



shaping tomorrow with you

Oracle Solaris 10

ZFS 構築/運用手順書

2011年11月

第1.2版

富士通株式会社

■使用条件

- 著作権・商標権・その他の知的財産権について

コンテンツ(文書・画像・音声等)は、著作権・商標権・その他の知的財産権で保護されています。

本コンテンツは、個人的に使用する範囲でプリントアウトまたはダウンロードできます。ただし、これ以外の利用(ご自分のページへの再利用や他のサーバへのアップロード等)については、当社または権利者の許諾が必要となります。

- 保証の制限

本コンテンツについて、当社は、その正確性、商品性、ご利用目的への適合性等に関して保証するものではなく、そのご利用により生じた損害について、当社は法律上のいかなる責任も負いかねます。本コンテンツは、予告なく変更・廃止されることがあります。

■商標について

- UNIX は、米国およびその他の国におけるオープン・グループの登録商標です。
- SPARC Enterprise, SPARC64 およびすべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している、同社の米国およびその他の国における商標または登録商標です。
- Oracle と Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。
- その他各種製品名は、各社の製品名称、商標または登録商標です。

はじめに

本書の内容

- 本書は、Oracle Solaris 10 の ZFS でのインストール後の管理、運用の手順について解説しています。インストール環境は SPARC Enterprise シリーズを対象としています。
- 本文中で使用される用語の詳細については下記ドキュメントを参考にしてください。
[Oracle オンラインドキュメント]『Oracle Solaris ZFS 管理ガイド』
<http://download.oracle.com/docs/cd/E19253-01/819-6260/819-6260.pdf>

留意事項

- 本書では、Oracle Solaris を Solaris と記載することがあります。
- 本手順書に記載の設定値(ホスト名、IP アドレス等)は参考例です。実際のシステム環境に応じて読み替えてください。

目 次

注意事項	1
1. ZFS 概要	2
1-1. ZFS 概要	3
1-1-1. ZFS の特長	3
1-1-2. 便利な機能	3
1-2. ZFS システム要件	4
2. ZFS ファイルシステムのインストール	5
2-1. ZFS ルートファイルシステムのインストール概要	5
2-2. ZFS ルートファイルシステムのインストール要件	5
3. OS セットアップ	7
3-1. ターミナルの選択	7
3-2. ネットワーク接続の有無	7
3-3. ネットワークインタフェースの構成	7
3-4. DHCP の使用有無	8
3-5. ホスト名の設定	8
3-6. IP アドレスの設定	8
3-7. サブネットの設定	8
3-8. IPv6 使用の有無	8
3-9. デフォルトルートの設定	8
3-10. 設定情報の確認	9
3-11. セキュリティポリシーの設定	9
3-12. ネームサービスの選択	9
3-13. NFSv4 ドメイン名の設定	10
3-14. 時間帯の設定	10
3-15. 日付と時刻の設定	10
3-16. 情報の確認	10
3-17. root パスワードの設定	10
3-18. Identify This System	11
3-19. インストール方法の選択	11

3-20. ライセンスへの同意.....	12
3-21. 使用文字コードの選択.....	12
3-22. インストールソフトウェアの選択.....	13
3-23. ファイルシステムの選択.....	14
3-24. ソフトウェアグループの選択.....	14
3-25. ディスク/ファイルシステムの設定.....	14
3-25-1. インストールディスクの選択.....	14
3-25-2. 既存データの保存.....	15
3-25-3. ZFS 設定の構成.....	15
3-25-4. リモートファイルシステムのマウント.....	15
3-26. boot-device の設定.....	16
4. ZFS の運用.....	18
4-1. ストレージプールの作成.....	18
4-2. ファイルシステムの作成.....	21
4-3. プロパティの設定.....	22
4-4. ZFS ファイルシステムのマウント、アンマウント.....	23
4-5. ZFS ファイルシステムの共有、共有解除.....	26
4-6. デバイスの追加(add).....	28
4-7. デバイスの接続/切り離し(attach/detach).....	30
4-7-1. デバイスの接続(attach).....	30
4-7-2. デバイスの切り離し(detach).....	32
4-8. デバイスのオンライン/オフライン(online/offline).....	33
4-8-1. デバイスのオフライン.....	33
4-8-2. デバイスのオンライン.....	34
4-9. エラーのクリア(clear).....	36
4-10. デバイスの置き換え(replace).....	36
4-11. ホットスペア(spare).....	39
4-12. ストレージプールの移行(export/import).....	42
4-12-1. エクスポート.....	42
4-12-2. インポート.....	43
4-13. ZFS スナップショット.....	44
4-13-1. スナップショットの作成.....	45
4-13-2. ロールバック(rollback).....	46
4-13-3. クローン(clone).....	46

4-14. ZFS データの送信/受信 (send/receive)	48
4-14-1. スナップショットの送信	48
4-14-2. ZFS スナップショットの受信.....	49
4-15. ZFS データの完全性をチェックする (scrub)	50
4-16. Oracle Solaris 10 OS 9/10 (U9)からの新機能.....	51
4-16-1. ミラーディスクの切り離し(split)	51
4-16-2. ZFS スナップショットの保護(hold)	54
4-17. トラブルシューティング	57
4-17-1. 損傷した ZFS 構成の修復	57
4-17-2. 見つからないデバイスを使用可能する.....	57
4-17-3. 損傷したデバイスを修復する.....	57
4-17-4. 損傷したデータを修復する	58
4-18. 起動できないシステムの修復	60
付録1. プロパティの紹介	61

注意事項

本手順書で説明されている ZFS 操作を実施する前に、最新の注意事項を確認してください。必要なパッチが適用されていないと、ZFS に関するコマンドが正常に実施できないなどの不具合が発生することがあります。

- SPARC Enterprise ハードウェアプラットフォームガイド
<http://primeserver.fujitsu.com/sparcenterprise/manual/solaris.html>
- システムに適用されているパッチ ID を調べる方法

```
# showrev -p
```

1. ZFS 概要

本手順書は Solaris 10 5/09 以降に対応しています。Solaris 10 6/06 以降の ZFS に関連する主な新機能として以下のものがあります。

【Solaris 10 6/06 からの新機能】

1) ZFS ファイルシステムの追加

【Solaris 10 11/06 からの新機能】

2) パフォーマンスの向上。以下の機能強化

- ・ホットスペア
- ・RAIDZ2
- ・クローンプロモーション
- ・高速スナップショット

【Solaris 10 8/07 からの新機能】

3) Solaris iSCSI ターゲットデバイスとして ZFS の指定が可能に

4) ファイルシステムの共有プロセス改善、zfs、zpool コマンドのコマンド実行履歴、および構成オプションとプロパティの追加

【Solaris 10 10/08 からの新機能】

5) ZFS ルートファイルシステム

【Solaris 10 5/09 からの新機能】

6) ZFS クローンを使用したゾーンのクローン化

【Solaris 10 10/09 からの新機能】

7) ファイルシステム、ユーザ、グループごとに割り当て制限を設定可能

ZFS ルートファイルシステムのフラッシュアーカイブを使用して JumpStart が可能

ZFS の L2ARC に高速なキャッシュとして SSD(Solid State Disk)を使用可能

【Solaris 10 9/10 からの新機能】

8) autoexpand プロパティ、zpool split コマンド、zfs hold コマンド、

zpool clear コマンドおよび zpool import コマンドにオプション-F の追加

zfs remove コマンドによる log device の切り離しが可能

ストレージプール単位に ZFS I/O 処理用プロセスを使用することで

各ストレージプールの CPU 使用状況が確認可能

logbias プロパティに throughout を指定することが可能

1-1.ZFS 概要

ZFS は拡張性を実現しながら、管理しやすいファイルシステムとして設計されています。また、高いデータの堅牢性を保持したファイルシステムです。

1-1-1.ZFS の特長

■ 拡張性

- ・ 世界初の 128bit のファイルシステム
- ・ 事実上無限大といえるファイルシステムを構築可能

■ 管理のしやすさ

- ・ 物理 DISK を集約し、1 つの仮想 DISK として管理・提供
- ・ ストレージプールが足りなくなれば、ディスクを追加することにより※EFI ラベルが適用されストレージプール自体を拡張可能
- ・ ボリューム管理は不要

※ラベルについて

ラベル	説明
SMI (VTOC)	従来のラベル。サイズが 1T バイトに満たないディスク用です。
EFI (Extensible Firmware Interface)	64 ビット Solaris カーネルを実行しているシステムで 1T バイトを超えるディスクをサポートします。

■ データの堅牢性

- ・ データとチェックサムが分離しているので、チェックサム自身の損傷にも対応可能。
- ・ ミラー、RAID-Z、RAID-Z2、RAID-Z3 の冗長機能を提供。

