol」](#)を参照してください。

BUI の値	CLI の値	説明
待機時間	latency	同期書き込みは、専用のログデバイスがあればそれを活用して、待機時間に関して最適化されます。

BUI の値	CLI の値	説明
スループット	throughput	同期書き込みはスループットに関して最適化されます。データはログデバイスではなくプライマリデータディスクに書き込まれ、システムの合計帯域幅が最適になるような方法で書き込みが実行されます。

## データベースレコードサイズ

ファイルシステムに格納するファイルの推奨ブロックサイズを指定します。このプロパティは、ファイルシステムに対してのみ有効であり、レコードサイズが固定されているファイルにアクセスするデータベースワークロードで使用するよう設計されています。このシステムでは、標準的なアクセスパターンに最適化された内部アルゴリズムに従って、ブロックサイズが自動的に調整されます。

作成されるファイルのサイズが非常に大きく、それらのファイルにさまざまなパターンの小さなブロック単位でアクセスするデータベースの場合には、このようなアルゴリズムが最適でないことがあります。データベースのレコードサイズ以上のレコードサイズを指定すると、パフォーマンスが大幅に向上することがあります。このプロパティを汎用目的のファイルシステムに使用することは、パフォーマンスが低下する可能性があるため、できるだけ避けてください。

デフォルトのレコードサイズは 128K バイトです。指定するサイズは、512 バイト - 1M バイトの 2 の累乗にしてください。ファイルシステムのレコードサイズの変更は、その後で作成したファイルにのみ適用され、既存のファイルと受信データには影響しません。

注: 128K より大きいブロックサイズがプロジェクトまたはシェアに使用される場合、大きなブロックサイズをサポートしていないシステムへの、それらのプロジェクトまたはシェアのアプリケーションは失敗します。

Oracle Intelligent Storage Protocol では、「データベースレコードサイズ」設定を省略できます。Oracle Intelligent Storage Protocol では、ファイルシステムで定義されているレコードサイズを使用する代わりに、Oracle Database NFSv4 クライアントによって提供されるブロックサイズ値を使用できます。Oracle Database NFSv4 クライアントによって提供されるブロックサイズは、新しいデータベースファイルまたはテーブルを作成する場合にのみ適用できます。既存のファイルやテーブルのブロックサイズは変更されません。詳細は、[488 ページの「Oracle Intelligent Storage Protocol」](#)を参照してください。

## 追加レプリケーション

ストレージプールの冗長性に加え、各ブロックのコピーをいくつ保存するかを制御します。メタデータは常に複数のコピーで保存されますが、このプロパティを使用すると、同じ動作をデータブロックにも適用できます。ストレージプールは、これらの追加ブロックを異なるデバイスに保存しようと試みますが、保証はされません。さらに、論理デバイス全体 (RAID ストライプやミラー化されたペアなど) が失われた場合、ストレージプールをインポートすることはできません。このプロパティは、ストレージプールでの正式なレプリケーションに代わるものではありませんが、こだわりの強い管理者に安心を与えることができます。

BUI の値	CLI の値	説明
通常 (単一コピー)	1	デフォルトの動作。データブロックのコピーを 1 つ保存します。
2 コピー	2	データブロックのコピーを 2 つずつ保存します。
3 コピー	3	データブロックのコピーを 3 つずつ保存します。

## ウイルススキャン

このファイルシステムに対してウイルススキャンを行うかどうかを制御します。このプロパティはファイルシステムに対してのみ有効です。このプロパティの設定は、ウイルススキャンサービスの状態には依存しません。ウイルススキャンサービスが有効になっていても、ファイルシステムのスキャンはこのプロパティで明示的に有効にする必要があります。同様に、ウイルススキャンサービス自体がオフになっていても、特定のシェアに対してウイルススキャンを有効化できます。ウイルススキャンの構成の詳細は、[245 ページの「ウイルススキャン」](#)のセクションを参照してください。

## 破棄の防止

設定されている場合、そのシェアまたはプロジェクトを破棄することはできません。これには、従属クローンを介したシェアの破棄、プロジェクト内のシェアの破棄、およびレプリケーションパッケージの破棄も含まれます。ただし、レプリケーションの更新を介して破棄されるシェアは、このプロパティに影響されません。レプリケーションのソースになっている ZFSSA 上のシェアが破棄される場合、このプロパティが設定されていても、ターゲット上の対応するシェアは破棄されます。



シェアを破棄するには、別の手順としてこのプロパティをまず明示的にオフにする必要があります。このプロパティはデフォルトでオフになっています。

## 所有権の変更の制限

デフォルトでは、(適切なクライアント上で root 対応のエクスポートで) root ユーザーが行う場合を除き、ファイルの所有権を変更することはできません。このプロパティをオフにすることによって、ファイルシステム単位またはプロジェクト単位でこのプロパティをオフにすることができます。オフの場合、ファイルまたはディレクトリの所有者がファイルの所有権を変更でき、自分のファイルを実質的に「譲渡」できます。所有権を変更すると、setuid ビットと setgid ビットがすべて取り除かれ、ユーザーがこの操作によって権限を上げることが防ぎます。

## カスタムプロパティ

必要に応じてカスタムプロパティを追加して、ユーザー定義のタグをプロジェクトとシェアに添付できます。詳細は、[368 ページの「スキーマ」](#)を参照してください。

# 「シェア」>「シェア」>「プロトコル」 - BUI ページ

## シェアのプロトコル

各シェアにはプロトコル固有のプロパティがあり、そのシェアに対してさまざまなプロトコルの動作を定義します。これらのプロパティは、シェアごとに定義することも、シェアのプロジェクトから継承することもできます。[202 ページの「NFS」](#)、[210 ページの「SMB」](#)、[228 ページの「HTTP」](#)、および [226 ページの「FTP」](#) のプロパティはファイルシステムにのみ適用されます。一方、[208 ページの「iSCSI」](#) のプロパティは LUN にのみ適用されます。

BUI では、各プロトコルにパスが表示されます。そのプロトコルを使用しているクライアントは、このパスを使用してシェアを参照します。たとえば、サーバー「twofish」上のファイルシステム「fs0」は、次の場所で使用できます。

表 12-6 シェアのプロトコル

プロトコル	場所
NFS	twofish:/export/fs0

プロトコル	場所
SMB	¥¥twofish¥fs0
HTTP	//twofish/shares/export/fs0/
FTP	ftp://twofish/export/fs0/
SFTP	/export/fs0/

iSCSI の場合、イニシエータは [第6章「Storage Area Network の構成」](#)で説明されているメカニズムのいずれかを使用すると、ターゲットを発見できます。

## シェアのプロトコル - NFS

表 12-7 シェアのプロトコル - NFS のプロパティ

BUI のプロパティ	CLI のプロパティ	説明
シェアモード	off/ro/rw	シェアが読み取り専用か、読み取り/書き込みに使用できるか、どちらにも使用できないかを決定します。CLI では、「on」は「rw」の別名です。
setuid/setgid ファイル作成を無効にする	nosuid	このオプションを選択すると、クライアントは setuid (S_ISUID) ビットと setgid (S_ISGID) ビットを設定してファイルを作成することも、chmod(2) システム呼び出しによって既存のファイルにこれらのビットを設定することもできなくなります。
クライアントがサブディレクトリをマウントできないようにします	nosub	このオプションを選択すると、クライアントがサブディレクトリを直接マウントすることはできなくなります。シェアのルートのみをマウントするように強制されます。注: これは NFSv2 プロトコルと NFSv3 プロトコルだけに適用され、NFSv4 には適用されません。
匿名ユーザーマッピング	anon	特定のクライアントで「root」オプションが有効になっている場合を除き、そのクライアントの root ユーザーは不明なユーザーとして扱われ、そのユーザーがシェアのファイルにアクセスを試みると、すべてこ

BUI のプロパティ	CLI のプロパティ	説明
		の uid のユーザーによる操作として扱われます。そのあと、ファイルのアクセス権ビットと ACL は通常どおり評価されます。
文字エンコーディング	下記を参照	すべてのクライアントのデフォルトの文字セットを設定します。詳細は、文字セットエンコーディングに関するセクションを参照してください。
セキュリティーモード	下記を参照	すべてのクライアントのセキュリティーモードを設定します。

全体的なシェアモードに対する例外は、クライアントごと、またはクライアントの集まりごとに定義できます。クライアントがアクセスを試みると、リスト内でそのクライアントに最初に一致した例外に従ってアクセスが許可されます。あるいは、そのような例外がない場合は、上記で定義されたグローバルなシェアモードに従ってアクセスが許可されます。このようなクライアントの集まりは、次の 3 つのタイプのいずれかを使用すると定義できます。

表 12-8 クライアントの集まりのタイプ

タイプ	CLI の接頭辞	説明	例
ホスト (FQDN) または ネットグループ	none	指定された完全修飾名に解決される IP アドレスを持つ単一のクライアント、または、クライアントの IP アドレスを解決した完全修飾名を含んでいる ネットグループ	caji.sf.example.com
DNS ドメイン	.	この接尾辞で終わる完全修飾名に解決される IP アドレスを持つすべてのクライアント	sf.example.com
ネットワーク	@	CIDR 表記で指定された IP サブネットに含まれる IP アドレスを持つすべてのクライアント	192.168.20.0/22

次に、指定したクライアントまたはクライアントの集まりに対して、それぞれ 2 つのパラメータを指定します。1 つは、シェアに対する読み取り専用アクセスまたは読み取り/書き込みアクセスをクライアントに許可するかどうか、もう 1 つは、クライアントの root ユーザーを root ユーザー (選択されている場合) として扱うか不明なユーザーとして扱うかです。

ネットグループを使用する場合、ネットグループはまず [248 ページの「NIS」](#) (有効になっている場合)、次に [250 ページの「LDAP」](#) (有効になっている場合) から解決されます。LDAP を使用する場合は、ネットグループがデフォルトの場所 ou=Netgroup、(ベース DN) にあることと、標準スキーマを使用していることが必要です。通常、ネットグループエントリのユーザー名コンポーネントは NFS に影響を与えず、ホスト名だけが意味を持ちます。ネットグループに含まれるホスト名は正規名であること、また、DNS で解決される場合は完全修飾名であることが必要です。つまり、NFS サブシステムは、リクエストしているクライアントの IP アドレスを解決すると、指定された FQDN または指定されたネットグループのいずれかのメンバーに一致する正規ホスト名になることを検証しようとしています。ドメインのコンポーネントも含めて正確に一致する必要があります。それ以外の場合、その例外には一致せず、次の例外が調べられます。ホスト名の解決の詳細は、[267 ページの「DNS」](#) を参照してください。ネットグループの管理は複雑になることがあります。可能な場合は、代わりに IP サブネット規則または DNS ドメイン規則を使用することを検討してください。

2013.1.0 ソフトウェアリリースでは、Unix クライアントユーザーはパフォーマンスを低下させることなく、最大 1024 グループに所属することが可能です。それ以前のリリースでは、Unix クライアントユーザーにつき最大 16 グループをサポートしていました。

## シェアのプロトコル - CLI

CLI では、すべての NFS シェアモードと例外を、「sharenfs」プロパティの 1 つのオプション文字列で指定します。この文字列は、上記の表の値をコンマで区切ったリストです。BUI に関して説明したグローバルなシェアモードに相当する、「ro」、「rw」、または「off」のいずれかで始める必要があります。例:

```
set sharenfs=ro
```

このコマンドは、すべてのクライアントのシェアモードを読み取り専用を設定します。すべてのクライアントで、root ユーザーは一般的な「nobody」ユーザーとしてシェアのファイルにアクセスします。

「nosuid」オプションまたは「anon」オプション、あるいはその両方を追加することもできます。CLI では、プロパティ値に「=」文字が含まれている場合、引用符で囲む必要があることを忘れないでください。したがって、不明なユーザーをすべて uid 153762 にマッピングするように定義するには、次のように指定します

```
set sharenfs="ro,anon=153762"
```

追加の例外は、「option=collection」という形式のテキストを追加することで指定できます。ここで、「option」は、クライアントの集まり (collection) に付与するアクセス権のタイプを定義する、「ro」、「rw」、「root」のいずれかです。collection の指定には、上記の表の接頭辞文字と、DNS ホスト名/ドメイン名または CIDR ネットワーク番号を使用します。た

たとえば、sf.example.com ドメイン内のすべてのホストに読み取り/書き込みアクセス権を付与し、192.168.44.0/24 ネットワーク内のホストに root アクセス権を付与するには、次のようなコマンドを使用します

```
set sharenfs="ro,anon=153762,rw=.sf.example.com,root=@192.168.44.0/24"
```

ネットグループ名は、個別の完全修飾ホスト名を使用できる場所であればどこでも使用できます。たとえば、「engineering」ネットグループに読み取り/書き込みアクセス権を付与するには、次のように指定します。

```
set sharenfs="ro,rw=engineering"
```

セキュリティーモードは、「option=mode」という形式のテキストを追加することで指定できます。ここで、「option」は「sec」、「mode」は「sys」、「krb5」、「krb5:krb5i」、「krb5:krb5i:krb5p」のいずれかです。

```
set sharenfs="sec=krb5"
```

## セキュリティーモード

セキュリティーモードはシェア単位で設定され、パフォーマンスに影響を与えることがあります。次の表では、Kerberos セキュリティー設定について説明します。

表 12-9 Kerberos セキュリティー設定

設定	説明
krb5	Kerberos V5 によるエンドユーザー認証
krb5i	krb5 に完全性保護を加えたもの (データパケットに改ざんがないことが保証される)
krb5p	krb5i にプライバシー保護を加えたもの (データパケットに改ざんがないことと暗号化されていることが保証される)

Kerberos フレーバの組み合わせをセキュリティーモード設定で指定できます。組み合わせのセキュリティーモードにより、クライアントは一覧表示されている任意の Kerberos フレーバでマウントできます。

表 12-10 セキュリティーモードの設定

設定	メニュー
sys	システム認証

設定	メニュー
krb5	Kerberos v5 のみ - クライアントはこのフレーバを使用してマウントする必要があります。
krb5:krb5i	Kerberos v5 に完全性を加えたもの - クライアントは一覧表示されている任意のフレーバを使用してマウントできます。
krb5i	Kerberos v5 完全性のみ - クライアントはこのフレーバを使用してマウントする必要があります。
krb5:krb5i:krb5p	Kerberos v5 に完全性またはプライバシーを加えたもの - クライアントは一覧表示されている任意のフレーバを使用してマウントできます。
krb5p	Kerberos v5 プライバシのみ - クライアントはこのフレーバを使用してマウントする必要があります。

NFS および Kerberos の詳細は、次を参照してください。

- <http://www.ietf.org/rfc/rfc2623.txt> (<http://www.ietf.org/rfc/rfc2623.txt>)  
(NFSv2 および NFSv3 セキュリティー)
- <http://www.ietf.org/rfc/rfc3530.txt> (<http://www.ietf.org/rfc/rfc3530.txt>)  
(NFSv4 プロトコル)

## 文字セットエンコーディング

通常、ファイル名に使用する文字セットエンコーディングは指定されません。NFSv3 プロトコルと NFSv2 プロトコルは文字セットを指定しません。NFSv4 では UTF-8 を使用することになっていますが、すべてのクライアントが従うわけではなく、この制限がサーバーによって強制されることもありません。シェアの「UTF-8 のみ」オプションが無効になっている場合、これらのファイル名は、エンコーディングが認識されることなく、文字どおりにファイルシステムに書き込まれます。したがって、同じエンコーディングを使用しているクライアントのみがこれらを解釈できます。ただし、SMB では、ファイル名をサーバー側で解釈できるように、ファイル名を UTF-8 で保存することが要求されます。これにより、任意のクライアントエンコーディングをサポートすることはできなくなりますが、SMB でのアクセスは引き続き可能です。

そのような構成をサポートするには、文字セットエンコーディングをシェア全体またはクライアント単位で設定します。サポートされている文字セットエンコーディングは次のとおりです。

- cp932
- euc-cn

- euc-jp
- euc-jpms
- euc-kr
- euc-tw
- iso8859-1
- iso8859-2
- iso8859-5
- iso8859-6
- iso8859-7
- iso8859-8
- iso8859-9
- iso8859-13
- iso8859-15
- koi8-r
- shift\_jis

デフォルトの動作では、文字セットエンコーディングは指定されません (パススルー)。BUI では、標準の例外リストメカニズムによって文字セットを選択できます。CLI では、各文字セット自体がオプションになり、そこに 1 つ以上のホストを指定します。「\*」は、シェア全体の設定であることを示します。たとえば、次のコマンドを見てください。

```
set sharenfs="rw,euc-kr=*"
```

このコマンドは、「euc-kr」をデフォルトのエンコーディングとしてファイルシステムをシェアします。次のコマンドを見てください。

```
set sharenfs="rw,euc-kr=host1.domain.com,euc-jp=host2.domain.com"
```

このコマンドは、host1 には「euc-kr」、host2 には「euc-jp」を使用し、ほかのすべてのクライアントにはデフォルトのエンコーディングを使用します。ホストのリストの形式は、ほかの CLI NFS オプションの形式に従います。

一部の NFS クライアントでは、代替ロケールが正しくサポートされません。詳細は、使用している NFS クライアントのドキュメントを参照してください。

## シェア - SMB

- 「リソース名」- [210 ページの「SMB」](#) クライアントがこのシェアを参照するときに使用する名前。リソース名「オフ」はどの [210 ページの「SMB」](#) クライアントもこのシェアに

アクセスできないことを示し、リソース名「on」はこのシェアがファイルシステムの名前でエクスポートされることを示します。

- 「アクセスベースの列挙を有効化」- このオプションを有効にすると、アクセスベースの列挙が実行されます。アクセスベースの列挙では、クライアントの資格に基づいてディレクトリのエントリがフィルタリングされます。クライアントがファイルまたはディレクトリに対するアクセス権を持っていない場合、クライアントに返されるエントリのリストでそのファイルは省略されます。このオプションはデフォルトで無効になっています。
- 「DFS 名前空間です」- このプロパティは、このシェアが独立した [210 ページの「DFS 名前空間」](#)としてプロビジョニングされるかどうかを示します。
- 「シェアレベル ACL」- シェア内のファイルまたはディレクトリの ACL と組み合わせられることで、そのファイルの実質的なアクセス権を決定する ACL。デフォルトでは、この ACL はすべてのユーザーに完全な制御を許可します。この ACL は、ファイルの ACL の上にもう 1 段のアクセス制御を提供して、より詳細なアクセス制御構成を可能にします。このプロパティは、SMB リソース名を構成することによってファイルシステムがエクスポートされたあとでのみ構成できます。ファイルシステムが SMB プロトコルでエクスポートされない場合は、シェアレベルの ACL を設定しても何も効果はありません。

同じシステムの 2 つの [210 ページの「SMB」](#) シェアに、同じリソース名を使用することはできません。プロジェクトから継承されるリソース名には特殊な動作があります。詳細は、[356 ページの「プロジェクト」](#)を参照してください。リソース名には、次の文字を除く任意の 80 文字未満の英数字を含めることができます。

" / \ [ ] : | < > + ; , ? \* =

アクセスベースの列挙を有効にすると、クライアントが開くことができないファイルがディレクトリエントリとしてクライアントに表示されることがあります。ディレクトリエントリがフィルタリングされるのは、クライアントがそのファイルに対して何もアクセス権を持っていない場合だけです。たとえば、ACL で読み取りアクセス権だけが付与されている場合に、クライアントが読み取り/書き込みアクセスのためにファイルを開こうとすると、そのリクエストは失敗しますが、ファイルはエントリのリストに含まれたままです。

## シェア - iSCSI

- 「ターゲットグループ」- この LUN がエクスポートされるターゲット。
- 「イニシエータグループ」- この LUN にアクセスできるイニシエータ。2013.1.0 ソフトウェアリリースでは、複数のイニシエータグループを LUN に割り当てることができません。イニシエータグループの編集時に、PERSIST チェックボックスにチェックマークを付けると (デフォルト)、対応するイニシエータグループの LUN 番号が保持されます。



チェックマークを外すと、ZFSSA は SAN 構成の変更またはリブート後に、LUN を再割り当てする可能性があります。

- 「LU (論理ユニット) 番号」- LUN はターゲットグループおよびイニシエータグループと関連付けられるため、ターゲットグループとイニシエータのペアごとに一意の論理ユニット番号が割り当てられます。イニシエータがターゲットグループ経由でアクセス可能な 2 つの LUN が、論理ユニット番号を共有することはできません。このプロパティは、論理ユニットが番号 0 または自動的に割り当てられた番号のどちらを保持する必要があるかを制御します。
- 「動作ステータス」- この LUN の動作ステータス。ターゲットやイニシエータの構成にかかわらず、オフライン状態の LUN にはイニシエータからアクセスできません。
- 「書き込みキャッシュ動作」- この設定は、LUN で書き込みをキャッシュするかどうかを制御します。この設定がオフの場合は、すべての書き込みが同期処理され、使用可能なログデバイスがない場合は、書き込みのパフォーマンスがかなり低下します。したがって、この設定をオンにすると、書き込みのパフォーマンスが大幅に向上する可能性があります。予期しないシャットダウンまたはフェイルオーバーでデータが破壊されることもあります。これを避けるには、クライアントアプリケーションが揮発性書き込みキャッシュのセマンティクスを理解し、必要なときに適切にキャッシュをフラッシュする必要があります。これをオンにする前に、使用しているクライアントアプリケーションのドキュメントを参照してください。
- 「GUID」- LUN の GUID は、SCSI デバイスを識別する、グローバルに一意の読み取り専用識別子です。この GUID は、さまざまなヘッドノードおよびレプリケートされた環境で一貫性を保ちます。

## シェア - HTTP

表 12-11 シェア - HTTP のプロパティ

プロパティ	説明
シェアモード	このファイルシステムの HTTP シェアモード。「なし」、「読み取り専用」、「読み取り/書き込み」のいずれかです。

## シェア - FTP

表 12-12 シェア - FTP のプロパティ

プロパティ	説明
シェアモード	このファイルシステムの FTP シェアモード。「なし」、「読み取り専用」、「読み取り/書き込み」のいずれかです。

## シェア - SFTP

表 12-13 シェア - SFTP のプロパティ

プロパティ	説明
シェアモード	このファイルシステムの SFTP シェアモード。「なし」、「読み取り専用」、「読み取り/書き込み」のいずれかです。

## 「シェア」>「シェア」>「アクセス」

### アクセス制御

このビューでは、ACL の動作を制御するオプションと、ファイルシステムのルートディレクトリに対するアクセスを制御するオプションを設定できます。このビューはファイルシステムに対してのみ使用できます。

### シェア - ルートディレクトリアクセス

ファイルシステムのルートディレクトリに対する基本的なアクセス制御を設定します。これらの設定は、使用されている任意のプロトコルを介して帯域内で管理できますが、便宜上ここで指定することもできます。読み取り専用ファイルシステムでは、これらのプロパティを変更するにはファイルシステムのルートディレクトリのメタデータを変更する必要があるため、これらのプロパティは変更できません。

### シェア - ユーザー

ルートディレクトリの所有者。これはユーザー ID またはユーザー名で指定できます。Unix ユーザーと Windows ユーザーのマッピングの詳細は、[260 ページの「アイデンティティマッピング」](#)サービスを参照してください。Unix ベースの NFS アクセスの場合、これはクライアントから `chown` コマンドを使用すると変更できます。

### シェア - グループ

ルートディレクトリのグループ。これはグループ ID またはグループ名で指定できます。Unix グループと Windows グループのマッピングの詳細は、[260 ページの「アイデンティティマッピング」](#)サービスを参照してください。Unix ベースの NFS アクセスの場合、これはクライアントから `chgrp` コマンドを使用すると変更できます。

## シェア - アクセス権

ルートディレクトリに対する標準の Unix アクセス権。Unix ベースの NFS アクセスの場合、これはクライアントから `chmod` コマンドを使用すると変更できます。アクセス権は 3 つのタイプに分かれています。

表 12-14 シェアのユーザー

アクセスタイプ	説明
ユーザー	ディレクトリの現在の所有者であるユーザー。
グループ	ディレクトリの現在のグループ。
その他	ほかのすべてのアクセス。

各アクセスタイプには、次のアクセス権を付与できます。

表 12-15 シェアのアクセス権

タイプ		説明
読み取り	R	ディレクトリの内容を一覧表示するアクセス権。
書き込み	W	ディレクトリにファイルを作成するアクセス権
実行	X	ディレクトリのエントリを検索するアクセス権。実行アクセス権はあるが読み取りアクセス権がない場合、ユーザーはファイル名を指定して明示的にファイルにアクセスすることはできますが、ディレクトリの内容を一覧表示することはできません。

- 2011.1 ソフトウェアリリースでは、次の追加動作がすべてのディレクトリに対する「書き込み」権と関連付けられます。
- ディレクトリにスティッキービットが設定されていない場合は、ディレクトリ内の子ファイルを削除することができ (ACL の D アクセス権と同等)、設定されている場合はファイルの所有者のリクエストにかぎり、子ファイルを削除できます
- ファイルまたはディレクトリに関連付けられた時間を変更できます (ACL の A アクセス権と同等)
- 拡張属性を作成したり、拡張属性ディレクトリに書き込んだりできます (ACL の W アクセス権と同等)

BUI では、個々のボックスをクリックしてアクセス権を選択します。または、ラベル（「ユーザー」、「グループ」、「その他」）をクリックすると、そのラベル内のアクセス権がすべて選択（または選択解除）されます。CLI では、標準の Unix 8 進数値で指定します。各桁は、順に「ユーザー」、「グループ」、「その他」に対応します。各桁は、読み取り (4)、書き込み (2)、および実行 (1) の合計です。したがって、アクセス権の値が 743 であれば、ユーザー RWX、グループ R、その他 WX に相当します。

シェアの作成時に POSIX アクセス権ビットを設定する代わりに、管理者は「Windows のデフォルトのアクセス権を使用」オプションを選択することもできます。これにより、後述の [338 ページの「ルートディレクトリ ACL」](#) のセクションで説明されている ACL が適用されます。これは Windows サーバーでのシェアの作成と似た動作を提供するためのもので、Windows に慣れたユーザーのみまたは主にそのようなユーザーが管理している環境では、これによって管理が簡単になります。

## シェア - ACL の動作

ACL およびその動作については、[338 ページの「ルートディレクトリ ACL」](#) のドキュメントを参照してください。

### モード変更時の ACL 動作

chmod(2) で標準的な Unix ユーザー/グループ/その他アクセス権を使用して ACL が変更される場合、このプロパティの設定に応じて、単純なモード変更リクエストが既存の ACL とさまざまに作用します。

表 12-16 モード変更の値

BUI の値	CLI の値	説明
ACL を破棄	discard	ディレクトリまたはファイルのモードを表さないすべての ACL エントリが破棄されます。これはデフォルトの動作です。
モードによる ACL のマスク	mask	グループアクセス権ビットより大きくならないように、アクセス権が縮小されます。ただし、ユーザーエントリの UID がファイルまたはディレクトリの所有者と同じである場合は例外です。この場合は、所有者アクセス権ビットよりも大きくならないように ACL アクセス権が縮小されます。マスク値によって、明示的な ACL 設定操作が実行されていない

BUI の値	CLI の値	説明
		い場合は、モード変更時も ACL が維持されます。
ACL を変更しない	passthrough	ファイルまたはディレクトリの新しいモードを表すのに必要な ACL エントリを生成する以外に、ACL は変更されません。

## ACL 継承動作

新しいファイルまたはディレクトリの作成時に、既存の ACL 設定を親ディレクトリから継承できます。このプロパティは、この継承の動作を制御します。通常、これらのプロパティの設定は、継承可能としてフラグがつけられている ACL エントリだけに影響を与えます。ほかのエントリは、このプロパティの設定にかかわらず、伝播されません。ただし SMB で使用されるときは、簡単な ACL エントリはすべて継承可能です。簡単な ACL とは、従来の Unix owner/group/other エントリのことです。

表 12-17 ACL 継承動作の値

BUI の値	CLI の値	説明
エントリを継承しない	discard	ACL のエントリは継承されません。ファイルやディレクトリは、使用されているクライアントとプロトコルに従って作成されます。
拒否エントリのみを継承する	noallow	「deny」アクセス権を指定している継承可能な ACL エントリだけが継承されます。
「ACL の書き込み」および「所有者の変更」以外のすべてを継承	restricted	ACL エントリを継承するときに「write_acl」アクセス権と「write_owner」アクセス権を削除します。ただし、それ以外の場合は、継承可能な ACL エントリを変更しません。これはデフォルトです。
すべてのエントリを継承	passthrough	継承可能な ACL エントリがすべて継承されます。通常、「passthrough」モードを使用するのは、ディレクトリツリー内のすべての「データ」ファイルが同一のモードで作成されるようにする場合です。管理者は、すべてのファイルが 0664 や 0666 などの同一モードで作成されるように ACL の継承を設定します。

BUI の値	CLI の値	説明
指定がない場合、「実行」以外のすべてを継承	passthrough-x	「passthrough」と同様ですが、相違点は、所有者、グループ、およびすべてのユーザーの ACL エントリが、ファイル作成モードも実行ビットをリクエストしている場合のみ、実行アクセス権を継承することです。「passthrough」設定はデータファイルに対して期待どおりに機能しますが、継承された ACL にファイル作成モードの実行ビットを必要に応じて含めたい場合もあります。cc や gcc などのツールから生成される出力ファイルはその一例です。継承された ACL に実行ビットが含まれていない場合、コンパイラから出力された実行可能ファイルは、chmod(1) を使用してファイルのアクセス権を変更するまで実行できません。

SMB を使用して簡単な ACL の設定されたディレクトリにファイルを作成すると、すべての ACL エントリが継承されます。この結果、次の動作が生じます。

- 継承ビットの表示方法は、SMB と NFS とで異なります。SMB で ACL ディレクトリを表示すると、継承ビットが表示されます。NFS では継承ビットが表示されません。
- SMB を使用してファイルをディレクトリに作成すると、その ACL エントリは継承されたとおりに表示されます。しかし NFS を使用して表示した場合、ディレクトリには継承可能な ACL エントリがありません。
- アクセス制御エントリ (ACE) を追加するなどして ACL が変更されて簡単ではなくなった場合、この動作は発生しません。
- SMB を使用して ACL を変更すると、得られる ACL では前の継承ビットを合成したものが実際の継承ビットになります。

上記のすべての動作は、将来のリリースで変更される可能性があります。

## ルートディレクトリ ACL

ファイルとディレクトリに対する詳細なアクセス権は、アクセス制御リストによって管理されます。ACL は、特定のユーザーまたはグループにどのようなアクセス権を付与するかを記述します。ZFS では NFSv4 形式の ACL がサポートされており、SMB でのアクセスも可能です。POSIX ドラフト ACL (NFSv3 で使用される) はサポートされていません。簡単

な ACL であれば NFSv3 で表現できる場合もありますが、ACL に複雑な変更を加えると、NFSv3 でアクセスしたときの動作が不定になることがあります。

「ルートディレクトリアクセス」と同様に、このプロパティはファイルシステムのルートディレクトリだけに影響を与えます。ACL は帯域内プロトコル管理によって制御できますが、BUI にはファイルシステムのルートディレクトリだけに ACL を設定する方法が用意されています。ルートディレクトリの ACL を CLI から設定する方法はありません。BUI を使用しない場合は、帯域内管理ツールを使用できます。この ACL を変更しても、ファイルシステム内の既存のファイルとディレクトリに影響はありません。新しく作成されるファイルとディレクトリにこれらの設定が継承されるかどうかは、ACL の継承動作によって異なります。ただし、SMB を使用して簡単な ACL の設定されたディレクトリにファイルを作成すると、すべての ACL エントリが継承されます。

ACL は任意の数の ACE (アクセス制御エントリ) から成ります。各 ACE は、タイプ/ターゲット、モード、一連のアクセス権、および継承フラグを記述します。ACE は ACL の先頭から順に適用され、特定のアクションを許可するかどうかを決定します。データプロトコルを介した帯域内構成 ACL については、該当するクライアントのドキュメントを参照してください。ACL を管理するための BUI インタフェースとルートディレクトリに対する影響については、ここで説明します。

表 12-18 シェア - ACL のタイプ

タイプ	説明
所有者	ディレクトリの現在の所有者。所有者が変更された場合、この ACE は新しい所有者に適用されます。
グループ	ディレクトリの現在のグループ。グループが変更された場合、この ACE は新しいグループに適用されます。
全員	任意のユーザー。
名前付きユーザー	「ターゲット」フィールドで指定されたユーザー。ユーザーは、ユーザー ID で指定したり、現在のネームサービス構成で解決可能な名前で指定したりできます。
名前付きグループ	「ターゲット」フィールドで指定されたグループ。グループは、グループ ID で指定したり、現在のネームサービス構成で解決可能な名前で指定したりできます。

表 12-19 シェア - ACL のモード

モード	説明
 許可	ACE のターゲットにアクセス権が明示的に許可されます。

モード	説明
拒否	ACE のターゲットにアクセス権が明示的に拒否されます。

表 12-20 シェア - ACL のアクセス権

	アクセス権	説明
	読み取り	
(r)	データの読み取り/ディレクトリをリスト表示	ディレクトリの内容を一覧表示するアクセス権。ファイルによって継承された場合は、ファイルのデータを読み取るアクセス権。
(x)	ファイルの実行/ディレクトリのトラバース	ディレクトリ内のエントリをたどる (検索する) アクセス権。ファイルによって継承された場合は、ファイルを実行するアクセス権。
(a)	属性の読み取り	ファイルの基本的な属性 (ACL 以外) を読み取るアクセス権。ファイルの基本的な属性は stat レベルの属性とみなされ、このアクセス権を許可すると、ユーザーは ls と stat に相当する操作を実行できるようになります。
(R)	拡張属性の読み取り	ファイルの拡張属性を読み取る、あるいは拡張属性ディレクトリの検索を実行するアクセス権。
	書き込み	
(w)	データを書き込み/ファイルを追加	ディレクトリに新しいファイルを追加するアクセス権。ファイルによって継承された場合は、ファイルのオフセット範囲内で任意の位置のデータを変更するアクセス権。これには、ファイルを拡大する機能や、任意のオフセットに書き込む機能も含まれます。
(p)	データの追加/サブディレクトリの追加	ディレクトリ内にサブディレクトリを作成するアクセス権。ファイルによって継承された場合は、ファイルの末尾以降のデータを変更するアクセス権。ファイルに適用される場合、このアクセス権は現在サポートされていません。



	アクセス権	説明
(d)	削除	ファイルを削除するアクセス権。
(D)	子の削除	ディレクトリ内のファイルを削除するアクセス権。2011.1 ソフトウェアリリースでは、スティッキービットが設定されている場合はファイルの所有者のみが子ファイルを削除できます。
(A)	属性を書き込む	ファイルまたはディレクトリに関連付けられた時間を変更するアクセス権。
(W)	拡張属性を書き込む	拡張属性を作成する、あるいは拡張属性ディレクトリに書き込むアクセス権。
	Admin	
(c)	ACL/アクセス権を読み取る	ACLを読み取るアクセス権。
(C)	ACL/アクセス権を書き込む	ACLを書き込む、あるいは基本的なアクセスモードを変更するアクセス権。
(o)	所有者を変更	所有者を変更するアクセス権。
	継承	
(f)	ファイルに適用	ディレクトリ内に新しく作成されるすべてのファイルに継承します。
(d)	ディレクトリに適用	ディレクトリ内に新しく作成されるすべてのディレクトリに継承します。
(i)	自己に適用しない	現在の ACE は現在のディレクトリには適用されませんが、子には適用されます。このフラグを設定するには、「ファイルに適用」または「ディレクトリに適用」のいずれかを設定する必要があります。
(n)	過去の子に適用しない	現在の ACE は、ツリーの 1 レベルのみ、つまり直接の子のみに継承されます。このフラグを設定するには、「ファイルに適用」または「ディレクトリに適用」のいずれかを設定する必要があります。

シェアの作成時に「Windows のデフォルトのアクセス権を使用」オプションを使用した場合は、次の 3 つのエントリを含む ACL がシェアのルートディレクトリに対して作成されます。

表 12-21 シェアのルートディレクトリのエントリ

タイプ	アクション	アクセス
所有者	許可	完全な制御
グループ	許可	読み取りと実行
全員	許可	読み取りと実行

## シェア - スナップショット

スナップショットとは、特定の時点におけるファイルシステムの読み取り専用コピーです。スナップショットおよびその動作の詳細は、[294 ページの「概念」](#)のページを参照してください。

## シェア - スナップショットのプロパティー

### .zfs/snapshot 表示

ファイルシステムのスナップショットはファイルシステムのルートの `.zfs/snapshot` にあり、データプロトコルを介してアクセスできます。このディレクトリにはファイルシステム上のすべてのスナップショットのリストが含まれ、これらのスナップショットには、通常のファイルシステムデータと同じように（読み取り専用モードで）アクセスできます。デフォルトでは、ディレクトリの内容を一覧表示するときに「.zfs」ディレクトリは非表示になっていますが、明示的に検索すればアクセスできます。これにより、バックアップソフトウェアが新規データだけでなく誤ってスナップショットもバックアップすることを防いでいます。

表 12-22 スナップショットの値

BUI の値	CLI の値	説明
非表示	hidden	ファイルシステムのルートのディレクトリの内容を一覧表示するときに <code>.zfs</code> ディレクトリは表示されません。これはデフォルトです。
表示	visible	この <code>.zfs</code> ディレクトリは、ファイルシステムのほかのディレクトリと同様に表示されます。

## 定期スナップショットラベル

このオプションプロパティは、ユーザー定義のラベルを各定期スナップショットに追加し、デフォルトは空白です。ラベルは、個別のシェアに対して設定するか、プロジェクトに対して設定してそのシェアによって継承するか、のいずれか一方だけで設定できます。スナップショットラベルは、スナップショットが取られたプロジェクトやシェアを識別するのに役立ちます。たとえば、「project1:share1」では、project1 内の share1 で取られた定期スナップショットを示すことができます。ラベルには 35 文字までの英数字を使用でき、特殊文字「\_-.:」を含めることができます。

## BUI を使用したスナップショットの一覧表示

「スナップショット」タブには、シェアのアクティブなスナップショットのリストが表示されます。このリストは 2 つのタブに分かれており、「スナップショット」タブはスナップショットの参照と管理に使用されます。「スケジュール」タブは、自動スナップショットのスケジュールを管理します。「スナップショット」タブでは、すべてのスナップショットの表示、手動スナップショットのみの表示、または定期スナップショットのみの表示を選択できます。スナップショットごとに、次のフィールドが表示されます。


フィールド	説明
名前	スナップショットの名前です。スナップショットには、手動と自動の 2 つのタイプがあります。
	手動スナップショット: 「名前」はスナップショットの作成時に指定した名前です。手動スナップショットの名前を変更するには、名前をクリックして新しい値を入力します。
	自動スナップショット: 3 つのタイプがあり、名前を変更することはできません。
	- .auto: カスタム保持ポリシーを使ってユーザーが構成した定期スナップショット ( <a href="#">346 ページの「定期スナップショット」</a> を参照)。
	- .ndmp: NDMP バックアップで使用され、自動的に取り除かれます。
	- .rr: リモートレプリケーションで使用され、自動的に取り除かれます。
作成	スナップショットが生成された日時。
一意	スナップショットで使用されている一意の領域の量。スナップショットは、最初はファイルシステムまたは

フィールド	説明
	LUN 自体と同じブロックを参照しています。アクティブなファイルシステムが変化すると、アクティブなシェア内で変更されているブロックが、1 つ以上のスナップショットで保持される場合があります。あるブロックが複数のスナップショットに含まれている場合、そのブロックはシェアのスナップショットの使用量に計上されますが、特定のスナップショットの一意的領域としては示されません。一意的領域は、ある特定のスナップショットだけが保持しているブロックであり、そのスナップショットが破棄された場合に解放される領域の量を表します。
合計	スナップショットで参照されている領域の合計量。これは、スナップショットが作成された時点のファイルシステムのサイズを表します。データブロックは書き換えられるため、どのスナップショットも理論上はこの合計サイズに等しい量の領域を使用する可能性があります。
クローン	スナップショットの <a href="#">294 ページ</a> の「クローン」の数を示します。クローンの数が 0 でないスナップショットの行にマウスを置くと、「表示...」リンクが表示されます。このリンクをクリックするとダイアログボックスが開き、すべてのクローンのリストが表示されます。


## BUI を使用した手動スナップショット

スナップショットには、プロジェクトレベルとシェア/LUN レベルの 2 つのタイプがあります。

### ▼ プロジェクトレベルのスナップショットの作成

1. スナップショットを作成するプロジェクトを開きます。
2. 「スナップショット」タブをクリックします。
3.  アイコンをクリックします。スナップショットのリストが表示されます。
4. ダイアログボックスで、スナップショットの名前を入力します。
5. スナップショットを作成するには、「適用」をクリックします。