種類	機能	ディスクの最低必要本数
ミラー	通常のみラー構成	2 本
RAID-Z	シングルパリティの RAID-Z 構成	2 本
RAID-Z2	ダブルパリティの RAID-Z 構成	3 本
※RAID-Z3	トリプルパリティの RAID-Z 構成	4 本

※ RAID-Z3 は、Solaris 10 OS 9/10 (U9)よりサポート

- ・ 実データを上書きせず、常にコピーブロックに対して変更する Copy on Write とトランザクション処理。

1-1-2.便利な機能

- ・ ルート(/)ファイルシステムにも ZFS が使用可能 (Solaris 10 10/08 以降)
- ・ 瞬時にファイルシステムを退避(スナップショット)、ロールバックも可能
- ・ 瞬時にファイルシステムを複製(クローン)

1 - 2 .ZFS システム要件

ZFS を使用する前に、次の要件が満たされていることを事前に確認してください。

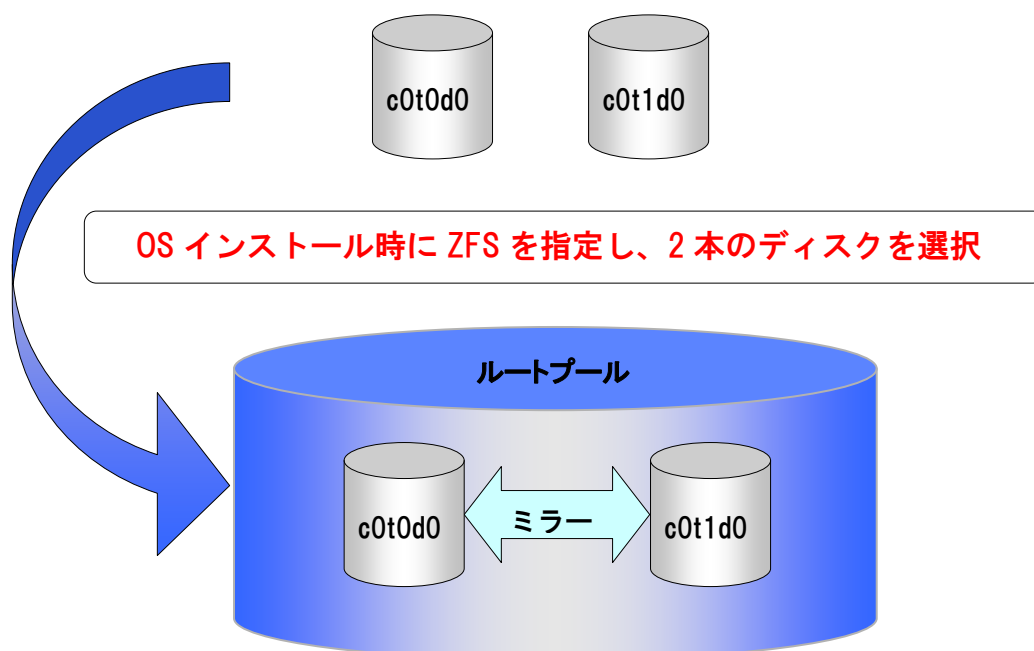
- Solaris 10 6/06 以降のリリースが稼働している SPARC システム。
- 最小ディスク容量は 128M バイトです。ストレージプールに必要な最小ディスク容量は、約 64M バイトです。
- Solaris システムのインストールに推奨される最小メモリ容量は 768M バイトです。ただし、十分な ZFS パフォーマンスを実現するために、1G バイト以上のメモリを搭載することをお勧めします。
- ミラーディスク構成を作成する場合は、複数のコントローラを使用することをお勧めします。

2. ZFS ファイルシステムのインストール

本手順書では、ZFS ファイルシステムで 2 本のディスクからなるミラー構成の Solaris10 インストールを実施します。SPARC Enterprise のコンソール接続環境 (ALOM、ILOM、XSCF) は、事前に設定しておいてください。

2-1.ZFS ルートファイルシステムのインストール概要

Solaris10 10/08 以降のリリースでは、ZFS ルートファイルシステムからインストールと起動を行うことができます。また、インストール時に 2 つ以上のディスクを選択することにより、ミラー化された ZFS ルートプールを設定できます。



2-2.ZFS ルートファイルシステムのインストール要件

システムを ZFS ルートファイルシステムでインストールする場合は、次の要件が満たされていることを事前に確認してください。

- ZFS ルートファイルシステムのインストールは Solaris10 10/08 以降のリリースで可能です。

「ルートプール」として使用する ZFS ストレージプールは、次の要件を満たす必要があります。

- ルートプールに使用するディスクには VTOC (SMI) ラベルが必要です。また、ディスクスライスを使用してプールを作成する必要があります。
- ルートプールは、ミラー化構成または単一ディスク構成として作成する必要があります。ディスクを追

加して複数のミラー化仮想デバイスを作成することはできませんが、zpool attach コマンドを使用して 1 つのミラー化仮想デバイスを拡張することができます。

- RAID-Z やストライプ化構成はサポートされていません。
- ルートプールに個別のログデバイスを使用することはできません。
- ZFS ルート環境にはスワップデバイスおよびダンプデバイスとして個別のデバイスが必要です。そのため ZFS ルートファイルシステムに最小限必要なプール容量は UFS ファイルシステムの場合よりも大きくなります。
- ZFS ルートファイルシステムのインストールに必要な最小メモリ容量は 768M バイトですが、全体的なパフォーマンスを向上させるには、1G バイトのメモリの搭載をお勧めします。
- 推奨される最小ディスク容量は 16G バイトです。容量は次のように消費されます。
 - ・ 初期インストール時のスワップボリュームのデフォルトサイズは物理メモリサイズの 1/2(ただし、2G バイト以下かつ 512M バイト以上)ですが、初期インストール時に調整が可能です。
 - ・ ダンプボリュームのデフォルトサイズは、dumpadm の情報と物理メモリのサイズに基づいて、カーネルによって計算されますが、初期インストール時に調整が可能です。
- プールは、1つのディスクスライス上または、ミラー化された複数のディスクスライス上に存在する必要があります。
- ZFS ルートプールで起動用として指定するディスクのサイズは 1T バイト以下でなければなりません。

3. OS セットアップ

本章では、2本のディスクからなるミラー構成の Solaris10 OS を ZFS でインストールする手順について解説します。

- 1) Solaris10 OS DVD を DVD-ROM に挿入し、CD ブートします。

```
ok boot cdrom
```

参考) ネットワークインストールの場合は、ネットワークブートします。

```
ok boot net
```

3-1. ターミナルの選択

- 1) 「Select a Language」画面では、ターミナルで表示する言語を選択します。「4」を入力して、[Enter] キーを押します。
- 2) 「どのタイプの端末を使用していますか？」画面では、使用している端末を選択します。「3」を入力して、[Enter] キーを押します。
- 3) 「Solaris インストールプログラム」画面では、インストール中の画面操作についての留意事項が表示されます。[Esc] キーを押して、[2] キーを押します。
- 4) 「システムを確認してください。」画面では、インストールの確認画面が表示されます。[Esc] キーを押して、[2] キーを押します。

3-2. ネットワーク接続の有無

- 1) 「ネットワーク接続性」画面では、ネットワークに接続するか否かを選択します。
※注意 ネットワークインストールの場合は下記の設定は省略されます。
- 2) 「はい」に「X」マークが付いたことを確認し、[Esc] キーを押して、[2] キーを押します。

3-3. ネットワークインタフェースの構成

- 1) 「複数のネットワークインタフェースの構成」画面では、OS インストールと同時に構成する NIC を設定する。1 つ以上選択する必要があります。
※注意 表示されるネットワークインタフェース情報は使用サーバの環境によって異なります。
- 2) 「e1000g0」に「X」マークが付いたことを確認し、[Esc] キーを押して、[2] キーを押します。