## ▼ シェア/LUN レベルのスナップショットの作成


1. スナップショットを作成するシェア/LUN を開きます。
2. 「スナップショット」タブをクリックします。
3.  アイコンをクリックします。スナップショットのリストが表示されます。
4. ダイアログボックスで、スナップショットの名前を入力します。
5. スナップショットを作成するには、「適用」をクリックします。

作成できるスナップショットの数に制限はありませんが、各スナップショットはメモリーを消費するため、多数のスナップショットを作成するとシステムが低速になる場合があります。システム全体での実用的なスナップショット数制限はシステム構成に依存しますが、十万を超える数にしてください。


## ▼ スナップショットの名前の変更 (BUI)

1. スナップショットの名前を変更するには、アクティブなスナップショットのリストでその名前をクリックします。これがテキスト入力ボックスに変わります。
2. テキストを入力して名前を変更したあとで、Return キーを押すかフォーカスを移動すると、変更が確定されます。

## ▼ スナップショットの破棄 (BUI)

1. スナップショットを破棄するには、そのスナップショットの行の上で  アイコンをクリックします。
2. スナップショットを破棄するには、そのクローンおよび子孫をすべて破棄する必要があります。この場合、影響を受けるクローンのリストが表示されます。


## ▼ スナップショットへのロールバック (BUI)

1. ファイルシステムをロールバックするには、目的のスナップショットの  アイコンをクリックします。

2. 確認ダイアログが開き、スナップショットのクローン、より新しいスナップショット、またはそれらの子孫が存在する場合は表示され、このプロセスの一環として破棄されることが通知されます。

ファイルシステムのスナップショットディレクトリにあるデータにアクセスできるほか、スナップショットを使用すると、ファイルシステムまたは LUN の以前のインスタンスにロールバックすることもできます。この操作では、そのスナップショットよりも新しいスナップショットおよびそのクローンをすべて破棄する必要があります。この操作により、シェアの内容は、スナップショットが作成された時点の内容に戻ります。シェアのプロパティの設定が影響を受けることはありませんが、ファイルシステムのルートディレクトリに対するアクセスに変更を加えていた場合、ファイルシステムのデータの一部なので、それらの変更は失われます。

## ▼ スナップショットのクローニング (BUI)

- クローンを作成するには、目的のスナップショットの  アイコンをクリックします。次の値の入力を求めるダイアログが表示されます。
  - 「プロジェクト」- 宛先プロジェクト。デフォルトでは、クローンは現在のプロジェクト内に作成されますが、別のプロジェクト内に作成することも、プロジェクト間で移動することもできます。
  - 「名前」- クローンの名前を入力します。
  - 「マウントポイント」- この値を使用するには、ロックアイコンをクリックします。クローンのマウントポイントを設定します。シェアで同じマウントポイントを保存することはできないため、「その他のローカル設定を保持」が設定されている場合はクローンに別のマウントポイントを指定する必要があります。
  - 「リソース名」- この値を使用するには、ロックアイコンをクリックします。クローンに使用するリソースを入力します。
  - 「その他のローカル設定を保持」- デフォルトでは、現在継承されるすべてのファイルシステムプロパティが、宛先プロジェクトからクローンに継承されます。ローカル設定は常に保持されます。このプロパティを設定すると、継承されるプロパティはすべて、新しいクローンのローカル設定として保持されます。

294 ページの「クローン」とは、スナップショットの書き込み可能コピーであり、ほかのシェアと同様に管理されます。ファイルシステムのスナップショットと同様に、最初は追加の領域を消費しません。クローンのデータが変更されるにつれ、消費する領域が増加します。元のスナップショットを破棄するには、クローンも破棄する必要があります。定期スナップショットのクローンは安全に作成できます。定期スナップショットは、クローンがなければ破棄されるような場合でも、クローンがあれば無視されます。

---

## BUI を使用した定期スナップショット


手動スナップショットに加え、次の表に従って自動スナップショットを構成することもできます。これらのスナップショットには「.auto-<タイムスタンプ>」という名前が付けられ、30 分ごと、毎時、毎日、毎週、または毎月のスケジュールで実行できます。スケジュールは、間隔と保持ポリシーから成るリストです。

時間は、ローカル (クライアントブラウザ) のタイムゾーンで表示されます。ただし、時間は UTC 形式で保存および実行され、サマータイムなどの慣習は考慮されません。たとえば、PST 午前 10 時 (UTC-8) にスケジュールされたスナップショットは、UTC 18:00 に保存および実行されます。

自動スナップショットは、プロジェクトまたはシェアのどちらか一方だけに設定できます。それ以外の場合、スケジュールや保持ポリシーの重複によって、両方のスケジュールを保証することは不可能になります。間隔を削除した場合やその保持ポリシーを変更した場合、新しいスケジュールに当てはまらない自動スナップショットはすべて、ただちに破棄されます。クローンが存在する自動スナップショットは無視されます。

以前のバージョンのソフトウェアでは、1 分ごとの自動スナップショットも可能でした。これは通常はあまり役に立たず、システムに過剰な負荷をかけることが判明しました。ユーザーがシステムに過剰な負荷をかけることを防ぐため、この機能は 2010.Q3 リリースで削除されました。現在は、スナップショットは 30 分以上の間隔でのみ指定できます。ソフトウェアをロールバックした場合、既存の 1 分ごとのスナップショットは保持され、古いインスタンスは既存のスケジュールに従って期限切れになりますが、新しいスナップショットが作成されることはありません。この間隔が指定されたシェアまたはプロジェクトが見つかった場合は、警告が通知されます。



新しい間隔を追加するには、「スケジュール」タブを表示して  アイコンをクリックします。各間隔には次のプロパティがあります。

プロパティ	説明
頻度	「30 分」、「時間」、「日」、「週」、「月」のいずれか。これは、スナップショットを作成する頻度を指定します。
オフセット	これは、頻度の中のオフセットを指定します。たとえば、1 時間ごとにスナップショットを作成するように選択した場合、正時からのオフセットを分単位で明示的に指定できます。日次スナップショットの場合は、オフセットを時間と分で指定でき、週次および月次スナッ

プロパティ	説明
最大保持時間	スナップショットの保持ポリシーを制御します。自動スナップショットは、無期限に保持することも (ただし、30 分ごとおよび毎時のスナップショットには、それぞれ 48 と 24 の上限がある)、特定の数に制限することもできます。この制限により、その間隔の自動スナップショットは、保持ポリシーよりも古くなると削除されます。この制限は、実際には絶対数ではなく、作成時間によって強制されます。したがって、毎時のスナップショットを設定している場合は、ZFSSA が 1 日停止したあとで復旧すると、毎時のスナップショットはすべて削除されます。複数の間隔に該当するスナップショットは、どの間隔でも保持するように指定されていない場合にかぎり、破棄されます。

## CLI を使用した手動スナップショット

シェアのスナップショットにアクセスするには、シェアおよびスナップショットのコンテキストに移動します。

```
clownfish:> shares select default select builds
clownfish:shares default/builds> snapshots
clownfish:shares default/builds snapshots>
```

### スナップショットの一覧表示 (CLI)

スナップショットは、標準の CLI コマンドを使用して一覧表示できます。

```
clownfish:shares default/builds snapshots> list
today
yesterday
clownfish:shares default/builds snapshots>
```

### 手動スナップショットの作成 (CLI)

手動プロジェクトレベルのスナップショットを作成するには、プロジェクトおよびスナップショットノードに移動して、snapshot コマンドを使用します。

```
clownfish:cd /
clownfish:shares select myproject snapshots
clownfish:shares myproject snapshots> snapshot cob_monday
```



---

個別のシェアの手動シェアレベルスナップショットを作成するには、そのシェアに移動し、そこで `snapshot` コマンドを使用します。

```
clownfish:cd /
clownfish:shares select myproject select share1 snapshots
clownfish:snapshot lunchtime
```

## スナップショットの名前の変更 (CLI)

スナップショットの名前を変更するには、`rename` コマンドを使用します。

```
clownfish:shares default/builds snapshots> rename test test2
clownfish:shares default/builds snapshots>
```

## スナップショットの破棄 (CLI)

スナップショットを破棄するには、`destroy` コマンドを使用します。

```
clownfish:shares default/builds snapshots> select test2
clownfish:shares default/builds@test2> destroy
This will destroy this snapshot. Are you sure? (Y/N)
clownfish:shares default/builds snapshots>
```

個別のスナップショットを選択せずに、シェアコンテキストから `destroy` コマンドを使用することもできます。

```
clownfish:shares default/builds snapshots> destroy test2
This will destroy this snapshot. Are you sure? (Y/N)
clownfish:shares default/builds snapshots>
```

## スナップショットへのロールバック (CLI)

スナップショットにロールバックするには、目的のスナップショットを選択し、`rollback` コマンドを実行します。

```
clownfish:shares default/builds snapshots> select today
clownfish:shares default/builds@today> rollback
Rolling back will revert data to snapshot, destroying newer data. Active
initiators will be disconnected.

Continue? (Y/N)
clownfish:shares default/builds@today>
```

---

## スナップショットのクローニング (CLI)

スナップショットのクローンを作成するには、`clone` コマンドを使用します。このコマンドを実行すると、シェアの作成に使用されたものと同じ未確定のシェアコンテキストに移動します。ここで、必要に応じてプロパティを調整してから、変更を確定してクローンを作成できます。

```
clownfish:shares default/builds snapshots> select today
clownfish:shares default/builds@today> clone testbed
clownfish:shares default/testbed (uncommitted clone)> get
  aclinherit = restricted (inherited)
  aclmode = discard (inherited)
  atime = true (inherited)
  checksum = Fletcher4 (inherited)
  compression = off (inherited)
  copies = 1 (inherited)
  mountpoint = /export/testbed (inherited)
  quota = 0 (default)
  readonly = false (inherited)
  recordsize = 128K (inherited)
  reservation = 0 (default)
  secondarycache = all (inherited)
  nbmand = false (inherited)
  sharesmb = off (inherited)
  sharenfs = on (inherited)
  snapdir = hidden (inherited)
  vscan = false (inherited)
  sharedav = off (inherited)
  shareftp = off (inherited)
  root_group = other (default)
  root_permissions = 777 (default)
  root_user = nobody (default)
  quota_snap = true (default)
  reservation_snap = true (default)
clownfish:shares default/testbed (uncommitted clone)> set quota=10G
  quota = 10G (uncommitted)
clownfish:shares default/testbed (uncommitted clone)> commit
clownfish:shares default/builds@today>
```

このコマンドには、クローンを作成するプロジェクトをオプションの第 1 引数として指定することもできます。デフォルトでは、クローンの作成元のシェアと同じプロジェクトにクローンが作成されます。

## CLI を使用した依存クローンの一覧表示

特定のスナップショット (依存クローン) から作成されたすべてのクローンを一覧表示するには、そのスナップショットに移動し、`list clones` コマンドを使用します。

```
clonefish:shares default/builds> snapshots
clonefish:shares default/builds snapshots> select today
clonefish:shares default/builds@today> list clones
```

---

Clones: 2 total

```
PROJECT      SHARE
default      testbed
default      production
clonefish:shares default/builds@today>
```

結果には、クローン名と、そのクローンが存在するプロジェクトが表示されます。

## CLI を使用した定期スナップショット

自動定期スナップショットは、個別シェアのプロジェクトレベルで、スナップショットコンテキストから `automatic` コマンドを使用すると構成できます。このコンテキストに移動したあと、`create` コマンドと `destroy` コマンドを使用すると、間隔の追加と削除を行うことができます。各間隔には、BUI ビューでの頻度、オフセット、および保持するスナップショット数に対応する、一連のプロパティがあります。スケジュールは UTC 形式で維持されます。

```
clonefish:shares default/builds snapshots> automatic
clonefish:shares default/builds snapshots automatic> create
clonefish:shares default/builds snapshots automatic (uncommitted)> set frequency=day
    frequency = day (uncommitted)
clonefish:shares default/builds snapshots automatic (uncommitted)> set hour=14
    hour = 14 (uncommitted)
clonefish:shares default/builds snapshots automatic (uncommitted)> set minute=30
    minute = 30 (uncommitted)
clonefish:shares default/builds snapshots automatic (uncommitted)> set keep=7
    keep = 7 (uncommitted)
clonefish:shares default/builds snapshots automatic (uncommitted)> get
    frequency = day (uncommitted)
    day = (unset)
    hour = 14 (uncommitted)
    minute = 30 (uncommitted)
    keep = 7 (uncommitted)
clonefish:shares default/builds snapshots automatic (uncommitted)> commit
clonefish:shares default/builds snapshots automatic> list
NAME          FREQUENCY    DAY          HH:MM KEEP
automatic-000  day          -            14:30  7
clonefish:shares default/builds snapshots automatic> done
clonefish:shares default/builds snapshots>
```

## CLI を使用した定期スナップショットラベルの設定

BUI では、プロジェクトまたはシェアに対して「定期スナップショットラベル」プロパティを設定できます。同様に、CLI では、最初にプロジェクトまたはシェアコンテキストに移動することで、ラベルを設定できます。定期スナップショットラベルを作成するには、`set snaplabel` コマンドを使用します。

```
clonefish:shares project1/share1> set snaplabel=project1:share1
```

## プロジェクト

シェア、ファイルシステム、および LUN はプロジェクトにまとめることができます。プロジェクトは、シェアを管理するための共通の管理制御点を定義します。プロジェクト内のシェアは共通の設定をシェアでき、割り当て制限はシェアレベルに加え、プロジェクトレベルでも強制できます。プロジェクトはまた、論理的に関連するシェアをグループ化して、1 か所から共通の属性 (累積された領域など) にアクセスできるようにするためだけでも使用できます。

デフォルトでは、ZFSSA はストレージプールが最初に構成されるときにデフォルトプロジェクトを 1 つ作成します。このデフォルトプロジェクト内にすべてのシェアを作成できますが、適度な大きさの環境に対しては、組織上の目的のためだけであっても、追加のプロジェクトを作成することを強くお勧めします。

## BUI を使用したプロジェクトの操作

プロジェクトの UI には、「シェア」->「プロジェクト」からアクセスします。これにより、システム上にあるすべてのプロジェクトのリストが表示されますが、プロジェクトパネルを使用するか、またはプロジェクト内でシェアの編集中にプロジェクト名をクリックするとプロジェクトを選択できます。

## プロジェクトのフィールド



プロジェクトビューに移動すると、システム上のプロジェクトのリストが表示されます。または、シェアの画面に移動し、プロジェクトパネルを開いてプロジェクトのショートカットを表示することもできます。このパネルは、多数のプロジェクトに対応して拡張することはできず、完全なプロジェクトリストに代わるものではありません。プロジェクトごとに、次のフィールドが表示されます。

表 12-23 プロジェクトのフィールド

フィールド	説明
名前	シェアの名前。シェア名は編集可能なテキストフィールドです。名前をクリックすると、プロジェクトの新しい名前を入力できます。Return キーを押すか、名前からフォーカスを移動すると、変更が確定されます。シェアの名前を変更するには、アクティブクライアントの接続を切断する必要があるため、このアクションの確認を求められます。
サイズ	プロジェクト内のすべてのシェアと未使用予約の合計サイズ。

各プロジェクトでは、次のツールを使用できます。

表 12-24 プロジェクトのアイコン

アイコン	説明
	個別のプロジェクトを編集します (行をダブルクリックする方法でも利用可能)。
	プロジェクトを破棄します。このアクションによってシェアのデータがすべて破棄され、元に戻すことはできないため、このアクションの確認を求められます。

## プロジェクトの編集

プロジェクトを編集するには、鉛筆アイコンをクリックするか、プロジェクトリストで行をダブルクリックするか、プロジェクトパネルで名前をクリックします。これでプロジェクトが選択され、いくつかの異なるタブが表示されます。これらのタブから選択して、プロジェクトのプロパティを編集できます。

プロジェクトの名前は、プロジェクトパネルの右側の左上隅に表示されます。プロジェクトの名前を変更するには、プロジェクト名をクリックして新しいテキストを入力する方法もあります。プロジェクトのアクティブクライアントの接続を切断する必要があるため、このアクションの確認を求められます。

## 使用状況統計

ビューの左側に (展開された場合はプロジェクトパネルの下に)、現在の領域の使用状況統計を示す表があります。値が 0 のプロパティがあれば、そのプロパティは表から除外されます。これらのプロパティの大部分はプロジェクトとシェアの間で同じものですが、プロジェクトのみで意味を持つ統計もいくつかあります。

- 「使用可能な容量」- 309 ページの「[「シェア」>「シェア」](#)」を参照してください。
- 「参照されるデータ」- プロジェクトに含まれているすべてのシェアの全参照データの合計に、少量のプロジェクトオーバーヘッドを加えた値。シェアの参照データを計算する方法の詳細は、309 ページの「[「シェア」>「シェア」](#)」を参照してください。
- 「スナップショットデータ」- すべてのシェアの全スナップショットデータの合計に、プロジェクトのスナップショットのオーバーヘッドを加えた値。シェアのスナップショットデータを計算する方法の詳細は、309 ページの「[「シェア」>「シェア」](#)」を参照してください。
- 「未使用予約」- プロジェクトの未使用予約。この値には、プロジェクトレベルの予約で現在使用されていないデータだけが含まれます。プロジェクトに含まれているシェアの未使用予約は、この値に含まれません。


- 「シェアの未使用予約」- すべてのシェアの未使用予約の合計。シェアの未使用予約を計算する方法の詳細は、[309 ページの「「シェア」>「シェア」](#)を参照してください。
- 「合計容量」- 参照データ、スナップショットデータ、未使用予約、およびシェアの未使用予約の合計。

## 静的プロパティ

シェアビューの左側には、特定のプロジェクトの編集集中に静的プロパティも表示されません。これらのプロパティは読み取り専用で、変更することはできません。

- 「圧縮率」- 詳細は、[309 ページの「「シェア」>「シェア」](#)を参照してください。

## ▼ プロジェクトの作成

1. プロジェクトを作成するには、プロジェクトのリストを表示し、 ボタンをクリックします。
2. または、プロジェクトパネルで「追加...」ボタンをクリックすると、同じダイアログが表示されます。プロジェクト名を入力し、「適用」をクリックしてプロジェクトを作成します。

## CLI を使用したプロジェクトの操作

プロジェクトの CLI は shares の下にあります

## ナビゲーション

プロジェクトを選択するには、select コマンドを使用します。

```
clownfish:> shares
clownfish:shares> select default
clownfish:shares default> get
    aclinherit = restricted
    aclmode = discard
    atime = true
    checksum = fletcher4
    compression = off
    compressratio = 100
    copies = 1
    creation = Thu Oct 23 2009 17:30:55 GMT+0000 (UTC)
    mountpoint = /export
    quota = 0
    readonly = false
    recordsize = 128K
```

```
reservation = 0
secondarycache = all
  nbmand = false
  sharesmb = off
  sharenfs = on
  snapdir = hidden
  snaplabel = project1:share1
  vscan = false
  sharedav = off
  shareftp = off
default_group = other
default_permissions = 700
default_sparse = false
default_user = nobody
default_volblocksize = 8K
default_volsize = 0
  space_data = 43.9K
space_unused_res = 0
space_unused_res_shares = 0
  space_snapshots = 0
space_available = 12.0T
space_total = 43.9K
clownfish:shares default>
```

## プロジェクトの操作

プロジェクトを作成するには、`project` コマンドを使用します。必要に応じてプロパティを変更してから、変更を確定します。

```
clownfish:shares> project home
clownfish:shares home (uncommitted)> get
  mountpoint = /export (default)
  quota = 0 (default)
  reservation = 0 (default)
  sharesmb = off (default)
  sharenfs = on (default)
  sharedav = off (default)
  shareftp = off (default)
  default_group = other (default)
  default_permissions = 700 (default)
  default_sparse = true (default)
  default_user = nobody (default)
  default_volblocksize = 8K (default)
  default_volsize = 0 (default)
  aclinherit = (default)
  aclmode = (default)
  atime = (default)
  checksum = (default)
  compression = (default)
  copies = (default)
  readonly = (default)
  recordsize = (default)
  secondarycache = (default)
  nbmand = (default)
  snapdir = (default)
  snaplabel = project1:share1
  vscan = (default)
  custom:contact = (default)
```

```

custom:department = (default)
clownfish:shares home (uncommitted)> set sharenfs=off
sharenfs = off (uncommitted)
clownfish:shares home (uncommitted)> commit
clownfish:shares>

```

プロジェクトを破棄するには、`destroy` コマンドを使用します。

```

clownfish:shares> destroy home
This will destroy all data in "home"! Are you sure? (Y/N)
clownfish:shares>

```

このコマンドは、プロジェクトを選択したあとでプロジェクトコンテキスト内から実行することもできます。

プロジェクトの名前を変更するには、`rename` コマンドを使用します。

```

clownfish:shares> rename default home
clownfish:shares>

```

## クラスタ内のプールの選択

アクティブ/アクティブクラスタ構成では、フェイルオーバー中に 1 つのノードが両方のプールを管理する場合があります。この場合、CLI コンテキストでは、現在のプールが括弧に囲まれて表示されます。プールを変更するには、最上位のシェアコンテキストから `set` コマンドを使用します。

```

clownfish:shares (pool-0)> set pool=pool-1
clownfish:shares (pool-1)>

```

プールコンテキストを選択したあとは、標準の CLI インタフェースを使用して、そのプール内でプロジェクトとシェアを管理します。

## プロジェクトのプロパティ

CLI で使用可能なプロパティおよび相当する BUI プロパティは次のとおりです。プロパティは、標準の CLI コマンド `get` および `set` を使用して設定できます。また、`unset` コマンドを使用して、プロパティを親プロジェクトから継承することもできます。

CLI 名	294 ページの「タイプ」	BUI 名	BUI の場所
<code>aclinherit</code>	継承	366 ページの「プロジェクトのアクセス」	アクセス
<code>aclmode</code>	継承	366 ページの「プロジェクトのアクセス」	アクセス



CLI 名	294 ページの「タイプ」	BUI 名	BUI の場所
atime	継承	363 ページの「プロジェクト - 一般」	一般
checksum	継承	363 ページの「プロジェクト - 一般」	一般
compression	継承	363 ページの「プロジェクト - 一般」	一般
compressratio	読み取り専用	356 ページの「プロジェクト」	静的
copies	継承	363 ページの「プロジェクト - 一般」	一般
creation	読み取り専用	-	-
dedup	継承	363 ページの「プロジェクト - 一般」	一般
default_group	作成のデフォルト	363 ページの「プロジェクト - 一般」	一般
default_permissions	作成のデフォルト	363 ページの「プロジェクト - 一般」	一般
default_sparse	作成のデフォルト	363 ページの「プロジェクト - 一般」	一般
default_user	作成のデフォルト	363 ページの「プロジェクト - 一般」	一般
default_volblocksize	作成のデフォルト	363 ページの「プロジェクト - 一般」	一般
default_volsize	作成のデフォルト	363 ページの「プロジェクト - 一般」	一般
mountpoint	継承	363 ページの「プロジェクト - 一般」	一般
nbmand	継承	363 ページの「プロジェクト - 一般」	一般
quota	スペース管理	363 ページの「プロジェクト - 一般」	一般
readonly	継承	363 ページの「プロジェクト - 一般」	一般
recordsize	継承	363 ページの「プロジェクト - 一般」	一般

CLI 名	294 ページの「タイプ」	BUI 名	BUI の場所
reservation	スペース管理	363 ページの「プロジェクト - 一般」	一般
secondary cache	継承	363 ページの「プロジェクト - 一般」	一般
sharedav	継承	365 ページの「プロジェクトのprotocols」	protocols
shareftp	継承	365 ページの「プロジェクトのprotocols」	protocols
share nfs	継承	365 ページの「プロジェクトのprotocols」	protocols
sharesmb	継承	365 ページの「プロジェクトのprotocols」	protocols
snapdir	継承	366 ページの「プロジェクトのスナップショット」	スナップショット
snaplabel	継承	366 ページの「プロジェクトのスナップショット」	スナップショット
space_available	読み取り専用	356 ページの「プロジェクト」	使用状況
space_data	読み取り専用	356 ページの「プロジェクト」	使用状況
space_snapshots	読み取り専用	356 ページの「プロジェクト」	使用状況
space_total	読み取り専用	356 ページの「プロジェクト」	使用状況
space_unused_res	読み取り専用	356 ページの「プロジェクト」	使用状況
space_unused_res_shares	読み取り専用	356 ページの「プロジェクト」	使用状況
vscan	継承	363 ページの「プロジェクト - 一般」	一般

## プロジェクト - 一般

### プロジェクト - 一般のプロパティー

BUI のこのセクションでは、特定のプロトコルに依存しない、またアクセス制御やスナップショットに関連しない、プロジェクト全体の設定を制御します。CLI ではすべてのプロパティーが 1 つのリストにまとめられていますが、このセクションでは、両方のコンテキストでのプロパティーの動作について説明します。

これらのプロパティーと CLI がどのように対応しているかについては、「[プロジェクトの CLI](#)」のセクションを参照してください。

### プロジェクト - 領域の使用

ストレージプール内の領域は、すべてのシェアの間でシェアされます。ファイルシステムは必要に応じて動的に拡大または縮小できますが、シェア単位で領域制限を強制することもできます。プール化されたストレージの詳細は、[294 ページの「概念」](#)のページを参照してください。

### プロジェクト - 割り当て制限

プロジェクト内のすべてのファイルシステムおよび LUN によって消費される領域の合計量に上限を設定します。詳細は、[320 ページの「シェアのセクション」](#)を参照してください。ファイルシステムとは異なり、プロジェクトの制限はスナップショットを除外することはできず、すべてのシェアとそのスナップショットにわたって強制されます。

### プロジェクト - 予約

プロジェクト内のすべてのファイルシステムおよび LUN によって使用できる領域の最小量を保証します。詳細は、[320 ページの「シェアのセクション」](#)を参照してください。ファイルシステムとは異なり、プロジェクトの予約はスナップショットを除外することはできず、すべてのシェアとそのスナップショットにわたって強制されます。

### プロジェクト - 継承されるプロパティー

これらは、プロジェクト内のシェアによって継承できる標準プロパティーです。これらのプロパティーの動作は、シェアレベルでの動作と同じで、シェアのセクションで詳細に説明されています。

- [320 ページの「マウントポイント」](#)
- [320 ページの「読み取り専用」](#)
- [320 ページの「読み取り時のアクセス時間の更新」](#)
- [320 ページの「ブロック不可の必須ロック」](#)
- [320 ページの「データ圧縮」](#)
- [320 ページの「データ複製解除」](#)
- [320 ページの「チェックサム」](#)
- [320 ページの「キャッシュデバイスの使用状況」](#)
- [320 ページの「データベースレコードサイズ」](#)
- [320 ページの「追加レプリケーション」](#)
- [320 ページの「ウイルススキャン」](#)

## プロジェクト - カスタムプロパティ

必要に応じてカスタムプロパティを追加して、ユーザー定義のタグをプロジェクトとシェアに添付できます。詳細は、[368 ページの「スキーマ」](#)を参照してください。

## ファイルシステム作成のデフォルト値

これらの設定は、ファイルシステムの作成時にデフォルト値を入力するために使用されます。これらを変更しても、既存のファイルシステムに影響はありません。詳細は、該当するシェアのセクションを参照してください。

- [338 ページの「ユーザー」](#)
- [338 ページの「グループ」](#)
- [338 ページの「アクセス権」](#)

## LUN 作成のデフォルト値

これらの設定は、LUN の作成時にデフォルト値を入力するために使用されます。これらを変更しても、既存の LUN に影響はありません。詳細は、該当するシェアのセクションを参照してください。

- [320 ページの「ボリュームサイズ」](#)
- [320 ページの「シンプロビジョニング済み」](#)
- [309 ページの「\[シェア\]>\[シェア\]」](#)

## プロジェクトのプロトコル

各プロジェクトにはプロトコル固有のプロパティがあり、そのプロジェクト内のシェアに対してさまざまなプロトコルの動作を定義します。一般に、[329 ページの「シェア」](#)はプロトコル固有のプロパティを直接継承します。例外と特殊事例については、ここで説明します。

- 「NFS」- [202 ページの「NFS」](#) シェアプロパティは、通常の方法で継承され、[329 ページの「シェアのドキュメント」](#)で説明されています。
- SMB
  - 「リソース名」- [210 ページの「SMB」](#) クライアントがこのシェアを参照するとき使用する名前。
  - 「アクセスベースの列挙を有効化」- このオプションを有効にすると、アクセスベースの列挙が実行されます。アクセスベースの列挙では、クライアントの資格に基づいてディレクトリのエントリがフィルタリングされます。クライアントがファイルまたはディレクトリに対するアクセス権を持っていない場合、クライアントに返されるエントリのリストでそのファイルは省略されます。このオプションはデフォルトで無効になっています。

同じシステムの 2 つの [210 ページの「SMB」](#) シェアに、同じリソース名を使用することはできません。ファイルシステムがプロジェクトからリソース名を継承する場合、シェアのリソース名は次の規則に従って構築されます。
  - 「オフ」- 含まれているファイルシステムは [210 ページの「SMB」](#) でエクスポートされません。
  - 「オン」- 含まれているファイルシステムは、そのファイルシステム名をリソース名として、[210 ページの「SMB」](#) でエクスポートされます。
  - 「オフ」または「オン」以外 - ファイルシステムごとに <プロジェクトのリソース名>\_<ファイルシステム名> という形式のリソース名が構築されます。
- 「iSCSI」- [208 ページの「iSCSI」](#) プロパティは継承されません。
- 「HTTP」- [228 ページの「HTTP」](#) シェアプロパティは、通常の方法で継承され、[329 ページの「シェアのドキュメント」](#)で説明されています。
- 「FTP」- [226 ページの「FTP」](#) シェアプロパティは、通常の方法で継承され、[329 ページの「シェアのドキュメント」](#)で説明されています。
- 「SFTP」- [240 ページの「SFTP」](#) シェアプロパティは、通常の方法で継承され、[329 ページの「シェアのドキュメント」](#)で説明されています。
- 「NFS」- [202 ページの「NFS」](#) シェアプロパティは、通常の方法で継承され、[329 ページの「シェアのドキュメント」](#)で説明されています。
- SMB

- 「リソース名」- [210 ページの「SMB」](#) クライアントがこのシェアを参照するときに使用する名前。
  - 「アクセスベースの列挙を有効化」- このオプションを有効にすると、アクセスベースの列挙が実行されます。アクセスベースの列挙では、クライアントの資格に基づいてディレクトリのエントリがフィルタリングされます。クライアントがファイルまたはディレクトリに対するアクセス権を持っていない場合、クライアントに返されるエントリのリストでそのファイルは省略されます。このオプションはデフォルトで無効になっています。

同じシステムの 2 つの [210 ページの「SMB」](#) シェアに、同じリソース名を使用することはできません。ファイルシステムがプロジェクトからリソース名を継承する場合、シェアのリソース名は次の規則に従って構築されます。
  - 「オフ」- 含まれているファイルシステムは [210 ページの「SMB」](#) でエクスポートされません。
  - 「オン」- 含まれているファイルシステムは、そのファイルシステム名をリソース名として、[210 ページの「SMB」](#) でエクスポートされます。
  - 「オフ」または「オン」以外 - ファイルシステムごとに <プロジェクトのリソース名>\_<ファイルシステム名> という形式のリソース名が構築されます。
- 「iSCSI」- [208 ページの「iSCSI」](#) プロパティは継承されません。

## プロジェクトのアクセス

- 「アクセス制御」- このビューでは、[338 ページの「ACL」](#) の動作に影響する継承可能なプロパティを制御できます。
- 「継承される ACL 動作」- これらのプロパティの動作は、シェアレベルでの動作と同じです。プロパティを変更すると、プロパティを現在継承しているすべてのファイルシステムで、対応する動作が変更されます。
  - [338 ページの「モード変更時の ACL 動作」](#)
  - [338 ページの「ACL 継承動作」](#)

## プロジェクトのスナップショット

スナップショットとは、特定の時点におけるファイルシステムの読み取り専用コピーです。スナップショットおよびその動作の詳細は、[294 ページの「概念」](#)のページを参照してください。プロジェクトのスナップショットは、プロジェクト内のすべてのファイルシステムと LUN のスナップショットから成り、これらの名前はすべて同じです。シェアはスナップショットを個別に削除でき、プロジェクトのスナップショットと同じ名前のスナップショットを作成するこ

ともできます。その場合、そのスナップショットは同じ名前のプロジェクトのスナップショットの一部とみなされるため、結果として動作が不定になることがあります。

## プロジェクトのスナップショットのプロパティ

### .zfs/snapshot 表示

ファイルシステムのスナップショットはファイルシステムのルートの `.zfs/snapshot` にあり、データプロトコルを介してアクセスできます。このディレクトリにはファイルシステム上のすべてのスナップショットのリストが含まれ、これらのスナップショットには、通常のファイルシステムデータと同じように（読み取り専用モードで）アクセスできます。デフォルトでは、ディレクトリの内容を一覧表示するときに「.zfs」ディレクトリは非表示になっていますが、明示的に検索すればアクセスできます。これにより、バックアップソフトウェアが新規データだけでなく誤ってスナップショットもバックアップすることを防いでいます。

表 12-25 プロジェクトのスナップショットの値

BUI の値	CLI の値	説明
非表示	hidden	ファイルシステムのルートのディレクトリの内容を一覧表示するときに <code>.zfs</code> ディレクトリは表示されません。これはデフォルトです。
表示	visible	この <code>.zfs</code> ディレクトリは、ファイルシステムのほかのディレクトリと同様に表示されます。

### 定期スナップショットラベル

このオプションプロパティは、ユーザー定義のラベルを各定期スナップショットに追加し、デフォルトは空白です。ラベルは、個別のシェアに対して設定するか、プロジェクトに対して設定してそのシェアによって継承するか、のいずれか一方だけで設定できます。スナップショットラベルは、スナップショットが取られたプロジェクトやシェアを識別するのに役立ちます。たとえば、「project1:share1」では、project1 内の share1 で取られた定期スナップショットを示すことができます。ラベルには 35 文字までの英数字を使用でき、特殊文字「`_ - . :`」を含めることができます。

プロジェクトレベルのスナップショットは、シェアレベルのスナップショットと同じ方法で管理されます。スナップショットの詳細は、[346 ページの「シェア:スナップショット」](#)を参照してください。

プロジェクトのスナップショットでは、ロールバック操作やクローン操作はサポートされていません。スナップショットの詳細は、[346 ページの「シェア:スナップショット」](#)を参照してください。

プロジェクトのスナップショットにアクセスするには、プロジェクトに移動し、`snapshots` コマンドを実行します。

```
clownfish:> shares select default
clownfish:shares default> snapshots
clownfish:shares default snapshots>
```

ここからは、スナップショットはシェアレベルのスナップショットと同じ方法で管理されます。スナップショットの詳細は、[346 ページの「シェア:スナップショット」](#)を参照してください。

プロジェクトのスナップショットでは、ロールバック操作やクローン操作はサポートされていません。スナップショットの詳細は、[346 ページの「シェア:スナップショット」](#)を参照してください。

## スキーマ

### シェアのカスタムプロパティ

標準の組み込みプロパティに加え、すべてのシェアとプロジェクトに使用できる追加のプロパティをいくつでも構成できます。これらのプロパティには、検証のための基本型が設定され、ほかのほとんどの標準プロパティと同様に継承されます。これらの値は、どのような方法でもソフトウェアで使用されることはなく、エンドユーザーだけが使用します。プロパティのスキーマはすべてのプールにわたってシステム全体に適用され、クラスタピア間で同期されます。

### BUI でのスキーマの操作

カスタムプロパティを定義するには、「シェア」->「スキーマ」ナビゲーション項目にアクセスします。現在のスキーマがリストとして表示され、必要に応じてエントリを追加または削除できます。各プロパティには次のフィールドがあります。

表 12-26 スキーマプロパティのフィールド

フィールド	説明
名前	このプロパティの CLI 名。これには英数字と文字「.」:「_」のみを含める必要があります。



フィールド	説明
説明	このプロパティの BUI 名。これは任意の文字を含むことができ、CLI のヘルプセクションで使用されま す
TYPE	検証のための、プロパティの型。これは、次に示すい ずれかの型でなければなりません。

プロパティに有効な型は次のとおりです

表 12-27 プロパティに有効な型

BUI の型	CLI の型	説明
文字列	String	任意の文字列データ。これは検証 しないことを意味します。
整数	Integer	正または負の整数
正の整数	PositiveInteger	正の整数
ブール値	Boolean	true または false の値。これ は、BUI ではチェックボックスで 表示され、CLI では「true」または 「false」のどちらかです。
電子メールアドレス	EmailAddress	電子メールアドレス。最小限の構文 検証だけが行われます。
ホスト名または IP	Host	有効な DNS ホスト名または IP (v4 または v6) アドレス。

定義されたプロパティは、[320 ページの「一般」](#)プロパティのタブで、プロパティ表の説明に従って使用できます。プロパティは CLI 名で識別されるため、プロパティの名前を変更すると、システム上の既存の設定がすべて削除されます。削除されたあとで元の名前に戻されたプロパティは、以前の設定値を引き続き参照します。プロパティの型を変更することは可能ですが、システム上の既存のプロパティに予期しない影響を与えることがあります。既存のプロパティは、新しいプロパティの型では無効な設定であっても、現在の設定を保持します。

## ▼ BUI を使用したスキーマの構成

1. 「シェア」->「スキーマ」ビューに移動します
2. 「+」アイコンをクリックし、スキーマプロパティリストに新しいプロパティを追加します

3. プロパティの名前を入力します (「contact」)
4. プロパティの説明を入力します (「所有者連絡先」)
5. 新しいプロパティの型を選択します (「電子メールアドレス」)
6. 「適用」ボタンをクリックします
7. 既存のシェアまたはプロジェクトに移動します
8. 「カスタムプロパティ」セクションで「所有者連絡先」プロパティを変更します。

## CLI を使用したスキーマの操作

スキーマコンテキストは「shares」->「schema」にあります。

```
carp:> shares schema
carp:shares schema> show
Properties:

NAME          TYPE          DESCRIPTION
owner         EmailAddress  Owner Contact
```

各プロパティは、プロパティの名前をトークンとして使用して、スキーマコンテキストの子になっています。プロパティを作成するには、create コマンドを使用します。

```
carp:shares schema> create department
carp:shares schema department (uncommitted)> get
      type = String
      description = department
carp:shares schema department (uncommitted)> set description="Department Code"
      description = Department Code (uncommitted)
carp:shares schema department (uncommitted)> commit
carp:shares schema>
```

特定のプロパティのコンテキスト内で、標準の CLI コマンドを使用してフィールドを設定できます。

```
carp:shares schema> select owner
carp:shares schema owner> get
      type = EmailAddress
      description = Owner Contact
carp:shares schema owner> set description="Owner Contact Email"
      description = Owner Contact Email (uncommitted)
carp:shares schema owner> commit
```

カスタムプロパティを定義したあとは、ほかのプロパティと同様に、「custom:<property>」という名前アクセスできます。

```
carp:shares default> get
...
      custom:department = 123-45-6789
      custom:owner =
...
carp:shares default> set custom:owner=bob@corp
      custom:owner = bob@corp (uncommitted)
carp:shares default> commit
```

## ▼ CLI を使用したスキーマの構成

1. スキーマコンテキストに移動します (shares schema)
2. 「contact」という名前の新しいプロパティを作成します (create contact)
3. プロパティの説明を設定します (set description="Owner Contact")
4. プロパティの型を設定します (set type=EmailAddress)
5. 変更を確定します (commit)
6. 既存のシェアまたはプロジェクトに移動します
7. 「custom:contact」プロパティを設定します



# ◆◆◆ 第 13 章

## レプリケーション

---

ライセンスについて: リモートレプリケーションとクローニングは無償で評価することが許可されていますが、本稼動環境で使用するには、各機能のライセンスを個別に購入する必要があります。評価期間を過ぎたら、これらの機能のライセンスを購入するか、機能を非アクティブ化する必要があります。オラクル社は、ライセンスが遵守されているかどうかをいつでも監査する権利を保持しています。詳細は、「オラクル社のソフトウェアライセンス契約書 (SLA)」および「ハードウェアシステムと組み込みのソフトウェアオプションの権利書」を参照してください。

### レプリケーションの概要

Oracle ZFS Storage Appliance では、ソース ZFSSA から任意の数のターゲット ZFSSA に、手動で、スケジュールに従って、あるいは連続して、プロジェクトとシェアをスナップショットベースでレプリケートできます。レプリケーションにはデータとメタデータの両方が含まれます。リモートレプリケーション (または単に「レプリケーション」) は、次のユースケースに最適な汎用機能です。

- 障害回復。レプリケーションを使用すると、障害回復のために ZFSSA をミラー化できます。プライマリ ZFSSA (またはデータセンター全体) のサービスに影響を与える障害イベント時に、管理者は障害時復元サイトでサービスをアクティブ化し、最新のレプリケートされたデータを使用して引き継ぎます。プライマリサイトが復元されたら、障害回復サイトでのサービス提供中に変更されたデータを、復元されたプライマリサイトと通常のサービスに移行できます。そのようなシナリオは、災害が発生する前に完全にテストできます。
- データの分散。レプリケーションを使用してデータ (仮想マシンのイメージやメディアなど) を世界中のリモートシステムに分散させることができます。ターゲット ZFSSA のクライアントがソース ZFSSA に通常は直接到達できない場合や、そのようなセットアップでは非常に長い待機時間が発生する場合などに利用します。たとえば、このスキームをローカルキャッシュに使用して、読み取り専用データ (ドキュメントなど) の待機時間を改善できます。
- ディスクからディスクへのバックアップ。テープバックアップを実行できない環境のバックアップソリューションとしてレプリケーションを使用できます。たとえば、使用可能な帯

域幅が不十分な場合や、回復の待機時間が長くなりすぎる場合には、テープバックアップを実行できないことがあります。

- データの移行。ハードウェアのアップグレードやストレージの負荷調整を行うときに、ZFSSA 間でレプリケーションを使用してデータと構成を移行できます。この目的には、シャドウ移行も使用できます。

リモートレプリケーション機能には、いくつかの重要な特性があります。

- スナップショットベース。レプリケーションサブシステムは、各更新操作の一部としてスナップショットを取得します。完全更新の場合は、スナップショットまでのプロジェクトの内容全体が送信されます。増分更新の場合は、同じアクションで最後にレプリケーションスナップショットを作成した時点以降の変更だけが送信されます。
- ブロックレベル。各更新処理は、ファイルシステムをブロックレベルでたどり、適切なファイルシステムデータとメタデータをターゲットに送信します。
- 非同期。レプリケーションでは、スナップショットを作成してから送信するため、データは送信開始前に安定したストレージに必ず確定されている必要があります。連続レプリケーションでは、ファイルシステムの変更が実質的に連続ストリームで送信されますが、NAS クライアントと SAN クライアントに関しては非同期のままです。
- メタデータも含む。基になるレプリケーションストリームでは、ユーザーデータと ZFS メタデータの両方が直列化されます。これには、「シェア」画面で構成されるプロパティのほとんども含まれます。これらのプロパティは、最初のレプリケーション更新が完了したあとでターゲット上で変更できます。ただし、レプリケーション接続が切断されるまでは変更が有効にならないものもあります。たとえば、ソースとは異なる一連のホストに対して NFS でシェアすることが可能になります。詳細は、[387 ページの「レプリケーションパッケージの管理」](#)を参照してください。
- セキュア。ZFS Storage Appliance 間で使用されるレプリケーション制御プロトコルは、SSL でセキュリティー保護されています。オプションで、データも SSL で保護できます。ほかの ZFSSA からのレプリケーション、またはほかの ZFSSA へのレプリケーションを行うには、最初に手動で認証処理を完了する必要があります。[380 ページの「ターゲットの作成と編集」](#)を参照してください。

レプリケーションには次のような既知の制限事項があります。

- ターゲットの IP アドレスを変更するとレプリケーションができなくなる
- アクションをプール間で移動できない
- プロジェクトレベルのレプリケーションごとに I/O が最大 200 MB/秒に制限される

## レプリケーションの理解

### レプリケーションの用語

- レプリケーションピア (このコンテキストでは単にピアとも呼ぶ): レプリケーションソースまたはレプリケーションターゲットとして構成されている ZFS Storage Appliance。
- レプリケーションソース (単にソースとも呼ぶ): 別の ZFSSA ピア (ターゲット) にレプリケーションするデータを保存している ZFSSA ピア。個々の ZFSSA は、ソースにもターゲットにもなり得ますが、特定のレプリケーションアクションのコンテキストではどちらか一方になります。
- レプリケーションターゲット (単にターゲットとも呼ぶ): 別の ZFSSA ピア (ソース) からレプリケートされるデータを受け取って保存する ZFSSA ピア。この用語は、別の ZFSSA へのレプリケーションを可能にする、ZFSSA 上の構成オブジェクトを指すこともあります。
- レプリケーショングループ (単にグループとも呼ぶ): 1 単位としてレプリケートされる一連のデータセット (正確に 1 つのプロジェクトといくつかのシェア)。379 ページの「プロジェクトレベルのレプリケーションとシェアレベルのレプリケーションの比較」を参照してください。
- レプリケーションアクション (単にアクションとも呼ぶ): プロジェクトまたはシェア、ターゲット ZFSSA、およびポリシーオプション (更新の送信頻度、ワイヤ上でのデータの暗号化の有無など) を指定する、ソース ZFSSA 上の構成オブジェクト。
- パッケージ: ターゲット側でアクションに相当するもの。特定のソースから特定のアクションの一環としてレプリケートされるデータを管理する、ターゲット ZFSSA 上の構成オブジェクト。ソース ZFSSA 上の各アクションには、ターゲット ZFSSA 上の正確に 1 つのパッケージが関連付けられます。逆も同様です。どちらかのオブジェクトが失われた場合は、アクションとパッケージのペア (および完全なレプリケーション更新) を新たに作成する必要があります。
- 完全同期 (完全更新とも呼ぶ): プロジェクトとそのいくつかのシェアの内容全体を送信するレプリケーション操作。
- 増分更新: 前回の更新 (完全更新か増分更新) 以降にプロジェクトとそのシェアに発生した差分だけを送信するレプリケーション操作。

### プロジェクトレプリケーションターゲット

ソース ZFSSA からターゲットへのレプリケーションを行う前に、2 つのシステムでレプリケーションピア接続を設定する必要があります。これにより、以降の通信で ZFSSA は互いに相手をセキュアに識別できます。この接続を設定するには、管理者がソース ZFSSA 上で「構成」>「サービス」>「リモートレプリケーション」画面を使用して、新しいレプリケーション

ターゲットを作成します。新しいターゲットを作成するには、管理者は次の 3 つのフィールドを指定します。

- 名前 (ソース ZFSSA の BUI および CLI でターゲットを識別するためだけに使用される)
- ネットワークアドレスまたはホスト名 (ターゲット ZFSSA に接続するため)
- ターゲット ZFSSA の root パスワード (管理者がターゲット ZFSSA で接続を設定できるようにするため)

次に、ZFSSA は、以降の通信で互いに相手をセキュアに識別するために使用される鍵を交換します。これらの鍵は ZFSSA の構成の一部として永続的に保存され、リブート後やアップグレード後も保持されます。ZFSSA を出荷時設定にリセットした場合や再インストールした場合には、鍵は失われます。root パスワードが永続的に保存されることはないため、どちらかの ZFSSA で root パスワードが変更された場合でも、レプリケーション構成を変更する必要はありません。最初のアイデンティティ交換は (すべてのレプリケーション制御操作と同様に) SSL で保護されているため、パスワードが平文で送信されることはありません。

デフォルトでは、レプリケーションターゲット接続は双方向ではありません。管理者がソース A からターゲット B へのレプリケーションを構成した場合、B が自動的に A をターゲットとして使用できるわけではありません。ただし、システムはレプリケーションの方向を逆にする機能もサポートしています。この機能は、B に A のターゲットを自動的に作成して (存在しない場合)、B から A へのレプリケーションを可能にします。

注: レプリケーションソースが NIS または LDAP サービスを使用してユーザーまたはユーザーグループをマップしており、これらのユーザーまたはユーザーグループがソースのシェア構成 (たとえば「シェアレベル ACL」や「シェアの領域の使用」) に含まれている場合、これらのユーザーまたはユーザーグループをレプリケーションターゲットで (たとえば同じ NIS または LDAP サーバーを使用して) 使用できる必要があります。使用できない場合、レプリケーションの切断/逆向きの操作は失敗する可能性があります。

レプリケーションターゲットを構成するには、[380 ページの「プロジェクトレプリケーションの構成」](#)を参照してください。

## プロジェクトレプリケーションアクションおよびパッケージ

ターゲットは、ZFSSA がレプリケーションのためにセキュアに通信できるようにする ZFSSA 間の接続を表しますが、レプリケーションの内容、頻度、オプションなどを指定するものではありません。このため、管理者はソース ZFSSA のレプリケーションアクションを定義する必要があります。アクションはレプリケーションのための主な管理制御点であり、それぞれが次の項目を指定します。



- レプリケーショングループ (1 つのプロジェクトといくつかのシェア)
- ターゲット ZFSSA
- ターゲット ZFSSA 上のストレージプール (最初の設定時のみ使用される)
- 頻度 (手動、定期、連続のいずれか)
- ワイヤ上でデータストリームを暗号化するかどうかなどの、追加のオプション

グループは、アクションを構成するプロジェクトまたはシェアによって暗黙的に指定されます (379 ページの「プロジェクトレベルのレプリケーションとシェアレベルのレプリケーションの比較」を参照)。アクションの作成後にターゲット ZFSSA とストレージプールを変更することはできませんが、ほかのオプションはいつでも変更できます。一般に、オプションが変更されたときにレプリケーション更新が進行中である場合、新しい値は次の更新が開始された時点でのみ有効になります。

アクションは ZFSSA のレプリケーション構成の主要な単位です。各アクションは、ターゲット ZFSSA 上のパッケージに対応します。このパッケージには、アクションが構成されているソースのプロジェクトとシェアの、最後のレプリケーション更新の開始時点での正確なコピーが含まれています。管理者は、対応するアクションのプロパティを変更することによって、レプリケーション更新の頻度などのオプションを構成します。ソース ZFSSA 上でアクションを作成すると、ターゲット ZFSSA 上の指定されたストレージプールにパッケージが作成されるため、アクションが最初に作成される時に、ソースがターゲットに接続できる必要があります。

各レプリケーションアクションの最初の更新では、**完全同期** (または**完全更新**) が送信されます。つまり、アクションのプロジェクトとシェアの内容全体がターゲット ZFSSA に送信されます。この最初の同期が完了すると、以降のレプリケーション更新は**増分**になります。つまり、前回の更新以降に発生した変更だけが送信されます。(ソース上の) アクションと (ターゲット上の) パッケージは、指定されたレプリケーションスナップショットを介してどの変更がターゲットにレプリケートされたかを追跡します。一般に、アクションの完全同期が少なくとも 1 回送信され、アクションとパッケージの接続がソフトウェア障害や管理アクションによって破壊されていない限り、レプリケーション更新は**増分**になります。

アクションとパッケージは互いに結合されています。パッケージが何らかの形で破壊または破棄されると、アクションに関連するデータとスナップショットがターゲットに残っていても、アクションはレプリケーション更新を送信できなくなります。同様に、アクションが破棄されると、(同じデータとスナップショットがソースに残っていても) パッケージは新しいレプリケーション更新を受信できなくなります。BUI と CLI では、管理者がアクションとパッケージの接続を破壊するような操作を実行しようとする、警告が表示されます。誤ってあるいは明示的な管理操作でアクションとパッケージの接続を破壊したために増分更新が不可能になった場合、管理者はパッケージとアクションを切断または破棄し、ソース上で新しいアクションを作成する必要があります。

注: 管理者が明示的にリクエストしないかぎり、ZFSSA はターゲット上のデータの破棄を防ぎます。その結果、アクションの最初のレプリケーション更新で一部のデータがレプリケートされたあとで更新が失敗し、パッケージ内に不完全なデータが残った場合、ZFSSA は以前に受信したデータを上書きできないため、同じアクションを使用する以降のレプリケーション更新は失敗します。これを解決するには、管理者は既存のアクションとパッケージを破棄し、新しいアクションとパッケージを作成してレプリケーションを再度開始するようにしてください。

2010.Q1 より前のソフトウェアリリースでは、アクションとレプリカの構成は (ターゲットの構成と同様に)、プロジェクトとシェアの構成の一部としてストレージプールに保存されるのではなく、コントローラに保存されていました。その結果、出荷時構成にリセットした場合に、構成は破棄されていました。2010.Q1 以降のリリースでは、アクションとパッケージの構成は、対応するプロジェクトおよびシェアとともにストレージプールに保存されるため、出荷時構成にリセットしたあとも使用可能です。ただし、ターゲットの情報は引き続き失われ、ターゲットを失ったアクションに別のターゲットを構成することは現在のところできません。

## プロジェクトレプリケーションのストレージプール

管理者は、アクションを最初に構成するときに、レプリケートされたデータをターゲットのどのストレージプールに保存するかを選択できます。アクションを作成したあとで、アクションを保存するストレージプールを変更することはできません。アクションを作成すると、ターゲット上の指定されたストレージプールに空のパッケージが作成されます。この処理のあとは、ソースがターゲット上のストレージ構成を認識することはありません。ソースは、どのプールにアクションがレプリケートされているかを追跡することも、ターゲット上のストレージ構成の変更で更新されることもありません。

ターゲットがクラスタ化されたシステムである場合、選択されたストレージプールは、ソースがレプリケーションに使用する IP アドレスを所有しているのと同じヘッドによって所有されるプールである必要があります。これらのプールが常にアクセス可能であることを保証されるのは、ソースがその IP アドレスを使用してターゲットに接続する場合だけであるためです。これは、NAS クライアント (NFS および SMB) の構成とちょうど似ています。NAS クライアントの構成では、マウント操作でリクエストする IP アドレスとパスが同じ制約に従っている必要があります。管理者は、クラスタ内のストレージプールおよび IP アドレスの所有者を変更する操作を実行する場合、そのクラスタにレプリケーションを行なっているソースに対する影響を考慮する必要があります。現在のところ、ストレージプール間でパッケージを移動する方法はありません。

## プロジェクトレベルのレプリケーションとシェアレベルのレプリケーションの比較

この ZFSSA では、管理者はプロジェクトレベルとシェアレベルの両方でリモートレプリケーションを構成できます。「シェア」画面で構成できるほかのプロパティと同様に、各シェアはその親プロジェクトの構成を継承するか、それをオーバーライドできます。構成を継承する場合、シェアはその親プロジェクトと同じスケジュールに従って、同じオプションで、同じターゲットにレプリケートされます。それだけではなく、プロジェクトの構成を継承しているほかのシェアと同じプロジェクトレベルのスナップショットを使用して、同じストリームでレプリケートされます。複数のシェアに保存されているデータの間には整合性を必要とするアプリケーションでは、これが重要になる場合があります。構成をオーバーライドする場合、シェアはプロジェクトレベルのアクションでレプリケートされることはありませんが、プロジェクトを含む独自のシェアレベルのアクションでレプリケートされることがあります。プロジェクトのレプリケーション構成の一部をオーバーライドし、残りを継承するということはできません。

より正確には、プロジェクトとそのシェアのレプリケーション構成により、いくつかのレプリケーショングループが定義されます。各グループは、同時に作成されたスナップショットを使用して、1つのストリームでレプリケートされます。すべてのグループには、プロジェクト自体が含まれています（プロジェクトは基本的にそのプロパティだけを含む）。1つのプロジェクトレベルのグループには、親プロジェクトのレプリケーション構成を継承しているすべてのシェアが含まれています。プロジェクトの構成をオーバーライドするシェアは、プロジェクトとシェア自体のみから成る新しいグループを形成します。

たとえば、次の場合について考えます。

- プロジェクト home とシェア bill, cindi, および dave があります。
- home には、いくつかのアクションによるレプリケーションが構成されています
- home/bill と home/cindi はプロジェクトのレプリケーション構成を継承します
- home/dave はプロジェクトのレプリケーション構成をオーバーライドし、いくつかのアクションによる独自の構成を使用します

この構成によって次のレプリケーショングループが定義され、その各グループは、プロジェクトとシェアに対して同時に作成されたスナップショットを使用して、アクションごとに1つのストリームでレプリケートされます。

- 1つのプロジェクトレベルのグループ。home, home/bill, および home/cindi が含まれます。
- 1つのシェアレベルのグループ。home と home/dave が含まれます。

現在の制限事項のため、プロジェクトレベルのレプリケーションとシェアレベルのレプリケーションを同じプロジェクト内に混在させないでください。これにより、レプリケーションの方向を逆にする場合やクローンを複製する場合に、予期しない結果が生じること

を回避できます。詳細は、387 ページの「レプリケーションパッケージの管理」および 408 ページの「クローンのレプリケーション」のセクションを参照してください。


## プロジェクトレプリケーションの構成

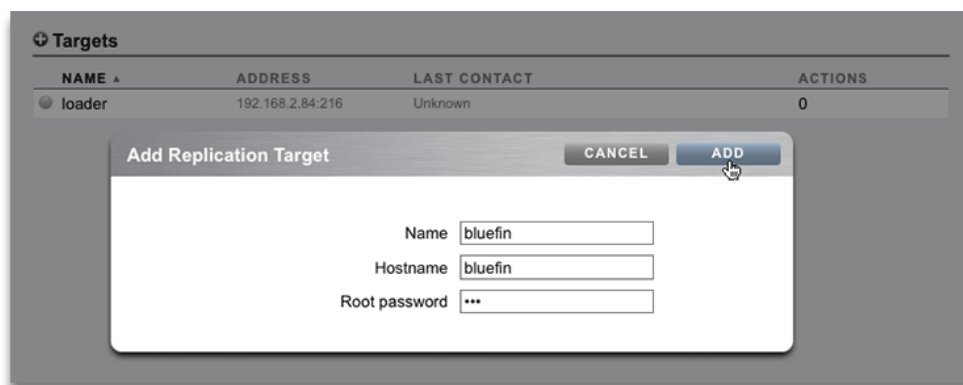
レプリケーションを構成する前に、レプリケーションのターゲット、アクション、およびパッケージに関する前述のセクションを必ず読み、理解してください。

### ターゲットの作成と編集

このセクションでは、ターゲットの作成および編集について説明します。

#### ▼ BUI でのターゲットの作成および編集

1. BUI でリモートレプリケーションターゲットを作成するには、「構成」>「サービス」>「リモートレプリケーション」>「ターゲット」に移動します。「 ターゲット」をクリックし、「名前」、「ホスト名」、および「パスワード」を構成します。
2. BUI でリモートレプリケーションターゲットを編集するには、「構成」>「サービス」>「リモートレプリケーション」>「ターゲット」に移動します。編集するターゲットで、カーソルをターゲット名の上に移動して、鉛筆アイコンをクリックし、「名前」または「ホスト名」、あるいはその両方を構成します。「ホスト名」は、以前と同じ ZFSSA に解決される必要があります (ターゲットのシリアル番号で確認)。以前に構成したのではない ZFSSA を指し示す場合は、新しいターゲットを作成して、新しい ZFSSA に対して認証を行う必要があります。



## ▼ CLI を使用したターゲットの作成および編集

1. CLI では、`targets` ノードに移動し、ターゲットの `hostname`、`root_password`、および `label` を設定または設定解除します。

```
knife:> configuration services replication targets
```

2. このコンテキストから、管理者は次の操作を実行できます。

- 新しいターゲットの追加
- 既存のターゲットで構成されているアクションの表示
- ターゲットの一意的識別子 (ラベル) またはホスト名、あるいはその両方の編集
- どのアクションでも使用されていないターゲットの破棄

3. アクションで使用されているターゲットは破棄しないようにしてください。破棄すると、アクションは破壊されたままになります。システムは、できるだけこれを防ごうとしますが、エクスポートされたストレージプールに特定のターゲットを使用しているアクションが存在しないとは保証できません。

## アクションの作成と編集

レプリケーションアクションには次のプロパティがあり、BUI と CLI では表示が多少異なっています。

図 13-1 レプリケーションアクションの追加

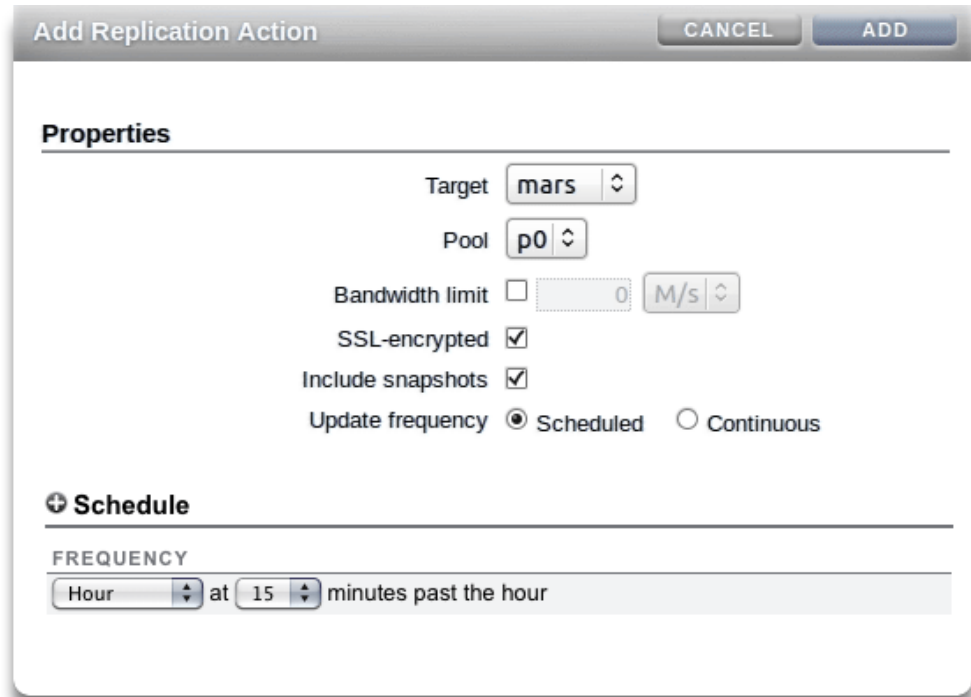


表 13-1 レプリケーションアクションの CLI プロパティ

プロパティ (CLI 名)	説明
ターゲット	レプリケーションのターゲットシステムを表す一意の識別子。このプロパティは、アクションを最初に構成するときに指定し、そのあとは変更できません。
プール	このプロジェクトのレプリケーション先となるターゲット上のストレージプール。このプロパティは、アクションを最初に構成するときに指定し、そのあとは表示されません。
モード (CLI: 連続) およびスケジュール	このアクションが連続してレプリケートされるか、手動でレプリケートされるか、定期的なレプリケートされるかを示します。詳細は、 <a href="#">386 ページの「レプリケーションモード: 「定期」または「連続」</a> を参照してください。
スナップショットを含める	レプリケーション更新に非レプリケーションスナップショットを含めるかどうか。詳細は、 <a href="#">386 ページの「レプリケーション - 中間スナップショットを含める」</a> を参照してください。

プロパティ (CLI 名)	説明
帯域幅の制限	このレプリケーション更新の最大速度を指定します (ネットワーク上で 1 秒あたりに転送されるデータ量で表す)。レプリケーション更新中にこのプロパティに行なった変更は、次回更新時まで反映されません。
送信バイト数	ターゲットに送信されたバイト数を示す読み取り専用プロパティ。
推定サイズ	レプリケートされるデータの推定サイズを示す読み取り専用プロパティ。
推定残り時間	完了までの推定残り時間を示す読み取り専用プロパティ。
平均スループット	レプリケーションの平均スループットを示す読み取り専用プロパティ。
SSL を使用	SSL を使用してワイヤ上でデータを暗号化するかどうか。この機能を使用すると、アクション単位のレプリケーションのパフォーマンスにかなり影響することがあります。
状態	アクションが現在アイドル状態か、更新を送信中か、更新を取り消し中かを示す読み取り専用プロパティ。
最終同期	更新が最後に正常に送信された時点を示す読み取り専用プロパティ。ブート以降に更新が正常に送信されていない場合、この値は不明になることがあります。
最後の試行	更新が最後に試行された時点を示す読み取り専用プロパティ。ブート以降に更新の送信が試行されていない場合、この値は不明になることがあります。
次の更新	次の試行がいつ行われるかを示す読み取り専用プロパティ。この値は、日付 (定期更新の場合)、「手動」、または「連続」になります。

## ▼ BUI でのアクションの作成および編集

1. レプリケーションターゲットを少なくとも 1 つ構成したあと、管理者はローカルプロジェクトまたはシェアに対するアクションを構成できます。BUI では、プロジェクトまたはシェアに移動して「レプリケーション」タブをクリックします。CLI では、プロジェクトまたはシェアに移動して「replication」ノードを選択します。これらのインタフェースは、プロジェクトまたはシェアで構成された既存のアクションのステータス、およびレプリケーションの進捗情報を示し、管理者が新しいアクションを作成することを可能にします。

TARGET ▲	UPDATES	STATUS
jupiter Manual	2013-7-15 17:22:04 Synced 2013-7-15 17:22:04 Attempted	Sync now
jupiter Continuous	2013-7-15 17:24:48 Synced 2013-7-15 17:25:21 Fail	86% of 3.5G @21MB/s (-00:00:24)
mars Scheduled	2013-7-15 17:23:32 Synced 2013-7-15 17:23:32 Attempted	2013-7-15 17:34:00 Next
venus Manual	2013-7-15 17:18:43 Synced 2013-7-15 17:18:43 Attempted	74% of 4.0G @24MB/s (-00:00:45)

- ターゲットへのレプリケート時に、2 行のステータス情報が表示されます。最初の行には、ターゲット名、前回同期に成功した日付と時間、および進行状況バーまたはサインポール型の進捗バー（レプリケーションが継続中の場合）が表示されます。2 番目の行には、レプリケーションタイプ（スケジュール、「手動」、または「連続」）、前回同期を試みるか同期に失敗した日付と時間、およびステータスの詳細が表示されます。レプリケーションが進行中の場合、ステータスの詳細には、完了した割合、レプリケートするデータの推定サイズ、レプリケーションの平均スループット、および推定完了時間が含まれます。レプリケーションが進行中でない場合、複製タイプに応じて、「ステータス」列にスケジュールされている次のレプリケーションまたは「今すぐ同期」のメッセージが表示されます。

## ▼ CLI でのアクションの作成および編集

- 同じ進捗情報を CLI に表示でき、進行中のレプリケーションについて `sending` という状態が表示されます。

```
otoro:shares otoro-proj-01 action-000> show
Properties:
    id = 80a96f4f-93fe-4abd-eb54-fb82e7f8c69f
    target = chutoro
    continuous = false
    include_snaps = true
    max_bandwidth = unlimited
    bytes_sent = 505M
    estimated_size = 3.0G
    estimated_time_left = 00:00:41
    average_throughput = 63MB/s
    use_ssl = false
    state = sending
    state_description = Sending update
    next_update = Sync now
    last_sync = Sun Jul 14 2013 06:04:38 GMT+0000 (UTC)
    last_try = Sun Jul 14 2013 06:04:38 GMT+0000 (UTC)
    last_result = success
```



2. 注: レプリケートするデータのサイズによっては、レプリケーションの完了までに長い時間がかかる場合があります。進捗情報を使用して、更新ステータスを判定します。初期レプリケーションは、中断しないことが重要であり、これには ZFSSA の再起動や更新の取り消しが含まれますが、これらに限定されません。さもないと、初期レプリケーション全体の再起動が必要になります。
3. `actions` という状態のレプリケーションターゲット情報を CLI に表示できます。

```
otoro:configuration services replication targets> show

Targets:
  TARGET      LABEL      ACTIONS
  target-000  oakmeal    1

otoro:configuration services replication targets> select target-000

otoro:configuration services replication target-000> show
Properties:
  address = 10.153.34.167:216
  label = oakmeal
  hostname = oakmeal-7320-167
  asn = 4913649f-7549-6d2a-866b-987ddbc4e163
  actions = 1

oakmeal-7320-167:configuration services replication target-000> actions
POOL      PROJECT    SHARE
pool1     project1   (multiple)
```

4. CLI を使用する場合は、新しく作成したレプリケーションアクションの ID を把握しておく役立ちます。ID は、あとで正しいレプリケーションアクションノードを選択する際に使用します。新しく作成したアクションの ID を表示するには、コマンド `last` を使用して、新しいレプリケーションアクションを持つノードに移動します。その後、コマンド `get id` を使用して、アクション ID を取得します。

```
otoro:> shares
otoro:shares> select p1
otoro:shares p1> replication
otoro:shares p1 replication> create
otoro:shares p1 action (uncommitted)> set target=oakmeal
      target = oakmeal (uncommitted)
otoro:shares p1 action (uncommitted)> set pool=p
      pool = p (uncommitted)
otoro:shares p1 action (uncommitted)> set use_ssl=false
      use_ssl = false (uncommitted)
otoro:shares p1 action (uncommitted)> commit
otoro:shares p1 replication> last
otoro:shares p1 action-001> get id
      id = fb1bb3fd-3361-42e1-e4a1-b06c426172fb
otoro:shares p1 action-001> done
otoro:shares p1 replication>
```

## レプリケーションモード: 「定期」または「連続」

レプリケーションアクションでは、更新をスケジュールに従って、または連続して送信するように構成できます。レプリケーション更新の処理自体はどちらの場合も同じです。このプロパティは間隔だけを制御します。


連続レプリケーションアクションはできるかぎり頻繁に更新を送信するため、ファイルシステムのすべての変更が連続したストリームでターゲットシステムに送信されることになります。変動の多い (多数のファイルの作成と破棄が短い間隔で発生する) ファイルシステムの場合は、これによって、実際に必要なデータよりはるかに大量のデータがレプリケートされることがあります。ただし、データの変化に遅れずにレプリケーションが行われるかぎり、ソースシステムでデータが失われる障害が発生した場合に失われるデータは最小限で済みます。


連続レプリケーションは非同期のままです。現在、ZFS Storage Appliance は同期レプリケーションをサポートしていません。同期レプリケーションでは、プライマリストレージシステムとセカンダリストレージシステムの両方でデータが安定したストレージに確定されるまで、データは安定したストレージに確定されたときみなされません。

## レプリケーション - 中間スナップショットを含める

「スナップショットを含める」プロパティが true の場合、レプリケーション更新には、前回のレプリケーション更新のあとで (あるいは、最初の完全更新の場合はシェアの作成以降に) 作成された非レプリケーションスナップショットが含まれます。これには、自動スナップショットと管理者によって作成されたスナップショットが含まれます。このプロパティを無効にすると、各更新でこのようなスナップショットを除外し、レプリケーションスナップショットの間の変更だけを送信できます。

## レプリケーション - 更新の送信と取り消し

定期レプリケーションまたは手動レプリケーションが構成されているターゲットに対して、管理者はレプリケーション更新をただちに送信できます。そのためには、BUI では  ボタンをクリックし、CLI では `sendupdate` コマンドを使用します。更新がアクティブに送信されている場合、これは使用できません (または機能しません)。更新を送信する前に、プロジェクト全体をレプリケーションするのに十分なディスク容量がターゲットにあることを確認してください。

更新が現在アクティブである場合、BUI では進捗バーが表示され、CLI では `sending` という状態が表示されます。更新を取り消すには、 ボタンをクリックするか、または `cancelupdate` コマンドを使用します。取り消しが完了するまでに数秒かかることがあります。

## レプリケーションパッケージの管理

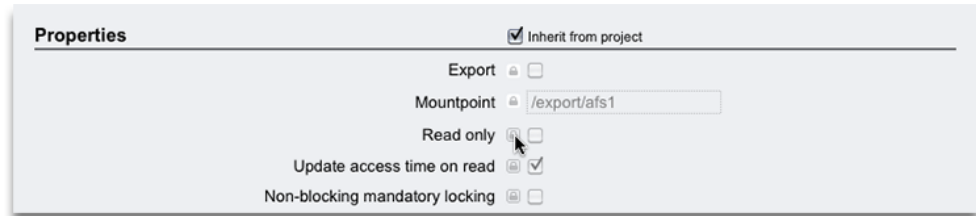
パッケージは、レプリケートされたプロジェクトとシェアのコンテナです。前述のとおり、ソース ZFSSA 上の各レプリケーションアクションは、ターゲット ZFSSA 上の 1 つのパッケージに対応します。BUI と CLI のどちらを使用しても、レプリケートされたプロジェクト、シェア、スナップショット、およびプロパティをローカルプロジェクトやシェアと同様に参照できます。ただし、レプリケートされたシェアは、ソース ZFSSA 上の対応するシェアと厳密に一致する必要があるため、レプリケーションパッケージ内では多くの管理操作が禁止されます。たとえば、プロジェクトとシェアの作成、名前の変更、破棄、スナップショットの作成と名前の変更、プロジェクトとシェアのほとんどのプロパティの変更などです。レプリケーションパッケージ内のスナップショットは、増分レプリケーションの基礎として使用されるもの以外は破棄できます。この操作はお勧めしませんが、追加の空き容量が必要になった場合に使用できます。

2009.Q3 以前のソフトウェアバージョンでは、レプリケートされたシェアのプロパティを変更することはできませんでした。2010.Q1 リリース (および関連する遅延アップグレード) を使用すると、レプリケートされたシェアのプロパティを変更するための限定的なサポートが追加され、ソース ZFSSA とターゲット ZFSSA で異なるポリシーを実装できるようになります。そのようなプロパティの変更は、レプリケーション更新後も保持されます。レプリケートされたプロジェクトとシェアのプロパティのうち、次のものだけが変更可能です。

- 予約、圧縮、コピー、複製解除、およびキャッシュ。これらのプロパティをレプリケーションターゲット上で変更することで、コスト、柔軟性、パフォーマンス、信頼性などに関してソース ZFSSA とは異なるポリシーをターゲット ZFSSA に適用できます。
- マウントポイントおよびシェアのプロパティ (たとえば sharenfs、SMB リソース名など)。これらのプロパティは、シェアが NAS クライアントにエクスポートされる方法を制御します。また、ソースとターゲット ZFSSA で異なるセキュリティポリシーまたは保護ポリシーを適用するよう変更できます。
- 自動スナップショットのポリシー。自動スナップショットのポリシーはターゲットシステム上で変更できますが、パッケージが切断されるまで変更は有効になりません。レプリケートされたプロジェクトおよびシェアでは自動スナップショットの作成や破棄は行われません。

BUI と CLI では、管理者は変更不可能なプロパティを変更できません。シェアの場合は、プロパティの継承を変更できないことを示すために、別のアイコンが使用されます。

図 13-2 レプリケーションパッケージのプロパティの管理



レプリケーションターゲットでプロパティを変更できるようにするには、2010.Q1 リリースで提供されている遅延更新をターゲットに適用する必要があります。2010.Q1 の遅延更新が適用されていないシステムでは、管理者は、レプリケーションパッケージ内のプロパティを変更できません。

現在のリリースでは、「連鎖」レプリケーション (つまり、レプリケートされたシェアを別の ZFSSA にレプリケーションする) 構成はサポートされていません。

## BUI でのレプリケーションパッケージの管理

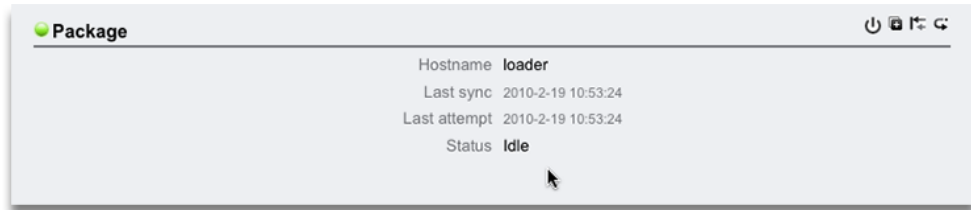
レプリケーションパッケージは、BUI ではプロジェクトとして「レプリカ」フィルタの下に表示されます。

図 13-3 「レプリカ」フィルタ



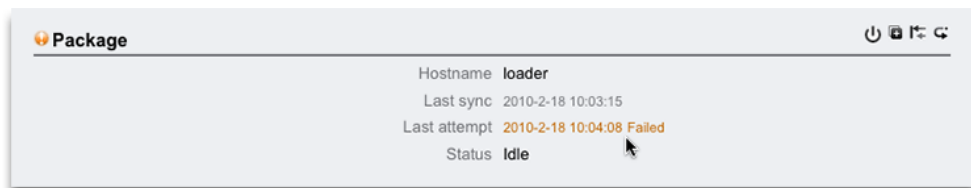
レプリケーションパッケージを編集用に選択すると、管理者には、パッケージのプロジェクトの「シェア」ビューが表示されます。前述の例外はありますが、管理者はここから、レプリケートされたシェアをローカルのシェアと同じ方法で管理できます。パッケージのプロパティ (ステータスも含む) は、「レプリケーション」タブで変更できます。

図 13-4 パッケージのプロジェクトの「シェア」ビュー



レプリケーションが失敗した場合は、左側のステータスアイコンが変化します。

図 13-5 失敗を示すステータスアイコン



パッケージは、最初のレプリケーション更新が開始したあとではじめて BUI に表示されます。最初の更新が完了したあと、しばらくはリストに表示されない場合があります。

## CLI を使用したレプリケーションパッケージの管理

レプリケーションパッケージは、CLI では `shares replication sources` の下でソース別に管理されます。管理者はまずソースを選択し、次にパッケージを選択します。パッケージレベルの操作はこのノードで実行できます。あるいは、ローカルのプロジェクトやシェアと同じ方法で、プロジェクトを選択してプロジェクトのプロパティとシェアを管理できます。ただし、前述の例外はあります。次に例を示します。

```
loader:> shares replication sources
loader:shares replication sources> show
Sources:

source-000 ayu
      PROJECT      STATE      LAST UPDATE
package-000 oldproj  idle      unknown
package-001 aproj1  receiving Sun Feb 21 2010 22:04:35 GMT+0000 (UTC)
```

```

loader:shares replication sources> select source-000
loader:shares replication source-000> select package-001
loader:shares replication source-000 package-001> show
Properties:
    enabled = true
    state = receiving
    state_description = Receiving update
    last_sync = Sun Feb 21 2010 22:04:40 GMT+0000 (UTC)
    last_try = Sun Feb 21 2010 22:04:40 GMT+0000 (UTC)

Projects:
    aproj1

loader:shares replication source-000 package-001> select aproj1
loader:shares replication source-000 package-001 aproj1> get mountpoint
    mountpoint = /export
loader:shares replication source-000 package-001 aproj1> get sharenfs
    sharenfs = on

```

configuration services replication からレプリケーションソースを表示できます。次に例を示します。

```

loader:configuration services replication> show
Properties:
    <status> = online

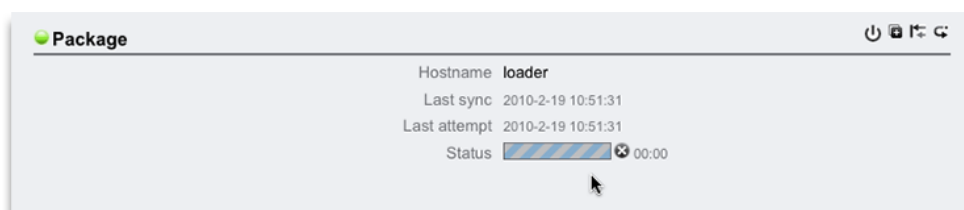
Children:
    targets => Configure replication targets
    sources => View and manage replication packages

```

## レプリケーション更新の取り消し

ターゲット上で進行中のレプリケーション更新を BUI を使用して取り消すには、レプリケーションパッケージに移動し (上記を参照)、「レプリケーション」タブをクリックします。更新が進行中の場合は、次に示すように、サインポール型の進捗バーとその横に取り消しボタン (⊗) が表示されます。

図 13-6 レプリケーションの取り消し




更新を取り消すには、このボタンをクリックします。

ターゲット上で進行中のレプリケーション更新を CLI を使用して取り消すには、レプリケーションパッケージに移動し (上記を参照)、cancelupdate コマンドを使用します。

ターゲットから更新を開始することはできません。手動更新を開始するには、管理者はソースシステムにログインする必要があります。

## パッケージの無効化


パッケージのレプリケーション更新を完全に無効にできます。その場合、進行中の更新は取り消され、ソース ZFSSA からの新しい更新は失敗します。

BUI からパッケージの有効/無効を切り替えるには、パッケージに移動し (上記を参照)、「レプリケーション」タブ、続いて  アイコンをクリックします。左側のステータスアイコンがパッケージのステータス (有効、無効、または失敗) を示して変化します。管理者がこの同じボタンまたは CLI を使用して明示的に有効にするまで、パッケージは無効のままになります。

CLI からパッケージの有効/無効を切り替えるには、パッケージに移動し (上記を参照)、enabled プロパティを変更し、変更を確定します。

## パッケージまたは個々のシェアのクローニング

レプリケートされたパッケージのクローンは、システム上のほかのプロジェクトと同様に管理できる、可変のローカルプロジェクトです。このクローンのシェアは、もっとも新しく受信されたスナップショットでレプリケートされたシェアのクローンです。このようなクローンは、シェアのスナップショットのクローンと同じ方法で、ストレージを元のスナップショットとシェアします (346 ページの「スナップショットのクローニング」を参照)。このメカニズムは、レプリケーションソースに壊滅的な問題が発生した場合のフェイルオーバーや、単にローカルで変更可能なデータを作成するために使用できます。