3-4.DHCP の使用有無

1) 「e1000g0 用の DHCP」画面では、e1000g0 の IP アドレス割り当てに DHCP を使用するか否かを選択します。

※注意 ネットワークインストールの場合、この設定は省略されます。

2) 「いいえ」に画面に「X」マークが付いたことを確認し、[Esc]キーを押して、[2]キーを押します。

3-5.ホスト名の設定

1) 「e1000g0 用のホスト名」画面では、e1000g0 に定義するホスト名を設定します。

※注意 ネットワークインストールの場合、この設定は省略されます。

2) ホスト名を入力し、[Esc]キーを押して、[2]キーを押します。

3-6.IP アドレスの設定

1) 「e1000g0 用の IP アドレス」画面では、e1000g0 に割り当てる IP アドレスを設定します。

※注意 ネットワークインストールの場合、この設定は省略されます。

2) IP アドレスを入力し、[Esc]キーを押して、[2]キーを押します。

3-7.サブネットの設定

1) 「e1000g0 用のサブネット」画面では、接続するネットワークのサブネット情報を設定します。

2) 「はい」に画面に「X」マークが付いたことを確認し、[Esc]キーを押して、[2]キーを押します。

3) 「e1000g0 用のネットマスク」画面では、e1000g0 用のサブネットのネットマスクを指定します。

4) ネットマスクを入力し、[Esc]キーを押して、[2]キーを押します。

3-8.IPv6 使用の有無

1) 「e1000g0 用の IPv6」画面では、IPv6 の使用の有無を設定します。矢印キーで「いいえ」にカーソルを合わせ、[Space]キーを押します。

2) 「いいえ」に「X」マークが付いたことを確認し、[Esc]キーを押して、[2]キーを押します。

3-9.デフォルトルートの設定

1) 「e1000g0 用のデフォルトルートの設定」画面では、ネットワークのデフォルトルートを設定します。矢印キーで「指定」にカーソルを合わせ、[Space]キーを押します。

※「なし」を選択しインストール後に設定することも可能です。

※注意 ネットワークインストールの場合は下記の設定は省略されます。

2) 「指定」に「X」マークが付いたことを確認し、[Esc]キーを押して、[2]キーを押します。

3) 「e1000g0 用のデフォルトルートの IP アドレス」画面では、デフォルトルートの IP アドレスを設定します。

※上記の手順で「なし」を選択した場合は下記の設定は省略されます。

※注意 ネットワークインストールの場合、この設定は省略されます。

4) IP アドレスを入力し、[Esc]キーを押して、[2]キーを押します。

3-1 0. 設定情報の確認

1) 「e1000g0 の情報の確認」画面では、ここまで設定した内容を確認します。

2) 内容が正しいことを確認し、[Esc]キーを押して、[2]キーを押します。

☛ 内容を修正する場合は[Esc]キーを押して、[4]キーを押すと、「3.2.1 ネットワーク接続の有無」まで戻り、設定変更が可能です。ネットワークインストール実行時は、ホスト名や IP アドレスの項目は設定済みなので表示されません。

3-1 1. セキュリティポリシーの設定

1) 「セキュリティポリシーの構成」画面では、Kerberos セキュリティ機構を使用するか、標準の UNIX セキュリティを使用するか、選択します。矢印キーで「いいえ」にカーソルを合わせ、[Space]キーを押します。

2) 「いいえ」に「X」マークが付いたことを確認し、[Esc]キーを押して、[2]キーを押します。

3) 「情報の確認」画面では、Kerberos セキュリティ構成の確認画面が表示されます。内容を確認し、[Esc]キーを押して、[2]キーを押します。

3-1 2. ネームサービスの選択

1) 「ネームサービス」画面では、ネームサービスを選択します。矢印キーで「None」にカーソルを合わせ、[Space]キーを押します。

2) 「None」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

3) 「情報の確認」画面では、ネームサービスの確認画面が表示されます。内容を確認し、[Esc]キーを押して、[2]キーを押します。

3-1 3 .NFSv4 ドメイン名の設定

1) 「NFSv4 ドメイン名」画面では、NFSv4 ドメイン名を指定する。「システムによって取得された NFSv4 ドメインを使用します」にカーソルを合わせ、[Space]キーを押します。

2) 「システムによって取得された NFSv4 ドメインを使用します」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

3) 「NFSv4 Domain の情報の確認」画面では、情報を確認して[Esc]キーを押して[2]キーを押します。

3-1 4 .時間帯の設定

1) 「時間帯」画面では、デフォルトの時間帯を指定します。矢印キーで「アジア」にカーソルを合わせ、[Space]キーを押します。

2) 「アジア」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

3) 「国または地域」画面では、国または地域を選択します。矢印キーで「日本」にカーソルを合わせ、[Space]キーを押します。

4) 「日本」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

3-1 5 .日付と時刻の設定

1) 「日付と時刻」画面では、システムの日付と時刻を設定します。時刻がずれている場合、変更します。

2) 時刻が正しいことを確認し、[Esc]キーを押して[2]キーを押します。

3-1 6 .情報の確認

1) 「情報の確認」画面では、設定内容を確認します。

2) 設定が正しいことを確認し、[Esc]キーを押して[2]キーを押します。

3-1 7 .root パスワードの設定

1) 「root パスワード」画面では、システム管理者(root)のパスワードを設定します。

2) を確認し、[Esc]キーを押して[2]キーを押します。

3-18. Identify This System

1) 「Identify This System」画面では、内容を確認し、[Esc]キーを押して[2]キーを押します。

2) 「Enabling remote services」画面では、リモートサービスを有効にするか設定します。ここでは、[Yes]にカーソルを合わせ[Space]キーを押します。

3) 「Yes」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

ここまでの設定を終えると下記のメッセージが表示されます。

システムの確認が完了しました。
:

3-19. インストール方法の選択

1) 「Solaris 対話式インストール」画面では、Solaris10 OS のインストール方法を選択します。標準インストールを実施するため、[Esc]キーを押して、[2]キーを押します。

<参考>

標準(標準インストール)

ローカルサーバのインストール情報を以降のウィザードで行うインストール。

フラッシュ(フラッシュアーカイブ)

あるサーバに格納された「インストール情報アーカイブ(ソフトウェア/ファイルシステムの構成など)」を参照し、以降のウィザードを使わず、自動的に行うインストール。

2) 「CD/DVD を自動的に取り出すようにしますか？」画面では、メディアの排出方法を選択します。「CD/DVD を自動的に取り出す」にカーソルを合わせ、[Space]キーを押します。

3) 「CD/DVD を自動的に取り出す」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

4) 「インストール後にリブートしますか？」画面では、インストール後の自動リブート設定を選択します。「自動リブート」にカーソルを合わせ、[Space]キーを押します。

5) 「自動リブート」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

6) 「媒体を選択」画面では、インストール媒体がネットワークファイルシステムの場合は、「ネットワークファイルシステム」にカーソルを合わせ、[Space]キーを押します。

※ネットワークインストールの場合に表示されます。

7) 「ネットワークファイルシステム」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

8) 「ネットワークファイルシステムのパスの指定」画面では、オペレーティングシステムを含むネットワークファイルシステムのパスを指定します。

※ネットワークインストールの場合に表示されます。

9) 「Solaris 対話式インストール」画面は、既に旧版数の SolarisOS がインストールされている環境の場合に表示されます。本手順では特にアップグレードインストールはせずに新規インストールを実施します。[Esc]キーを押して[4]キーを押します。

ここまでの設定を終えると下記のメッセージが表示されます。

```
- 初期化中 -----  
  
システムを初期化中です。  
  
インストール媒体を読み込んでいます。しばらくお待ちください...
```

3-2 0. ライセンスへの同意

1) 「ライセンス」画面では、ライセンスへ同意します。

2) 内容を確認し、[Esc]キーを押して[2]キーを押します。

3-2 1. 使用文字コードの選択

1) 「地域の選択」画面では、ユーザインタフェースで表示する言語を選択します。矢印キーで「アジア」の左側にカーソルを合わせて[Space]キーを押します。

2) 矢印キーで「日本語 EUC(ja)」「日本語 PC 漢字(ja_JP.PCK)」「日本語 UTF-8(ja_JP.UTF-8)」にそれぞれカーソルを合わせ、[Space]キーを押します。

- 3) 「システムロケールの選択」画面では選択可能なロケールが表示されます。
- 4) 「日本語 EUC(ja)」に「X」マークが付いたことを確認し、[Esc]キーを押して、[2]キーを押します。

3-2 2. インストールソフトウェアの選択

- 1) 「追加製品の指定」画面では、追加してインストールする製品を選択します。「なし」にカーソルを合わせ、[Space]キーを押します。
- 2) 「なし」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

3-2 3.ファイルシステムの選択

1) 「ファイルシステムタイプの選択」画面では、「ZFS」にカーソルを合わせ、[Space]キーを押します。

ファイルシステムタイプの選択

Solaris インストールに使用するファイルシステムを選択します

[] UFS
[X] ZFS

Esc-2_継続 F3_前に戻る F5_終了

2) 「ZFS」に「X」マークがついたことを確認し、[ESC] キーを押して[2]キーを押します。

3-2 4.ソフトウェアグループの選択

1) 「ソフトウェアの選択」画面では、インストールする Solaris OS グループを選択します。「全体ディストリビューションと OEM サポート」にカーソルを合わせ、[Space]キーを押します。