BUI の  ボタンまたは CLI の clone コマンドを (パッケージのコンテキストで) 使用して、もっとも新しく受信されたレプリケーションスナップショットに基づいてパッケージクローンを作成します。CLI と BUI のどちらのインターフェースでも、管理者は新しいクローンプロジェクトの名前を指定する必要があります。また、プロジェクトまたはそのシェアのマウントポイントをオーバーライドして、システム上のほかのシェアとの競合を防止できます。

2009.Q3 以前は、レプリケートされたプロジェクトのクローニングが、そのデータにアクセスするための唯一の方法であり、したがって障害回復フェイルオーバーを実装するための唯一の方法でした。2010.Q1 以降は、クローンを作成することなく、個々のファイルシステムを読み取り専用としてエクスポートできます。さらに、フェイルオーバー処理の一環として、レ

アプリケーションパッケージを書き込み可能なローカルプロジェクトに直接変換できます。その結果、これらの代替方法によって同様の機能をより簡単な操作で実現でき、クローンとその依存関係を管理する必要もなくなるため、パッケージのクローニングは不要または推奨されなくなりました。

特に、クローンが存在している間は、元のスナップショットを破棄することができません。(スナップショットが属しているシェア、プロジェクト、またはレプリケーションパッケージの破棄などの結果として) スナップショットが破棄される場合、その操作によって破棄される従属クローンがあれば、管理者に警告されます。ソース上のスナップショットがいつでも破棄される可能性もあり、そのようなスナップショットは、それ以降のレプリケーション更新の一環としてターゲット上で破棄されます。そのようなスナップショットにクローンが存在する場合は、代わりにスナップショットの名前が変更され、一意の名前 (通常は `recv-xxx`) が付けられます。

管理者は、通常の BUI および CLI インタフェースを使用して、レプリケートされたシェアのスナップショットのクローンを個別に作成することもできます。

## レプリケートされたファイルシステムのエクスポート

レプリケートされたファイルシステムを読み取り専用として NAS クライアントにエクスポートできます。この機能は、レプリケートされたデータを確認する場合や、バックアップなどの負荷がかかる操作を (ソース ZFSSA にそのような負荷をかけないために) レプリケートされたデータに対して実行する場合に使用できます。

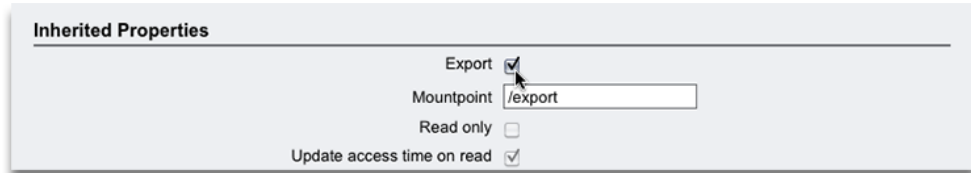
ファイルシステムの内容は、そのファイルシステムのもっとも新しく受信されたレプリケーションスナップショットと常に一致します。これはパッケージ全体のもっとも新しく受信されたスナップショットよりも新しい場合があり、同じパッケージに含まれているほかのシェアの最新スナップショットと一致しない場合があります。詳細は、[406 ページの「スナップショットとデータ整合性」](#)を参照してください。

レプリケーション更新はファイルシステムレベルで原子的に適用されます。レプリケートされたファイルを見ているクライアントには、レプリケーション更新は配下のファイルシステムの瞬時変化に見えます。最新の更新で削除されたファイルを操作しているクライアントには、エラーが表示されます。最新の更新で変更されたファイルを操作しているクライアントには、更新後の内容がただちに表示されます。

レプリケートされたファイルシステムは、デフォルトではエクスポートされません。エクスポートするには、BUI または CLI を使用してプロジェクトまたはシェアの「`exported`」プロパティを変更します。



図 13-7 継承されるプロパティ




このプロパティは、ほかのシェアプロパティと同様に継承されます。ローカルプロジェクトとシェアは常にエクスポートされるため、このプロパティは表示されません。また、レプリケーションを切断すると（パッケージがローカルプロジェクトに変換され）、パッケージのシェアがエクスポートされます。

現在、レプリケートされた LUN をエクスポートすることはできません。それらの内容をエクスポートするには、まずクローンを作成するか、レプリケーションパッケージを切断する必要があります。

## レプリケーションの切断

レプリケーション接続を切断することによって、レプリケーションパッケージを、ほかのローカルプロジェクトと同様に動作する（つまり、レプリケーションパッケージに適用される管理上の制限がない）書き込み可能なローカルプロジェクトに変換できます。この操作のあとは、このパッケージにレプリケーション更新を受信することはできなくなります。そのため、これ以降にソースから同じプロジェクトのレプリケーション更新を行うには、新しいアクションで完全更新を（新しいパッケージに）送信する必要があります。これ以降に同じアクションを使用してレプリケーション更新を行うと、対応するパッケージがターゲットに存在しないため失敗します。

このオプションが主に役立つのは、レプリケーションを使用して ZFSSA 間でデータを移行する場合など、受信したデータをレプリケートしてソースに戻す処理を伴わないシナリオです。典型的な 2 システムによる障害回復計画にはこのような処理が含まれます。

BUI からレプリケーションを切断するには、レプリケーションパッケージに移動し（上記を参照）、「レプリケーション」タブ、続いて  ボタンをクリックします。表示されるダイアログで、管理者は新しいローカルプロジェクトの名前を指定できます。

CLI からレプリケーションを切断するには、レプリケーションパッケージに移動し（上記を参照）、`sever` コマンドを使用します。このコマンドは、新しいローカルプロジェクトの名前を指定するオプションの引数を取ります。引数が指定されていない場合は、元の名前が使用されます。

ローカルのシェアはすべてエクスポートされるため、パッケージを切断すると、それまでにエクスポートされたかどうかにかかわらずパッケージ内のすべてのシェアがエクスポートされます (上記を参照)。レプリケートされたファイルシステムとシステム上のほかのファイルシステムの間でマウントポイントが競合している場合、切断操作は失敗します。切断する前に、関連するシェアのマウントポイントを再構成して、これらの競合を解決する必要があります。

## レプリケーションの方向を逆にする

レプリケーションの方向を逆にして、典型的な 2 システムによる障害回復計画をサポートできます。この操作は、前述の切断操作に似ていますが、新しいローカルプロジェクトを増分レプリケートしてソースシステムに戻すレプリケーションアクションを追加で構成します。この操作が完了した時点ではソースシステムに変更は加えられませんが、このアクションを使用して最初の更新が試行されると、ソースシステム上の元のプロジェクトがレプリケーションパッケージに変換され、そのシステムからのレプリケーション更新が最後に正常に実行された時点以降の変更がすべてロールバックされます。

この機能は、データのプライマリコピーとセカンダリコピーの読み書きステータスを変更する以外は、本番負荷のリダイレクト、IP アドレスのフェイルオーバー、障害回復フェイルオーバーに関連するその他の処理を自動的に実行するわけではありません。

(現在はターゲットとして動作している) 元のソースシステムで元のソースプロジェクトをレプリケーションパッケージに変換する一環として、現在逆向きになっているアクション/パッケージの一部としてレプリケートされたシェアが、新しいレプリケーションパッケージに移動され、アンエクスポートされます。元のプロジェクトはローカルコレクションに残りますが、アクション/パッケージがそのすべてのシェアを含んでいた場合には、空になることがあります。シェアレベルのレプリケーションを逆向きにした場合、元のプロジェクト内のほかのシェアはどれも変更されません。


ある ZFSSA から別の ZFSSA へのシェアレベルのレプリケーションを確立したあとで、ターゲット ZFSSA 上でそのレプリケーションを逆向きにすると、レプリケーションスケジュールが破棄されます。そのあと、プロジェクトレベルでレプリケーションアクションが作成されます。これには、正しいターゲット ZFSSA が設定され、スケジュールは含まれていません。

前述のとおり、この機能は通常、2 システムによる障害回復構成を実装するために使用されます。この構成では、プライマリシステムが本番データを処理し、それをセカンダリ (DR) システムにレプリケートします。このシステムは、別のデータセンターに置かれている場合が多く、プライマリサイトで障害が発生した場合に本番トラフィックを引き継ぐために待機しています。プライマリサイトで障害が発生した場合は、本番トラフィックをセカンダリサイトにリダイレクトし、セカンダリサイトのコピーを書き込み可能にして「プライマリ」にする必要があります。プライマリサイトが修復されたら、セカンダリサイトに蓄積された変更をレプリケートしてプライマリサイトに戻し、そのサイトで本番負荷の処理を再開できます。

そのような計画では、通常は次のような順序でイベントが発生します。

- プライマリシステムが本番負荷を処理し、それをセカンダリシステムにレプリケートしています。
- プライマリサイトで完全なシステム障害になるような災害が発生します。管理者がセカンダリサイトでレプリケーションの方向を逆にし、レプリケートされたシェアを新しいプロジェクトにエクスポートします。このプロジェクトは、プライマリサービスが復元されたときにプライマリサイトにレプリケートして戻すように構成されています。その一方で、本番負荷はセカンダリサイトにリダイレクトされます。
- プライマリサイトがオンラインに戻ったら、管理者がセカンダリサイトからプライマリサイトへのレプリケーション更新を開始します。これにより、プライマリのコピーがレプリケーションパッケージに変換され、ターゲットへの (障害の前の) 最後の正常な更新のあとに加えられた変更がすべてロールバックされます。プライマリサイトのコピーが最新の状態になったら、管理者はプライマリサイトのコピーを書き込み可能にして、レプリケーションの方向を再度逆にします。本番トラフィックがプライマリサイトにリダイレクトされて戻されます。プライマリサイトからセカンダリサイトへのレプリケーションが再開され、プライマリコピーとセカンダリコピーの間の最初の関係が復元されます。

パッケージのレプリケーションの方向を逆にするときは、その前にソースからそのプロジェクトのレプリケーションを停止することを管理者に強くお勧めします。レプリケーション更新の進行中に管理者がプロジェクトのレプリケーションの方向を逆にした場合、管理者は、以前のターゲット ZFSSA (現在のソース ZFSSA) に作成されたプロジェクトに、どの整合性のあるレプリケーションスナップショットが使用されたのかわからなくなります。

BUI からレプリケーションを逆向きにするには、レプリケーションパッケージに移動し (上記を参照)、「レプリケーション」タブ、続いて  ボタンをクリックします。表示されるダイアログで、管理者は新しいローカルプロジェクトの名前を指定できます。

CLI からレプリケーションを逆向きにするには、レプリケーションパッケージに移動し (上記を参照)、`reverse` コマンドを使用します。このコマンドは、新しいローカルプロジェクトの名前を指定するオプションの引数を取ります。引数が指定されていない場合は、元の名前が使用されます。

ローカルのシェアはすべてエクスポートされるため、パッケージを逆向きにすると、それまでにエクスポートされたかどうかにかかわらずパッケージ内のすべてのシェアがエクスポートされます (上記を参照)。レプリケートされたファイルシステムとシステム上のほかのファイルシステムの間でマウントポイントが競合している場合、逆向きにする操作は失敗します。切断する前に、関連するシェアのマウントポイントを再構成して、これらの競合を解決する必要があります。通常、この操作は本番用のサービスの復元におけるクリティカルパスの一部なので、このようなマウントポイントの競合は、DR フェイルオーバーの時点ではなく、システムを最初に設定するときに解決することを強くお勧めします。

## レプリケーションパッケージの破棄

パッケージ内のプロジェクトとシェアを破棄するには、パッケージ全体を破棄する必要があります。BUI からパッケージ全体を破棄するには、対応するプロジェクトを破棄します。CLI からパッケージを破棄するには、shares replication sources ノードで destroy コマンドを使用します。

パッケージが破棄されると、対応するアクションからの以降のレプリケーション更新は失敗します。レプリケーションを再開するには、ソース上でアクションを作成し直して、データの新しいコピーを受信するための新しいパッケージをターゲット上に作成します。



## レプリケーションタスク




次のタスクは、レプリケーションの手順例です。

### レプリケーションを逆向きにする - レプリケーションの確立

次の例は、レプリケーションの方向を逆にして、典型的な 2 システムによる障害回復をサポートします。この例で、M11 は本番システム、M5 は回復システムです。

#### ▼ レプリケーションを逆向きにする

1. 本番システム M11 で、「構成」>「サービス」に移動します。
2. 「データサービス」の下の「SMB」行でステータスが「無効」である場合、「サービスを有効化」をクリックします。
3. 「構成」>「サービス」>「リモートレプリケーション」に移動します。
4. 「 ターゲット」をクリックし、名前、ホスト名、およびパスワードの各設定を構成します。「名前=M5」、「ホスト名=192.168.1.17」、「ルートパスワード=pppp\$1234」
5. 「プール=Pool1」を選択します。
6. 「シェア」>「プロジェクト」に移動します。
7. 「 プロジェクト」をクリックします。「名前」=P1

8. 「シェア」>「プロジェクト」>「P1」>「プロトコル」に移動します。
9. SMB セクションで、「リソース名=on」を設定します。
10. 「シェア」>「プロジェクト」>「P1」>「シェア」に移動します。
11. 「 ファイルシステム」をクリックします。「名前=S1」、「ユーザー=root」、「グループ=other」、「アクセス権=RWX RWX RWX」
12. 「シェア」>「プロジェクト」>「P1」>「シェア」>「S1」>「プロトコル」に移動します。「SMB」セクションに、SMB を使用して ¥¥192.168.1.7¥S1 で S1 に到達できることが示されます。
13. 「シェア」>「プロジェクト」>「P1」>「レプリケーション」に移動します。
14. 「 アクション」をクリックし、ターゲットとプールを設定します。「ターゲット=M5」、「プール=Pool1」
15. 「 スケジュール」をクリックし、頻度を設定します。「頻度=Half-Hour at 00 minutes past the hour」。
16. SMB クライアントシステムで、ネットワークドライブ ¥¥192.168.1.7¥S1 (user=root, password=pppp\$1234) をマップします。
17. ファイル F1.txt を作成します。
18. 本番システム M11 で、「シェア」>「プロジェクト」>「P1」>「レプリケーション」に移動します。
19. アクション行 TARGET=M5 で、「今すぐ更新」をクリックします。
20. レプリケーションが完了したら「無効」をクリックします。「ステータス」が「無効」に変更されます。

## レプリケーションを逆向きにする - 障害からの回復のシミュレート

本番システムへのアクセスを妨げる障害からの回復をシミュレートするには、回復システムを使用してレプリケーションを逆向きにします。レプリケーションを逆向きにすると、ターゲット上に存在するレプリケーションパッケージがローカルプロジェクトに変換され、このローカルプロジェクトを増分レプリケートして元のソースシステムに戻すレプリケーションアクションが

追加で構成されます。このレプリケーションアクションはデフォルトで無効になっています。管理者は更新を手動で送信するようにしてください。

## ▼ レプリケーションを逆向きにする

1. SMB クライアントシステム上でのプライマリシステム M11 との連絡の損失をシミュレートするには、「ネットワークドライブの切断」を選択します。
2. 障害回復システム M5 で、「プール=Pool1」を選択します。
3. 「シェア」>「プロジェクト」>「レプリカ」に移動します。プロジェクト M11:P1 がリストに表示されます
4. 「シェア」>「プロジェクト」>「レプリカ」>「M11:P1」>「レプリケーション」に移動します。パッケージは「ステータス=Idle」になっています
5. 「レプリケーションの方向を逆にする」をクリックし、新しいプロジェクト名を設定します。「新しいプロジェクト名=P1」
6. 「シェア」>「プロジェクト」>「レプリカ」に移動します。ターゲット上に存在するレプリケーションパッケージがローカルプロジェクトに変換されたため、プロジェクト M11:P1 はリストに表示されなくなりました。
7. 「構成」>「サービス」を選択します。
8. 「データサービス」の下の「SMB」行でステータスが「無効」である場合、「サービスを有効化」をクリックします。
9. 「シェア」>「プロジェクト」>「ローカル」に移動します。プロジェクト P1 がリストに表示されます。
10. 「シェア」>「プロジェクト」>「P1」>「プロトコル」に移動します。
11. SMB セクションで、「リソース名=on」を設定します。
12. 「シェア」>「プロジェクト」>「P1」>「シェア」>「S1」>「プロトコル」に移動します。「SMB」セクションに、SMB を使用して ¥¥192.168.1.17¥¥S1 で S1 に到達できることが示されます。
13. SMB クライアントシステムで、ネットワークドライブ ¥¥192.168.1.17¥¥S1 (user=root, password=pppp\$1234) をマップします。


14. ファイル F1.txt を編集し、F2.txt として保存します。注: 実際の障害回復シーケンスでは、本番システム M11 との通信が復元されたら、アプリケーションが障害回復システム M5 上でデータへのアクセスを継続中に、手動、定期、または連続レプリケーション更新をトリガーする機会があります。
15. 本番システムにふたたび移行する準備を行うには、「ネットワークドライブの切断」を選択します。
16. 障害回復システム M5 で、「シェア」>「プロジェクト」>「P1」>「レプリケーション」に移動します。
17. 「アクション」行 TARGET=M11 で、「今すぐ更新」をクリックします。
18. レプリケーションが完了したら「無効」をクリックします。

## レプリケーションを逆向きにする - 本番システムからのレプリケーションの再開

レプリケーションを逆向きにするたびに、レプリケーションパッケージを新しいローカルプロジェクトに変換するとき使用する新しいプロジェクト名を指定します。同じ名前を使用する場合、かつ以前のレプリケーション逆転によりその名前の空のローカルプロジェクトが残されている場合は、次の逆転で同じ名前のプロジェクトを作成できるように、既存の空のプロジェクトを削除する必要があります。

### ▼ レプリケーションを逆向きにする

1. 本番システム M11 で、「シェア」>「プロジェクト」>「ローカル」>「P1」に移動します。新しいソースシステム (元のターゲット) からの最初の更新により、元のソースシステム上の元のプロジェクトがレプリケーションパッケージに変換されるため、P1 は空です。この例では、プロジェクト内のすべてのシェアをレプリケートする、プロジェクトレベルのレプリケーションアクションを使用します。このため、このローカルプロジェクトの下にあったすべてのシェアは、現在はレプリケーションパッケージの下にあり、ローカルプロジェクトは空のままになります。
2. 「シェア」>「プロジェクト」>「ローカル」に移動します。
3. P1 を削除します。元のターゲットからの逆方向レプリケーションの結果として、その内容はすでにレプリケーションパッケージに移動されているため、この空のプロジェクトを削除しても安全です。


4. 「シェア」>「プロジェクト」>「レプリカ」>「M5:P1」>「レプリケーション」に移動します。
5. 「レプリケーションの方向を逆にする」をクリックし、プロジェクト名を設定します。「新しいプロジェクト名=P1」
6. SMB クライアントシステムで、ネットワークドライブ ¥¥192.168.1.7¥S1 (user=root, password=pppp\$1234) をマップします。ファイル F1.txt と F2.txt がリストに表示されます。
7. 本番システム M11 で、「シェア」>「プロジェクト」>「P1」>「レプリケーション」に移動します。
8. 「アクション」行 TARGET=M5 で、「エントリを編集」をクリックします。
9. 「 スケジュール」をクリックし、頻度を設定します。「頻度=Half-Hour at 00 minutes past the hour」。
10. 「アクション」行 TARGET=M5 で、「今すぐ更新」をクリックします。これは、障害回復システム M5 に対し、ローカルプロジェクト P1 をレプリケーションパッケージ M11:P1 内に変換して戻すよう指示します。
11. アクション行 TARGET=M5 の更新の列をモニターして、レプリケーション更新が完了するのを待ちます。
12. 障害回復システム M5 で、「シェア」>「プロジェクト」>「ローカル」>「P1」に移動します。プロジェクトがレプリケーションパッケージ内に変換して戻されたため、P1 は空です。
13. 「シェア」>「プロジェクト」>「ローカル」に移動します。
14. 次の逆転でレプリケーションパッケージを P1 という名前のプロジェクトに変換できるようにするには、P1 を削除します。

## レプリケーションでの静的ルートの強制的な使用

レプリケーショントラフィックを特定のネットワークインターフェースに統合するには、静的ルートを使用してソースおよびターゲットの ZFS ZFSSA を接続する必要があります。静的ルートを設定するには、次の手順を使用します。




## ▼ レプリケーションでの静的ルートの強制的な使用

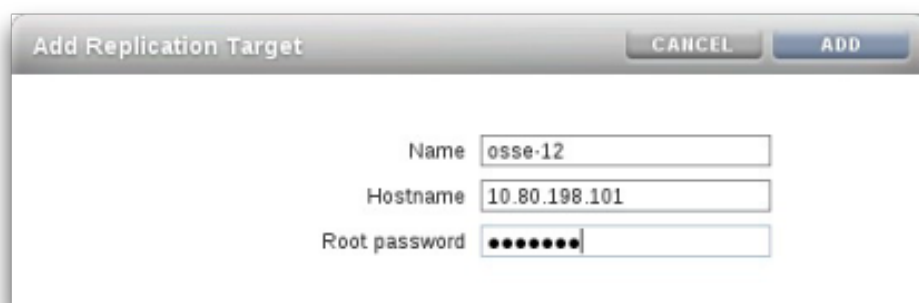
1. 静的ルートを設定するには、「構成」>「ネットワーク」>「ルーティング」ページで  追加アイコンをクリックします。
2. 「静的ルートの挿入」ボックスで「ファミリー」および「種類」を選択し、「宛先 IP」、「ゲートウェイ」、「および「インタフェース」を入力します。



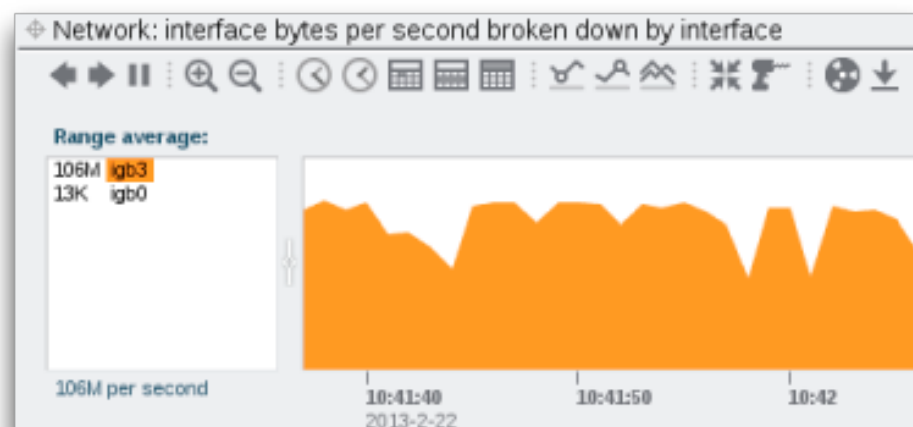
3. 「追加」をクリックします。
4. トラフィックがソースおよびターゲットを經由していることを確認するには、CLI で `tracertoute` を使用します。`tracertoute` の使用方法の詳細は、[79 ページの「ネットワークルーティングの構成」](#)を参照してください。次の例では、`10.80.219.124 @ igb0` で `igb0` をインタフェースとして識別しています。これは、正しいインタフェースが使用されていることを確認する簡単な方法です。

```
brmv01sn02:> tracertoute poc7330-050
tracertoute: Warning: Multiple interfaces found; using 10.80.219.124 @ igb0
tracertoute
to poc7330-050 (10.80.219.117), 30 hops max, 40 byte packets
 1  poc7330-050.us.oracle.com (10.80.219.117)  0.446 ms  0.115 ms  0.104 ms
```

5. 新しいレプリケーションターゲットを追加するには、「構成」>「サービス」>「レプリケーション」ページで  追加アイコンをクリックします。
6. 「レプリケーションターゲットの追加」ボックスで、ターゲットの名前、ネットワークインタフェースのホスト名 IP アドレス、およびパスワードを入力します。



7. 「追加」をクリックします。
8. 定義した静的ルートをトラフィックが経由していることを確認するには、レプリケーションの開始後に、『Oracle ZFS Storage Appliance Analytics ガイド』の「ネットワークインタフェースバイト数」を使用します。



9. 「設定」ページで、147 ページの「設定のプロパティ」がオンに切り替えられていることを確認します。
10. ソースからターゲットへのレプリケーションで正しいインタフェースが使用されていることを確認したあと、レプリケーションを逆方向にします。レプリケーションを逆方向にする詳細は、394 ページの「レプリケーションの方向を逆にする」を参照してください。

## 受信したレプリケーションプロジェクトのクローニング

これは、プロジェクトと 1 つのシェアの両方のマウントポイントをオーバーライドして、受信したレプリケーションプロジェクトをクローニングする CLI 例です。

```
perch:> shares
perch:shares> replication
perch:shares replication> sources
perch:shares replication sources> select source-000
perch:shares replication source-000> select package-000
perch:shares replication source-000 package-000> clone
perch:shares replication source-000 package-000 clone> set target_project=my_clone
      target_project = my_clone
perch:shares replication source-000 package-000 clone> list
CLONE PARAMETERS
      target_project = my_clone
      original_mountpoint = /export
      override_mountpoint = false
      mountpoint =

      SHARE                MOUNTPOINT
      bob                  (inherited)
      myfs1                (inherited)
perch:shares replication source-000 package-000 clone> set override_mountpoint=true
      override_mountpoint = true
perch:shares replication source-000 package-000 clone> set mountpoint=/export/my_clone
      mountpoint = /export/my_clone
perch:shares replication source-000 package-000 clone> select bob
perch:shares replication source-000 package-000 clone bob> set override_mountpoint=true
      override_mountpoint = true
perch:shares replication source-000 package-000 clone bob> set mountpoint=/export/bob
      mountpoint = /export/bob
perch:shares replication source-000 package-000 clone bob> done
perch:shares replication source-000 package-000 clone> commit
CLONE PARAMETERS
      target_project = my_clone
      original_mountpoint = /export
      override_mountpoint = true
      mountpoint = /export/my_clone

      SHARE                MOUNTPOINT
      bob                  /export/bob (overridden)
      myfs1                (inherited)
Are you sure you want to clone this project?
There are no conflicts.
perch:shares replication source-000 package-000 clone>
```

## リモートレプリケーションの詳細

### 承認

「サービス」スコープの「リモートレプリケーション」フィルタを使用して管理者はレプリケーションサービスの停止、開始、および再開を実行できるほか、レプリケーションサブシステムに

よって「プロジェクトとシェア」スコープの下に 2 つの[138 ページの「ユーザー承認」](#)が提供されています。

承認	詳細
rrsource	レプリケーションターゲットとアクションの作成、編集、破棄、およびレプリケーションアクションの更新の送信と取り消しを管理者に許可します。
rrtarget	レプリケートされたパッケージの管理を管理者に許可します。これには、パッケージレベルでのレプリケーションの無効化、パッケージやそのメンバーのクローニング、受信されたデータセットのプロパティの変更、レプリケーションの切断またはレプリケーションの向きを逆にする操作が含まれます。これらの操作の一部には、ほかの承認が必要になる場合もあります (プロパティの設定、個々のシェアのクローニングなど)。詳細は、「プロジェクトとシェア」スコープで使用可能な承認を参照してください。

ZFSSA にレプリケーションターゲットを構成するには、「リモートレプリケーション」サービス画面で構成する場合でも、rrsource 承認が必要です。承認の詳細は、[138 ページの「ユーザー承認」](#)を参照してください。

## 警告

次のいずれかのイベントが発生すると警告が通知されます。

- 手動または定期のレプリケーション更新が正常に開始または終了した (ソースとターゲットの両方)。
- 管理者が明示的に取り消した場合も含め、レプリケーション更新が失敗した (ソースとターゲットの両方)。
- 同じアクションの別の更新がすでに進行中のため、定期レプリケーション更新が省略された (上記を参照)。
- 連続レプリケーションをはじめて開始した場合。
- 連続レプリケーションが失敗した場合。
- 連続レプリケーションをはじめて開始した場合、失敗した場合、または障害のあとに再開した場合。

## レプリケーション監査イベント

システムは、次のレプリケーションイベントを監査して、『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「ログ」に記録します。

- レプリケーションアクションの作成、変更、または破棄
- レプリケーショングループに対するシェアの追加または除去
- ターゲット上のレプリケーションパッケージの作成、変更、クローニング、逆転、切断、または破棄
- レプリケーションターゲットの作成、変更、または破棄

## レプリケーションとクラスタ化

任意の ZFS Storage Appliance から別の任意の ZFS Storage Appliance に対してレプリケーションを構成できます。これは、それぞれがクラスタに属しているかどうか、また、ZFSSA のクラスタピアにいずれかの方向のレプリケーションが構成されているかどうかには依存しませんが、次の制約があります。

- クラスタの両方のピアから同じレプリケーションターゲットへのレプリケーションを構成することはできません。ただし、同じターゲット ZFSSA に 2 つの異なる IP アドレスを使用することで、そのような構成を実現できます。管理者はターゲット ZFSSA の複数の IP アドレスを使用することで、各クラスタヘッドに 1 つのレプリケーションターゲットを作成し、そのヘッドで使用できます。
- クラスタピア間のレプリケーションを構成するときは、両方のコントローラを CLUSTERED 状態にしてレプリケーションを構成してください。プライベートネットワークアドレスは使用せず、各コントローラのプールに個別のレプリケーションターゲットを使用してください。

クラスタ構成でのレプリケーションの動作は、次の規則に従います。

- プロジェクトとシェアのレプリケーション更新は、それを格納しているストレージプールをインポートしたいずれかのクラスタピアから送信されます。
- レプリケーション更新は、ソース上のレプリケーションアクションに構成されている IP アドレスをインポートしたいずれかのピアによって受信されます。管理者は必ず、この IP アドレスを使用しているヘッドが、インポートされたレプリカを格納するストレージプールを持つようにしてください。クラスタを構成する際にプールおよび IP アドレスのリソースを同じヘッドに割り当てることで、この条件が保証されます。
- ZFSSA が (テイクオーバーまたはフェイルバックの一部として) 対応するストレージプールまたは IP アドレスをエクスポートしたときに進行中のレプリケーション更新 (ZFSSA へ、ZFSSA からの両方) は失敗します。テイクオーバーまたはフェイルバッ

ク操作の影響を受けないストレージプールおよび IP アドレスを使用しているレプリケーション更新は、その操作の影響を受けません。

クラスタ化およびクラスタの用語の詳細は、[第10章「クラスタ構成」](#)を参照してください。

## スナップショットとデータ整合性

ZFSSA はスナップショットをレプリケートし、各スナップショットはターゲットで原子的に受信されるため、ターゲット上のシェアのレプリカの内容は、スナップショットが作成された時点におけるソース上のシェアの内容と常に一致します。特定のグループで送信されるすべてのシェアのスナップショットは同時に作成されるため (上記を参照)、レプリケーション更新が正常に完了したあとは、パッケージの内容全体が、ソース上でスナップショットが作成された時点 (レプリケーション更新の開始時) のグループの内容と厳密に一致します。

ただし、各シェアのスナップショットは個別に (および順次に) レプリケートされるため、パッケージ内の一部のシェアが、同じパッケージ内のほかのシェアよりも新しいスナップショットで更新されていることはありえます。このようになるのは、レプリケーション更新の進行中 (一部のシェアは更新済みだが、ほかのシェアはまだ更新されていない)、およびレプリケーション更新の失敗後 (一部のシェアは更新済みだが、ほかのシェアは更新されていない可能性がある) です。

まとめると、次のようになります。

- 各シェアには、ターゲット上で常にポイントインタイムの整合性があります (自己整合性)。
- 前回のレプリケーション更新が成功した場合で、レプリケーション更新が進行中でないときは、パッケージの各シェア相互にもポイントインタイムの整合性があります (パッケージ整合性)。
- レプリケーション更新が進行中の場合、または前回の更新が失敗した場合、パッケージのシェア相互には整合性がない可能性があります。それぞれの自己整合性は保たれます。パッケージの整合性がアプライアンスに重要となる場合は、レプリケーションパッケージのクローンを作成する必要があります。これにより、各シェアのもっとも新しく正常に受信されたスナップショットのクローンが常に作成されます。

## スナップショットの管理

スナップショットは増分レプリケーションの基礎です。増分レプリケーションを継続するためには、ソースとターゲットが常に共通のスナップショットをシェアする必要があります。また、ソースは、どれがターゲット上で最新のスナップショットかを把握する必要があります。これを簡単にするために、レプリケーションサブシステムは独自のスナップショットを作成して管

理します。通常、管理者がこれらに注意を払う必要はありませんが、スナップショットはストレージの利用率にかなり影響することがあるため、ここで詳細に説明します。

特定のアクションのレプリケーション更新は、それぞれ次の手順で行われます。

- 以前にこのアクションのレプリケーションを試みたかどうか、およびターゲットが増分更新に必要なスナップショットをすでに持っているかどうかに基づいて、これが増分更新か完全更新かを判定します。
- 新しいプロジェクトレベルのスナップショットを作成します。
- 更新を送信します。完全更新の場合は、この新しいスナップショットまでのグループの内容全体を送信します。増分更新の場合は、前回のスナップショット (基礎) とこの新しいスナップショットの間の差分を送信します。
- この新しいスナップショットを次回の更新のベーススナップショットとして記録し、前回のベーススナップショットを破棄します (増分更新の場合)。ベーススナップショットは、次回の更新を受信するまでターゲット上に残ります。次回の更新の時点で、そのベーススナップショットは破棄される最初の項目となります。

これは、スナップショットの管理にいくつかの影響を与えます。

- 最初のレプリケーション更新の進行中、および最初の更新のあとレプリケーションがアクティブでないときは、グループのプロジェクトまたはシェアに構成されている各アクションに、プロジェクトレベルのスナップショットが正確に 1 つ存在します。レプリケーションアクションで、そのアクションによりレプリケートされているグループ内のシェアと同じプロジェクト内にあるが、そのグループ用の更新の一部として送信されていないシェア上にスナップショットが作成されることがあります。
- 特定のアクションによる以降のレプリケーション更新の進行中には、そのアクションに関連するプロジェクトレベルのスナップショットが 2 つ存在することがあります。ソースで、ターゲットが新しいスナップショットを正常に受信したかどうかを判定できなかった場合 (更新中にネットワークの機能停止で障害が発生した場合など)、更新の完了後に両方のスナップショットが残ることがあります。
- レプリケーションアクションに関連付けられたスナップショットを管理者が破棄すると、増分レプリケーションが破壊されます。管理者は、ソースまたはターゲット上にある、増分レプリケーションに必要なスナップショットを破棄することはできません。ソース上のそのようなスナップショットを破棄するには、アクションを破棄する必要があります (これにより、そのアクションに関連付けられたスナップショットが破棄される)。ターゲット上のそのようなスナップショットを破棄するには、まずパッケージを切断する必要があります (これにより、そのパッケージに増分更新を受信できなくなる)。
- 管理者はレプリケーションスナップショットより前に作成されたスナップショットにロールバックしてはいけません。この操作を行うと、それよりあとのレプリケーションスナップショットが破棄され、それらのスナップショットを使用するすべてのアクションの増分レプリケーションが破壊されます。

- レプリケーションでスナップショットを使用するには、管理者は、ZFSSA の298 ページの「スペース管理」、特にその298 ページの「スナップショットへの適用」について理解している必要があります。
- LUN のレプリケーションのスペース管理の詳細は、298 ページの「LUN のレプリケーションのスペース管理」を参照してください

## iSCSI 構成のレプリケーション

前述のとおり、レプリケーション更新には、プロジェクトとそのシェアの「シェア」画面で指定された構成のほとんどが含まれます。これには、レプリケートされる LUN に関連付けられたすべてのターゲットグループとイニシエータグループが含まれます。デフォルト以外のターゲットグループとイニシエータグループを使用する場合、管理者は、プロジェクト内の LUN で使用されるターゲットグループとイニシエータグループがレプリケーションターゲット上にも存在することを確認する必要があります。グループは同じ名前が存在していればよく、同じ構成を定義している必要はありません。これを確認できない場合、レプリケートされた LUN のクローン作成とエクスポートが失敗することがあります。

LUN に関連付けられた SCSI GUID は、LUN とともにレプリケートされます。その結果、ターゲット ZFSSA 上の LUN の SCSI GUID は、ソース ZFSSA 上の LUN と同じになります。ただし、(ローカル LUN のクローンの GUID が元の GUID とは異なるように) レプリケートされた LUN のクローンの GUID は異なります。

## クローンのレプリケーション

2009.Q3 以前は、レプリケーションはプロジェクトレベルのみで、プロジェクトの外部にあるスナップショットから作成されたクローンを含んでいるプロジェクトをレプリケーションすることは明示的に禁止されていました。2010.Q1 以降のシェアレベルのレプリケーションでは、この制限は緩和されていますが、管理者はレプリケートされるクローンの元のスナップショットを引き続き考慮する必要があります。特に、クローンを最初にレプリケーションするときは、元のスナップショットがターゲットにすでにレプリケートされているか、同じ更新の一環としてレプリケートされる必要があります。この制限が ZFSSA 管理ソフトウェアによって強制されることはありませんが、元のスナップショットがターゲットに存在していないときにクローンをレプリケートしようとするとうまく失敗します。

実際には、クローンのレプリケーションを確実に成功させる方法がいくつかあります。

- クローンの元のスナップショットが同じプロジェクト内にある場合は、プロジェクトレベルのレプリケーションのみを使用します。
- クローンの元のスナップショットが同じプロジェクト内にない場合や、元のスナップショットを含むプロジェクトレベルのレプリケーションがほかの理由で望ましくない場合



は、シェアレベルのレプリケーションを使用して元のシェアをまずレプリケートし、次にプロジェクトレベルまたはシェアレベルのレプリケーションを使用してクローンをレプリケートします。

- クローン自体も破棄する場合を除き、ターゲットシステム上のクローンの元のスナップショットを破棄しないでください。

どの場合も、元のスナップショットが実際にターゲットに送信されるようにするために、元のスナップショットのアクションの「スナップショットを含める」プロパティを true にしてください。

## レプリケーションの監視

次の『Oracle ZFS Storage Appliance Analytics ガイド』の「Analytics」をレプリケーションで使用できます。

- 『Oracle ZFS Storage Appliance Analytics ガイド』の「データ移動レプリケーション操作」
- 『Oracle ZFS Storage Appliance Analytics ガイド』の「データ移動レプリケーション (バイト)」
- 『Oracle ZFS Storage Appliance Analytics ガイド』の「統計」 are also available.