2) 「全体ディストリビューションと OEM サポート」に「X」マークが付いたことを確認し、[Esc]キーを押して[2]キーを押します。

3-2 5.ディスク/ファイルシステムの設定

3-2 5-1.インストールディスクの選択

1) 「ディスクの選択」画面では、Solaris インストールに使用するディスクを選択します。選択したディスクはミラー構成となります。矢印キーで「c0t0d0」、「c0t1d0」にカーソルを合わせて、[Space]キーを押します。それぞれのディスクに対して設定を行います。

ディスクの選択

この画面では、Solaris ソフトウェアをインストールするディスクを選択します。
まず「推奨する最小値」フィールドの値を確認してください。
これは、選択したソフトウェアをインストールするために必要な容量の概算値です。
ZFS の場合は複数のディスクがミラーとして構成されるため、選択したディスクまたはディスク内のスライスが「推奨する最小値」の値を超えていなければなりません。
注: ** は現在のブートディスクを表します

ディスクデバイス	空き領域
[X] c0t0d0	69994 MB
[X] c0t1d0	69994 MB

最大ルートサイズ: 0 MB
推奨する最小値: 8688 MB

Esc-2_継続 F3_前に戻る F4_編集 F5_終了 F6_ヘルプ

※ ディスク番号、ディスク数、ディスク容量はハード構成により異なります。

2) [Esc]キーを押して、[2]キーを押します。

3-2 5-2. 既存データの保存

1) 「データを保存しますか?」画面では、ディスク内にデータが存在している場合、既存データのバックアップを取るか、否かを選択出来ます。新規インストールとなるので保存は行いません。[Esc]キーを押して、[2]キーを押します。

3-2 5-3. ZFS 設定の構成

1) 「ZFS 設定の構成」画面では、作成するプールの名前、データセットの名前、プールのサイズ、/var の配置の設定を行います。

- ZFS 設定の構成 -

選択したディスクから作成するプールの名前を指定します。また、ファイルシステムのルートディレクトリとして使用するプール内に作成するデータセットの名前を指定します。

ZFS プール名: rpool
 ZFS ルートデータセット名: s10s_u7wos_08
 ZFS Pool Size (in MB): 69995
 Size of Swap Area (in MB): 2048
 Size of Dump Area (in MB): 1536
 (プールサイズは6640 MB から 69995 MB の範囲でなければなりません)

/ と /var を統合したままにする
 /var を別のデータセットに配置する

Esc-2_継続 F3_前に戻る F5_終了 F6_ヘルプ

※ 各項目はご利用の環境に合わせて設定してください。

2) 各項目を確認し、[Esc]キーを押して[2]キーを押します。

3-2 5-4. リモートファイルシステムのマウント

1) 「リモートファイルシステムをマウントしますか?」画面では、リモートファイルサーバからソフトウェアをマウントするか、否かを設定します。[Esc]キーを押し[2]キーを押します。

2) 「プロファイル」画面では今までの設定が表示されます。内容を確認し、[Esc]キーを押し[2]キーを押します。

☞ 内容が正しくない場合は、[Esc]キーを押してから [4]キーを押し前に戻ります。設定内容が正しくなるまで設定を繰り返します。

3) 警告メッセージが出力される場合がありますが、無視して構いません。[Esc]キーを押して[2]キーを押します。

ここまでの設定を終えると下記のメッセージが表示されます。

Solaris をインストールするためにシステムを準備しています

ディスク (c0t0d0) を構成中です

- Solaris ディスクラベル (VTOC) を作成中です

ディスク (c0t1d0) を構成中です

- Solaris ディスクラベル (VTOC) を作成中です

- プール rpool を作成しています

- プール rpool のスワップ zvol を作成しています

- プール rpool のダンプ zvol を作成しています

ファイルシステムの作成および検査をしています

- rpool/ROOT/s10s_u7wos_08 データセットを作成しています

- データセット rpool/ROOT/s10s_u7wos_08/var を作成しています

Solaris ソフトウェアのインストールを開始します

4) ファイルコピーの進行状況がわかるインジケータが表示されます。インジケータが右端まで行くとファイルコピーが終了し、自動でシステムが再起動します。

5) ファイルコピー終了後、再起動前にメッセージが表示されます。途中、一時停止するか否か選択が出ますが、そのまま続けます。

3-2 6 .boot-device の設定

ZFS を選択し、システムディスクをミラーリングしても、そのままではブート時にミラー先ディスクを指定しないとミラー先ディスクからブートできません。そのため、ミラー先ディスクを boot-device に追加します。

1) OBP への移動

OBP 環境変数を操作するため、OBP へ移動します。

```
# /usr/sbin/shutdown -y -i0 -g0
```

2) ミラー先ディスクのエイリアス確認

製品マニュアルのアドミニストレーションガイドを参考にミラー先ディスクのエイリアスを確認します。

```
[?] ok devalias
.. 略 ..
disk1 /pci@400/pci@0/pci@8/scsi@0/disk@1
disk0 /pci@400/pci@0/pci@8/scsi@0/disk@0
disk /pci@400/pci@0/pci@8/scsi@0/disk@0
.. 略 ..
```

3) boot-device の確認

boot-device の設定を確認します。

```
[?] ok printenv boot-device
boot-device = disk
```

4) ミラー先ディスクの追加

boot-device の設定にミラー先ディスクを追加します。

```
[?] ok setenv boot-device <元の boot-device 値> <ミラー先ディスク> ..
```

例えば、ミラー先ディスクが disk1 の場合、以下のように設定します。

```
[?] ok setenv boot-device disk disk1
```

5) 設定の確認

boot-device が正しく設定されているか、確認します。

```
[?] ok printenv boot-device
boot-device = disk disk1
```

4. ZFS の運用

本章では、ZFS の基本的な運用について解説します。

本章で解説している環境は、PTF、最新のパッチクラスタ(RSPC)およびパッチ、ESF が適用されているものとします。PTF や個々のパッチ、EFS の適用についてはそれぞれの説明書や README を参照の上、実行してください。

4-1. ストレージプールの作成

ストレージプールはディスク全体(c0t1d0 など)または個別のスライス(c0t1d0s0 など)をストレージデバイスとして利用可能ですが、ディスク全体を使用する方法を推奨します。この場合、ディスクをフォーマットする必要はなく、ZFS によって EFI ラベルの 1 つの大きなスライスのディスクとしてフォーマットされます。format コマンドで表示されるパーティションテーブルは次のようになります。

Current partition table (original):

Total disk sectors available: 143358320 + 16384 (reserved sectors)

Part	Tag	Flag	First Sector	Size	Last Sector
0	usr	wm	256	68.36GB	143358320
1	unassigned	wm	0	0	0
2	unassigned	wm	0	0	0
3	unassigned	wm	0	0	0
4	unassigned	wm	0	0	0
5	unassigned	wm	0	0	0
6	unassigned	wm	0	0	0
8	reserved	wm	143358321	8.00MB	143374704

ZFS コンポーネントに名前をつけるときの規則は以下の通りです。

- 空のコンポーネントは許可されません。
- 各コンポーネントに使用できる文字は、英数字および次の 4 つの特殊文字だけです。
 - ・ 下線(_)
 - ・ ハイフン(-)
 - ・ コロン(:)
 - ・ ピリオド(.)
- プール名の先頭は英字にする必要があります。ただし、次の制限事項があります。
 - ・ c[0-9] の順序で始まる名前は許可されません。
 - ・ log という名前は予約されています。
 - ・ mirror、raidz、または spare で始まる名前は予約されているため、使用できません。
 - ・ プール名にはパーセント記号(%) を含めないでください。
 - ・

- データセット名の先頭は英数字にする必要があります。データセット名にはパーセント記号(%) を含めないでください。

1) ディスク c0t1d0 と c0t2d0 からなるミラー構成のストレージプールを「testpool」という名前で作成します。

```
# zpool create testpool mirror c0t1d0 c0t2d0
```

- ☞ mirror の代わりに raidz と指定することによって raidz 構成のストレージプールを作成することも可能です。また、構成を指定しないで作成した場合は、ストライプ構成のストレージプールが作成されます。
- ☞ プールが作成されるときに、root データセットのデフォルトマウントポイントは/pool-name になります。

《参考》

ディスク c0t1d0 と c0t2d0 からなる raidz 構成の「test_raid_pool」というストレージプールを作成する場合。

```
# zpool create test_raid_pool raidz c0t1d0 c0t2d0
```