## レプリケーションの失敗

個々のレプリケーション更新はいくつかの理由で失敗することがあります。可能な場合、ZFSSA は、ソース ZFSSA またはターゲット ZFSSA で通知される警告か、失敗したアクションの「レプリケーション」画面で、失敗の理由を報告します。アクションのステータスを表すオレンジ色の警告アイコンをクリックすると、失敗の詳細を取得できます。もっとも一般的な失敗の種類は次のとおりです。

失敗	詳細
取り消されました	レプリケーション更新は管理者によって取り消されました。レプリケーションはソース上またはターゲット上で取り消すことができ、一方のピアには他方のピアが操作を取り消したことがわからない場合があります。
ネットワーク接続失敗	ネットワークの問題のため、ZFSSA はターゲット ZFSSA に接続できませんでした。ソース、ターゲット、またはネットワークの構成に誤りがある可能性があります。

失敗	詳細
ピア検証失敗	ZFSSA はターゲットのアイデンティティを確認できませんでした。これは、ターゲットを再インストールまたは出荷時設定にリセットした場合によく発生します。ターゲットを再インストールまたは出荷時構成にリセットした場合は、新しい認証鍵セットを生成するために、そのターゲットに対する新しいレプリケーションターゲットをソース ZFSSA 上で構成する必要があります。 <a href="#">375 ページの「プロジェクトレプリケーションターゲット」</a> を参照してください。
ピア RPC 失敗	ターゲットシステムでリモート手続き呼び出しが失敗しました。これは、ターゲット ZFSSA で互換性のないソフトウェアが実行されている場合によく発生します。詳細は、 <a href="#">412 ページの「2009.Q3 以前からのアップグレード」</a> を参照してください。
パッケージがありません	レプリケートされたデータを格納するパッケージがターゲット上に存在しないため、レプリケーションが失敗しました。パッケージはアクションの構成時に作成されるため、このエラーは通常、管理者がターゲット上のパッケージを破棄したあとに発生します。パッケージを格納しているストレージプールがターゲットシステムにインポートされていない場合にも、このエラーが発生する可能性があります。プールに障害が発生したときや、ターゲット ZFSSA でストレージまたはネットワークの再構成が行われたときに、このような状況になることがあります。
空でないパッケージが存在します	ターゲットパッケージに以前の失敗したレプリケーション更新のデータが含まれているため、レプリケーションが失敗しました。このエラーは、アクションの最初のレプリケーション更新が一部のデータをレプリケートしたあとで失敗した場合に、そのアクションのレプリケーション更新を送信しようとするとき発生します。ターゲット ZFSSA では、明示的な管理操作で指示しないかぎりデータは破棄されないため、部分的に受信されたデータは上書きされません。管理者は既存のアクションとパッケージを削除し、ソース上で新しいアクションを作成してレプリケーションを再度開始するようにしてください。
無効	ターゲット上でレプリケーションが無効になっているため、レプリケーションが失敗しました。ターゲット上でレプリケーションサービスが無効になっているか、レプリケートされている特定のパッケージに対してレプリケーションが無効になっています。
ターゲットビジー	ターゲットシステムが同時レプリケーション更新の最大数に到達したため、レプリケーションが失敗しました。システムでは、リソース不足を防ぐために、進行中のレ

失敗	詳細
	レプリケーション処理の最大数が制限されます。この制限に到達した場合、それ以降は、更新の受信の試行はこのエラーで失敗しますが、更新の送信の試行はリソースが使用可能になるまでキューに入れられます。
容量不足	ソースシステムに新しいスナップショットを作成するための領域が不足しているため、レプリケーションが失敗しました。ストレージプールに使用可能な物理的領域がないか、スナップショットを含まない予約のためプロジェクトまたはそのシェアの1つが割り当て制限を超えていることが、この原因になっている場合があります。
互換性のないターゲット	ターゲットシステムがソースシステムのデータストリーム形式を受信できないため、レプリケーションが失敗しました。ソースシステムにはアップグレードと遅延更新の適用を行なったが、ターゲットにはアップグレードおよび同じ更新の適用を行っていない場合に、これが発生する可能性があります。ソースシステムのソフトウェアバージョンのリリースノートで遅延更新のリストを調べ、リモートレプリケーションに関連するものがないか確認してください。
その他	レプリケーションが失敗しましたが、ソースには使用可能な追加情報はありません。ターゲットシステムの警告ログを確認し、必要に応じてサポートに問い合わせてください。現在、このカテゴリに入る障害モードとしては、ターゲットに更新を受信するためのディスク容量が不足している、クローンの元のスナップショットがターゲットシステムに存在しない状態でクローンをレプリケートしようとした、などがあります。

レプリケーション更新の一部が失敗した場合、その更新は失敗になります。現在の実装では、プロジェクト内のシェアが順次にレプリケートされ、失敗した更新から変更がロールバックされることはありません。そのため、更新が失敗した場合、ターゲット上の一部のシェアは最新の状態で、ほかのシェアは最新の状態でない可能性があります。詳細は、前述の「スナップショットとデータ整合性」を参照してください。

失敗した更新で一部のデータは正常にレプリケートされている可能性はありますが、現在の実装では、前回の (失敗した) 更新で送信されたすべてのデータが再度送信されます。つまり、失敗した更新は、中止したところから再開するのではなく、失敗した更新が開始したところから開始します。

手動更新または定期更新が失敗した場合、次の定期更新 (存在する場合) まで、自動的に再試行されることはありません。連続レプリケーションが失敗した場合、システムは数分

待機してから再試行します。失敗した連続レプリケーションの再試行は無限に繰り返されません。

レプリケーション更新の進行中に、別の更新の実行がスケジュールされている場合、後者の更新は、前の更新が完了した直後に開始されるのではなく、完全に省略されます。次の更新は、次の更新の実行がスケジュールされている場合にのみ送信されます。この理由で更新が省略された場合は、警告が通知されます。

## レプリケーションの互換性

レプリケーション更新を実行する前に、レプリケーションサービスは、ターゲットシステムがソースからの新規データと互換性があることを検証します。

- ターゲットと互換性のないソース上で使用中の機能が存在し、その機能を安全に無効にできる場合は、レプリケーションサービスはその機能を無効にして更新を実行し、警告を発行します。
- ターゲットと互換性のないソース上で使用中の機能が存在し、その機能を無効にできない場合は、レプリケーションサービスは更新を実行せずに、エラーを発行します。

注: ターゲットはできるかぎり早くアップグレードするのが常に最善です。

レプリケーションの互換性を壊す更新は、遅延更新として配布されます。最新のリストおよび説明については、『[Oracle ZFS Storage Appliance 顧客サービスマニュアル](#)』の「[更新](#)」および最新リリースの Oracle ZFS Storage Appliance リリースノートを参照してください。

## 2009.Q3 以前からのアップグレード

2009.Q3 リリースと 2010.Q1 リリースの間では、レプリケーションの実装が大幅に変更されています。2009.Q3 以前からのアップグレードを開始する前に、ZFSSA へのレプリケーションおよび ZFSSA からのレプリケーションを一時停止することを引き続き強くお勧めします。順次アップグレードを使用するクラスターでは、これは必須です。

2010.Q1 以降へのアップグレードに関連して、ユーザーの目に触れる重要な変更が 3 つあります。

- レプリケーションに使用されるネットワークプロトコルが拡張されました。2009.Q3 システムは、任意のリリース (2010.Q1 以降も含む) を実行しているシステムにレプリケートできますが、2010.Q1 以降を実行しているシステムは、2010.Q1 以降を実行しているほかのシステムにのみレプリケートできます。実際には、プロトコルのバージョンに

互換性がないことから発生する障害を避けるために、レプリケーションターゲットのアップグレードをレプリケーションソースよりも前または同時に行う必要があります。

- レプリケーションアクションの構成は、ヘッドシステム上ではなく、ストレージプール自体に保存されるようになりました。そのため、2009.Q3 以前から 2010.Q1 にアップグレードしたあとで、レプリケーション構成を移行するために管理者が遅延更新を適用する必要があります。
- \* これらの更新を適用するまでは、既存のレプリケーションのレプリケーション更新が着信しても失敗し、2009.Q3 以前で構成されていたアクションのレプリケーション更新は送信されません。さらに、BUI や CLI からは管理できない未移行のレプリカのためにストレージプールの領域が使用されます。
- \* すべての遅延更新と同様に、これらの更新を適用したあとでシステムのソフトウェアをロールバックすると、結果は不定になります。その古いリリースでは、レプリケートされたデータにアクセスできなくなり、レプリケーションアクションはすべて未構成になり、着信するレプリケーション更新は完全更新になることが想定されます。
- レプリケーションの承認は、専用のスコープから「プロジェクトとシェア」スコープに移動されました。2009.Q3 以前で構成されていたレプリケーションの承認は、2010.Q1 には存在しなくなります。レプリケーションに詳細なアクセス制御を使用している管理者は、アップグレード後に新しいレプリケーションの承認を適切な管理者に委任するようにしてください。



# ◆◆◆ 第 14 章

## シャドウ移行

---

このセクションでは ZFSSA のシャドウ移行について説明します。

### データの移行

管理者の一般的なタスクの 1 つは、ある場所から別の場所にデータを移動することです。もっとも抽象的な意味では、この問題は、サーバー間でデータをレプリケートすることからラップトップ上のユーザーデータをサーバーと同期された状態に保つことまで、多数のユースケースを含んでいます。このタスクには多数の外部ツールを利用できますが、ZFSSA には、もっとも一般的なユースケースに対応するデータ移行用の総合的な解決方法が 2 つ用意されています。1 つめは第13章「レプリケーション」です。これは、1 つ以上の ZFSSA 間でデータをレプリケートするためのもので、これについては別途説明します。2 つめはシャドウ移行で、ここで説明します。

シャドウ移行とは、移行が完了した時点で元のデータを置換または廃棄するために、外部の NAS ソースからデータを移行するためのプロセスです。これは、新しい ZFSSA を既存の環境に導入する際に別のサーバーのファイルシェア業務を引き継ぐためにもっともよく使われますが、下記で説明するように、ほかにもいくつかの新しい使い方が可能です (下記参照)。

### 従来のデータ移行

通常、従来のファイル移行は反復同期または外部介入のどちらかの方法で行われます。

#### 同期による移行

この方法では、アクティブなホスト X を取り、X をアクティブにしたままデータを新しいホスト Y に移行します。この移行の進行中も、クライアントは引き続き元のホストに対して読み取り/書き込みを行います。データが最初に移行されると、差分が 1 回の停止時間帯で送信できるほど小さくなるまで増分変更が繰り返し送信されます。この時点で、元のシェアは読み取り専用になり、最後の差分が新しいホストに送信されて、すべてのクライアントが新

しい場所を指すように更新されます。これを実行するもっとも一般的な方法は再同期ツールを使用することですが、ほかの統合ツールも存在します。このメカニズムにはいくつかの欠点があります。

- 差分が小さくなる間、予測される停止時間を簡単に定量化できません。スケジュールされた停止時間の直前にユーザーが大量の変更を確定した場合は、停止時間帯が長くなる可能性があります。
- 移行中は、新しいサーバーがアイドル状態になります。通常、新しいサーバーでは新しい機能が追加されたりパフォーマンスが向上したりしているため、長期間に及ぶ可能性のある移行中にリソースが浪費されることとなります。
- 複数のファイルシステム間での調整には手間がかかります。何十または何百ものファイルシステムを移行する場合、各移行にかかる時間はさまざまで、ファイルシステム全体にわたって停止時間をスケジュールする必要があります。

## 外部介入による移行

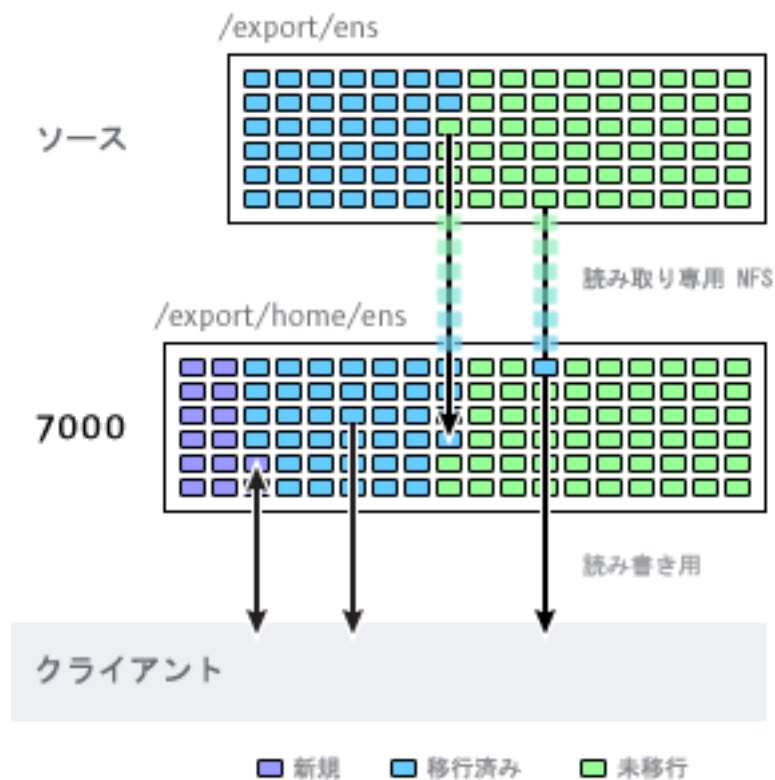
この方法では、アクティブなホスト X を取り、データを新しいホスト Y に移行する新しい ZFSSA M を挿入します。すべてのクライアントはただちに M を指すように更新され、データはバックグラウンドで自動的に移行されます。これにより、移行オプションの柔軟性が向上し (将来は停止時間なしに新しいサーバーに移行できるなど)、移行済みのデータに新しいサーバーを利用できますが、重大な欠点もあります。

- 移行 ZFSSA は新しい物理マシンを表し、関連コスト (初期投資、サポート費用、電力、および冷却) と追加の管理オーバーヘッドがかかります。
- 移行 ZFSSA は、システム内の新しい障害点を表します。
- 移行 ZFSSA は移行済みのデータに介入するため、多くの場合、永続的に追加の待機時間を発生させます。通常、これらの ZFSSA は所定の場所に置かれますが、別の停止時間帯をスケジュールしたり、移行 ZFSSA を廃止したりできます。



## シャドウ移行

図 14-1 シャドウ移行



シャドウ移行は介入を使用しますが、ZFSSA に組み込まれているため、個別の物理マシンは必要ありません。シェアが作成されると、それらはオプションで、ローカルに、または NFS を介して既存のディレクトリの「シャドウを作成」できます。このシナリオでは停止時間が一度スケジュールされ、その時間にソース ZFSSA X が読み取り専用モードになり、シャドウプロパティが設定されたシェアが作成され、Sun Storage 7000 ZFSSA でクライアントがその新しいシェアを指すように更新されます。その後、クライアントは読み取り/書き込みモードで ZFSSA にアクセスできます。

シャドウプロパティが設定されると、データはローカルでソース ZFSSA からバックグラウンドで透過的に移行されます。まだ移行されていないファイルのリクエストがクライアントから出された場合、ZFSSA はそのリクエストに回答する前にこのファイルをローカルサーバーに自動的に移行します。この操作では、一部のクライアントリクエストで初期遅延が発生することがありますが、ファイルが移行されると、すべてのアクセスが ZFSSA に対してローカル

になり、パフォーマンスが回復します。ファイルシステムの現在の作業セットは合計サイズに比べてかなり小さいことが多いため、この作業セットが移行されると、ソース上の元の合計サイズに関係なく、パフォーマンスへの影響は認められなくなります。

シャドウ移行の欠点はデータの移行が完了する前に確定が必要なことですが、このことはどの介入方法にも当てはまります。移行中は、データ部分は 2 つの場所に存在するため、バックアップがより複雑になり、スナップショットが不完全になったり、1 つのホストにしか存在しなかったりすることがあります。このため、2 つのホスト間の移行を最初に十分にテストして、アイデンティティ管理やアクセス制御が正しく設定されていることを確認することがきわめて重要です。データ移行全体をテストする必要はありませんが、新しいシステム上で、ワールドリーダブルでないファイルまたはディレクトリが正しく移行されていること、ACL (存在する場合) が保持されていること、およびアイデンティティが適切に表現されていることを検証するようにしてください。

シャドウ移行はファイルシステム内のディスク上のデータを使用して実装されるため、ストレージプールの外側にローカルに格納される外部データベースやデータはありません。クラスター内でプールがフェイルオーバーされた場合、または両方のシステムディスクが故障して新しいヘッドノードが必要になった場合、割り込みなしでシャドウ移行を続行するのに必要なすべてのデータはストレージプールで保持されます。

## シャドウ移行の動作

### シャドウソースの制限事項

- データを適切に移行するためには、ソースのファイルシステムまたはディレクトリが「読み取り専用である必要があります」。ファイルソースに加えられた変更が反映されるかどうかはタイミングによって異なり、ディレクトリ構造に対する変更は ZFSSA で回復不可能なエラーを発生させる可能性があります。
- シャドウ移行は NFS ソースからの移行のみをサポートします。NFSv4 シェアは最適な結果をもたらします。NFSv2 および NFSv3 移行も可能ですが、その過程で ACL が失われ、NFSv2 に対して大きすぎるファイルはそのプロトコルを使って移行できません。SMB ソースからの移行はサポートされません。
- LUN のシャドウ移行はサポートされません。

### 移行時のシャドウファイルシステムのセマンティクス

クライアントがまだ移行されていないファイルまたはディレクトリにアクセスする場合、次のような動作への影響が確認できます。

- ディレクトリの場合、クライアントリクエストはディレクトリ全体が移行されるまでブロックされます。ファイルの場合、ファイルのリクエストされている部分だけが移行されるため、複数のクライアントが同時にファイルのさまざまな部分を移行できます。
- シャドウファイルシステムでのファイルおよびディレクトリの名前変更、削除、または上書きは、移行プロセスに影響を及ぼすことなく任意に実行できます。
- ハードリンクを持つファイルの場合、移行が完了するまでハードリンク数がソースと一致しないことがあります。
- ファイル属性のほとんどはディレクトリが作成されるときに移行されますが、ディスク上のサイズ (UNIX stat 構造体の `st_nblocks`) はそのファイルに対する読み取りまたは書き込み操作が完了するまでわかりません。論理サイズは正しくなりますが、ファイルの内容が実際に移行されるまで、`du(1)` などのコマンドはサイズ 0 を報告します。
- ZFSSA をリブートした場合、移行はそれがもともと中止された場所を取り出します。データを再移行する必要はありませんが、場合によってはローカルファイルシステムの移行済みの部分をトラバースする必要があり、その割り込みのために移行の合計時間にいくらか影響が及ぶ可能性があります。
- データ移行では、ファイルに対して非公開の拡張属性を使用します。これらは通常、ファイルシステムのルートディレクトリ上で、またはスナップショットを通じてしか確認できません。SUNWshadow で始まる拡張属性の追加、変更、または削除は、移行プロセスに予測できない影響を及ぼし、不完全な状態または破壊された状態をもたらします。また、ファイルシステム全体の状態はファイルシステムのルートにある `.SUNWshadow` ディレクトリに格納されます。この内容に何らかの変更を加えると、同様の影響があります。
- ファイルシステムの移行が完了すると、警告が送信され、シャドウ属性が適切なメタデータとともに削除されます。その後、このファイルシステムは通常のファイルシステムと見分けがつかなくなります。
- NFSv4 自動クライアントマウント (「ミラーマウント」と呼ばれることもある) または入れ子のローカルマウントを使用すると、データを複数のファイルシステムから 1 つのファイルシステムに移行できます。

## ID と ACL の移行

ファイルのアイデンティティ情報 (ACL を含む) を適切に移行するには、次の規則を守る必要があります。

- 移行元と移行先の ZFSSA のネームサービス構成が同じである必要があります。
- 移行元と移行先の ZFSSA の NFSv4 mapid ドメインが同じである必要があります。

- 移行元で NFSv4 がサポートされている必要があります。NFSv3 も使用できますが、いくらかの情報の損失が発生します。基本的なアイデンティティ情報 (所有者とグループ) および POSIX アクセス権は保持されますが、ACL はすべて失われます。
- 移行元が root アクセス権を使って ZFSSA にエクスポートされる必要があります。

「nobody」によって所有されているファイルまたはディレクトリが見つかった場合は、ZFSSA のネームサービス設定が正しくなかったり、NFSv4 mapid ドメインが異なっていたりする可能性があります。クライアントが本来ならアクセスできるはずのファイルシステムをトラバースしているときに「アクセス権が拒否されました」エラーが発生した場合、もっとも起こりうる問題は root アクセス権を使った移行元のエクスポートに失敗することです。

## シャドウ移行の管理

### シャドウファイルシステムの作成

シャドウ移行ソースは、ファイルシステムを作成するときのみ設定できます。BUI では、ファイルシステムの作成ダイアログで使用できます。CLI では、shadow プロパティとして使用できます。このプロパティは、次のいずれかの書式を取ります。

- ローカル - file:///<path>
- NFS - nfs://<host>/<path>

BUI では、NFS マウントに対して別の書式 <host>:/<path> も使用できます。この書式は UNIX システムで使用される構文に一致します。BUI では、プルダウンメニューを使用すると、設定のプロトコル部分 (file:// または nfs://) も指定できます。ファイルシステムの作成時に、サーバーは、そのパスが存在し、マウントできることを検証します。

### バックグラウンド移行の管理

シェアが作成されると、インラインリクエストが処理されるだけでなく、移行がバックグラウンドで自動的に開始されます。この移行は、[240 ページの「シャドウ移行サービス」](#)によって制御されます。このタスク専用のスレッドの数を示すチューニング可能なグローバルプロパティが 1 つあります。スレッドの数を増やすと、追加のリソースが必要になりますが、そのおかげで並列性が向上します。

シャドウ移行サービスは無効にできますが、テスト目的の場合、またはアクティブなシャドウ移行によってシステムの一時的停止が必要になるまでシステムに負荷がかかっている場合のみ無効にしてください。シャドウ移行サービスを無効にすると、同期リクエストは引き続き必要に応じて移行されますが、バックグラウンド移行は行われません。このサービスが無効

になっていると、ファイルシステムのすべての内容が手動で読み取られても、シャドウ移行が完了することはありません。このサービスは常に有効にしておくことを強くお勧めします。

## 移行エラーの処理

シャドウ移行では、移行が完了する前にサーバーへの新しい書き込みを確定する必要があるため、移行をテストしエラーがないかどうかをモニターすることが非常に重要です。バックグラウンド移行中に発生したエラーは保持され、BUI でシャドウ移行ステータスの一部として表示されます。ほかの同期移行中に発生したエラーは追跡されませんが、影響を受けたファイルにバックグラウンドプロセスがアクセスするとエラーとして表示されます。ファイルごとに、リモートファイル名とそれぞれのエラーが保持されます。エラー件数の横にある情報アイコンをクリックすると、エラーの詳細なリストが表示されます。エラーリストはエラーが修正されても更新されず、移行が正常に完了することによってのみクリアされます。

シャドウ移行は、すべてのファイルが正常に移行されるまで完了しません。エラーがある場合は、バックグラウンド移行で移行が成功するまで継続的に再試行されます。これにより、管理者はエラー（アクセス権の問題など）を修正し、移行を完了させ、確実に成功させることができます。永続的なエラーのために移行を完了できない場合は、移行を取り消すと、移行できたデータとともにローカルファイルシステムをそのままにしておくことができます。これは最後の手段としてのみ使用するようしてください。移行は取り消すと再開できません。

## 移行の進行状況のモニタリング

シャドウ移行の進行状況をモニターすることは、この操作の実行コンテキストを考えると困難です。1つのファイルシステムは、ファイルシステムのすべてまたは一部のシャドウを作成することも、入れ子のマウントポイントを使って複数のファイルシステムのシャドウを作成することもできます。このため、ソースに関する統計をリクエストし、その統計が正しいことを確認する方法がありません。さらに、1つのファイルシステムの移行であっても、利用可能なサイズの計算に用いられる方法がシステム全体で一貫していません。たとえば、リモートファイルシステムで圧縮を使用する場合もあれば、メタデータのオーバーヘッドを含めたり含めなかったりする場合があります。このような理由から、特定の移行の正確な進行状況は表示できません。

ZFSSA は、正確であることが保証されている次の情報を提供します。

- これまでのローカルファイルシステムのローカルサイズ
- これまでにコピーされたデータの論理サイズ
- これまでにデータの移行に費やされた時間

これらの値は、BUI および CLI で、標準のファイルシステムプロパティとシャドウ移行ノード (UI パネル) 両方のプロパティを通じて利用できます。リモートファイルシステムのサ

イズがわかっている場合は、それを使って進行状況を予測できます。コピーされたデータのサイズは、ソースから移行される必要のあるブレンファイルの内容でのみ構成されます。ディレクトリ、メタデータ、および拡張属性は、この計算には含まれません。これまでに移行されたデータのサイズにはリモートで移行されたデータのみが含まれますが、バックグラウンド移行を再開すると、ファイルシステムの移行済みの部分がトラバースされることがあります。このため、このような初期ディレクトリの処理中は実行速度が大きく上昇し、ファイルシステムのまだ移行されていない部分に到達すると速度が低下する可能性があります。

進行状況を示す正確な指標はありませんが、ZFSSA は、ディレクトリツリーが比較的均一であるとの想定の下で残りのデータの予測を試みます。この予測は、データセットによっては、かなり正確になることもあればまったく役に立たないこともあるため、情報提供のためだけにのみ提供されています。たとえば、比較的浅いファイルシステムツリーがあるものの、最後に参照される単一ディレクトリに大量のデータが含まれているとします。このシナリオでは、移行がほとんど完了したように思われたあと、この新しいツリーの発見に伴って進行状況の割合がごくわずかな値にまで急激に低下します。反対に、そのような大きなディレクトリが最初に処理された場合、ほかのすべてのディレクトリにも同様に大量のデータが含まれていると想定される可能性があります。それらのディレクトリがほぼ空であることがわかると、予測される進行状況の割合が小さな値から 100% 近くまでただちに上昇します。進行状況を測定する最良の方法は、テスト移行を設定してそれを完了するまで実行し、その値を使用して同様のレイアウトやサイズのファイルシステムの進行状況を予測することです。

## 移行の取り消し

移行の取り消しは可能ですが、ソースが利用できなくなるような危機的な状況でのみ行うようにしてください。移行を取り消すと、再開することはできません。主な目的は、ソースに訂正不能なエラーがあるときに移行を完了できるようにすることです。少数のファイルまたはディレクトリを除いてファイルシステム全体の移行が完了し、これらのエラーを修正する方法がない（ソースが完全に壊れている）場合は、移行を取り消すことで、ローカルファイルシステムを「通常の」ファイルシステムのステータスに戻すことができます。

BUI で移行を取り消すには、問題になっているシェアの左側の列で、進行状況バーの横にある閉じるアイコンをクリックします。CLI では、ファイルシステムの下の shadow ノードに移動して、cancel コマンドを実行します。

## シャドウファイルシステムのスナップショット

シャドウファイルシステムのスナップショットの取得は可能ですが、どの項目のステータスをスナップショットに含めるかは任意に選択できます。まだ移行されていないファイルは存在しないため、実装の詳細 (SUNWshadow 拡張属性など) をスナップショットで確認で

ることがあります。このスナップショットを使用すると、最初の移行が開始されたあとで移行または変更された個々のファイルも復元できます。このため、移行が完了するまですべてのスナップショットをソースに保持しておくことをお勧めします。そうすることで、移行されていないファイルを必要に応じて引き続きソースから取得できます。保持ポリシーによっては、サービス要件を満たすためにソースでの保持期間を延長することが必要な場合があります。

スナップショットの取得は可能ですが、このスナップショットをロールバックしたり、クローンのソースにしたりすることはできません。これは、移行中のディスク上のデータの矛盾状態を反映しています。

## シャドウファイルシステムのバックアップ

シャドウデータをアクティブに移行しているファイルシステムは、ほかのファイルシステムと同様に NDMP を使用してバックアップできます。シャドウ設定はバックアップストリームで保持されますが、ファイルシステムの完全な復元が行われ、シェアが存在しなくなった場合にのみ復元されます。そのようなバックアップストリームから個々のファイルを復元したり、既存のファイルシステムに復元したりすると、矛盾した状態やデータの破壊が発生する可能性があります。ファイルシステム全体の復元中には、ファイルシステムが (部分的な復元による通常の矛盾以上に) 矛盾した状態となり、シャドウ移行はアクティブになりません。復元が完了したときにのみシャドウ設定が復元されます。シャドウソースが存在しなくなったかまたは移動した場合、管理者はエラーがないかどうかを監視し、必要に応じてエラーを修正できます。

## シャドウファイルシステムのレプリケート

シャドウデータをアクティブに移行しているファイルシステムは、通常のメカニズムを使用してレプリケートできますが、移行されたデータのみがデータストリームに送信されます。このため、リモート側には部分的なデータだけが含まれ、これらのデータは矛盾した状態を示す場合があります。シャドウ設定はレプリケーションストリームとともに送信されるため、リモートターゲットがフェイルオーバーされると、同じシャドウ設定が保持されます。NDMP バックアップストリームの復元と同様に、この設定はリモートターゲットのコンテキストでは正しくないことがあります。ターゲットのフェイルオーバー後、管理者はエラーがないかどうかを監視し、必要に応じてシャドウ設定を新しい環境に合わせて修正できます。

## シャドウ移行の分析

シェアごとの標準的なモニタリングのほかに、『[Oracle ZFS Storage Appliance Analytics ガイド](#)』の「[Analytics](#)」を使用してシステム全体のシャドウ移行をモニターす

することもできます。シャドウ移行の分析は「データ移動」カテゴリの下で利用できます。次の 2 つの基本的な統計を利用できます。

## シャドウ移行リクエスト

この統計は、キャッシュされておらず、ファイルシステムのローカルにあることがわかっているファイルまたはディレクトリに対するリクエストを追跡します。これは移行が済んだものと移行が済んでいないものの両方のファイルおよびディレクトリを考慮に入れ、シャドウ移行の一部として発生する待機時間を追跡したり、バックグラウンド移行の進捗状況を追跡するために使用できます。この統計は、ファイル、シェア、プロジェクト、または待機時間ごとに分類できます。現在は同期と非同期の両方の (バックグラウンド) 移行が含まれるため、クライアントに表示される待機時間だけを表示することはできません。

## シャドウ移行バイト数

この統計は、ファイルまたはディレクトリの内容の移行の一部として転送されるバイト数を追跡します。これはメタデータには当てはまりません (拡張属性、ACL など)。これは転送されるデータの概算を示しますが、ソースデータセットに大量のメタデータがある場合、帯域幅が不釣り合いなほど小さくなります。完全な帯域幅は、ネットワーク分析を表示することによって観察できます。この統計は、ローカルファイル名、シェア、またはプロジェクトごとに分類できます。

## シャドウ移行の操作

この統計は、ソースファイルシステムに到達する必要がある操作を追跡します。この統計を使用すると、シャドウ移行のソースからのリクエストの待機時間を追跡できます。この統計は、ファイル、シェア、プロジェクト、または待機時間ごとに分類できます。

## ローカルファイルシステムの移行

リモートソースからデータを移行するという主な目的のほかに、同じメカニズムを使用してローカルファイルシステムから ZFSSA 上の別のファイルシステムにデータを移行することもできます。これは、ファイルシステムの圧縮バージョンを作成したり、ファイルシステムのレコードを作成したあとでレコードサイズを変更したりするなど、本来なら変更できない設定を変更するために使用できます。このモデルでは、古いシェア (またはシェア内のサブディレクトリ) は読み取り専用になるか、または使用されなくなり、file プロトコルを使用してシャドウプロパティが設定された新しいシェアが作成されます。クライアントがこの新しいシェアにアクセスすると、データがその新しいシェアの設定を使用して書き込まれます。



## シャドウ移行のタスク

完全な移行を試みる前に、移行をテストして、ZFSSA に適切なアクセス権があることやセキュリティ属性が正しく変換されることを確認することが重要です。基本設定が機能していることを確認できたら、最終移行用にシェアを設定できます。

### ▼ シャドウ移行の可能性のテスト

1. ZFSSA がシェアへの root アクセス権を持つようにソースを構成します。通常、このためには NFS ホストベースの例外を追加したり、匿名ユーザーのマッピングを設定したりする必要があります (後者はセキュリティ面でより重要な意味があります)。
2. シャドウ属性を CLI で「nfs://<host>/<snapshotpath>」に設定したり、BUI で「<host>/<snapshotpath>」に設定したりして (プロトコルには「NFS」を選択)、ローカルファイルシステム上にシェアを作成します。スナップショットは、ソースの読み取り専用コピーにするようにしてください。使用できるスナップショットがない場合、読み取り/書き込み可能なソースを使用できますが、未定義のエラーが発生する可能性があります。
3. ファイル構造をトラバースして、ファイルの内容およびアイデンティティマッピングが正しく保持されていることを検証します。
4. データソースが (スナップショットと同様に) 読み取り専用の場合、移行を完了させて、転送中にエラーが発生しなかったことを確認します。

### ▼ アクティブな NFS サーバーからのデータの移行

1. クライアントを休止させて、新しいサーバーを指すように再構成するための停止時間をスケジュールします。
2. ZFSSA がシェアへの root アクセス権を持つようにソースを構成します。通常、このためには NFS ホストベースの例外を追加したり、匿名ユーザーのマッピングを設定したりする必要があります (後者はセキュリティ面でより重要な意味があります)。
3. ソースを読み取り専用になるように構成します。この手順は技術的にはオプションですが、移行の進行中に誤った構成のクライアントがソースへの書き込みを実行できない場合、適合性を保証するのが非常に簡単になります。

4. シャドウ属性を CLI で「nfs://<host>/<path>」に設定したり、BUI で「<host>/<path>」に設定したりして (プロトコルには「NFS」を選択)、ローカルファイルシステム上にシェアを作成します。
5. クライアントが SS7000 上のローカルシェアを指すように再構成します。

この時点で、シャドウ移行がバックグラウンドで実行され、クライアントリクエストが必要に応じて処理されます。上で説明したように、進行状況を監視できます。CLI をスクリプト化することによって、スケジュールされた 1 回の停止時間中に複数のシェアを作成できます。

# ◆◆◆ 第 15 章

# 15

## CLI のスクリプト化

---

CLI は、繰り返しタスクを実行するための強力なスクリプト環境を提供するように設計されています。

### アクセスの自動化

427 ページの「コマンドのバッチ処理」または428 ページの「コマンドのスクリプト化」(あるいはなんらかの組み合わせ)を使用できますが、どの場合でも自動化インフラストラクチャーにはこのアプライアンスへの自動アクセスが必要になります。そのためには、第7章「ユーザー構成」、138 ページの「ユーザー承認」、および 148 ページの「CLI を使用した SSH 公開鍵の設定」を行う必要があります。

### コマンドのバッチ処理

もっとも単純なスクリプト化のメカニズムは、アプライアンスのシェルコマンドをバッチ処理することです。たとえば、プロジェクト「myproj」とファイルシステム「myfs」内に「newsnap」と呼ばれるスナップショットを自動的に作成するには、ファイル内に次のコマンドを記述します。

```
shares
select myproj
select myfs
snapshots snapshot newsnap
```

次に、このファイルを標準入力としてリダイレクトして、アプライアンスに `ssh` を実行します。

```
% ssh root@dory < myfile.txt
```

多くのシェルでは、「ヒアドキュメント」を使用してこれを省略できます。この場合は、トークンまでの入力が標準入力に送信されます。ヒアドキュメントを使用した場合の上の例を次に示します。

```
% '''ssh root@dory << EOF
shares
select myproj
select myfs
snapshots snapshot newsnap
```

```
EOF'''
```

このメカニズムは、もっとも単純な種類の自動化には十分であり、またクライアント上のより高レベルのシェルスクリプト言語のプログラムロジックでラップされている場合は十分である可能性があります。一般にはかなり不十分な方法です。

## コマンドのスクリプト化

もっとも単純な操作にはコマンドのバッチ処理で十分ですが、プログラムロジックでラップすることは面倒になる場合があります。たとえば、すべてのシェアの領域使用状況に関する情報を取得する場合は、特定のコマンドの出力を解析したクライアント上のより高レベルの言語でラップされた、CLI のさまざまな呼び出しを行う必要があります。これにより、低速で、脆弱な自動化インフラストラクチャーがもたらされます。より高速で、もっとも堅牢な自動化を可能にするために、このアプライアンスには ECMAScript 3 に基づいた、機能の豊富なスクリプト環境が用意されています。ECMAScript のチュートリアルはこのドキュメントの範囲を超えていますが、ECMAScript は C に似た構文を持つ動的に型付けされる言語であり、次のことが可能になります。

- 条件付きコードフロー (if/else)
- 繰り返しのコードフロー (while, for など)
- 優れたオブジェクト型および配列型による構造的なデータや配列データの操作
- Perl に似た正規表現および文字列操作 (split(), join() など)
- 例外
- クロージャージャーなどの高度な言語機能

## スクリプト環境

CLI で、script コマンドを使用してスクリプト環境に入ります。

```
dory:> script  
("." to run)>
```

スクリプト環境のプロンプトが表示されたら、スクリプトを入力し、最後に 1 行に「.」だけを入力してそのスクリプトを実行できます。

```
dory:> script  
("." to run)> for (i = 10; i > 0; i--)  
("." to run)>   printf("%d... ", i);  
("." to run)> printf("Blastoff!\n");  
("." to run)> .  
10... 9... 8... 7... 6... 5... 4... 3... 2... 1... Blastoff!
```

スクリプトが 1 行の場合は、単にそのスクリプトを script コマンドへの引数として指定できます。これは、スクリプトを調べるための簡単な方法になります。

```
dory:> script print("It is now " + new Date())
It is now Tue Oct 14 2009 05:33:01 GMT+0000 (UTC)
```

## システムとの対話

当然ながら、システムと詳細に対話できないかぎり、スクリプトはほとんど役立ちません。スクリプトがシステムと対話できるようにするための組み込み関数がいくつか存在します。

表 15-1 システムとの対話をサポートするための組み込み関数

関数	説明
get	指定されたプロパティの値を取得します。この関数がネイティブ形式の値を返すことに注意してください。たとえば、日付は Date オブジェクトとして返されます。
list	現在のコンテキストの動的な子に対応するトークンの配列を返します。
run	指定されたコマンドをシェルで実行し、すべての出力を文字列として返します。出力に複数の行が含まれている場合、返される文字列には、埋め込まれた復帰改行が含まれることに注意してください。
props	現在のノードのプロパティ名の配列を返します。
set	2 つの文字列引数を受け取り、指定されたプロパティを指定された値に設定します。
choices	値のセットが知られていて列挙可能であるすべてのプロパティの有効なプロパティ値の配列を返します。

## run 関数

スクリプトがより大規模なシステムと対話するためのもっとも単純な方法は、「run」関数を使用することです。この関数は、実行するコマンドを受け取り、そのコマンドの出力を文字列として返します。次に例を示します。

```
dory:> configuration version script dump(run('get boot_time'))
'
      boot_time = 2009-10-12 07:02:17\n'
```

組み込みの dump 関数は、埋め込まれた復帰改行を展開することなく引数をダンプ出力します。この出力を分割するために、ECMAScript の文字列処理機能を使用できます。たとえば、空白に基づいて上の出力を分割すると、次のようになります。

```
dory:> configuration version script dump(run('get boot_time').split(/\s+/))
[&#39;', 'boot_time', '=', '2009-10-12', '07:02:17', &#39;']
```

## get 関数

run 関数は十分に強力であるため、システムに関する情報を取得するために、もっぱら出力の解析に頼りたくなることがあります。しかしこれには、将来変更されるかどうかわからない、人間が読める形式の出力をスクリプトがいつまでも解析し続けるという決定的な欠点があります。システムに関する情報をより安定して収集するには、組み込みの「get」関数を使用します。boot\_time プロパティーの場合は、この関数によって文字列ではなく、ECMAScript の Date オブジェクトが返されるため、プログラムからプロパティー値を操作できます。たとえば、boot\_time プロパティーを現在の時間と組み合わせて使用して、ブート時からの時間を特定することもできます。

```
script
  run('configuration version');
  now = new Date();
  uptime = (now.valueOf() - get('boot_time').valueOf()) / 1000;
  printf('up %d day%s, %d hour%s, %d minute%s, %d second%s\n',
    d = uptime / 86400, d < 1 || d >= 2 ? 's' : '',
    h = (uptime / 3600) % 24, h < 1 || h >= 2 ? 's': '',
    m = (uptime / 60) % 60, m < 1 || m >= 2 ? 's': '',
    s = uptime % 60, s < 1 || s >= 2 ? 's': '');
```

上のスクリプトを「uptime.aksh」として保存したとすると、それを次の方法で実行できます。

```
% ssh root@dory < uptime.aksh
Pseudo-terminal will not be allocated because stdin is not a terminal.
Password:
up 2 days, 10 hours, 47 minutes, 48 seconds
```

仮想端末の割り当てに関するメッセージは、ssh クライアントによるものです。このメッセージが示す問題は、ssh に「-T」オプションを指定することによって対処できます。

## list 関数

動的な子が存在するコンテキストでは、これらの子をプログラムから繰り返し処理すると非常に有効です。これを行うには、動的な子の配列を返す list 関数を使用します。たとえば、すべてのプロジェクト内のすべてのシェアを繰り返し処理し、消費されている容量と空きスペースを出力するスクリプトを次に示します。

```
script
  run('shares');
  projects = list();

  for (i = 0; i < projects.length; i++) {
    run('select ' + projects[i]);
    shares = list();

    for (j = 0; j < shares.length; j++) {
      run('select ' + shares[j]);
      printf("%s/%s %1.64g %1.64g\n", projects[i], shares[j],
        get('space_data'), get('space_available'));
      run('cd ..');
    }
  }
}
```

```

    }
    run('cd ..');
}

```

このスク립トが「space.aksh」という名前のファイルに保存されたと仮定して、スク립トを実行した出力を次に示します。

```

% ssh root@koi < space.aksh
Password:
admin/accounts 18432 266617007104
admin/exports 18432 266617007104
admin/primary 18432 266617007104
admin/traffic 18432 266617007104
admin/workflow 18432 266617007104
aleventhal/hw_eng 18432 266617007104
bcantrill/analytx 1073964032 266617007104
bgregg/dashbd 18432 266617007104
bgregg/filesys01 26112 107374156288
bpijewski/access_ctrl 18432 266617007104
...

```

この結果の「きれいに出力された」バージョン (ただし、プログラムの処理はより困難になる) がほしい場合は、get コマンドの出力を直接解析することもできます。

```

script
    run('shares');
    projects = list();

    printf('%-40s %-10s %-10s\n', 'SHARE', 'USED', 'AVAILABLE');

    for (i = 0; i < projects.length; i++) {
        run('select ' + projects[i]);
        shares = list();

        for (j = 0; j < shares.length; j++) {
            run('select ' + shares[j]);

            share = projects[i] + '/' + shares[j];
            used = run('get space_data').split(/\s+/)[3];
            avail = run('get space_available').split(/\s+/)[3];

            printf('%-40s %-10s %-10s\n', share, used, avail);
            run('cd ..');
        }
    }

    run('cd ..');
}

```

このスク립トが「prettyspace.aksh」という名前であると仮定して、この新しいスク립トを実行した出力の一部を次に示します。

```

% ssh root@koi < prettyspace.aksh
Password:
SHARE                                USED          AVAILABLE
admin/accounts                       18K           248G
admin/exports                       18K           248G
admin/primary                       18K           248G
admin/traffic                       18K           248G
admin/workflow                       18K           248G

```

aleventhal/hw_eng	18K	248G
bcantrill/analytx	1.00G	248G
bgregg/dashbd	18K	248G
bgregg/filesys01	25.5K	100G
bpijewski/access_ctrl	18K	248G
...		

## children 関数

静的な子が存在するコンテキストであっても、これらの子をプログラムから繰り返し処理すると有効です。これを行うには、静的な子の配列を返す children 関数を使用します。たとえば、すべてのサービスを繰り返し処理し、そのサービスのステータスを出力するスクリプトを次に示します。

```
configuration services
script
    var svcs = children();
    for (var i = 0; i < svcs.length; ++i) {
        run(svcs[i]);
        try {
            printf("%-10s %s\n", svcs[i], get('<status>'));
        } catch (err) { }
        run("done");
    }
}
```

このスクリプトが「svcinfo.aksh」という名前のファイルに保存されたと仮定して、スクリプトを実行した出力を次に示します。

```
% ssh root@koi < space.aksh
Password:
cifs      disabled
dns       online
ftp       disabled
http      disabled
identity  online
idmap     online
ipmp      online
iscsi     online
ldap      disabled
ndmp      online
nfs       online
nis       online
ntp       online
scrk      online
sftp      disabled
smtp      online
snmp      disabled
ssh       online
tags      online
vscan     disabled
```

## choices 関数

choices 関数は、値のセットが知られていて列挙可能であるすべてのプロパティの有効なプロパティ値の配列を返します。たとえば、次のスクリプトは、choices 関数を使用して



シェアード上のすべてのプールのリストを取得してから、すべてのプールを反復処理してプロジェクトのリストを作成し、使用可能な領域とともにシェアします。

```
fmt = '%-40s %-15s %-15s\n';
printf(fmt, 'SHARE', 'USED', 'AVAILABLE');
run('cd /');
run('shares');
pools = choices('pool');
for (p = 0; p < pools.length; p++) {
  set('pool', pools[p]);
  projects = list();
  for (i = 0; i < projects.length; i++) {
    run('select ' + projects[i]);
    shares = list();
    for (j = 0; j < shares.length; j++) {
      run('select ' + shares[j]);
      share = pools[p] + ':' + projects[i] + '/' + shares[j];
      printf(fmt, share, get('space_data'),
            get('space_available'));
      run('cd ..');
    }
    run('cd ..');
  }
}
```

これは、スクリプトを実行した出力です。

SHARE	USED	AVAILABLE
pond:projectA/fs1	31744	566196178944
pond:projectA/fs2	31744	566196178944
pond:projectB/lun1	21474836480	587670999040
puddle:deptA/share1	238475	467539219283
puddle:deptB/share1	129564	467539219283
puddle:deptB/share2	19283747	467539219283

## 出力の生成

システムの状態を報告するには、出力を生成する必要があります。出力を生成するために、スクリプトで使用可能な組み込み関数がいくつか存在します。

表 15-2 出力を生成するための組み込み関数

関数	説明
dump	埋め込まれた復帰改行を展開することなく、指定された引数を端末にダンプします。オブジェクトは、JSON に似た形式で表示されます。デバッグに有効です。
print	指定されたオブジェクトを文字列として出力し、そのあとに復帰改行を出力します。そのオブジェクトに <code>to String</code> メソッドがない場合は、不透明に出力されません。
printf	C の <code>printf(3C)</code> と同様に、指定された引数を指定された書式設定文字列に従って出力します。

## エラーの処理

エラーが生成されると、例外がスローされます。この例外は一般に、次のメンバーを含むオブジェクトです。

- `code` - エラーに関連付けられた数値コード
- `message` - エラーに関連付けられた、人間が読める形式のメッセージ

例外はキャッチして処理できます。または、スクリプト環境からスローされる可能性があります。スクリプト環境にキャッチされていない例外がある場合は、CLI にその詳細が表示されます。次に例を示します。

```
dory:> script run('not a cmd')
error: uncaught error exception (code EAKSH_BADCMD) in script: invalid command
      "not a cmd" (encountered while attempting to run command "not a cmd")
```

例外をキャッチしてダンプすることによって、例外に関するより詳細な情報を表示できます。

```
dory:> script try { run('not a cmd') } catch (err) { dump(err); }
{
  toString: <function>,
  code: 10004,
  message: 'invalid command "not a cmd" (encountered while attempting to
           run command "not a cmd")'
}
```

また、これにより、豊富なエラー処理も可能になります。次に例を示します。

```
#!/usr/bin/ksh -p

ssh -T root@dory <<EOF
script
  try {
    run('shares select default select $1');
  } catch (err) {
    if (err.code == EAKSH_ENTITY_BADSELECT) {
      printf('error: "$1" is not a share in the ' +
            'default project\n');
      exit(1);
    }

    throw (err);
  }

  printf("default/$1: compression is %s\n", get('compression'));
  exit(0);
EOF
```

このスクリプトが「share.ksh」という名前であり、無効なシェア名で実行された場合は、次のような豊富なエラーメッセージが生成されます。

```
% ksh ./share.ksh bogus
error: "bogus" is not a share in the default project
```

# ◆◆◆ 第 16 章 16

## 保守のワークフロー

---

ワークフローとは、ZFSSA にアップロードされ、その ZFSSA 自体で管理される [第15章「CLI のスクリプト化」](#) のことです。ワークフローは、ブラウザインタフェースまたはコマンド行インタフェースのどちらからも優れた方法でパラメータ化し、実行できます。ワークフローはまた、オプションで [第9章「警告の構成」](#) として実行したり、指定された時間に実行したりすることもできます。そのため、ワークフローを使用すると、特定のポリシーや手順を取り込むような方法で ZFSSA を拡張できます。また、たとえば、特定の組織またはアプリケーションのベストプラクティスを正式にエンコードすることもできます。

## ワークフローの使用

ワークフローは、1 つの大域変数 `workflow` が含まれている、有効な ECMAScript ファイルに組み込まれています。これは、少なくとも次の 3 つのメンバーを含む必要のあるオブジェクトです。

表 16-1 オブジェクトの必須メンバー

必須メンバー	タイプ	説明
<code>name</code>	文字列	ワークフローの名前
<code>description</code>	文字列	ワークフローの説明
<code>execute</code>	関数	ワークフローを実行する関数

次に、標準的な簡単なワークフローを示します。

```
var workflow = {
  name: 'Hello world',
  description: 'Bids a greeting to the world',
  execute: function () { return ('hello world!') }
};
```

このワークフローをアップロードすると「Hello world」という名前の新しいワークフローが作成され、このワークフローを実行すると「hello world!」が出力されます。

## ワークフローの実行コンテキスト

ワークフローは、ZFSSA のシェル内で非同期に (デフォルトでは) そのワークフローを実行しているユーザーとして実行されます。そのため、ワークフローは [第15章「CLI のスクリプト化」](#)を自由に使用でき、さらに ZFSSA のシェルのほかの任意のインスタンスと同じように ZFSSA と対話できます。つまり、ワークフローはコマンドを実行したり、出力を解析したり、状態を変更したりできます。run 関数を使用して現在の CPU 使用率を返す、より複雑な例を次に示します。

```
var workflow = {
  name: 'CPU utilization',
  description: 'Displays the current CPU utilization',
  execute: function () {
    run('analytics datasets select name=cpu.utilization');
    cpu = run('csv 1').split('\n')[1].split(',');
    return ('At ' + cpu[0] + ', utilization is ' + cpu[1] + '%');
  }
};
```

## ワークフローのパラメータ

入力に従って動作しないワークフローの範囲は限られます。有効に使用するには、多くのワークフローをパラメータ化する必要があります。これを行うには、グローバルな workflow オブジェクトに parameters メンバーを追加します。この parameters メンバーがさらに、パラメータごとにメンバーを持つことが想定されるオブジェクトになっています。各 parameters メンバーには、次のメンバーが必要です。

表 16-2 ワークフローパラメータの必須メンバー

必須メンバー	タイプ	説明
label	文字列	ワークフローパラメータの入力を飾るためのラベル
type	文字列	ワークフローパラメータのタイプ

type メンバーは、次の型のいずれかに設定されている必要があります。

表 16-3 メンバーの型名

型名	説明
Boolean	ブール値
ChooseOne	いくつかの指定された値のいずれか

型名	説明
EmailAddress	電子メールアドレス
File	ZFSSA に転送されるファイル
Host	有効なホスト (名前またはドット区切り 10 進数のどちらか)
HostName	有効なホスト名
HostPort	有効な、使用可能なポート
Integer	整数
NetAddress	ネットワークアドレス
NodeName	ネットワークノードの名前
NonNegativeInteger	0 以上の整数
Number	任意の数値 (浮動小数点を含む)
Password	パスワード
Permissions	POSIX アクセス権
Port	ポート番号
Size	サイズ
String	文字列
StringList	文字列のリスト

指定された型に基づいて、ワークフローの実行時に適切な入力フォームが生成されます。たとえば、ビジネス単位の名前 (プロジェクトとして使用される) と、シェアの名前 (シェア名として使用される) の 2 つのパラメータを持つワークフローを次に示します。

```
var workflow = {
  name: 'New share',
  description: 'Creates a new share in a business unit',
  parameters: {
    name: {
      label: 'Name of new share',
      type: 'String'
    },
    unit: {
      label: 'Business unit',
      type: 'String'
    }
  },
  execute: function (params) {
    run('shares select ' + params.unit);
    run('filesystem ' + params.name);
    run('commit');
  }
}
```

```

        return ('Created new share "' + params.name + '"');
    }
};

```

このワークフローをアップロードして実行すると、シェアとビジネス単位の名前を入力するためのダイアログボックスが表示されます。シェアが作成されると、それを示すメッセージが生成されます。

## 制約付きのパラメータ

パラメータによっては、任意の文字列は許可せず、入力を少数の候補のいずれかに制限する場合があります。これらのパラメータは型 `ChooseOne` として指定すべきであり、そのパラメータを含むオブジェクトには次の 2 つの追加のメンバーが必要です。

表 16-4 制約付きのパラメータの必須メンバー

必須メンバー	タイプ	説明
<code>options</code>	配列	有効なオプションを指定する文字列の配列
<code>optionlabels</code>	配列	<code>options</code> で指定されたオプションに関連付けられているラベルを指定する文字列の配列

`ChooseOne` パラメータ型を使用すると、前の例を、ビジネス単位が少数の定義済みの値のいずれかに制限されるように拡張できます。

```

var workflow = {
  name: 'Create share',
  description: 'Creates a new share in a business unit',
  parameters: {
    name: {
      label: 'Name of new share',
      type: 'String'
    },
    unit: {
      label: 'Business unit',
      type: 'ChooseOne',
      options: [ 'development', 'finance', 'qa', 'sales' ],
      optionlabels: [ 'Development', 'Finance',
        'Quality Assurance', 'Sales/Administrative' ],
    }
  },
  execute: function (params) {
    run('shares select ' + params.unit);
    run('filesystem ' + params.name);
    run('commit');
    return ('Created new share "' + params.name + '"');
  }
};

```

このワークフローが実行されたときに、unit パラメータは手動では入力されず、可能性のあるオプションの指定されたリストから選択されます。

## オプションのパラメータ

一部のパラメータは、ワークフローの実行を許可するためにそのパラメータがいずれかの値に設定されていることが UI によって義務付けられないという点で、*オプション*とみなすことができます。このようなパラメータは、parameters メンバーの optional フィールドによって示されます。

表 16-5 オプションのパラメータの必須メンバー

オプションメンバー	タイプ	説明
optional	ブール値	true に設定されている場合は、このパラメータを設定する必要がないことを示します。このパラメータの値が指定されてなくても、UI によってワークフローの実行が許可されます。

パラメータがオプションであり、未設定の場合、execute 関数に渡される parameters オブジェクト内のそのメンバーは undefined に設定されます。

## ワークフローのエラー処理

ワークフローの実行中にエラーが検出された場合は、例外がスローされます。例外がワークフロー自体によってキャッチされない場合 (または、ワークフローが別の方法ではキャッチされない例外をスローした場合)、ワークフローは失敗し、その例外に関連した情報がユーザーに表示されます。エラーを正しく処理するには、例外をキャッチして処理すべきです。たとえば、前の例では、存在しないプロジェクト内にシェアを作成しようとすると、キャッチされない例外が発生します。この例は、問題のあるエラーをキャッチし、プロジェクトが存在しない場合はそのプロジェクトを作成するように変更できます。

```
var workflow = {
  name: 'Create share',
  description: 'Creates a new share in a business unit',
  parameters: {
    name: {
      label: 'Name of new share',
      type: 'String'
    },
    unit: {
      label: 'Business unit',
```

```
    type: 'ChooseOne',
    options: [ 'development', 'finance', 'qa', 'sales' ],
    optionlabels: [ 'Development', 'Finance',
        'Quality Assurance', 'Sales/Administrative' ],
  }
},
execute: function (params) {
  try {
    run('shares select ' + params.unit);
  } catch (err) {
    if (err.code != EAKSH_ENTITY_BADSELECT)
      throw (err);

    /*
     * We haven't yet created a project that corresponds to
     * this business unit; create it now.
     */
    run('shares project ' + params.unit);
    run('commit');
    run('shares select ' + params.unit);
  }

  run('filesystem ' + params.name);
  run('commit');
  return ('Created new share "' + params.name + '"');
}
};
```

## ワークフローの入力の検証

ワークフローは、そのワークフローのパラメータがメンバーとして含まれているオブジェクトをパラメータとして受け取る `validate` メンバーを追加することによって、必要に応じてその入力を検証できます。`validate` 関数からは、各メンバーの名前が検証に失敗したパラメータの名前であり、各メンバーの値がユーザーに表示される検証失敗メッセージであるオブジェクトが返されます。前の例を拡張して、ユーザーがすでに存在するシェアを作成しようとした場合に明確なエラーを表示するには、次のようになります。

```
var workflow = {
  name: 'Create share',
  description: 'Creates a new share in a business unit',
  parameters: {
    name: {
      label: 'Name of new share',
      type: 'String'
    },
  },
  unit: {
    label: 'Business unit',
    type: 'ChooseOne',
    options: [ 'development', 'finance', 'qa', 'sales' ],
    optionlabels: [ 'Development', 'Finance',
        'Quality Assurance', 'Sales/Administrative' ],
  }
},
validate: function (params) {
  try {
    run('shares select ' + params.unit);
```



```

    run('select ' + params.name);
  } catch (err) {
    if (err.code == EAKSH_ENTITY_BADSELECT)
      return;
  }

  return ({ name: 'share already exists' });
},
execute: function (params) {
  try {
    run('shares select ' + params.unit);
  } catch (err) {
    if (err.code != EAKSH_ENTITY_BADSELECT)
      throw (err);

    /*
     * We haven't yet created a project that corresponds to
     * this business unit; create it now.
     */
    run('shares project ' + params.unit);
    set('mountpoint', '/export/' + params.unit);
    run('commit');
    run('shares select ' + params.unit);
  }

  run('filesystem ' + params.name);
  run('commit');
  return ('Created new share "' + params.name + '"');
}
};

```

## ワークフローの実行の監査

ワークフローは、`audit` 関数を呼び出すことによって監査レコードを発行できます。`audit` 関数の唯一の引数は、監査ログに記録される文字列です。

## ワークフローの実行の報告

実行にある程度の時間が必要な複雑なワークフローの場合は、そのワークフローを実行しているユーザーに明確な進捗状況を表示すると有効です。ワークフローの実行をこの方法で報告できるようにするには、`execute` メンバーが手順の配列を返すべきです。各配列要素には、次のメンバーが必要です。

表 16-6 実行の報告に必要なメンバー

必須メンバー	タイプ	説明
<code>step</code>	文字列	実行手順の名前を示す文字列
<code>execute</code>	関数	ワークフローの手順を実行する関数

全体としてのワークフロー上の `execute` 関数と同様に、各手順の `execute` メンバーは、そのワークフローへのパラメータを含むオブジェクトを引数として受け取ります。例として、3 つの手順で新しいプロジェクト、シェア、および監査レコードを作成するワークフローを次に示します。

```
var steps = [ {
  step: 'Checking for associated project',
  execute: function (params) {
    try {
      run('shares select ' + params.unit);
    } catch (err) {
      if (err.code !== EAKSH_ENTITY_BADSELECT)
        throw (err);

      /*
       * We haven't yet created a project that corresponds to
       * this business unit; create it now.
       */
      run('shares project ' + params.unit);
      set('mountpoint', '/export/' + params.unit);
      run('commit');
      run('shares select ' + params.unit);
    }
  }, {
  step: 'Creating share',
  execute: function (params) {
    run('filesystem ' + params.name);
    run('commit');
  }, {
  step: 'Creating audit record',
  execute: function (params) {
    audit('created "' + params.name + '" in "' + params.unit);
  }
} ];

var workflow = {
  name: 'Create share',
  description: 'Creates a new share in a business unit',
  parameters: {
    name: {
      label: 'Name of new share',
      type: 'String'
    },
  },
  unit: {
    label: 'Business unit',
    type: 'ChooseOne',
    options: [ 'development', 'finance', 'qa', 'sales' ],
    optionlabels: [ 'Development', 'Finance',
      'Quality Assurance', 'Sales/Administrative' ],
  },
},
validate: function (params) {
  try {
    run('shares select ' + params.unit);
    run('select ' + params.name);
  } catch (err) {
    if (err.code === EAKSH_ENTITY_BADSELECT)
      return;
  }
}
```

```

    return ({ name: 'share already exists' });
  },
  execute: function (params) { return (steps); }
};

```

## バージョン管理

ワークフローに関するバージョン管理には 2 つの側面があります。1 つめはワークフローが依存する ZFSSA ソフトウェアのバージョンの表現であり、2 つめはワークフロー自体のバージョンの表現です。バージョン管理は、ワークフローへの次の 2 つのオプションメンバーによって表されます。

表 16-7 バージョン管理用のオプションのメンバー

オプションメンバー	タイプ	説明
required	文字列	このワークフローを実行するために必要な ZFSSA ソフトウェアの最小のバージョン。最小の年、月、日、ビルド、およびブランチを含みます。
version	文字列	ドット区切り 10 進数 (major.minor.micro) 形式での、このワークフローのバージョン。

## アプライアンスのバージョン管理

ZFSSA ソフトウェアの最小限必要なバージョンを表すには、使用しているワークフローにオプションの `required` フィールドを追加します。この ZFSSA は、ソフトウェアがビルドされた年、月、日のあとにビルド番号とブランチ番号を付加した「`year.month.day.build-branch`」によってバージョン管理されます。たとえば、「`2009.04.10,12-0`」は、2009 年 4 月 10 日に最初にビルドされたソフトウェアの 12 番目のビルドです。現在の ZFSSA キットソフトウェアのバージョンを取得するには、CLI コマンド「`configuration version get version`」を実行するか、または BUI の『[Oracle ZFS Storage Appliance 顧客サービス マニュアル](#)』の「システム」にある「バージョン」フィールドを参照します。`required` フィールドを使用した例を次に示します。

```

var workflow = {
  name: 'Configure FC',
  description: 'Configures fibre channel target groups',
  required: '2009.12.25,1-0',
  ...

```

あるワークフローに、ZFSSA 上にロードされているバージョンより新しいバージョンのソフトウェアが必要な場合は、そのワークフローをアップロードしようとする時失敗し、不一致を説明したメッセージが表示されます。

## ワークフローのバージョン管理

ZFSSA ソフトウェアの必要なバージョンの指定に加えて、ワークフロー自体も version フィールドでバージョン管理できます。この文字列は、ワークフローのバージョンのメジャー、マイナー、およびマイクロ番号を示し、同じワークフローの複数のバージョンがマシン上に存在できるようにします。ワークフローをアップロードするとき、同じワークフローの互換性がある、古いバージョンはすべて削除されます。ワークフローは、メジャー番号が同じ場合に互換性があると見なされ、バージョン番号が低い場合に古いと見なされます。そのため、バージョン「2.1」のワークフローをアップロードすると、バージョン「2.0」(またはバージョン「2.0.1」) の同じワークフローが削除されますが、「1.2」や「0.1」は除去されません。

## 警告アクションとしてのワークフロー

ワークフローは、オプションで [第9章「警告の構成」](#)として実行できます。ワークフローを警告アクションとして使用できるようにするには、その alert アクションが true に設定されている必要があります。

## 警告アクションの実行コンテキスト

警告アクションとして実行される場合、ワークフローは、そのワークフローを作成したユーザーの識別情報を引き継ぎます。このため、警告アクションとして使用できるワークフローはすべて、setid が true に設定されている必要があります。警告アクションには、次のメンバーを持つ 1 つのオブジェクトパラメータがあります。

表 16-8 警告の実行コンテキストの必須メンバー

必須メンバー	タイプ	説明
class	文字列	警告のクラス。
code	文字列	警告のコード。
items	オブジェクト	警告を記述したオブジェクト。
timestamp	日付	警告の時間。

parameters オブジェクトの items メンバーには、次のメンバーがあります。

表 16-9 items メンバーに必要なメンバー

必須メンバー	タイプ	説明
url	文字列	警告を記述した Web ページの URL
action	文字列	警告に対応してユーザーが実行すべきアクション。
impact	文字列	警告の原因となったイベントの影響。
description	文字列	警告を記述した、人間が読める形式の文字列。
severity	文字列	警告の原因となったイベントの重要度。

## 警告アクションの監査

警告アクションとして実行されているワークフローは、`audit` 関数を使用して監査ログエントリを生成できます。関連するすべてのデバッグ情報を `audit` 関数を經由して監査ログに生成することをお勧めします。たとえば、クラスタ化された状態にある場合はフェイルオーバーを実行するが、リブートの失敗をすべて監査するワークフローを次に示します。

```
var workflow = {
  name: 'Failover',
  description: 'Fail the node over to its clustered peer',
  alert: true,
  setid: true,
  execute: function (params) {
    /*
     * To failover, we first confirm that clustering is configured
     * and that we are in the clustered state. We then reboot,
     * which will force our peer to takeover. Note that we're
     * being very conservative by only rebooting if in the
     * AKCS_CLUSTERED state: there are other states in which it
     * may well be valid to failback (e.g., we are in AKCS_OWNER,
     * and our peer is AKCS_STRIPPED), but those states may also
     * indicate aberrant operation, and we therefore refuse to
     * failback. (Even in an active/passive clustered config, a
     * FAILBACK should always be performed to transition the
     * cluster peers from OWNER/STRIPPED to CLUSTERED/CLUSTERED.)
     */
    var uuid = params.uuid;
    var clustered = 'AKCS_CLUSTERED';

    audit('attempting failover in response to alert ' + uuid);

    try {
      run('configuration cluster');
    } catch (err) {
      audit('could not get clustered state; aborting');
      return;
    }
  }
}
```

```
    }  
  
    if ((state = get('state')) != clustered) {  
        audit('state is ' + state + '; aborting');  
        return;  
    }  
  
    if ((state = get('peer_state')) != clustered) {  
        audit('peer state is ' + state + '; aborting');  
        return;  
    }  
  
    run('cd /');  
    run('confirm maintenance system reboot');  
}  
};
```

## スケジュールされたワークフローの使用

ワークフローは、ワークフローのスケジュールを設定し、タイマーイベント経由で開始できます。プロパティ `scheduled` をワークフローオブジェクトに追加して `true` に設定する必要があります。ワークフローが ZFSSA にロードされたあとで CLI を使用してスケジュールを作成することも、オブジェクトワークフローに配列型のプロパティ `schedule` をオブジェクトワークフローに追加することもできます。

## CLI の使用

ワークフローが ZFSSA にロードされると、次のように CLI インタフェースを使用してスケジュールを定義できます。

```
dory:> maintenance workflows  
dory:maintenance workflows> "select workflow-002''"  
dory:maintenance workflow-002> schedules  
dory:maintenance workflow-002 schedules>create  
dory:maintenance workflow-002 schedule (uncommitted)> set frequency=day  
frequency = day (uncommitted)  
dory:maintenance workflow-002 schedule (uncommitted)> set hour=10  
hour = 10 (uncommitted)  
dory:maintenance workflow-002 schedule (uncommitted)> set minute=05  
minute = 05 (uncommitted)  
dory:maintenance workflow-002 schedule (uncommitted)> commit  
dory:maintenance workflow-002 schedules> list  
NAME          FREQUENCY      DAY           HH:MM  
schedule-001  day            -            10:05  
dory:maintenance workflow-002 schedules> create  
dory:maintenance workflow-002 schedule (uncommitted)> set frequency=week  
frequency = week (uncommitted)  
dory:maintenance workflow-002 schedule (uncommitted)> set day=Monday  
day = Monday (uncommitted)  
dory:maintenance workflow-002 schedule (uncommitted)> set hour=13  
hour = 13 (uncommitted)  
dory:maintenance workflow-002 schedule (uncommitted)> set minute=15  
minute = 15 (uncommitted)
```

```
dory:maintenance workflow-002 schedule (uncommitted)> commit
dory:maintenance workflow-002 schedules> list
NAME          FREQUENCY    DAY          HH:MM
schedule-001  day          -            10:05
schedule-002  week        Monday      13:15
dory:maintenance workflow-002 schedules>
```

各スケジュールエントリは、次のプロパティで構成されています。

表 16-10 スケジュールのプロパティ

プロパティ	タイプ	説明
NAME	文字列	スケジュールの名前 (システムが生成)
frequency	文字列	minute、halfhour、hour、day、week、month
day	文字列	特定の曜日を指定します。設定できるのは Monday、Tuesday、Wednesday、Thursday、Friday、Saturday、または Sunday です。frequency が week または month に設定されているときに設定できます
hour	文字列	00 - 23 の範囲でスケジュールの時間部分を指定します。frequency が day、week、または month に設定されているときに設定できます。
minute	文字列	00 - 59 の範囲で、スケジュールの分部分を指定します。

## スケジュールのコーディング

スケジュールは、オブジェクトワークフローのプロパティとして、ワークフローコードで指定することもできます。ここで使用されるプロパティの構文は、CLI でスケジュールを作成する場合と異なります。使用されるプロパティを次に示します。

表 16-11 スケジュールのプロパティ

プロパティ	タイプ	説明
offset	数値	定義された期間の開始時点を指定します
period	数値	スケジュールの頻度を定義します

プロパティ	タイプ	説明
unit	文字列	offset および period の定義で単位として秒と月のどちらを使用するかを指定します

次のコード例は、プロパティの使用方法を示しています。インライン演算を使用することで、offset や period の宣言が読みやすくなっていることに注意してください。

```
// Example of using Schedule definitions within a workflow
var MyTextObject = {
  MyVersion: '1.0',
  MyName: 'Example 9',
  MyDescription: 'Example of use of Timer',
  Origin: 'Oracle'
};
var MySchedules = [
  // half hr interval
  { offset: 0, period: 1800, units: "seconds" },
  // offset 2 days, 4hr, 30min , week interval
  {offset: 2*24*60*60+4*60*60+30*60, period: 604800,units: "seconds" }
];
var workflow = {
  name: MyTextObject.MyName,
  description: MyTextObject.MyDescription,
  version: MyTextObject.MyVersion,
  alert: false,
  setid: true,
  schedules: MySchedules,
  scheduled: true,
  origin: MyTextObject.Origin,
  execute: function () {
    audit('workflow started for timer; ');
  }
};
```

オブジェクト MySchedules のプロパティ units は、プロパティ offset および period で使用される単位の種類を指定します。これらは、seconds または month に設定できます。プロパティ period はイベントの頻度を指定し、offset はその期間内の単位数を指定します。上記の例では、2 番目のスケジュールの期間が 1 週に設定されていて、2 日目の 4:30 に開始します。プロパティ schedules には複数のスケジュールを定義できます。

この例のオブジェクト MySchedules では、次の 3 つのプロパティが使用されています。

- offset - これは、1970 年 1 月 1 日からの、スケジュールの開始オフセットです。このオフセットは、プロパティ「units」で定義された単位で表されます。
- period - これはスケジュールが繰り返される期間であり、同様にプロパティ「units」で定義された単位で表されます。
- units - これは秒数または月数で定義できます。



週次スケジュールの開始時点は木曜日です。これは、エポックの開始基準として定義されている 1970 年 1 月 1 日が木曜日だったからです。

上記の例では、2 番目のスケジュールの期間に、2 日と 4 時間と 30 分という開始オフセットが使われています。これにより、開始日は 1970 年 1 月 3 日の午前 4:30 になります。このスケジュールは、毎週土曜日の午前 4:30 に無期限で繰り返されます。CLI でのスケジュールの表示を次に示します。

```
<small>dory:> maintenance workflows
dory:maintenance workflows> list
WORKFLOW      NAME                               OWNER SETID ORIGIN          VERSION
workflow-000  Configure for Oracle Solaris Cluster NFS root false Oracle Corporation 1.0.0
workflow-001  Unconfigure Oracle Solaris Cluster NFS root false Oracle Corporation 1.0.0
workflow-002  Configure for Oracle Enterprise Manager Monitoring root false Sun Microsystems,
              Inc. 1.1
workflow-003  Unconfigure Oracle Enterprise Manager Monitoring root false Sun Microsystems,
              Inc. 1.0</small>
```

```
dory:maintenance workflow-002 schedules>
```

NAME	FREQUENCY	DAY	HH:MM
schedule-000	halfhour	-	--:00
schedule-001	week	Saturday	04:30

## 例: デバイスタイプの選択

指定されたドライブタイプに基づいてワークシートを作成するワークフローの例を次に示します。

```
var steps = [ {
  step: 'Checking for existing worksheet',
  execute: function (params) {
    /*
     * In this step, we're going to see if the worksheet that
     * we're going to create already exists. If the worksheet
     * already exists, we blow it away if the user has indicated
     * that they desire this behavior. Note that we store our
     * derived worksheet name with the parameters, even though
     * it is not a parameter per se; this is explicitly allowed,
     * and it allows us to build state in one step that is
     * processed in another without requiring additional global
     * variables.
     */
    params.worksheet = 'Drilling down on ' + params.type + ' disks!';

    try {
      run('analytics worksheets select name="' +
        params.worksheet + '"');

      if (params.override) {
        run('confirm destroy');
        return;
      }
    }

    throw ('Worksheet called "' + params.worksheet +
      '" already exists!');
```

```
} catch (err) {
  if (err.code != EAKSH_ENTITY_BADSELECT)
    throw (err);
}
}, {
step: 'Finding disks of specified type',
execute: function (params) {
  /*
   * In this step, we will iterate over all chassis, and for
   * each chassis iterates over all disks in the chassis,
   * looking for disks that match the specified type.
   */
  var chassis, name, disks;
  var i, j;

  run('cd /');
  run('maintenance hardware');

  chassis = list();
  params.disks = [];

  for (i = 0; i < chassis.length; i++) {
    run('select ' + chassis[i]);

    name = get('name');
    run('select disk');
    disks = list();

    for (j = 0; j < disks.length; j++) {
      run('select ' + disks[j]);

      if (get('use') == params.type) {
        params.disks.push(name + '/' +
          get('label'));
      }

      run('cd ..');
    }

    run('cd ../../');
  }

  if (params.disks.length === 0)
    throw ('No ' + params.type + ' disks found');
  run('cd /');
}
}, {
step: 'Creating worksheet',
execute: function (params) {
  /*
   * In this step, we're ready to actually create the worksheet
   * itself: we have the disks of the specified type and
   * we know that we can create the worksheet. Note that we
   * create several datasets: first, I/O bytes broken down
   * by disk, with each disk of the specified type highlighted
   * as a drilldown. Then, we create a separate dataset for
   * each disk of the specified type. Finally, note that we
   * aren't saving the datasets -- we'll let the user do that
   * from the created worksheet if they so desire. (It would
   * be straightforward to add a boolean parameter to this
   * workflow that allows that last behavior to be optionally
   * changed.)
   */
}
```

```
    */
    var disks = [], i;

    run('analytics worksheets');
    run('create "' + params.worksheet + '"');
    run('select name="' + params.worksheet + '"');
    run('dataset');
    run('set name=io.bytes[disk]');

    for (i = 0; i < params.disks.length; i++)
        disks.push('"' + params.disks[i] + '"');

    run('set drilldowns=' + disks.join(', '));
    run('commit');

    for (i = 0; i < params.disks.length; i++) {
        run('dataset');
        run('set name="io.bytes[disk=' +
            params.disks[i] + ']'');
        run('commit');
    }
}
}
} ]];

var workflow = {
    name: 'Disk drilldown',
    description: 'Creates a worksheet that drills down on system, ' +
        'cache, or log devices',
    parameters: {
        type: {
            label: 'Create a new worksheet drilling down on',
            type: 'ChooseOne',
            options: [ 'cache', 'log', 'system' ],
            optionlabels: [ 'Cache', 'Log', 'System' ]
        },
        overwrite: {
            label: 'Overwrite the worksheet if it exists',
            type: 'Boolean'
        }
    },
    execute: function (params) { return (steps); }
};
```

## BUI

ワークフローは、プラスアイコンをクリックすることによって ZFSSA にアップロードされ、そのワークフローを指定する行をクリックすることによって実行されます。

図 16-1

NAME ^	DESCRIPTION	VERSION
Clear locks	Clear locks held on behalf of an NFS client	1.0.0
Configure for Oracle Enterprise Manager Monitoring	Sets up environment to be monitored by Oracle Enterprise Manager	1.1
Configure for Oracle Solaris Cluster NFS	Sets up environment for Oracle Solaris Cluster NFS	1.0.0
Unconfigure Oracle Enterprise Manager Monitoring	Removes the artifacts from the appliance used by Oracle Enterprise Manager	1.0
Unconfigure Oracle Solaris Cluster NFS	Removes the artifacts from the appliance used by Oracle Solaris Cluster NFS	1.0.0

## CLI

ワークフローは、CLI の maintenance workflows セクションで操作されます。

### ワークフローのダウンロード

ワークフローは、download コマンドを使用して ZFSSA にダウンロードされますが、これは『Oracle ZFS Storage Appliance 顧客サービスマニュアル』の「システム」と同様です。

```
dory:maintenance workflows> download
dory:maintenance workflows download (uncommitted)> get
      url = (unset)
      user = (unset)
      password = (unset)
```

そのワークフローに有効な URL になるように「url」プロパティを設定する必要があります。これはネットワークにローカル、またはインターネット経由のどちらでもかまいません。URL には、HTTP (「http://」で始まる) または FTP (「ftp://」で始まる) のどちらかを指定できます。ユーザー認証が必要な場合は、URL の一部にユーザー認証を含めることも (たとえば、「ftp://myusername:mypasswd@myserver/export/foo」)、URL からユーザー名とパスワードを除き、代わりにユーザープロパティとパスワードプロパティを設定することもできます。

```
dory:maintenance workflows download (uncommitted)> set url=
      ftp://foo/example1.akwf
      url = ftp://foo/example1.akwf
dory:maintenance workflows download (uncommitted)> set user=bmc
      user = bmc
dory:maintenance workflows download (uncommitted)> set password
Enter password:
      password = *****
dory:maintenance workflows download (uncommitted)> commit
Transferred 138 of 138 (100%) ... done
```

## ワークフローの表示

ワークフローを一覧表示するには、maintenance workflows コンテキストから list コマンドを使用します。

```
<small>dory:maintenance workflows> list
WORKFLOW  NAME                               OWNER SETID ORIGIN                VERSION
workflow-000 Configure for Oracle Solaris Cluster NFS root false Oracle Corporation  1.0.0
workflow-001 Unconfigure Oracle Solaris Cluster NFS root false Oracle Corporation  1.0.0
workflow-002 Configure for Oracle Enterprise Manager Monitoring root false Sun Microsystems,
Inc. 1.1
workflow-003 Unconfigure Oracle Enterprise Manager Monitoring root false Sun Microsystems,
Inc. 1.0</small>
```

ワークフローを表示するには、maintenance workflows コンテキストから show コマンドを使用します。

```
dory:maintenance workflows> select workflow-001
dory:maintenance workflow-001> show
Properties:
    name = Configure for Oracle Solaris Cluster NFS
    description = Sets up environment for Oracle Solaris Cluster NFS
    owner = root
    origin = Oracle Corporation
    setid = false
    alert = false
    version = 1.0.0
    scheduled = false
```

ワークフローを選択するには、select コマンドを使用します。

```
dory:maintenance workflows> select workflow-000
dory:maintenance workflow-000>
```

ワークフローのプロパティを取得するには、選択されたワークフローのコンテキスト内から get コマンドを使用します。

```
dory:maintenance workflow-000> get
    name = Hello world
    description = Bids a greeting to the world
    owner = root
    origin = <local>
    setid = false
    alert = false
    scheduled = false
```

## ワークフローの実行

ワークフローを実行するには、選択されたワークフローのコンテキスト内から execute コマンドを使用します。ワークフローがパラメータを取らない場合は、単純に実行されます。

```
dory:maintenance workflow-000> execute
hello world!
```

ワークフローがパラメータを取る場合、コンテキストは、パラメータを指定する必要があるキャプティブコンテキストになります。

```
dory:maintenance workflow-000> execute
dory:maintenance workflow-000 execute (uncommitted)> get
      type = (unset)
      overwrite = (unset)
```

最初に必要なパラメータを設定しないでワークフローの実行をコミットしようとする、常に明示的に失敗します。

```
dory:maintenance workflow-000 execute (uncommitted)> commit
error: cannot execute workflow without setting property "type"
```

ワークフローを実行するには、指定されたパラメータを設定したあと、commit コマンドを使用します。

```
dory:maintenance workflow-000 execute (uncommitted)> set type=system
      type = system
dory:maintenance workflow-000 execute (uncommitted)> set overwrite=true
      overwrite = true
dory:maintenance workflow-000 execute (uncommitted)> commit
```

ワークフローに指定された手順がある場合は、それらの手順が CLI に表示されます。次に例を示します。

```
dory:maintenance workflow-000 execute (uncommitted)> commit
Checking for existing worksheet ... done
Finding disks of specified type ... done
Creating worksheet ... done
```

## 統合

---

Oracle ZFS Storage Appliance では、広範なアプリケーションホストと通信するためにデータプロトコルのフルスイートを用意しています。アプリケーションのパフォーマンスを向上させたり、アプリケーション環境とより緊密に統合するには、NAS ストレージのドキュメントページにあるホワイトペーパーおよびソリューション概要に示されているベストプラクティスに従います。

- [「Symantec DMP/Storage Foundation」](#)

一部のアプリケーションでは、アプリケーションホストにソフトウェアをインストールすると相互運用性が強化されます。次の記事では、ソフトウェア統合によってストレージ管理者の操作向上を実現する方法について概要を説明しています。包括的なドキュメントが各ダウンロードに同梱されています。

- [489 ページの「Oracle Solaris Cluster Geographic Edition 用 Sun ZFS Storage Appliance プラグイン」](#)
- [489 ページの「Oracle Solaris Cluster 用 Sun ZFS Storage Appliance ネットワークファイルシステムプラグイン」](#)
- [492 ページの「ボリュームシャドウコピーサービスソフトウェア用 Sun ZFS Storage Appliance プロバイダ」](#)
- [490 ページの「Oracle Enterprise Manager Grid Controller 用 Sun ZFS Storage Management プラグイン」](#)
- [492 ページの「Sun ZFS Storage Appliance 用 Oracle Virtual Machine Storage Connect プラグイン」](#)
- [494 ページの「Sun ZFS Storage 7000 Storage Replication Adapter for VMware Site Recovery Manager」](#)
- [488 ページの「 Oracle Intelligent Storage Protocol 」](#)

また、ほかの Oracle 製品とシームレスに統合するように、アプライアンスは独自の機能を備えています。たとえば以降のセクションでは、ZFS Storage Appliance を Oracle Exadata Database Machine および Oracle SPARC SuperCluster のバックアップ先として構成する方法について説明します。

- [456 ページの「Oracle Exadata Database Machine Backup」](#)

- [484 ページの「ZFS Storage Appliance のバックアップのための Oracle SPARC SuperCluster の構成」](#)

詳細は、NAS ストレージのドキュメントページにアクセスしてください。

## Oracle Exadata Database Machine Backup

ネイティブな QDR InfiniBand および 10Gb Ethernet 接続オプションを搭載している ZFS Storage Appliance は、Oracle Exadata の信頼性あるバックアップに最適です。コマンドラインツールを使用した配備用に Oracle Exadata Backup Configuration Utility が提供されています。または以降のセクションの手順を使用して、アプライアンスを手動で構成することもできます。