マウントポイントを「/export/test」に設定してディスク c0t1d0 と c0t2d0 からなるミラー構成の「testpool」ストレージプールを作成する場合。

```
# zpool create -m mountpoint=/export/test testpool mirror c0t1d0 c0t2d0
```

ディスク c0t1d0 の EFI ラベルを SMI ラベルに戻す場合。

```
# format -e c0t1d0
selecting c0t1d0
[disk formatted]

FORMAT MENU:
  disk      - select a disk
  type      - select (define) a disk type
  partition - select (define) a partition table
  current   - describe the current disk
  format    - format and analyze the disk
  repair    - repair a defective sector
  label     - write label to the disk
  analyze   - surface analysis
  defect    - defect list management
  backup    - search for backup labels
  verify    - read and display labels
  save      - save new disk/partition definitions
  inquiry   - show vendor, product and revision
  scsi      - independent SCSI mode selects
  cache     - enable, disable or query SCSI disk cache
  volname   - set 8-character volume name
  !<cmd>    - execute <cmd>, then return
  quit
```

```
format> label                                ←「label」を入力
[0] SMI Label
[1] EFI Label
Specify Label type[0]: 0                    ←「0」を入力
Ready to label disk, continue? y           ←「y」を入力
Auto configuration via format.dat[no]?     ←「Enter」キーを押す
Auto configuration via generic SCSI-2[no]? ←「Enter」キーを押す
format>
```

2) ストレージプールが作成されたことを確認します。(30G バイトの 2 本のディスクからなるミラー構成の表示例)

```
# zpool list
NAME      SIZE  USED  AVAIL  CAP      HEALTH  ALTROOT
testpool  29.8G  94K   29.7G  0%      ONLINE  -
```

NAME	プールの名前。
SIZE	プールの合計サイズ。最上位レベルにあるすべての仮想デバイスの合計サイズになります。
USED	すべてのデータセットおよび内部メタデータから割り当てられた容量。この容量は、df コマンドなどのファイルシステムレベルで報告される容量とは一致しません。
AVAIL	プール内で割り当てられていない容量。
CAP	使用されている容量。総容量に対するパーセントで表現されます。
HEALTH	プールの現在の状態。
ALTROOT	プールの代替ルート(存在する場合)。

```
# zpool status
プール: testpool
状態: ONLINE
スクラブ: 何も要求されませんでした
構成:

      NAME      STATE      READ WRITE CKSUM
testpool  ONLINE      0    0    0
  mirror      ONLINE      0    0    0
    c0t1d0     ONLINE      0    0    0
    c0t2d0     ONLINE      0    0    0
```

エラー: 既知のデータエラーはありません

- 👉 -x オプションを使うとエラーが発生しているか、使用不可能なプールの状態だけを表示します。
- 👉 -v オプションを使うと詳細なデータエラー情報を表示し、前回のプールのスクラブの完了後のデータエラーすべての完全なリストを出力します。

各デバイスは、次のいずれかの状態になります。

ONLINE	デバイスは正常に機能しています。
DEGRADED	仮想デバイスに障害が発生していますが、引き続き利用できます。例えば、ミラーデバイスまたは RAID-Z デバイスを構成するデバイスのうち、1 つ以上のデバイスが失われたときなどに表示されます。
FAULTED	仮想デバイスへのアクセスが完全にできない状態です。最上位レベルの仮想デバイスがこの状態の場合には、そのプールへのアクセスはまったくできません。
OFFLINE	管理者がその仮想デバイスを明示的にオフラインにしています
UNAVAILABLE	デバイスまたは仮想デバイスを開くことができません。最上位レベルの仮想デバイスが使用できない場合は、そのプールのデバイスには一切アクセスできません。

4 - 2 .ファイルシステムの作成

ZFS ファイルシステムの作成には、`zfs create` コマンドを使用します。ファイルシステムの作成時にはファイルシステムのプロパティを設定でき、正常に作成されると自動的にマウントされます。

1) 「testpool/home」という名前の ZFS ファイルシステムを作成します。

```
# zfs create testpool/home
```

👉 プロパティの設定はファイルシステム作成後でも行えます。

《参考》

マウントポイントを「/export/zfs」に設定して「testpool/home」ファイルシステムを作成する場合。

```
# zfs create -o mountpoint=/export/zfs testpool/home
```

ファイルシステム「testpool/home」の名前を「testpool/zfs/home_old」に変更する場合。

```
# zfs rename -p testpool/home testpool/zfs/home_old
```

👉 `-p` オプションを使用することで、中間のデータセット `testpool/zfs` が存在しない場合でも、作成をします。

👉 `-p` オプションは `create`、`clone`、`rename` コマンドで使用可能です。

「testpool/home」を削除する場合。

```
# zfs destroy testpool/home
```

👉 `-r` オプションを使用すると、ファイルシステムとその子孫を再帰的に削除します。

5G バイトの ZFS ボリューム `testpool/home/user1` を作成する場合。

```
# zfs create -V 5gb testpool/home/user1
```

👉 `-V` オプションでサイズを指定すると、予約が自動的に設定されます。

2) 作成したファイルシステムを確認します。

```
# zfs list
NAME          USED    AVAIL    REFER    MOUNTPOINT
testpool      130K    29.3G           18K    /testpool
testpool/home 18K     29.3G    18K     /testpool/home
```

4-3. プロパティの設定

作成したファイルシステムにプロパティを設定することによって、動作の制御をすることができます。設定可能なプロパティの中には親に設定したプロパティがそのすべての子孫に継承されるものがあります。(各プロパティについては、「付録 1. プロパティの紹介」を参照してください。)

1) quota のプロパティの設定例。

・「testpool/home」が使用できるディスク容量 (quota) を 10G バイトに設定します。

```
# zfs set quota=10g testpool/home
```

☞ 設定を解除する場合は、「quota=none」と設定します。

2) 「testpool/home」の「quota」のプロパティを確認する場合。

```
# zfs get quota testpool/home
NAME          PROPERTY    VALUE    SOURCE
testpool/home quota        10G     local
```

☞ -r オプションを使用すると、子孫のプロパティも確認することができます。

4 番目の列の SOURCE は、プロパティ値がどこに設定されているかを示します。出力されるソース値の意味は下記の通りです。

default	プロパティのデフォルト値が使用されています。
inherited from <i>dataset-name</i>	<i>dataset-name</i> に指定されている親から継承されています。
local	zfs set を使って、このデータセットに明示的に設定されたものです。
temporary	このプロパティ値は、zfs mount -o オプションを使って設定され、マウントのされている間だけ有効です。
- (なし)	このプロパティは読み取り専用プロパティです。値は ZFS によって生成されます。

3) プロパティの継承

割り当てと予約を除いて、すべての設定可能なプロパティは親から値を継承します。明示的な値が祖先に設定されていない場合は、デフォルト値が設定されます。

testpool/home/user1 の圧縮の「on」の設定を解除して、親からの設定(default)を継承する場合。

testpool の compression の設定値を確認します。

```
# zfs get -r compression testpool
NAME          PROPERTY    VALUE    SOURCE
testpool      compression off       default
testpool/home compression off       default
```

```
testpool/home/user1  compression  on  local
```

☞ -r オプションを使用すると、inherit サブコマンドが再帰的に適用されます。

compression の設定を親から継承します。

```
# zfs inherit compression testpool/home/user1
```

設定後に testpool の compression の設定値を確認します。

```
# zfs get -r compression testpool
NAME                PROPERTY           VALUE  SOURCE
testpool            compression        off    default
testpool/home       compression        off    default
testpool/home/user1 compression        off    default
```

☞ -r オプションを使用すると、inherit サブコマンドが再帰的に適用されます。

4-4 .ZFS ファイルシステムのマウント、アンマウント

デフォルトでは、すべての ZFS ファイルシステムは、ZFS の起動時にサービス管理機能(SMF) の svc:/system/filesystem/local サービスを使用してマウントされます。/etc/vfstab ファイルを編集する必要はありません。

zfs set コマンドを使用して mountpoint プロパティを特定のパスに設定することで、デフォルトのマウントポイントを上書きできます。また、mountpoint プロパティは継承されます。

ファイルシステムは、必要に応じてレガシーマウントインタフェースを使って明示的に管理できます。そのためには、zfs set で mountpoint プロパティを legacy に設定します。legacy に設定することで、そのファイルシステムは自動的にマウントおよび管理されなくなります。代わりに、mount や umount コマンドや /etc/vfstab ファイルで管理する必要があります。

1) マウントポイントのプロパティの設定例。

・「testpool/home」ファイルシステムのマウントポイントを「/export/zfs」に設定する場合。

マウントポイントのプロパティの設定値を確認します。

```
# zfs get mountpoint
NAME                PROPERTY           VALUE                SOURCE
testpool            mountpoint         /testpool           default
testpool/home       mountpoint         /testpool/home      default
testpool/home/user1 mountpoint         /testpool/home/user1 default
```