- [456 ページの「Sun ZFS Storage Appliance の手動構成」](#)
- [460 ページの「Sun ZFS Storage Appliance 用の Oracle Exadata の構成」](#)

Oracle Exadata からバックアップを実行する方法など、包括的なドキュメントがユーティリティーに同梱されています。手動かユーティリティーの使用かにかかわらず、それらの設定のほかに、アプライアンスのネットワークおよびストレージプールの構成も必要です。

Oracle Exadata のバックアップ先として ZFS Storage Appliance を使用方法の詳細は、NAS ストレージのドキュメントページで、Sun ZFS Storage Appliance を使用した Oracle Exadata の保護: 構成のベストプラクティスに関するホワイトペーパーを参照してください。また、セットアップの複雑さを最小限に抑えるために、Oracle ZFS Storage ZS3-BA としてディスクシェルフに事前にラックされている Oracle ZFS Storage ZS3-4 クラスタも使用可能です。このアプライアンスを Oracle Exadata と統合する方法は、前述のプロセスと同一です。

## Sun ZFS Storage Appliance の手動構成

このセクションでは、ZFS Storage Appliance を Oracle Exadata と使用するために手動で構成する場合の一般的なガイドラインについて説明します。詳細は NAS ストレージのドキュメントページで、Sun ZFS Storage Appliance を使用した Oracle Exadata の保護: 構成のベストプラクティスに関するホワイトペーパーを参照してください。



## ネットワーク、プール、およびシェアの構成

以降のセクションでは、バックアップおよび復元処理をサポートするように ZFS Storage Appliance ネットワーク、ストレージプール、およびシェアの構成を最適化するベストプラクティスについてサマリーを示します。

### ネットワーク構成

このセクションでは、IP ネットワークマルチパス (IPMP) グループを構成する方法、および ZFS Storage Appliance でルーティングを構成する方法について説明します。

注: Oracle Exadata Backup Configuration Utility を使用した場合は、このセクションの説明に従ってネットワークを構成してください。詳細は、ベストプラクティスのホワイトペーパーを参照してください。

追加の IB 接続を必要としている場合は、IB HCA をさらにインストールして構成できません。詳細は、Oracle ZFS Storage Appliance の設置ガイドを参照してください。

このセクションの原則は、ネットワーク構成を ibp インタフェースではなく ixgbe インタフェースに適用することで、10G ビット Ethernet 実装に適用できます。10G ビット Ethernet 実装は、アクティブ/アクティブ IPMP として構成できます。ZFS Storage Appliance が Oracle Exadata と異なるサブネット上にある場合は、ZFS Storage Appliance から Oracle Exadata への静的ルートを作成することが必要な場合があります。詳細は、ネットワーク管理者に問い合わせてください。

### ▼ 基本ネットワーク構成

1. ZFS Storage Appliance が Oracle Exadata に接続されていることを確認します。
2. `ibp0`、`ibp1`、`ibp2`、および `ibp3` をアドレス `0.0.0.0/8` (IPMP に必要)、接続モード、およびパーティションキー `ffff` で構成します。Oracle Exadata システムで使用されるパーティションキーを指定するには、root ユーザーとしてコマンド `<br/># cat /sys/class/net/ib0/pkey` を実行します
3. アクティブ/スタンバイ IPMP グループを `ibd0` および `ibd3` で構成します (`ibd0` がアクティブで `ibd3` がスタンバイ)。
4. アクティブ/スタンバイ IPMP グループを `ibd1` および `ibd2` で構成します (`ibd2` がアクティブで `ibd1` がスタンバイ)。

5. アダプティブルーティングを有効にして、同じサブネット上の複数の IP アドレスが同じヘッドに所有されているときに、トラフィックが適切に負荷分散されるようにします。これは、クラスタのフェイルオーバーのあとで発生します。

## プールの構成

このセクションでは、データ保護およびパフォーマンスの要件に基づいて Oracle Recovery Manager (RMAN) のバックアップおよび復元操作に ZFS Storage Appliance の最適なプール構成を決定するために、設計に関する考慮事項について説明します。

注: Oracle Exadata Backup Configuration Utility を使用した場合は、このセクションの説明に従ってプールを構成してください。詳細は、ベストプラクティスのホワイトペーパーを参照してください。

システム計画担当者は、次のガイドラインに基づいてプールの保護を検討する必要があります。

- 汎用および容量用に最適化されたシステムには、パリティに基づく保護を使用します。
- \* ランダムなワークロードの対象となるシステムの 1 ドライブ障害から保護するには RAID-Z。
- \* ストリーミングのワークロードのみであるシステムの 2 ドライブ障害から保護するには RAID-Z2。
- 増分で適用されるバックアップと高パフォーマンスには、ミラー化を使用します。
- パフォーマンス要件に基づいてプールを構成します。
- \* 管理用に最適化されたシステムには、1 つのプールを構成します。
- \* パフォーマンス用に最適化されたシステムには、2 つのプールを構成します。各トレイのドライブの半分を使用して、2 つのプールシステムを構成してください。
- ログデバイスの保護を構成します。
- \* RAID-Z およびミラー化プール構成にはログデバイスをストライプ化します。
- \* RAID-Z2 プール構成にはログデバイスをミラー化します。

注: Oracle Exadata Backup Configuration Utility を使用した場合は、次のトピック、460 ページの「Sun ZFS Storage Appliance 用の Oracle Exadata の構成」に進みます。

## シェアの構成

ZFS Storage Appliance のシェアでは、デフォルトオプションを汎用ワークロードの開始点として使用できます。ZFS Storage Appliance のシェアは、次のようにして Oracle RMAN バックアップおよび復元操作に最適化できます。

- 単一データベースのバックアップおよび復旧に関連するすべてのシェアを格納するためのプロジェクトを作成します。2 プール実装の場合は、2 つのプロジェクト (各プールに 1 つ) を作成します。
- Oracle RMAN のバックアップおよび復元ワークロードをサポートするシェアを次の値を使用して構成します。
- \* データベースレコードサイズ (recordsize): 128 K バイト
- \* 同期書き込みバイアス (logbias): 「スループット」 (バックアップセットおよびイメージコピーを処理する場合) または「待機時間」 (増分で適用されるバックアップの場合)
- \* デバイス使用をキャッシュ (secondary cache): 「なし」 (バックアップセットの場合) または「すべて」 (増分で適用されるバックアップまたはデータベースのクローン操作をサポートする場合)
- \* データ圧縮 (compression): 「オフ」 (パフォーマンスが最適化されたシステムの場合)、 「LZJB」または「gzip-2」 (容量が最適化されたシステムの場合)
- \* プールごとのシェアの数: 1 (管理が最適化されたシステムの場合)、2 または 4 (パフォーマンスが最適化されたシステムの場合)

Oracle Exadata のバックアップおよび復元をサポートするために使用されるシェアに対して、ユーザー要件の要求に従い、追加のシェア構成オプション (高次の gzip 圧縮やレプリケーションなど) を適用できます。

追加の ZFS Storage Appliance データサービスを実装するユーザーは、前述のプラクティスから逸脱する部分の影響を検証するために、実装固有のテストを検討する必要があります。

## Oracle RMAN および Oracle Database インスタンスの構成

Oracle RMAN は、Oracle Exadata の内容を保護するために不可欠なコンポーネントです。Oracle RMAN を使用すると、ZFS Storage Appliance に Oracle Exadata の内容のバックアップセット、イメージコピー、増分更新されたバックアップを作成できます。Oracle Exadata から ZFS Storage Appliance への Oracle RMAN バックアップのパフォーマンスを最適化するために、データベース管理者は次のベストプラクティスを適用する必要があります。

- データベースマシンのノード全体に均等に Oracle RMAN チャンネルを負荷分散します。
- ZFS Storage Appliance のシェアおよびコントローラ全体に均等に Oracle RMAN チャンネルを負荷分散します。

ZFS Storage Appliance に対する Oracle RMAN チャンネルのバッファリングを最適化するために、隠しインスタンスパラメータのいくつかについて、値を調整できます。Oracle Database 11g リリース 2 では、次のパラメータを調整できます。

- バックアップおよび復元セットの場合:
  - \*\_backup\_disk\_bufcnt=64
  - \*\_backup\_disk\_bufsz=1048576
- イメージコピーのバックアップおよび復元の場合:
  - \*\_backup\_file\_bufcnt=64
  - \*\_backup\_file\_bufsz=1048576

これらのパラメータの調整や Oracle Database ソフトウェアの以前のバージョンにおける同等パラメータの調整に関する詳細は、<http://support.oracle.com> (<http://support.oracle.com>) でバッファメモリーパラメータを使用した RMAN パフォーマンスの調整に関する記事番号 1072545.1 を参照してください。

Oracle Direct NFS (dNFS) は、高パフォーマンスの NFS クライアントで、Oracle RMAN のバックアップおよび復元操作に優れたパフォーマンスを発揮します。バックアップおよび復元操作で最大限のスループットを求めるユーザーは、dNFS を構成する必要があります。

## 次のステップ

[460 ページの「Sun ZFS Storage Appliance 用の Oracle Exadata の構成」](#)

## Sun ZFS Storage Appliance 用の Oracle Exadata の構成

このセクションでは、ZFS Storage Appliance を Oracle Exadata に接続する方法を示すスクリプト例について説明します。これらのスクリプトは、dbname という名前のデータベースを 1 プールおよび 2 プール ZFS Storage Appliance 構成でサポートするように設計されています。

## Exadata の構成 Sun ZFS Storage Appliance 用の Oracle Exadata の構成

### 一般的な実装手順

実装手順は次のとおりです。

1. ホスト上にシェアをマウントするためのディレクトリ構造 (マウントポイント) を設定します。
2. ZFS Storage Appliance からエクスポートされたシェアが適切なマウントポイントにマウントされるように `/etc/fstab` を更新します。
3. シェアのマウントおよびアンマウントの処理を自動化する `init.d` サービスを作成します。
4. ZFS Storage Appliance でエクスポートされたシェアにアクセスするように `orantab` ファイルを更新するか、`/etc/fstab` でブート時マウントを設定します。
5. ホスト上にシェアをマウントします。
6. `ORACLE_HOME` のアクセス権設定と一致するように、マウントされたシェアのアクセス権を変更します。
7. Oracle Database インスタンスを再起動して、`orantab` ファイルに対する変更を有効にします (オプション)。

注: Oracle Exadata Backup Configuration Utility を使用した場合は、手順 4 および手順 7 を除き、すべての手順がすでに実行されています。次のセクション「詳細な実装手順」では、`orantab` の更新による ZFS Storage Appliance のエクスポートへのアクセス、およびマウントされたシェアの所有権設定の手順 2 をオプションで実行できます。

### 詳細な実装手順

このセクションのトピック:

- ホスト上にシェアをマウントするためのディレクトリ構造の設定
- `/etc/fstab` ファイルの更新
- `init.d` サービスの作成
- ZFS Storage Appliance のエクスポートにアクセスするための `orantab` の更新
- ホスト上にシェアのマウント
- マウントされたシェアの所有者の設定

## ホスト上にシェアをマウントするためのディレクトリ構造の設定

ホスト上にシェアをマウントするためのマウントポイントを次のように設定します。

```
mkdir -p /zfssa/dbname/backup1
mkdir -p /zfssa/dbname/backup2
mkdir -p /zfssa/dbname/backup3
mkdir -p /zfssa/dbname/backup4
```

## /etc/fstab ファイルの更新

/etc/fstab ファイルを更新するには、次のいずれかのオプションを使用します。

注: 下記のリストで UNIX の改行エスケープ文字 (\) は、1 行分のコードが 2 行目に折り返していることを表します。折り返された 1 行を fstab に入力するときは、\ 文字を削除して空白で区切られた 2 つの行の断片を 1 行に結合してください。

1 プール構成の場合:

```
192.168.36.200:/export/dbname/backup1 /zfssa/dbname/backup1 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
192.168.36.200:/export/dbname/backup2 /zfssa/dbname/backup2 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
192.168.36.200:/export/dbname/backup3 /zfssa/dbname/backup3 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
192.168.36.200:/export/dbname/backup4 /zfssa/dbname/backup4 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
```

2 プール構成の場合:

```
192.168.36.200:/export/dbname/backup1 /zfssa/dbname/backup1 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
192.168.36.201:/export/dbname/backup2 /zfssa/dbname/backup2 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
192.168.36.200:/export/dbname/backup3 /zfssa/dbname/backup3 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
192.168.36.201:/export/dbname/backup4 /zfssa/dbname/backup4 nfs \<br/>
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,tcp,nfsvers= \<br/> 3,timeo=600 0 0
```

## init.d サービスの作成

次の適切なオプションを使用して、init.d サービスを作成します。

```
# !/bin/sh
#
# zfssa_dbname: Mount ZFSSA project dbname for database dbname
#
# chkconfig: 345 61 19
# description: mounts ZFS Storage Appliance shares
#
```

```

start()
{
    mount /zfssa/dbname/backup1
    mount /zfssa/dbname/backup2
    mount /zfssa/dbname/backup3
    mount /zfssa/dbname/backup4
    echo "Starting $prog: "
}

stop()
{
    umount /zfssa/dbname/backup1
    umount /zfssa/dbname/backup2
    umount /zfssa/dbname/backup3
    umount /zfssa/dbname/backup4
    echo "Stopping $prog: "
}

case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    restart)
        stop
        start
        ;;
    status)
        mount
        ;;
    *)
        echo "Usage: $0 {start|stop|restart|status}"
        exit 1
esac

```

(オプション) 次を入力して、init.d サービスのブート時起動を有効にします。

```
# chkconfig zfssa_dbname on
```

(オプション) サービスコマンドを使用してサービスを手動で起動および停止します。

```
# service zfssa_dbname start<br/># service zfssa_dbname stop
```

## ZFS Storage Appliance のエクスポートにアクセスするための oranfstab の更新

ZFS Storage Appliance のエクスポートにアクセスするように oranfstab ファイルを更新するには、次の適切なオプションを使用します。

注: Oracle Exadata Backup Configuration Utility を使用した場合は、オプションでこの手順を実行できます。

## 1 プール構成の場合:

```
server: 192.168.36.200
path: 192.168.36.200
export: /export/dbname/backup1 mount: /zfssa/dbname/backup1
export: /export/dbname/backup2 mount: /zfssa/dbname/backup2
export: /export/dbname/backup3 mount: /zfssa/dbname/backup3
export: /export/dbname/backup4 mount: /zfssa/dbname/backup4
```

## 2 プール構成の場合:

```
server: 192.168.36.200
path: 192.168.36.200
export: /export/dbname/backup1 mount: /zfssa/dbname-2pool/backup1
export: /export/dbname/backup3 mount: /zfssa/dbname-2pool/backup3
server: 192.168.36.201
path: 192.168.36.201
export: /export/dbname/backup2 mount: /zfssa/dbname-2pool/backup2
export: /export/dbname/backup4 mount: /zfssa/dbname-2pool/backup4
```

## ホスト上にシェアのマウント

ホスト上にシェアをマウントするには、次の 2 つのうちいずれかのオプションを使用します。

```
# service mount_dbname start
```

または

```
# dcli -l root -g /home/oracle/dbs_group service mount_dbname start
```

## マウントされたシェアの所有者の設定

ORACLE\_HOME のアクセス権設定と一致するように、マウントされたシェアのアクセス権の設定を変更します。この例では、ユーザーおよびグループの所有者が oracle:dba に設定されています。

注: Oracle Exadata Backup Configuration Utility を使用した場合は、オプションで手順 2 を実行できます。手順 1 はすでに実行されています。

1. 次の 2 つのうちいずれかのオプションを入力します。  
# chown oracle:dba /zfssa/dbname/\*  
または  
# dcli -l root -g /home/oracle/dbs\_group chown oracle:dba/zfssa/dbname/\*
2. 次のいずれかのオプションを使用して Oracle Database インスタンスを再起動して、oranfstab ファイルに対する変更を有効にします。

■ 一度に 1 つのインスタンスを再起動します (順次アップグレード)。例:

```
■ :$ srvctl stop instance -d dbname -i dbname1
```



- `:$ srvctl start instance -d dbname -i dbname1`
- `:$ srvctl stop instance -d dbname -i dbname2`
- `:$ srvctl start instance -d dbname -i dbname2`
- `:$ srvctl stop instance -d dbname -i dbname3`
- `:$ srvctl start instance -d dbname -i dbname3`
- `:$ srvctl stop instance -d dbname -i dbname4`
- `:$ srvctl start instance -d dbname -i dbname4`
- `:$ srvctl stop instance -d dbname -i dbname5`
- `:$ srvctl start instance -d dbname -i dbname5`
- `:$ srvctl stop instance -d dbname -i dbname6`
- `:$ srvctl start instance -d dbname -i dbname6`
- `:$ srvctl stop instance -d dbname -i dbname7`
- `:$ srvctl start instance -d dbname -i dbname7`
- `:$ srvctl stop instance -d dbname -i dbname8`
- `:$ srvctl start instance -d dbname -i dbname8`
- データベース全体を再起動します。例:
- `:$ srvctl stop database -d dbname`
- `:$ srvctl start database -d dbname`

## Oracle SPARC SuperCluster のバックアップ

ネイティブな QDR InfiniBand および 10Gb Ethernet 接続オプションを搭載している ZFS Storage Appliance は、Oracle SPARC SuperCluster の信頼性あるバックアップに最適です。以降のセクションの手順を使用してシステムを構成します。

- [「バックアップ用の ZFS Storage Appliance の構成」](#)
- [「ZFS Storage Appliance のバックアップのための Oracle SPARC SuperCluster の構成」](#)

Oracle SPARC SuperCluster のバックアップ先として ZFS Storage Appliance を使用する方法の詳細は、NAS ストレージのドキュメントページで、Oracle SPARC SuperCluster を使用した Sun ZFS Backup Appliance の構成に関するホワイトペーパーを参照してください。また、セットアップの複雑さを最小限に抑えるために、Oracle ZFS Storage ZS3-BA としてディスクシェルフに事前にラックされている Oracle ZFS Storage ZS3-4 クラスタも使用可能です。このアプライアンスを Oracle SPARC SuperCluster と統合する方法は、前述のプロセスと同一です。

## バックアップ用の ZFS Storage Appliance の構成

このセクションでは、Oracle SPARC SuperCluster で ZFS Storage Appliance をバックアップに使用するために構成する場合の一般的なガイドラインについて説明します。詳細は NAS ストレージのドキュメントページで、Oracle SPARC SuperCluster を使用した Sun ZFS Backup Appliance の構成に関するホワイトペーパーを参照してください。2 つのコントローラ (ヘッド) および 4 つのディスクシェルフを備えた ZFS Storage Appliance を例として示しています。

このセクションのトピック:

- [466 ページの「ZFS Storage Appliance InfiniBand データリンクの構成」](#)
- [467 ページの「ZFS Storage Appliance を追加する Oracle SPARC SuperCluster InfiniBand スイッチの構成」](#)
- [470 ページの「シングル IP 接続用の ZFS Storage Appliance ネットワークの構成」](#)
- [471 ページの「アクティブ/アクティブ構成用の ZFS Storage Appliance ネットワークの構成」](#)
- [473 ページの「ZFS Storage Appliance ストレージプールの構成」](#)
- [473 ページの「ZFS Storage Appliance のシェアの構成」](#)
- [474 ページの「ZFS Storage Appliance DTrace Analytics の構成」](#)
- [475 ページの「クライアント NFS マウントの構成」](#)
- [476 ページの「Solaris 11 ネットワークおよびカーネルのチューニング」](#)
- [476 ページの「Oracle Direct NFS \(dNFS\) の構成」](#)
- [477 ページの「Oracle RMAN バックアップおよび復元用の Oracle Database インスタンスの調整」](#)
- [「Oracle RMAN 操作の専用サービスの作成」](#)[479 ページの「Oracle RMAN 操作の専用サービスの作成」](#)
- [480 ページの「Oracle RMAN の構成」](#)

## ZFS Storage Appliance InfiniBand データリンクの構成

このセクションの手順に従って、各 ZFS Storage Appliance InfiniBand 接続を構成します。次の手順では、この手順の間に記録する InfiniBand HBA ポートの 8 つの GUID を使用して、Oracle SPARC SuperCluster InfiniBand スイッチを構成します。

1. NAS ストレージのドキュメントページにある、Oracle SPARC SuperCluster を使用した Sun ZFS Backup Appliance の構成に関するホワイトペーパーの説明に従って、ZFS Storage Appliance を Oracle SPARC SuperCluster に接続します。
2. ヘッド 1 のブラウザユーザーインターフェース (BUI) にログインして、「構成」>「ネットワーク」に移動します。
3. データリンクの横のプラスアイコンをクリックします。「ネットワークデータリンク」ダイアログボックスが開きます。
4. 次のようにダイアログボックスの操作を完了します。
  - 「IB パーティション」ボックスにチェックマークを付けます。
  - データリンク名にわかりやすい名前を入力します。
  - 「パーティションキー」を「8503」に設定します。
  - 「リンクモード」に「接続済みモード」を選択します。
  - 「LACP アグリゲーション」ボックスにチェックマークを付けないでください。
  - 「パーティションデバイス」の「ibp0」を選択します。
  - GUID 番号 (たとえば、21280001ef43bb) を記録して、「適用」をクリックします。
5. 残りの各 InfiniBand インタフェース (ibp1、ibp2、および ibp3) に対して手順 3 および 4 を繰り返します。
6. ヘッド 2 に対して手順 2 から 5 を繰り返します。

## ZFS Storage Appliance を追加する Oracle SPARC SuperCluster InfiniBand スイッチの構成

この手順では、ZFS Storage Appliance Infiniband HBA ポートの GUID を、既存の Oracle SPARC SuperCluster InfiniBand 構成に追加します。これらのポートを追加してパーティションキーの 8503 を使用することで、2 つのデバイス間の通信を行うことができます。

1. Oracle SPARC SuperCluster InfiniBand スパインスイッチに root としてログインします。デフォルトでは、スパインスイッチには <sscid>sw- ib1 (<sscid> は Oracle SPARC SuperCluster システム全体に与えられる接頭辞名) のホスト名が付けられます。次の例では、<sscid> は aiessc です。

```
login as: root
root@aiesscsw-ib1's password:
Last login: Tue Sep 25 08:19:01 2013 from dhcp-brm-bl5-204-3e
```

```
east-10-135-75-254.usdhcp.oraclecorp.com
```

2. コマンド `enablesm` を入力して、スイッチで Subnet Manager が実行されていることを確認します (またはこのコマンドにより Subnet Manager が起動します)。<br/>

```
[root@aiesscsw-ib1 ~]# enablesm
opensm (pid 15906) is already running...
Starting partitiond daemon
/usr/local/util/partitiond is already running
(You may also perform a 'restart' if wanted)
```

3. コマンド `getmaster` を入力して、これが構成のマスタースイッチであることを確認します。スパインスイッチでマスタースイッチが実行されていない場合は、ログアウトして、指定したマスタースイッチにログインして残りの手順を行います。<br/>

```
[root@aiesscsw-ib1 ~]# getmaster
Local SM enabled and running
20130913 10:16:51 Master SubnetManager on sm lid 13 sm guid
0x2128e8ac27a0a0 : SUN DCS 36P QDR aiesscsw-ib1.us.oracle.com
[root@aiesscsw-ib1 ~]#
```

4. 説明されているバックアップ手順 ([http://docs.oracle.com/cd/E26698\\_01/index.html](http://docs.oracle.com/cd/E26698_01/index.html)) に従って、スイッチ構成をバックアップします。

5. コマンド `smpartition list active` を入力して、パーティションキー `0x0503` がパーティション名「sto」に割り当てられていることを確認します (`sto = 0x0503`)。<br/> ZFS Storage Appliance データリンクではパーティションキーが `8503` に設定されましたが、InfiniBand スイッチでは `0503` と報告されます。InfiniBand プロトコルは自身で使用するために 16 進パーティションキー (pkey) の最上位ビット (`0x8000`) を予約するため、これは意図的なものです。したがって、pkey の `0x8503` と `0x0503` は同じです。<br/>

```
[root@aiesscsw-ib1 ~]# smpartition list active
# Sun DCS IB partition config file
# This file is generated, do not edit
#! version_number : 11
Default=0x7fff, ipoib : ALL_CAS=full, ALL_SWITCHES=full, SELF=full;
SUN_DCS=0x0001, ipoib : ALL_SWITCHES=full;
ic1s10 = 0x0501, ipoib, defmember=full:
0x002128001ef30f7,
0x002128001ef33bf,
0x002128001ef30b7,
0x002128001ef314b;
ic2s10 = 0x0502, ipoib, defmember=full:
0x002128001ef30f8,
0x002128001ef33c0,
0x002128001ef30b8,
0x002128001ef314c;
sto = 0x0503, ipoib, defmember=full:
0x002128001ef43f8,
```

```
0x0021280001ef43b7,
0x0021280001cf90c0,
0x0021280001ef43bb,
...more...
```

6. ZFS Storage Appliance を InfiniBand 構成に追加します。

- コマンド `smpartition start` を入力して、再構成セッションを開始します。<br/>

```
# smpartition start<br/>
[root@aiesscsw-ib1 ~]# smpartition start
```

- コマンド `smpartition add` を入力して、8 つの新しい GUID を構成に追加します。<br/>

```
# smpartition add -n sto -port <GUID1> <GUID2> <GUID3> ... <GUID8><br/>
[root@aiesscsw-ib1 ~]# smpartition add -n sto -port
21280001ef43bb 21280001ef43bc 21280001cf90bf 21280001cf90c0
21280001ef43f7 21280001ef43f8 21280001ef43b7 21280001ef43b8
```

- コマンド `smpartition list modified` を入力して、新しい GUID が正しく追加されていることを確認します。<br/>

```
# smpartition list modified<br/>
[root@aiesscsw-ib1 ~]# smpartition list modified
# Sun DCS IB partition config file
<nowki># This file is generated, do not edit
#! version_number : 11
Default=0x7fff, ipoib : ALL_CAS=full, ALL_SWITCHES=full, SELF=
full;
SUN_DCS=0x0001, ipoib : ALL_SWITCHES=full;
ic1s10 = 0x0501, ipoib, defmember=full:
0x0021280001ef30f7,
0x0021280001ef33bf,
0x0021280001ef30b7,
0x0021280001ef314b;
ic2s10 = 0x0502, ipoib, defmember=full:
0x0021280001ef30f8,
0x0021280001ef33c0,
0x0021280001ef30b8,
0x0021280001ef314c;
sto = 0x0503, ipoib, defmember=full:
0x0021280001ef43f8,
0x0021280001ef43b7,
0x0021280001cf90c0,
0x0021280001ef43bb,
0x0021280001ef43bc,
0x0021280001cf90bf,
0x0021280001ef43b8,
0x0021280001ef43f7,
0x0021280001ef3048,
0x0021280001ef30af,
0x0021280001ef30f8,
0x0021280001ef30f7,
0x0021280001ef33c0,
0x0021280001ef33bf,
```

```
0x0021280001ef30cc,  
0x0021280001ef342b,  
0x0021280001ef30b8,  
0x0021280001ef30b7,  
0x0021280001ef314c,  
0x0021280001ef314b,  
0x0021280001efec65,  
0x0021280001efec66,  
0x0021280001efecb1,  
0x0021280001efecb2;
```

- コマンド `smpartition commit` を入力して新しい構成を適用し、構成内のすべての InfiniBand スイッチに構成の変更を伝播します。<br/>

```
# smpartition commit<br/>  
[root@aiesscsw-ib1 ~]# smpartition commit  
[root@aiesscsw-ib1 ~]#
```

7. InfiniBand スイッチからログオフします。

8. 説明されているバックアップ手順 ([http://docs.oracle.com/cd/E26698\\_01/index.html](http://docs.oracle.com/cd/E26698_01/index.html) ([http://docs.oracle.com/cd/E26698\\_01/index.html](http://docs.oracle.com/cd/E26698_01/index.html))) に従って、InfiniBand 構成をバックアップします。

## シングル IP 接続用の ZFS Storage Appliance ネットワークの構成

この構成は、外部リーフスイッチのない Oracle SPARC SuperCluster T5 にのみ使用します。最適なフェイルオーバーおよびパフォーマンスのためには、その他すべての構成でアクティブ/アクティブ構成 (次のセクション) を使用します。

次の手順を使用してポート 1 に目的の IP アドレスを設定することで、ネットワーク接続およびシンプルクラスタフェイルオーバー用の ZFS Storage Appliance InfiniBand ポートを構成します。

1. ヘッド 1 の BUI にログインして、「構成」>「ネットワーク」に移動します。
2. インタフェースの横のプラスアイコンをクリックします。「ネットワークインタフェース」ダイアログボックスが開きます。
3. 次のようにダイアログボックスの操作を完了します。
  - ネットワークインタフェースのわかりやすい名前を入力します。
  - 「インタフェースを有#化」にチェックマークが付いていることを確認します。
  - 「管理を許可」にチェックマークが付いていることを確認します。
  - 「IPv4 プロトコルを使用」にチェックマークが付いていることを確認します。

- 「構成」メニューで「#的アドレスリスト」が選択されていることを確認します。
  - その下のボックスに、目的の IP アドレスと適切なネットマスクを入力します。
  - 「IPv6 プロトコルを使用」にチェックマークが付いていないことを確認します。
  - `ibp0` のデータリンクを選択して、「適用」をクリックします。
4. データリンクとして `ibp2` を使用して、ヘッド 2 で手順 1 から 3 を繰り返します。

## アクティブ/アクティブ構成用の ZFS Storage Appliance ネットワークの構成

ZFS Storage Appliance で IP マルチパス用の InfiniBand ポートを構成します。インタフェースはアクティブ/アクティブ構成で実行されるため、プライベートストレージサブネット上で各 ZFS Storage Appliance ヘッド用に 4 つの IP アドレス (つまり合計で 8 つのアドレス) が必要です。

1. 各 InfiniBand データリンクを、自身のネットワークインタフェースとして構成します。
  - ヘッド 1 の BUI にログインして、「構成」>「ネットワーク」に移動します。
  - インタフェースの横のプラスアイコンをクリックします。「ネットワークインタフェース」ダイアログボックスが開きます。
  - 次のようにダイアログボックスの操作を完了します。
  - \* ネットワークインタフェースのわかりやすい名前を入力します。
  - \* 「インタフェースを有#化」にチェックマークが付いていることを確認します。
  - \* 「管理を許可」にチェックマークが付いていることを確認します。
  - \* 「IPv4 プロトコルを使用」にチェックマークが付いていることを確認します。
  - \* 「構成」メニューで「静的アドレスリスト」が選択されていることを確認します。
  - \* その下のボックスに「`0.0.0.0/8`」と入力します。
  - \* 「IPv6 プロトコルを使用」にチェックマークが付いていないことを確認します。
  - \* `ibp0` のデータリンクを選択して、「適用」をクリックします。
  - 残りのデータリンク (`ibp1`, `ibp2`, および `ibp3`) に対して 2 番目と 3 番目のサブ手順を繰り返します。
  - ヘッド 2 で 1 番目から 4 番目のサブ手順を繰り返します。
2. ヘッド 1 で IPMP インタフェースを構成します。
  - ヘッド 1 の BUI にログインして、「構成」>「ネットワーク」に移動します。
  - インタフェースの横のプラスアイコンをクリックします。「ネットワークインタフェース」ダイアログボックスが開きます。

- 次のようにダイアログボックスの操作を完了します。
- \* IPMP ネットワークインタフェースのわかりやすい名前を入力します。
- \* 「インタフェースを有#化」にチェックマークが付いていることを確認します。
- \* 「管理を許可」にチェックマークが付いていることを確認します。
- \* 「IPv4 プロトコルを使用」にチェックマークが付いていることを確認します。
- \* 「構成」メニューで「#的アドレスリスト」が選択されていることを確認します。
- \* 空の 4 つのボックスが表示されるように、空のボックスの横のプラス記号を 3 回クリックします。
- \* 空の各ボックスに、InfiniBand 接続用に予約されている IP アドレスの 1 つを、それぞれの /24 ネットマスク指定とともに入力します。ベストプラクティスとして、ブロックの連続する IP アドレスを使用するのではなく、1 つおきの IP アドレス (たとえば、すべて奇数またはすべて偶数) を使用します。
- \* 「IPv6 プロトコルを使用」にチェックマークが付いていないことを確認します。
- \* 「IP マルチパスグループ」ボックスにチェックマークを付けます。
- \* データリンク ibp0 および ibp3 に対応するインタフェースの横のボックスにチェックマークを付けます。
- \* 2 つのインタフェースのそれぞれが「アクティブ」に設定されていることを確認して、「適用」をクリックします。
- 「構成」>「ネットワーク」から「ルーティング」をクリックします。
- 「適#可能」に対応するマルチホーミングモデルをクリックします。

### 3. ヘッド 2 で IPMP インタフェースを構成します。

- ヘッド 2 の BUI にログインして、「構成」>「ネットワーク」に移動します。
- インタフェースの横のプラスアイコンをクリックします。「ネットワークインタフェース」ダイアログボックスが開きます。
- 次のようにダイアログボックスの操作を完了します。
- \* IPMP ネットワークインタフェースのわかりやすい名前を入力します。
- \* 「インタフェースを有#化」にチェックマークが付いていることを確認します。
- \* 「管理を許可」にチェックマークが付いていることを確認します。
- \* 「IPv4 プロトコルを使用」にチェックマークが付いていることを確認します。
- \* 「構成」メニューで「#的アドレスリスト」が選択されていることを確認します。
- \* 空の 4 つのボックスが表示されるように、空のボックスの横のプラス記号を 3 回クリックします。
- \* 空の各ボックスに、InfiniBand 接続用に予約されている残りの 4 つの IP アドレスの 1 つを、それぞれの /24 ネットマスク指定とともに入力します。これにはヘッド 1 で使用されていないものを使用します。



- \* 「IPv6 プロトコルを使用」にチェックマークが付いていないことを確認します。
- \* 「IP マルチパスグループ」ボックスにチェックマークを付けます。
- \* データリンク ibp1 および ibp2 に対応するインタフェースの横のボックスにチェックマークを付けます。
- \* 2 つのインタフェースのそれぞれが「アクティブ」に設定されていることを確認して、「適用」をクリックします。
- 「構成」>「ネットワーク」から「ルーティング」をクリックします。
- 「適#可能」に対応するマルチホーミングモデルをクリックします。

4. Oracle SPARC SuperCluster ノードとの接続を確認します。各ノードから、ZFS Storage Appliance の IPMP グループで使用されている 8 つのアドレスのそれぞれに ping を実行できることを確認します。これらの IP アドレスを、各ノードの `/etc/inet/hosts` テーブルに追加します。

## ZFS Storage Appliance ストレージプールの構成

プール構成では、物理ディスクドライブのリソースを、バックアップデータストレージの論理ストレージプールに割り当てます。システムのスループットを最大限にするために、各ドライブトレイの物理ドライブの半分を各ストレージプールに割り当てることで、サイズの等しい 2 つのストレージプールを構成します。

同じ RAID 保護プロファイルを使用して 2 つのプールを構成すると、ZFS Storage Appliance 管理ソフトウェアに効率に関する警告メッセージが表示されます。高パフォーマンスの Oracle RMAN バックアップソリューション用の構成の場合、このメッセージは無視して構いません。

## ZFS Storage Appliance のシェアの構成

シェアの構成は、クライアントアクセス用の NFS マウントポイントを設定および実行するプロセスです。Oracle SPARC SuperCluster 構成用に 2 つのプロジェクト (プールごとに 1 つのプロジェクト) を作成する必要があります。プロジェクトは、シェアの集合のためにより高いレベルの管理インタフェースポイントを提供するエンティティです。シェアの管理を最適化するには、プロジェクトに含まれるシェアのデフォルトのマウントポイントを、データベース名を参照するように更新します (`/export/dbname` など)。パフォーマンスが最適化されたシステムのためには、各プールのプロジェクトごとに 4 つのシェア、合計で 8 つのシェア (ヘッドごとに 4 つ) を作成します。プロジェクトを構成するには次の手順を実行します。

1. ヘッド 1 の BUI にログインして、「シェア」>「プロジェクト」に移動します。

2. プロジェクトの横のプラスアイコンをクリックしてプロジェクトのわかりやすい名前を入力し、「適用」をクリックします。別のヘッドで類似したプロジェクトが作成されるため、プロジェクトにはヘッド 1 の独自の名前 (H1-mydb など) を付けます。
3. 新しいプロジェクト名の横の鉛筆アイコンをクリックして、プロジェクトを編集します。
4. 「一般」をクリックして、プロパティを次のように指定します。
  - 「マウントポイント」をデータベース名を含むように変更します (たとえば、/export/H1-mydb)。
  - 「同期書き込みバイアス」を「待機時間」から「スループット」に変更して、「適用」をクリックします。
5. 「プロトコル」をクリックして、次のように NFS 例外を追加します。
  - NFS 例外の横のプラスアイコンをクリックします。
  - 「タイプ」を「ネットワーク」に変更します。
  - InfiniBand ネットワークのサブネットおよびネットマスク (たとえば、/24) を入力します。
  - 「アクセスモード」を「読み取り/書き込み」に変更します。
  - 「Charset」が「デフォルト」に設定されていることを確認します。
  - 「ルートアクセス」ボックスにチェックマークを付けて、「適用」をクリックします。
6. 「一般」の横で「シェア」をクリックします。
7. ヘッド 1 の 4 つのファイルシステムを作成し、ヘッド 2 とは異なる名前になるように独自の名前を付けます。バックアップストリームをインターリーブしてデータを 2 つのヘッドに分散させ、それによってパフォーマンスを向上させるため、ヘッド 1 には奇数番号の名前 (backup1、backup3、backup5、および backup7 など) を使用し、ヘッド 2 には偶数番号の名前 (backup2、backup4、backup6、および backup8 など) を使用します。ファイルシステムを作成するには、ファイルシステムの横のプラスアイコンをクリックしてファイルシステムの名前 (backup1) を入力し、「適用」をクリックします。この手順を繰り返して、残りの 3 つのファイルシステム (backup3、backup5、および backup7) を作成します。
8. ヘッド 2 に対して手順 1 から 7 を繰り返します。忘れずに一意のプロジェクト名 (たとえば、H2-mydb) を使用し、ファイルシステム名として偶数番号のバックアップ ID (backup2、backup4、backup6、および backup8) を指定します。

## ZFS Storage Appliance DTrace Analytics の構成

ZFS Storage Appliance には、DTrace Analytics という包括的なパフォーマンス分析ツールがあります。DTrace Analytics は、重要なサブシステムのパフォーマンスのアカウ

ンティング統計をモニターするフレームワークです。Oracle RMAN のバックアップおよび復元のワークロードの有効性とパフォーマンスに関する包括的なデータを提供するために、使用可能なアカウンティング統計のサブセットをモニターする必要があります。

ZFS Storage Appliance で高度な分析を構成 (「構成」>「設定」>「高度な分析を有効化」) していると、次の分析を使用できます。

- CPU: CPU モード別使用率
- ディスク: 操作の状態別平均 I/O 操作回数
- ディスク: 操作タイプ別 1 秒あたりの I/O バイト数
- ディスク: 待機時間別 1 秒あたりの I/O 操作回数
- ディスク: ディスク別の 95 % 以上の使用率のディスク内訳
- ネットワーク: 方向別 1 秒あたりのインタフェースバイト数
- ネットワーク: インタフェース別 1 秒あたりのインタフェースバイト数
- プロトコル: サイズ別 1 秒あたりの NFSv3 操作回数
- プロトコル: 待機時間別 1 秒あたりの NFSv3 操作回数
- プロトコル: 待機時間別 1 秒あたりの読み取りタイプ NFSv3 操作回数
- プロトコル: 待機時間別 1 秒あたりの書き込みタイプ NFSv3 操作回数
- プロトコル: サイズ別 1 秒あたりの読み取りタイプ NFSv3 操作回数
- プロトコル: サイズ別 1 秒あたりの書き込みタイプ NFSv3 操作回数

これらのアカウンティング統計を実装すると、エンドユーザーは特定の実装について、リソース消費およびサービス品質 (QoS) を即時におよび履歴から数量的に理解することができます。

## クライアント NFS マウントの構成

ZFS Storage Appliance を構成する際、アプライアンスにアクセスするすべてのサーバー (Oracle SPARC SuperCluster ノードを含む) はクライアントと見なされます。クライアントの NFS マウントの構成には、ZFS Storage Appliance にアクセスするためのターゲットディレクトリ構造の作成、および最適なシステムパフォーマンスのために必要な特定の NFS マウントオプションの作成が含まれます。Solaris クライアントのマウントオプションは次のとおりです。

```
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto=tcp,vers=3,forcedirectio
```

ZFS Storage Appliance に作成したディレクトリのマウントポイントを、Oracle SPARC SuperCluster の各ノード上に作成し、それぞれの `/etc/inet/hosts` テーブルに追加する必要があります。

## Solaris 11 ネットワークおよびカーネルのチューニング

Oracle SPARC SuperCluster の各ノードの `/etc/system` ファイルに、次のエントリを追加する必要があります。

```
set rpcmod:clnt_max_conns = 8
set nfs:nfs3_bsize = 131072
```

また、Oracle SPARC SuperCluster の各ノードをリブートするたびに、各ノードで次のコマンドを実行する必要があります。

```
/usr/sbin/ndd -set /dev/tcp tcp_max_buf 2097152
/usr/sbin/ndd -set /dev/tcp tcp_xmit_hiwat 1048576
/usr/sbin/ndd -set /dev/tcp tcp_recv_hiwat 1048576
```

最適なパフォーマンスを実現するために、追加の調整が必要になる場合があります。最新情報は、<http://support.oracle.com> (<http://support.oracle.com>) で入手可能な Oracle SPARC SuperCluster Tunables のドキュメント 1474401.1 を参照してください。また、2013 年 1 月の QFSDP リリースには、調整可能値を自動的に設定する「`ssctuner`」ツールが追加されています。詳細は、Oracle SPARC SuperCluster のリリースノートを参照してください。

## Oracle Direct NFS (dNFS) の構成

Oracle SPARC SuperCluster の各ノードで、次のように dNFS を構成します。

1. Oracle Database ソフトウェアの実行中のインスタンスをシャットダウンします。
2. ディレクトリを `$ORACLE_HOME/rdbms/lib` に変更します。
3. dNFS を有効にします。<br/>

```
make -f $ORACLE_HOME/rdbms/lib/ins_rdbms.mk dnfs_on
```

4. (`/ORACLE_HOME/dbs` にある) `oranfstab` ファイルを、構成に固有のサーバー、パス、およびエクスポート名で更新します。ここでは:<br/>

- サーバーパラメータは、InfiniBand ネットワーク上の ZFS Storage Appliance ヘッドのローカル名です。<br/>
- パスパラメータは、構成時に指定した、そのヘッドのアドレスを反映する必要があります。<br/>
- エクスポートパラメータは、`/etc/vfstab` に作成したエントリと似たマウントポイントを反映する必要があります。エントリは次のようになります。<br/>

シングル IP 構成 (外部リーフスイッチのない Oracle SPARC SuperCluster T5 のみ) の場合:

```
server: aie-zba-h1-stor
path: 192.168.30.100
export: /export/test1/backup1 mount: /zba/test1/backup1
export: /export/test1/backup3 mount: /zba/test1/backup3
export: /export/test1/backup5 mount: /zba/test1/backup5
export: /export/test1/backup7 mount: /zba/test1/backup7
server: aie-zba-h2-stor
path: 192.168.30.101
export: /export/test1/backup2 mount: /zba/test1/backup2
export: /export/test1/backup4 mount: /zba/test1/backup4
export: /export/test1/backup6 mount: /zba/test1/backup6
export: /export/test1/backup8 mount: /zba/test1/backup8<br/>
```

IPMP グループ構成 (その他すべて) の場合:

```
server: aie-zba-h1-stor
path: 192.168.30.100
path: 192.168.30.102
path: 192.168.30.104
path: 192.168.30.106
export: /export/test1/backup1 mount: /zba/test1/backup1
export: /export/test1/backup3 mount: /zba/test1/backup3
export: /export/test1/backup5 mount: /zba/test1/backup5
export: /export/test1/backup7 mount: /zba/test1/backup7
server: aie-zba-h2-stor
path: 192.168.30.101
path: 192.168.30.103
path: 192.168.30.105
path: 192.168.30.107
export: /export/test1/backup2 mount: /zba/test1/backup2
export: /export/test1/backup4 mount: /zba/test1/backup4
export: /export/test1/backup6 mount: /zba/test1/backup6
export: /export/test1/backup8 mount: /zba/test1/backup8
```

5. Oracle Database ソフトウェアインスタンスを再起動します。

## Oracle RMAN バックアップおよび復元用の Oracle Database インスタンスの調整

Oracle RMAN および ZFS Storage Appliance を使用した高帯域幅のバックアップおよび復元操作を最適化するには、I/O バッファリングを制御するインスタンスパラメータの調整が必要です。これらのパラメータを調整する方法の詳細は、記事 ID 1072545.1: バッファメモリーパラメータを使用した RMAN パフォーマンスの調整 (<http://support.oracle.com>. (<http://support.oracle.com>)) を参照してください。

Oracle SPARC SuperCluster については、次の 4 つのパラメータの調整を検討する必要があります。

- `_backup_disk_bufcnt` - バックアップセットの処理に使用するバッファの数

- `_backup_disk_bufsz` - バックアップセットの処理に使用するバッファのサイズ
- `_backup_file_bufcnt` - イメージコピーの処理に使用するバッファの数
- `_backup_file_bufsz` - イメージコピーの処理に使用するバッファのサイズ

バックアップセットおよびイメージコピーのバックアップおよび復元操作の場合、バッファの数を 64 に、バッファサイズを 1 M バイトに設定します。

```
SQL> alter system set "_backup_disk_bufcnt"=64;
SQL> alter system set "_backup_file_bufcnt"=64;
SQL> alter system set "_backup_disk_bufsz"=1048576;
SQL> alter system set "_backup_file_bufsz"=1048576;
```

SPFILE を追加することでこれらのコマンドを永続的に構成することも、バックアップおよび復元操作の実行に使用される Oracle RMAN 実行ブロック内で動的に設定することもできます。

次のコード部分は、バックアップおよび復元操作のバッファのサイズおよび数を動的に調整する方法を示しています。

- バックアップセットのバックアップ:

```
run
{<br/>
  sql 'alter system set "_backup_disk_bufcnt"=64';<br/>
  sql 'alter system set "_backup_disk_bufsz"=1048576';<br/>
  allocate channel...
...<br/>
  backup as backupset database;
}
```

- バックアップセットの復元:

```
run
{<br/>
  sql 'alter system set "_backup_disk_bufcnt"=64';<br/>
  sql 'alter system set "_backup_disk_bufsz"=1048576';<br/>
  allocate channel...
...<br/>
  restore database;
}
```

- イメージコピーのバックアップ:

```
run
{<br/>
  sql 'alter system set "_backup_file_bufcnt"=64';<br/>
  sql 'alter system set "_backup_file_bufsz"=1048576';<br/>
  allocate channel...
...<br/>
  backup as copy database;
}
```

## ■ イメージコピーの復元:

```
run
{<br/>
  sql 'alter system set "_backup_file_bufcnt"=64';<br/>
  sql 'alter system set "_backup_file_bufsz"=1048576';<br/>
  allocate channel...
...<br/>
  restore database;
}
```

増分で適用するバックアップを実行するには、増分バックアップセットを読み取り、イメージコピーに書き込む必要があります。増分で適用するバックアップ用にバッファを調整するには、次を実行します。<br/>

```
run
{<br/>
  sql 'alter system set "_backup_disk_bufcnt"=64';<br/>
  sql 'alter system set "_backup_disk_bufsz"=1048576';<br/>
  sql 'alter system set "_backup_file_bufcnt"=64';<br/>
  sql 'alter system set "_backup_file_bufsz"=1048576';<br/>
  allocate channel...
...<br/>
  recover copy of database;
}
```

## Oracle RMAN 操作の専用サービスの作成

Oracle RMAN 処理専用の 2 つのサービスを構成して、負荷分散、高可用性、およびアップグレードの管理を最適化できます。これらのサービスを、Oracle SPARC SuperCluster システムの全ノードにわたって均等に負荷分散できます。可用性およびパフォーマンスは、優先インスタンスで実行されるようにサービスを構成するとともに、それらのサービスがクラスタ内のインスタンスにフェイルオーバーされるように準備することで最適化できます。これらのサービスを構成すると、クォータラックまたはハーフラックの Oracle SPARC SuperCluster システムをアップグレードする際に、Oracle RMAN 実行ブロックの接続文字列を変更する必要がありません。

Oracle RMAN 処理用のサービスをインストールするには、`srvctl` ユーティリティを使用します。次のコード部分は、4 ノードのクラスタに均等に分散され、クラスタ内のほかのノードにフェイルオーバーされるように設定された、2 つのサービスを作成する方法を示しています。この例では、これらのサービスは `dbname` というデータベース用にインストールされ、サービスの名前は `dbname_bkup` です。

```
srvctl add service -d dbname -r dbname1 -a dbname2 -s dbname_bkup1
srvctl start service -d dbname -s dbname_bkup1
srvctl add service -d dbname -r dbname2 -a dbname1 -s dbname_bkup2
srvctl start service -d dbname -s dbname_bkup2
```

## Oracle RMAN の構成

Oracle RMAN のチャンネルおよび並列性の構成には、Oracle RMAN バックアップチャンネルのファイルシステムターゲット、およびバックアップと復元操作に使用されるチャンネルの合計数の指定が含まれます。パフォーマンスのメリットは、ZFS Storage Appliance の使用可能なシェアにまたがる 16 の Oracle RMAN チャンネルを構成することで実現できます。Oracle RMAN チャンネルを、RAC クラスタ内の Oracle Database インスタンスとノード間に均等に分散され、ZFS Storage Appliance からエクスポートされたシェア間に均等に分散されるように構成します。

次のコード部分は、バックアップセットおよびイメージコピーのバックアップと復元操作を実行し、イメージコピーに増分マージを適用するための Oracle RMAN 実行ブロックのサンプルを示しています。このサンプルコードは、次のデータベース構成をベースにしています。

- データベース名: dbname
- SYSDBA ログイン: sys/welcome
- スキャンアドレス: ad01-scan
- バックアップのサービス名: dbname\_bkup

ZFS Storage Appliance は 1 プール構成で構成でき、この構成では 8 つのマウントポイントとして使用される 8 つのシェアがアプライアンスによってエクスポートされます。

バックアップセットおよびイメージコピーを使用するバックアップおよび復元用の Oracle RMAN 実行ブロックを、以降のセクションの例で示しています。これらの例では、4 シェア構成のマウントポイントは /zfssa/dbname/backup1 から /zfssa/dbname/backup4 としてアクセスされます。またこれらは、ZFS Storage Appliance が 16 の Oracle RMAN チャンネル用の 4 つのマウントポイントとして使用される 4 つのシェアをエクスポートする構成の例です。

バックアップセットのレベル 0 バックアップ:

```
run
{<br/>
  sql 'alter system set "_backup_disk_bufcnt"=64 scope=memory';<br/>
  sql 'alter system set "_backup_disk_bufsz"=1048576 scope=memory';<br/>
  allocate channel ch01 device type disk connect 'sys/welcome@ad01-<br/>
  scan/dbname_bkup1' format '/zfssa/dbname/backup1/%U';<br/>
  allocate channel ch02 device type disk connect 'sys/welcome@ad01-<br/>
  scan/dbname_bkup2' format '/zfssa/dbname/backup2/%U';<br/>
  allocate channel ch03 device type disk connect 'sys/welcome@ad01-<br/>
  scan/dbname_bkup1' format '/zfssa/dbname/backup3/%U';<br/>
  allocate channel ch04 device type disk connect 'sys/welcome@ad01-<br/>
  scan/dbname_bkup2' format '/zfssa/dbname/backup4/%U';<br/>
  allocate channel ch05 device type disk connect 'sys/welcome@ad01-<br/>
  scan/dbname_bkup1' format '/zfssa/dbname/backup1/%U';<br/>
  allocate channel ch06 device type disk connect 'sys/welcome@ad01-<br/>
```



```

scan/dbname_bkup2' format '/zfssa/dbname/backup2/%U';<br/>
allocate channel ch07 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup3/%U';<br/>
allocate channel ch08 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup4/%U';<br/>
allocate channel ch09 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup2/%U';<br/>
allocate channel ch10 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup1/%U';<br/>
allocate channel ch11 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup4/%U';<br/>
allocate channel ch12 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup3/%U';<br/>
allocate channel ch13 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup2/%U';<br/>
allocate channel ch14 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup1/%U';<br/>
allocate channel ch15 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup4/%U';<br/>
allocate channel ch16 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup3/%U';<br/>
configure snapshot controlfile name to<br/>
'/zfssa/dbname/backup1/snapcf_dbname.f';<br/>
backup as backupset incremental level 0 section size 32g database<br/>
tag 'FULLBACKUPSET_L0' plus archivelog tag 'FULLBACKUPSET_L0';
}

```

バックアップセットのレベル 1 バックアップ:

```

run
{<br/>
sql 'alter system set "_backup_disk_bufcnt"=64 scope=memory';<br/>
sql 'alter system set "_backup_disk_bufsz"=1048576 scope=memory';<br/>
allocate channel ch01 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup1/%U';<br/>
allocate channel ch02 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup2/%U';<br/>
allocate channel ch03 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup3/%U';<br/>
allocate channel ch04 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup4/%U';<br/>
allocate channel ch05 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup1/%U';<br/>
allocate channel ch06 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup2/%U';<br/>
allocate channel ch07 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup3/%U';<br/>
allocate channel ch08 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup4/%U';<br/>
allocate channel ch09 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup2/%U';<br/>
allocate channel ch10 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup1/%U';<br/>
allocate channel ch11 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup4/%U';<br/>
allocate channel ch12 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup3/%U';<br/>
allocate channel ch13 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup2/%U';<br/>
allocate channel ch14 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup1/%U';<br/>
allocate channel ch15 device type disk connect 'sys/welcome@ad01-<br/>

```

```

scan/dbname_bkup1' format '/zfssa/dbname/backup4/%U';<br/>
allocate channel ch16 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup3/%U';<br/>
configure snapshot controlfile name to<br/>
'/zfssa/dbname/backup1/snapcf_dbname.f';<br/>
backup as backupset incremental level 1 database tag<br/>
'FULLBACKUPSET_L1' plus archivelog tag 'FULLBACKUPSET_L1';
}

```

イメージコピーのバックアップ:

```

run
{<br/>
  sql 'alter system set "_backup_file_bufcnt"=64 scope=memory';<br/>
  sql 'alter system set "_backup_file_bufsz"=1048576 scope=memory';<br/>
  allocate channel ch01 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup1/%U';<br/>
  allocate channel ch02 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup2/%U';<br/>
  allocate channel ch03 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup3/%U';<br/>
  allocate channel ch04 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup4/%U';<br/>
  allocate channel ch05 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup1/%U';<br/>
  allocate channel ch06 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup2/%U';<br/>
  allocate channel ch07 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup3/%U';<br/>
  allocate channel ch08 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup4/%U';<br/>
  allocate channel ch09 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup2/%U';<br/>
  allocate channel ch10 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup1/%U';<br/>
  allocate channel ch11 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup4/%U';<br/>
  allocate channel ch12 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup3/%U';<br/>
  allocate channel ch13 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup2/%U';<br/>
  allocate channel ch14 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup1/%U';<br/>
  allocate channel ch15 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1' format '/zfssa/dbname/backup4/%U';<br/>
  allocate channel ch16 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2' format '/zfssa/dbname/backup3/%U';<br/>
  configure snapshot controlfile name to<br/>
'/zfssa/dbname/backup1/snapcf_dbname.f';<br/>
  backup incremental level 1 for recover of copy with tag 'IMAGECOPY'<br/>
  database;
}

```

イメージコピーへの増分マージ:

```

run
{<br/>
  sql 'alter system set "_backup_disk_bufcnt"=64 scope=memory';<br/>
  sql 'alter system set "_backup_disk_bufsz"=1048576 scope=memory';<br/>
  sql 'alter system set "_backup_file_bufcnt"=64 scope=memory';<br/>
}

```

```

sql 'alter system set "_backup_file_bufsz"=1048576 scope=memory';<br/>
allocate channel ch01 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch02 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch03 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch04 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch05 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch06 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch07 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch08 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch09 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch10 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch11 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch12 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch13 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch14 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch15 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch16 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
configure snapshot controlfile name to<br/>
'/zfsa/dbname/backup1/snapcf_dbname.f';<br/>
recover copy of database with tag 'IMAGECOPY';
}

```

#### 復元の検証:

```

run
{<br/>
sql 'alter system set "_backup_disk_bufcnt"=64 scope=memory';<br/>
sql 'alter system set "_backup_disk_bufsz"=1048576 scope=memory';<br/>
sql 'alter system set "_backup_file_bufcnt"=64 scope=memory';<br/>
sql 'alter system set "_backup_file_bufsz"=1048576 scope=memory';<br/>
allocate channel ch01 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch02 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch03 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch04 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch05 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch06 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch07 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch08 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
}

```

```
allocate channel ch09 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch10 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch11 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch12 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch13 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch14 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
allocate channel ch15 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup1';<br/>
allocate channel ch16 device type disk connect 'sys/welcome@ad01-<br/>
scan/dbname_bkup2';<br/>
configure snapshot controlfile name to<br/>
'/zfssa/dbname/backup1/snapcf_dbname.f';<br/>
restore validate database;
}
```

## 次のステップ

[465 ページの「Oracle SPARC SuperCluster のバックアップ」](#)

# ZFS Storage Appliance のバックアップのための Oracle SPARC SuperCluster の構成

このセクションでは、ZFS Storage Appliance を Oracle SPARC SuperCluster に接続する方法を示すスクリプト例について説明します。これらのスクリプトは、dbname という名前のデータベースを 1 プールおよび 2 プール ZFS Storage Appliance 構成でサポートするように設計されています。

## SSC の構成 ZFS Storage Appliance のバックアップのための Oracle SPARC SuperCluster の構成

### 一般的な実装手順

実装手順は次のとおりです。

1. ホスト上にシェアをマウントするためのディレクトリ構造 (マウントポイント) を設定します。
2. ZFS Storage Appliance からエクスポートされたシェアが適切なマウントポイントにマウントされるように /etc/vfstab を更新します。

3. NFS シェアがリブート時にマウントされ、シェアのマウントおよびアンマウント処理が自動化されるように、NFS クライアントサービスを有効化します。
4. ZFS Storage Appliance からエクスポートされたシェアにアクセスするように `orandstab` ファイルを更新します。
5. ホスト上にシェアをマウントします。
6. `ORACLE_HOME` のアクセス権設定と一致するように、マウントされたシェアのアクセス権を変更します。
7. Oracle Database インスタンスを再起動して、`orandstab` ファイルに対する変更を有効にします。

## 詳細な実装手順

このセクションのトピック:

- [485 ページの「ホスト上にシェアをマウントするためのディレクトリ構造の設定」](#)
- [485 ページの「/etc/vfstab ファイルの更新」](#)
- [486 ページの「NFS クライアントサービスの有効化」](#)
- [486 ページの「ZFS Storage Appliance のエクスポートにアクセスするための `orandstab` の更新」](#)
- [487 ページの「ホスト上にシェアのマウント」](#)
- [487 ページの「マウントされたシェアの所有者の設定」](#)

### ホスト上にシェアをマウントするためのディレクトリ構造の設定

ホスト上にシェアをマウントするためのマウントポイントを次のように設定します。

```
mkdir -p /zfssa/dbname/backup1
mkdir -p /zfssa/dbname/backup2
mkdir -p /zfssa/dbname/backup3
mkdir -p /zfssa/dbname/backup4
```

### /etc/vfstab ファイルの更新

/etc/vfstab ファイルを更新するには、次のいずれかのオプションを使用します。

注: 下記のリストで UNIX の改行エスケープ文字 (`\n`) は、1 行分のコードが 2 行目に折り返していることを表します。折り返された 1 行を `fstab` に入力するときは、`\n` 文字を削除して空白で区切られた 2 つの行の断片を 1 行に結合してください。

### 1 プール構成の場合:

```
192.168.36.200:/export/dbname/backup1 - /zfssa/dbname/backup1 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
192.168.36.200:/export/dbname/backup2 - /zfssa/dbname/backup2 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
192.168.36.200:/export/dbname/backup3 - /zfssa/dbname/backup3 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
192.168.36.200:/export/dbname/backup4 - /zfssa/dbname/backup4 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
```

### 2 プール構成の場合:

```
192.168.36.200:/export/dbname/backup1 - /zfssa/dbname/backup1 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
192.168.36.201:/export/dbname/backup2 - /zfssa/dbname/backup2 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
192.168.36.200:/export/dbname/backup3 - /zfssa/dbname/backup3 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
192.168.36.201:/export/dbname/backup4 - /zfssa/dbname/backup4 \<br/>
nfs - yes rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto= \<br/>
tcp,vers=3,forcedirectio
```

## NFS クライアントサービスの有効化

次のコマンドを使用して、Solaris 11 ホスト上の NFS クライアントサービスを有効化します。

```
svcadm enable -r nfs/client
```

## ZFS Storage Appliance のエクスポートにアクセスするための oranfstab の更新

ZFS Storage Appliance のエクスポートにアクセスするように oranfstab ファイルを更新するには、次の適切なオプションを使用します。

### 1 プール構成の場合:

```
server: 192.168.36.200
path: 192.168.36.200
path: 192.168.36.201
path: 192.168.36.202
path: 192.168.36.203
```

```
export: /export/dbname/backup1 mount: /zfssa/dbname/backup1
export: /export/dbname/backup2 mount: /zfssa/dbname/backup2
export: /export/dbname/backup3 mount: /zfssa/dbname/backup3
export: /export/dbname/backup4 mount: /zfssa/dbname/backup4
```

2 プール構成の場合:

```
server: 192.168.36.200
path: 192.168.36.200
path: 192.168.36.202
export: /export/dbname/backup1 mount: /zfssa/dbname-2pool/backup1
export: /export/dbname/backup3 mount: /zfssa/dbname-2pool/backup3
server: 192.168.36.201
path: 192.168.36.201
path: 192.168.36.203
export: /export/dbname/backup2 mount: /zfssa/dbname-2pool/backup2
export: /export/dbname/backup4 mount: /zfssa/dbname-2pool/backup4
```

## ホスト上にシェアのマウント

標準の Solaris mount コマンドを使用して、シェアを手動でマウントします。

```
# mount /zfssa/dbname/backup1
# mount /zfssa/dbname/backup2
# mount /zfssa/dbname/backup3
# mount /zfssa/dbname/backup4
```

## マウントされたシェアの所有者の設定

ORACLE\_HOME のアクセス権設定と一致するように、マウントされたシェアのアクセス権の設定を変更します。この例では、ユーザーおよびグループの所有者が oracle:dba に設定されています。

1. 次のように入力します。  
# chown oracle:dba /zfssa/dbname/\*
2. 次のいずれかのオプションを使用して Oracle Database インスタンスを再起動して、oranfstab ファイルに対する変更を有効にします。

■ 一度に 1 つのインスタンスを再起動します (順次アップグレード)。例:

```
■ :$ srvctl stop instance -d dbname -i dbname1
■ :$ srvctl start instance -d dbname -i dbname1
■ :$ srvctl stop instance -d dbname -i dbname2
■ :$ srvctl start instance -d dbname -i dbname2
■ :$ srvctl stop instance -d dbname -i dbname3
■ :$ srvctl start instance -d dbname -i dbname3
```

- `:$ srvctl stop instance -d dbname -i dbname4`
- `:$ srvctl start instance -d dbname -i dbname4`
- `:$ srvctl stop instance -d dbname -i dbname5`
- `:$ srvctl start instance -d dbname -i dbname5`
- `:$ srvctl stop instance -d dbname -i dbname6`
- `:$ srvctl start instance -d dbname -i dbname6`
- `:$ srvctl stop instance -d dbname -i dbname7`
- `:$ srvctl start instance -d dbname -i dbname7`
- `:$ srvctl stop instance -d dbname -i dbname8`
- `:$ srvctl start instance -d dbname -i dbname8`
- データベース全体を再起動します。例:
- `:$ srvctl stop database -d dbname`
- `:$ srvctl start database -d dbname`

## Oracle Intelligent Storage Protocol

Oracle Database の階層化されたアーキテクチャーには、Oracle Disk Manager (ODM) が含まれます。ODM の提供するファイル管理モジュールは、Oracle Database がローカルファイルシステム、raw ディスクパーティション、または NFS サーバーを使ってデータベース情報を格納することを可能にします。

データベースのパフォーマンスを向上させるため、ODM インタフェースは Oracle Database が各入出力要求に合わせて情報を渡すことを可能にします。この情報は、入出力要求に関連付けられたファイルタイプなど、入出力に関連する複数の属性を定義します。これにより、データファイルおよびデータベースログファイルの書き込みは異なる方法で処理されます。

新しい OISP を使用すると、Oracle Database NFSv4 クライアントが ODM 最適化情報を ZFS Storage Appliance の NFSv4 サーバーに渡すことができます。ZFS Storage Appliance は、ODM 最適化情報を活用してデータベース構成を簡略化し、データベースのパフォーマンスを一層向上させます。

Oracle Intelligent Storage Protocol には 2 つの機能があります。

- 新しいデータベースファイルに最適なファイルレコードサイズを自動的に設定します
- 各書き込み要求に最適な書き込みバイアスを自動的に使用します (ZFS の「待機時間」または「スループット」)



## 最適なファイルレコードサイズを設定します

Oracle dNFS クライアントは、NFSv4 書き込み要求ごとに最適なレコードサイズを ZFS Storage Appliance に設定します。ZFS Storage Appliance NFSv4 サーバーは、入出力要求付きのレコードサイズを ZFS ファイルシステムに渡します。次に、ZFS ファイルシステムは、デフォルトのファイルシステムレコードサイズを無視して、入出力要求付きで渡されたレコードサイズ値を使用します。レコードサイズは、新たに作成されたファイルでのみ設定できます。ファイルがすでに存在する場合は、レコードサイズは変更されません。

## 要求ごとに ZFS の「待機時間」または「スループット」書き込みモードのいずれかを使用します

Oracle dNFS クライアントは、NFSv4 書き込み要求ごとに最適な書き込みバイアスを ZFS Storage Appliance に設定します。ZFS Storage Appliance NFSv4 サーバーは、入出力要求付きの書き込みバイアスを ZFS ファイルシステムに渡します。次に、ZFS ファイルシステムは、デフォルトのファイルシステム書き込みバイアスを無視して、入出力要求付きで渡された書き込みバイアス値の使用を試みます。ZFS ファイルシステムの状態によっては、入出力要求付きで送信された書き込みバイアスが無視されることがあります。

## Oracle Solaris Cluster 用 Sun ZFS Storage Appliance ネットワークファイルシステムプラグイン

Oracle Solaris Cluster (OCS) は、Solaris オペレーティングシステム用の高可用性クラスタソフトウェア製品です。

Oracle Solaris Cluster 用 Sun ZFS Storage Appliance ネットワークファイルシステムプラグインを使用すると、NFS プロトコルを使用して OSC を Sun ZFS Storage Appliance で有効にできます。このプラグインと readme ファイルは、Oracle Technology Network で Oracle Solaris Cluster 用 Sun ZFS Storage Appliance ネットワークファイルシステムプラグインの一部として入手できます。

## Oracle Solaris Cluster Geographic Edition 用 Sun ZFS Storage Appliance プラグイン

Oracle Solaris Cluster Geographic Edition ソフトウェアは、Oracle Solaris Cluster ソフトウェアの階層化された拡張です。Geographic Edition ソフトウェアは、長距離を隔

てられた複数のクラスタを使用することにより、およびこれらのクラスタサイト間でデータをレプリケートする冗長インフラストラクチャーを使用することにより、アプリケーションを予期しない中断から保護します。このプラグインは、Sun ZFS Storage Appliance のリモートレプリケーションサービスを使用して、リモート Oracle Solaris Cluster サイト間のデータレプリケーションを調整します。

このプラグインパッケージは、Oracle Technology Network の Sun NAS ストレージ情報ページから入手できます。

## Oracle Enterprise Manager Grid Controller 用 Sun ZFS Storage Management プラグイン

Oracle Enterprise Manager Grid Controller 用 Sun ZFS Storage プラグインは、Sun ZFS Storage アプライアンスファミリ用のグリッドコントローラ環境に対して、次のような優れたモニタリング機能を提供します。

- Sun ZFS Storage アプライアンスのモニター
- ストレージシステム情報と、アクセス可能なストレージコンポーネントの構成情報およびパフォーマンス情報の収集
- ツールによって収集されたモニタリング情報としきい値に基づく警告および違反の表示
- 分析機能を補完する既製のレポートの提供
- リモートエージェントによるモニタリングのサポート。

アプライアンスがグリッドコントローラによってモニターされるように構成されると、分析機能のワークシートおよびデータセットが作成されます。これにより、グリッドコントローラ管理者の視点が、アプライアンス内部で使用可能なリアルタイム分析によって提供される詳細なレベルまで到達します。

管理プラグインは次のリンク (Oracle Technology Network) から入手できます。

プラグインと一緒にパッケージされているインストールガイドは、グリッドコントローラ管理者と、モニター対象アプライアンスのストレージ管理者の両方が読む必要があります。

各アプライアンスには 2 つの[435 ページの「ワークフロー」](#)が付属しており、これらはそれぞれモニタリング対象システムの準備と、モニタリング環境用に作成された成果物の削除に使用されます。

- Oracle Enterprise Manager Monitoring の構成
- Oracle Enterprise Manager Monitoring の構成解除

これらのワークフローは、ブラウザのユーザーインターフェースの[435 ページ](#)の「[「保守」](#) > [「ワークフロー」](#)」ページからアクセスできます。

## Oracle Enterprise Manager Grid Controller 用 Oracle Grid Controller Sun ZFS Storage Management プラグイン

### Oracle Enterprise Manager Monitoring の構成

このワークフローは、環境をモニタリング用に準備したり、ワークフローによって作成された成果物が操作中にストレージ管理者によって変更された場合に成果物を元の状態にリセットしたりするために使用されます。このワークフローを実行すると、システムに次の変更が加えられます。

- システムへの制限付きアクセス権を持つ *oracle\_agent* [140 ページ](#)の「[ロールのプロパティ](#)」が作成されます。Oracle Enterprise Manager Grid Controller エージェントはモニタリングに必要な情報を取得できますが、システムに変更を加えることはできません。*oracle\_agent* [第7章「ユーザー構成」](#)が作成されて、このロールが割り当てられます。このロールとユーザーを使用することは、エージェントがアプライアンスにアクセスした時期と方法についての明確な監査記録を維持する上で極めて重要です。
- 高度な分析が使用可能になり、Sun ZFS Storage アプライアンスのすべてのユーザーが、拡張された統計のセットを使用できるようになります。
- *Oracle Enterprise Manager* というワークシートが作成され、グリッドコントローラ管理者とストレージ管理者の間のコミュニケーションが促進されます。グリッドコントローラによってモニターされるすべてのメトリックをこのワークシートから使用できます。

### Oracle Enterprise Manager Monitoring の構成解除

このワークフローは、*Oracle Enterprise Manager Monitoring の構成* によって作成された成果物を削除します。特に、次のことを行います。

- *oracle\_agent* ロールおよびユーザーを削除します
- *Oracle Enterprise Manager* ワークシートを削除します。

このワークシートは、高度な分析や、収集目的でアクティブ化されたあらゆるデータセットを無効にするものではありません。

## Sun ZFS Storage Appliance 用 Oracle Virtual Machine Storage Connect プラグイン

Oracle VM 3.0 内で導入された多くの新機能の 1 つが Storage Connect フレームワークです。このフレームワークを使用すると、Oracle VM 3.0 Manager がストレージサーバーに直接アクセスしてリソースをプロビジョニングできます。このフレームワークを使用すると、ストレージサーバーの登録、既存ストレージリソースの発見、物理ディスクの作成とサーバープールでの参照、ストレージリポジトリおよび仮想マシンの共有を行えます。

Sun ZFS Storage Appliance 用 Oracle Virtual Machine Storage Connect プラグインは、Oracle VM が仮想化のために Sun ZFS Storage Appliance をプロビジョニングおよび管理できる Oracle VM ソフトウェアスイートのコンポーネントです。プラグインは Oracle VM サーバーにインストールされて、ZFSSA にインストールされているワークフローを介してストレージサーバーと通信します。

このプラグインおよび readme ファイルは、Oracle Technology Network で入手できません。

## ボリュームシャドウコピーサービスソフトウェア用 Sun ZFS Storage Appliance プロバイダ

Microsoft オペレーティングシステム用のボリュームシャドウコピーサービス (VSS) は、システム上のアプリケーションがボリュームへの書き込みを続行している間にボリュームバックアップを実行できるようにするためのフレームワークを提供します。VSS が提供する一貫性のあるインタフェースによって、ディスク上のデータを更新するユーザーアプリケーション (VSS 書き込み側) と、アプリケーションをバックアップするユーザーアプリケーション (VSS リクエスト側) との間で調整が可能になります。VSS は特に次のものを提供します。

- アプリケーションとファイルシステムのアクティビティを調整するバックアップインフラストラクチャ
- シャドウコピーと呼ばれる、特定の時点での合体したコピーを作成するための場所

Sun ZFS Storage Appliance Provider For Volume Shadow Copy Service Software は、Sun ZFS Storage Appliance が、ブロックターゲットを使用している Windows ホストの一貫性のあるスナップショットを作成できるようにする VSS ハードウェアプロバイダです。VSS は、ブロックデータが一貫性を保つようにスナップショットを調整します。プロバイダはアプライアンス上のワークフローのセットと通信して、アプリケーションから見たスナップショットの取得を調整します。これは iSCSI とファイバチャネルの両方で動作します。

Sun ZFS Storage Appliance Provider For Volume Shadow Copy Service Software は、この機能とアプリケーション間の調整とを必要とするホスト上にインストールします。このアプリケーション統合のためのドキュメント全体は、ダウンロードしたコンポーネントに ReadMe ファイルの形式でパッケージされています。プロバイダソフトウェアと readme ファイルは、Oracle Technology Network の Sun ZFS Storage 7000 ソフトウェアプロバイダ、およびプラグインパッチの一部として入手できます。VSS の詳細な情報は Microsoft Web サイトにあり、この [<http://msdn.microsoft.com/en-us/library/aa384649> (<http://msdn.microsoft.com/en-us/library/aa384649>)%28VS.85%29.aspx 概要] の説明も記載されています。

## Symantec の「DMP」/Storage Foundation での FC サポート

- SF - Symantec Storage Foundation 5.1
- SF HA - Storage Foundation High Availability 5.1
- SFCFS/SF Oracle RAC - Storage Foundation Cluster File System/Storage Foundation for Oracle RAC 5.1
- SFCFS/SFCFS Oracle RAC - Storage Foundation Cluster File System/Storage Foundation Cluster File System for Oracle RAC 5.1

## 次の OS バージョンにおける Symantec Storage Foundation 5.1RP2 以降の FC サポート

- Solaris 10 SPARC
- Solaris 10 x86
- Linux RedHat5
- Oracle Enterprise Linux (OEL)

Symantec の HCL (<http://www.symantec.com/business/support/index> (<http://www.symantec.com/business/support/index>)?page=content&id=TECH74012) を参照してください

次の制限に注意してください。

- Symantec では 7000 ASL (<https://vos.symantec.com/asl> からダウンロード可能) がインストールされていることを「要求」しています
- Symantec では SF 5.1 VM パッチレベル 5.1RP2 以上 (<https://vos.symantec.com/patch/matrix> からダウンロード可能) がインストールされていることも要求しています

- Symantec では次の DMP パラメータ設定も要求しています (「クラスタ化」された 7000 のみ)。
- :dmp\_health\_time=0
- :dmp\_path\_age=0
- :dmp\_lun\_retry\_timeout=200

「クラスタ化」された 7000 設定について説明している Symantec の HW テクニカルノート (<http://www.symantec.com/business/support/index> (<http://www.symantec.com/business/support/index>)? page=content&id=TECH47728) を参照してください

Symantec の Storage Foundation 5.1SP2 for Windows では、次の Windows バージョンにおける 7000 シリーズに対する FC 接続をサポートしています。

- Windows Server 2003
- Windows Server 2008
- Windows Server 2008 R2

SF 5.1SP2 HCL (<http://www.symantec.com/business/support/index> (<http://www.symantec.com/business/support/index>)? page=content&id=TECH138719) を参照してください

## Sun ZFS Storage 7000 Storage Replication Adapter for VMware Site Recovery Manager

Sun ZFS Storage 7000 Storage Replication Adapter (SRA) for VMware vCenter Site Recovery Manager (SRM) は、複数のサイトにまたがっていて、保護されているサイトのサービス中断時には迅速な復旧が必要な VMware の配備に、Sun ZFS Storage 7000 アプライアンスを統合します。SRA を既存の VMware vCenter SRM 環境に導入すると、復旧計画がテストされ実行されるときに VMware vCenter SRM の発見、テスト、フェイルオーバーシーケンスを通して Sun ZFS Storage 7000 アプライアンスを管理できます。SRA の使用は、完全に VMware vCenter SRM アプリケーション内で発生します。

VMware 管理者は、VMware データストアをホストしているアプライアンスを担当する Sun ZFS Storage 7000 アプライアンス管理者と密に協力する必要があります。詳細は、SRA に同梱されている Sun ZFS Storage 7000 SRA for VMware SRM の管理に関するガイドを参照してください。

注: SRA は Oracle Technology Network からダウンロードできます。SRA を入手するには、Sun ZFS Storage 7000 アプライアンスの有効な Oracle サポート契約が必要です。





# 索引

---

## 数字・記号

Active Directory, 254, 254, 254, 255, 255, 256  
ドメインへの参加, 258  
ワークグループへの参加, 258  
DNS, 267, 267, 267, 269, 269, 269  
DTrace, 474  
FC, 114, 114, 114, 115, 115, 116, 117, 117, 117, 118, 120  
FTP  
シェアへの FTP アクセスの許可, 228  
HTTP, 228, 229, 229, 229  
シェアへの HTTP アクセスの許可, 230  
iSCSI  
分析ワークシートの作成, 127  
LDAP, 250, 250, 251, 253  
アプライアンス管理者の追加, 253  
NFS  
NFS を介したファイルシステムのシェア, 207  
NIC, 71, 79, 79, 93, 94  
NIS, 248, 249, 249  
NIS からのアプライアンス管理者の追加, 249  
NTP  
BUI のクロック同期, 274  
SFTP  
シェアへの SFTP アクセスの許可, 241  
リモートアクセス用の SFTP サービスの構成, 242  
Share, 310  
SMB  
Active Directory の構成, 225  
SMB データサービスの構成, 226  
初期構成, 223  
プロジェクトとシェアの構成, 225  
SNMP  
アプライアンスのステータスを配信するよう SNMP を構成, 284  
トラップを送信するよう SNMP を構成, 284  
SRP  
iSER ターゲット構成, 128  
SRP ターゲット構成, 133

## SSH

root による SSH アクセスの無効化, 292  
TFTP  
シェアへの TFTP アクセスの許可, 244

## あ

アイデンティティマッピング  
アイデンティティマッピングの構成, 266  
マッピングの表示またはフラッシュ, 267  
ウイルススキャン  
シェアに対するウイルススキャンの構成, 247

## か

警告, 151, 151, 155, 156  
警告アクションの追加, 156, 157  
しきい値警告の追加, 155, 156

## さ

シェア, 309, 314  
ストレージ  
既存のプールへのキャッシュデバイスの追加, 105, 105  
ストレージプールの構成, 104  
スナップショット, 346, 347, 347, 347, 351, 352  
設定  
アクティビティのしきい値の変更, 58  
表示されるアクティビティ統計情報の変更, 57

## た

ダッシュボード, 46, 46, 46, 48, 49, 50, 53, 54  
ダッシュボードの連続実行, 54

## な

ネットワーク

Infiniband パーティションのデータリンクとインタフェースの作成, 92  
IPMP グループの拡張, 92  
LACP アグリゲーションの拡張, 92  
LACP 集合リンクインタフェースの作成, 90  
インタフェースの変更, 89  
クラスタ化されたコントローラでの VLAN ID なしの VNIC の作成, 93  
クラスタ化されたコントローラでの同じ VLAN ID の VNIC の作成, 94  
シングルポートインタフェースの作成, 89  
シングルポートインタフェースの作成、ドラッグ & ドロップ, 89  
静的ルートの削除, 95, 95  
静的ルートの追加, 95, 95  
プローブベースのリンク状態障害検出を使用した IPMP グループの作成, 90  
マルチホーミングプロパティの「厳しい」への変更, 96  
リンク状態のみの障害検出を使用した IPMP グループの作成, 91

## は

プール, 97  
プロジェクト, 356

## ま

マストヘッド, 24, 24

## や

ユーザー  
管理者の追加, 141, 144  
ダッシュボードのみを表示できるユーザーの追加, 142  
ロールからの承認の削除, 142, 146  
ロールの追加, 141, 145  
ロールへの承認の追加, 142, 145

## ら

リモートレプリケーション, 375, 380, 380, 403