「testpool/home」のマウントポイントのプロパティを「/export/zfs」に設定します。

```
# zfs set mountpoint=/export/zfs testpool/home
```

☞ mountpoint のプロパティは自動で継承されます。

- mountpoint プロパティを変更すると、ファイルシステムが古いマウントポイントから自動的にマウント解除されて、新しいマウントポイントに再マウントされます。

設定後にマウントポイントのプロパティの設定値を確認します。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
testpool            170K    19.6G    18K      /testpool
testpool/home       37K     19.6G    19K      /export/zfs
testpool/home/user1 18K     19.6G    18K      /export/zfs/user1
# zfs get mountpoint
NAME                PROPERTY    VALUE                SOURCE
testpool            mountpoint  /testpool            default
testpool/home       mountpoint  /export/zfs          local
testpool/home/user1 mountpoint  /export/zfs/user1    inherited from testpool/home
```

・「testpool/home」ファイルシステムのマウントポイントを「legacy」に設定する場合。

マウントポイントのプロパティの設定値を確認します。

```
# zfs get mountpoint
NAME                PROPERTY    VALUE                SOURCE
testpool            mountpoint  /testpool            default
testpool/home       mountpoint  /testpool/home       default
testpool/home/user1 mountpoint  /testpool/home/user1 inherited from testpool/home
```

マウントポイントのプロパティを「legacy」に設定します。

```
# zfs set mountpoint=legacy testpool/home
```

設定後にマウントポイントのプロパティの設定値を確認します。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
testpool            182K    19.6G    18K      /testpool
testpool/home       37K     19.6G    19K      legacy
testpool/home/user 118K    19.6G    18K      legacy
# zfs get mountpoint
NAME                PROPERTY    VALUE    SOURCE
testpool            mountpoint  /testpool default
testpool/home       mountpoint  legacy   local
testpool/home/user1 mountpoint  legacy   inherited from testpool/home
```

《参考》

レガシーファイルシステムをマウント/アンマウントする場合。

```
# mount -F zfs testpool/home /mnt
# umount /mnt
```

2) マウント/アンマウント実行

ZFS では、ファイルシステムが作成されるときまたはシステムが起動するときに、ファイルシステムが自動的にマウントされます。zfs mount コマンドを使用する必要があるのは、マウントオプションを変更するとき、またはファイルシステムのマウントまたはマウント解除を行うときだけです。

zfs mount コマンドを引数なしで実行すると、ZFS が管理していて、現在マウントされているファイルシステムがすべて表示されます。レガシーで管理されているマウントポイントは表示されません。

デフォルトでは、空でないディレクトリの最上位にマウントすることは許可されません。空でないディレクトリの最上位に強制的にマウントする場合は、-O オプションを使用する必要があります。

・ZFS ファイルシステムのマウント例。

testpool/home をマウントする場合。

現在の ZFS ファイルシステムのマウントの状態を確認します。

```
# zfs mount
testpool                /testpool
```

「testpool/home」をマウントします。

```
# zfs mount testpool/home
```

マウント実行後に ZFS ファイルシステムのマウントの状態を確認します。

```
# zfs mount
testpool                /testpool
testpool/home          /export/zfs
```

👉 -a オプションを使用すると、ZFS が管理しているファイルシステムをすべてマウントできます。レガシー管理されているファイルシステムはマウントされません。

👉 -O オプションを使用すると、空でないディレクトリへのマウントが許可されます。

《参考》

testpool/home を読み取り専用としてマウントする場合。

```
# zfs mount -o ro testpool/home
```

👉 ファイルシステムがマウント解除されるときに元の設定に戻ります。

・ZFS ファイルシステム のアンマウント例。

「testpool/home」ファイルシステムをアンマウントする場合。

```
# zfs unmount testpool/home
または
# zfs unmount /export/zfs
```

👉 -a オプションを使用すると、ZFS が管理しているファイルシステムをすべてアンマウントできます。

👉 アンマウントはファイルシステム名またはマウントポイントを指定して実行できます。

4-5.ZFS ファイルシステムの共有、共有解除

ZFS では sharenfs プロパティを使って自動的にファイルシステムを共有できます。この方法を使用すれば、新規ファイルシステムを追加するときに/etc/dfs/dfstab ファイルを変更する必要はありません。値を on に設定するとすべてのユーザに read/write 権を許可します。値を off にするとファイルシステムが ZFS によって管理されなくなり、従来の方法(/etc/dfs/dfstab)で共有できるようになります。sharenfs プロパティが off でないすべてのファイルシステムは、起動時に共有されます。

1) ファイルシステム共有プロパティの設定例。

・「testpool/home」ファイルシステムの共有を on に設定する場合。

sharenfs の設定値を確認します。

```
# zfs get sharenfs
NAME                PROPERTY    VALUE    SOURCE
testpool            sharenfs    off      default
testpool/home       sharenfs    off      default
testpool/home/user1 sharenfs    off      default
```

「testpool/home」の sharenfs の設定値を「on」に設定します。

```
# zfs set sharenfs=on testpool/home
```

- 🟢 sharenfs のプロパティは自動で継承されます。
- 🟢 zfs ファイルシステムの共有の状態は share コマンドで確認ができます。
- 🟢 sharenfs の設定を「on」に設定すると即座に共有が開始されます。

設定後に、sharenfs の設定値を確認します。

```
# zfs get sharenfs
NAME                PROPERTY    VALUE    SOURCE
testpool            sharenfs    off      default
testpool/home       sharenfs    on       local
testpool/home/user1 sharenfs    on       inherited from testpool/home
```

sharenfs の設定が on になっているすべての ZFS ファイルシステムを共有する場合。

```
# zfs share -a
```

- 🟢 -a オプションを使用すると、sharenfs の設定が on になっているすべての ZFS ファイルシステムを共有します。

・「testpool/home」ファイルシステムの共有を off にする場合。

sharenfs の設定値を確認します。

```
# zfs get sharenfs
NAME                PROPERTY    VALUE    SOURCE
testpool            sharenfs    off      default
testpool/home       sharenfs    on       local
```



```
testpool/home/user1 sharenfs      on      inherited from testpool/home
```

「testpool/home」の sharenfs の設定値を「off」に設定します。

```
# zfs set sharenfs=off testpool/home
```

☞ sharenfs を「off」すると即座に共有が解除されます。

設定後に、sharenfs の設定値を確認します。

```
# zfs get sharenfs
NAME                PROPERTY    VALUE    SOURCE
testpool            sharenfs    off      default
testpool/home       sharenfs    off      local
testpool/home/user1 sharenfs    off      inherited from testpool/home
```

☞ sharenfs のプロパティは自動で継承されます。

・「testpool/home」ファイルシステムの共有を読み取り専用にする場合。

sharenfs の設定値を確認します。

```
# zfs get sharenfs
NAME                PROPERTY    VALUE    SOURCE
testpool            sharenfs    off      default
testpool/home       sharenfs    off      local
testpool/home/user1 sharenfs    off      inherited from testpool/home
```

☞ sharenfs のプロパティは自動で継承されます。

「testpool/home」の sharenfs の設定値を「ro」に設定します。

```
# zfs set sharenfs=ro testpool/home
```

設定後に、sharenfs の設定値を確認します。

```
# zfs get sharenfs
NAME                PROPERTY    VALUE    SOURCE
testpool            sharenfs    off      default
testpool/home       sharenfs    ro      local
testpool/home/user1 sharenfs    ro      inherited from testpool/home
```

☞ sharenfs のプロパティは自動で継承されます。

2) ファイルシステムの共有/共有解除例。

・testpool/home/user1 の共有を解除する場合。

```
# zfs unshare testpool/home/user1
```

☞ sharenfs のプロパティは自動で継承されます。

☞ -a オプションを使用すると、共有されているすべての ZFS ファイルシステムの共有を解除します。

☞ zfs ファイルシステムの共有の状態は share コマンドで確認ができます。

shareコマンドによるZFSファイルシステムの共有

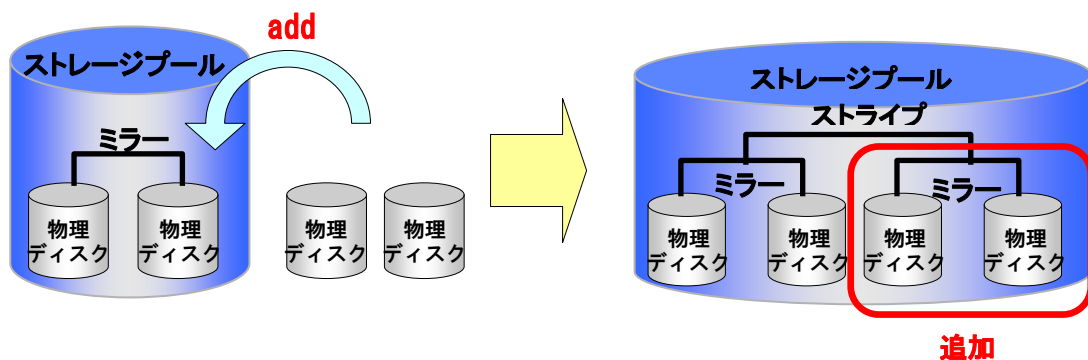
sharenfs プロパティが off の場合は、どのような場合にも zfs share または zfs unshare コマンドでファイルシステムを共有または共有解除することができません。この設定を利用して share または unshare コマンド、および/etc/dfs/dfstab ファイルなどを従来の方法で管理することができます。

ZFS ファイルシステムは従来の share および unshare コマンドでの操作も可能です。このため、sharenfs プロパティの設定とは異なるオプションを使って、ファイルシステムを手動で共有することもできますが、この管理モデルは推奨されていません。

ZFSを使用して NFS 共有を完全に管理するか、または/etc/dfs/dfstab ファイルを使用して完全に管理する方法を選択してください。

4-6. デバイスの追加 (add)

最上位レベルの新しい仮想デバイスを追加することで、プールに領域を動的に追加できます。プール内のすべてのデータセットは、この領域をすぐに利用できます。



1) ディスクを追加する前に testpool の構成を確認します。

# zpool list						
NAME	SIZE	USED	AVAIL	CAP	HEALTH	ALTROOT
testpool	29.8G	94K	29.7G	0%	ONLINE	-

```
# zpool status
プール: testpool
状態: ONLINE
スクラブ: 何も要求されませんでした
構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	ONLINE	0	0	0

エラー: 既知のデータエラーはありません

2) zpool add コマンドで、testpool に c0t3d0 と c1t3d0 からなるミラー構成のディスクを追加します。

```
# zpool add testpool mirror c0t3d0 c1t3d0
```

3) zpool list、zpool status コマンドで確認をします。

```
# zpool list
```

NAME	SIZE	USED	AVAIL	CAP	HEALTH	ALTROOT
testpool	59.5G	150K	59.5G	0%	ONLINE	-

```
# zpool status
プール: testpool
状態: ONLINE
スクラブ: 何も要求されませんでした
構成:
```

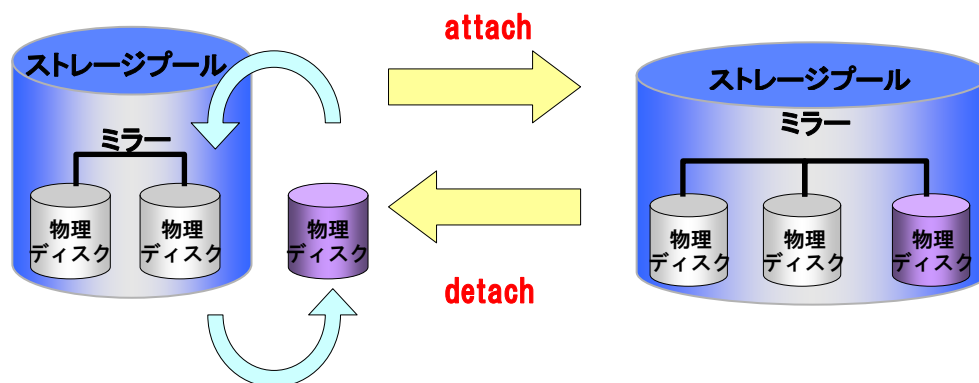
NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t3d0	ONLINE	0	0	0
c1t3d0	ONLINE	0	0	0

エラー: 既知のデータエラーはありません

4-7. デバイスの接続/切り離し(attach/detach)

新しいデバイスを既存のミラー化されたデバイスまたはミラー化されていないデバイスに接続や切り離しができます。たとえば、既存のデバイスが2方向ミラーである場合は、新規デバイスを接続すると、3方向ミラーが作成されます。

新しいデバイスを接続すると、すぐに再同期化が開始されます。また、attach/detach は raid 構成では使用できません。



4-7-1. デバイスの接続(attach)

1) プールの状態を確認します。

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 10:58:50 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:

  NAME          STATE      READ WRITE CKSUM
  testpool      ONLINE    0     0     0
    c0t1d0      ONLINE    0     0     0
    c0t2d0      ONLINE    0     0     0
```

2) zpool attach コマンドで、testpool のデバイス c0t1d0 に c0t3d0 を接続します。

```
# zpool attach testpool c0t1d0 c0t3d0
```

3) attach 後にプールの状態を確認します。

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 10:58:50 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	ONLINE	0	0	0
c0t3d0	ONLINE	0	0	0

《参考》

単一構成の ZFS ストレージプール(c0t1d0)にディスク(c0t2d0)を attach し、ミラー化された ZFS ストレージプールに変換する場合。

```
# zpool attach testpool c0t1d0 c0t2d0
```

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 11:12:44 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	ONLINE	0	0	0

エラー: 既知のデータエラーはありません

 attach を行うと自動で mirror 構成になります。

4-7-2. デバイスの切り離し(detach)

1) プールの状態を確認します。

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 11:12:44 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:

  NAME                STATE  READ  WRITE  CKSUM
  testpool             ONLINE  0      0      0
    mirror
      c0t1d0           ONLINE  0      0      0
      c0t2d0           ONLINE  0      0      0

エラー: 既知のデータエラーはありません
```

2) testpool からディスク c0t2d0 を切り離します。

```
# zpool detach testpool c0t2d0
```

3) detach 後に、プールの状態を確認します。

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 11:12:44 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:

  NAME                STATE  READ  WRITE  CKSUM
  testpool             ONLINE  0      0      0
    c0t1d0             ONLINE  0      0      0

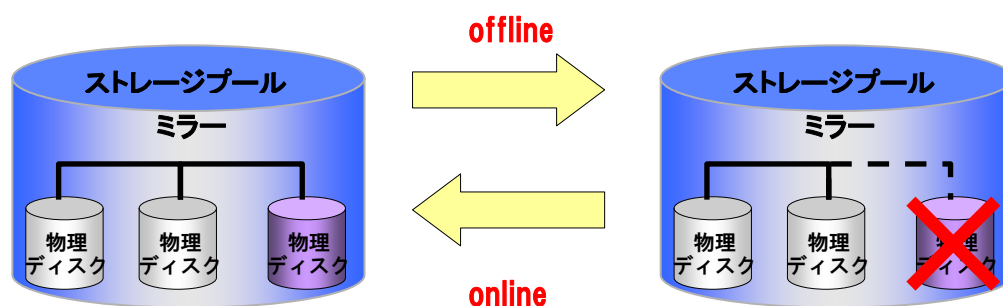
エラー: 既知のデータエラーはありません
```

4-8. デバイスのオンライン/オフライン (online/offline)

ZFS では、デバイスを個別にオフラインまたはオンラインにできます。

オフラインにしたデバイスには、要求はまったく送信されません。

ハードウェアに障害が発生している場合でも、ZFS ではその状態を一時的な状態と見なして、デバイスへのデータの読み取りおよび書き込みを続行しますが、継続的に障害が発生している場合は、デバイスをオフラインにして無視するように設定できます。また、オンラインに戻したあとは自動で再同期化されます。



デバイスをオフラインにするときは、以下の点に注意してください。

- プールをオフラインにすることはできません。
- デフォルトでは、オフライン状態はシステムを再起動しても、デバイスはオフラインのままです。一時的にオフラインにするには `-t` オプションを使用してください。この場合、システムを再起動すると、このデバイスは自動的に ONLINE 状態に戻ります。
- デバイスはオフラインになっても、ストレージプールから切り離されません。元のプールが破棄されても、オフラインのデバイスを別のプールで使用しようとすると、エラーが表示され使用できません。元のストレージプールを破棄した後で、オフラインのデバイスを別のストレージプールで使用する場合は、まずデバイスをオンラインに戻してから、元のストレージプールを破棄します。

4-8-1. デバイスのオフライン

1) デバイスの状態を確認します。

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 11:12:44 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	ONLINE	0	0	0

```
エラー: 既知のデータエラーはありません
```

2) testpool からディスク c0t2d0 を切り離します。

```
# zpool offline testpool c0t2d0
```

3) 切り離し後に、デバイスの状態を確認します。

```
# zpool status testpool
プール: testpool
状態: DEGRADED
状態: 1 つまたは複数のデバイスが管理者によってオフラインにされました。
十分な複製が存在するため、プールは縮退状態で
動作を継続できます。
アクション: 'zpool online' を使用してデバイスをオンラインにするか、
'zpool replace' を使用してデバイスを置換してください。
スクラプ: Tue Sep  8 10:58:50 2009
上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	DEGRADED	0	0	0
mirror	DEGRADED	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	OFFLINE	0	0	0

```
エラー: 既知のデータエラーはありません
```

《参考》

ディスク c0t2d0 を一時的にオフラインにする場合。

```
# zpool offline -t testpool c0t2d0
```

➡ `-t` オプションを使用すると一時的にデバイスをオフラインにすることが可能です。この場合、システムを再起動するとデバイスはオンラインに戻ります。

4-8-2. デバイスのオンライン

オフラインにしたデバイスは、`zpool online` コマンドを使って復元できます。デバイスがオンラインになると、プールに書き込まれたすべてのデータは、新しく使用可能になったデバイスに再同期化されます。デバイスを `zpool online` コマンドを利用して、障害のあるディスクを交換用のディスクに置き換えることはできません。交換用のディスクを挿入した後に、`zpool replace` を実行後に、`zpool online` コマンドを実行する必要があります。

1) デバイスの状態を確認します。

```
# zpool status testpool
  プール: testpool
  状態: DEGRADED
  状態: 1 つまたは複数のデバイスが管理者によってオフラインにされました。
         十分な複製が存在するため、プールは縮退状態で
         動作を継続できます。
  アクション: 'zpool online' を使用してデバイスをオンラインにするか、
              'zpool replace' を使用してデバイスを置換してください。
  スクラブ: Tue Sep  8 10:58:50 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	DEGRADED	0	0	0
mirror	DEGRADED	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	OFFLINE	0	0	0

エラー: 既知のデータエラーはありません

2) testpool のディスク c0t2d0 をオンラインにします。

```
# zpool online testpool c0t2d0
```

3) 接続後に、デバイスの状態を確認します。

```
# zpool status testpool
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 11:45:45 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t1d0	ONLINE	0	0	0
c0t2d0	ONLINE	0	0	0

エラー: 既知のデータエラーはありません

4-9. エラーのクリア (clear)

デバイスに障害が発生した場合、zpool status の出力にエラーが表示されます。デバイスのエラーが一時的である場合は、エラーを無視してクリアすることができますが、クリア後もデバイスのエラーが出力される場合は、デバイスの交換が必要です。デバイスの交換後は、エラー出力は自動的に消えないため、本手順によりエラー出力をクリアします。

1) testpool のエラーをクリアします。

```
# zpool clear testpool
```

《参考》

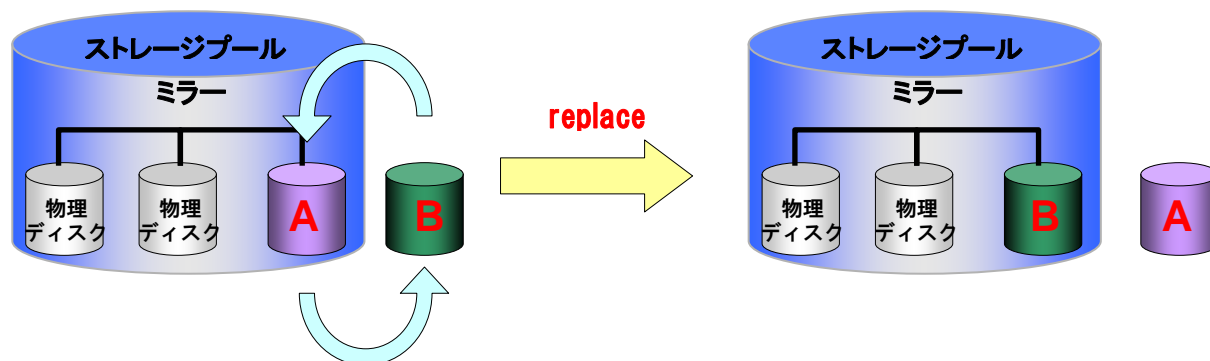
デバイスを指定してこのコマンドを実行する場合。

```
# zpool clear testpool c0t2d0
```

この場合、指定したデバイスに関連づけられたエラーだけがクリアされます。

4-10. デバイスの置き換え (replace)

zpool replace コマンドを使用して、ストレージプール内のデバイスを物理的に置き換えることができます。



ZFS ストレージプール内のデバイスを置き換える場合は、次のことを考慮してください。

- プールのプロパティ autoreplace を on に設定した場合、そのプールに以前属していたデバイスと物理的に同じ位置に新しいデバイスが検出されると、そのデバイスは自動的にフォーマットされ、zpool replace コマンドを使用せずに置き換えられます。ハードウェアの種類によっては、この機能を使用できない場合があります。
- デバイスを置き換えるときは、ミラー化構成または RAID-Z 構成内のすべてのデバイスの最小サイズと等しいかそれ以上のデバイスを使用する必要があります。
- 置き換えたあとのデバイスの方が大きい場合は、置き換えが完了すると、プールの容量が大きくなります。拡張された容量を確認するには、プールをエクスポートしてインポートする必要があります。また、

ストレージプールの一部の既存の LUN のサイズが大きくなる時も、拡張された容量を確認するには、エクスポートとインポートの手順を実行する必要があります。

※Solaris 10 9/10(U9)以降では、autoexpand プロパティを on に設定した場合、プールをエクスポートしてインポートする必要はありません。

- 大規模プール内で多数のディスクを置き換える場合は、新しいディスク上にデータを再同期化するために時間がかかります。また、ディスクの置き換えの合間に `zpool scrub` コマンドを実行して、置き換えたあとのデバイスが動作可能であり、データが正しく書き込まれることを確認することもできます。
- 障害の発生したディスクがホットスペアに自動的に置き換えられた場合は、障害の発生したディスクが置き換えられたあとでスペアの切り離しが必要になることがあります。

1) デバイスを置き換えます。

- ・冗長化されているストレージプール内のディスクを置き換える場合は、障害の発生したディスクを別のディスクと交換した後に、`zpool replace` コマンドを実行します。

ディスク `c0t1d0` を置き換える場合。

```
# zpool replace testpool c0t1d0
```

- ・1 つのデバイスで構成されるストレージプール内のデバイスを置き換える場合は、両方のデバイスを指定する必要があります。

testpool 内のディスク `c0t2d0` を `c0t3d0` と置き換える場合。

```
# zpool replace testpool c0t2d0 c0t3d0
```

2) `zpool status` コマンドで確認をします。

```
# zpool status
プール: testpool
状態: ONLINE
スクラブ: Tue Sep  8 11:59:35 2009
上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:

    NAME                STATE      READ WRITE CKSUM
    testpool             ONLINE     0    0    0
      mirror             ONLINE     0    0    0
        c0t1d0           ONLINE     0    0    0
        replacing       ONLINE     0    0    0
        c0t2d0           ONLINE     0    0    0
        c0t3d0           ONLINE     0    0    0
```

エラー: 既知のデータエラーはありません

 同期処理が完了するまでは「replacing」と表示されます。

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 11:59:35 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c0t1d0	ONLINE	0	0	0
c0t3d0	ONLINE	0	0	0

エラー: 既知のデータエラーはありません

 同期処理が完了すると「replacing」の表示が消えます。

《参考》

ディスクを置き換えるための基本的な手順

- 必要に応じて、zpool offline コマンドでディスクをオフラインにします。
- 置き換えるディスクを取り外します。
- 交換用ディスクを挿入します。
- zpool replace コマンドを実行します。
- zpool online コマンドでディスクをオンラインに戻します。

ZFS ルートプールのディスクを置き換える場合

ディスクに障害が発生しているためシステムが起動しない場合は、CD やネットワークなどの代替メディアから起動します。

- 1) 現在のルートプールのディスク(c0t0d0)と交換用起動ディスク(c1t0d0)を置き換えます。

```
# zpool replace rpool c0t0d0 c1t0d0
```

- 2) ディスクの置き換え状態を確認します。

```
# zpool status rpool
```

- 3) ディスクの置き換えと再同期化が完了したら、新しいディスクにブートブロックを適用します。

```
# installboot -F zfs /usr/platform/`uname -i`/lib/fs/zfs/bootblk c1t0d0
```

- 4) 交換用ディスクから起動します。

```
ok boot /pci@8,700000/~
```

5) 新しいディスクからシステムが起動した場合は、古いディスクを切り離します。

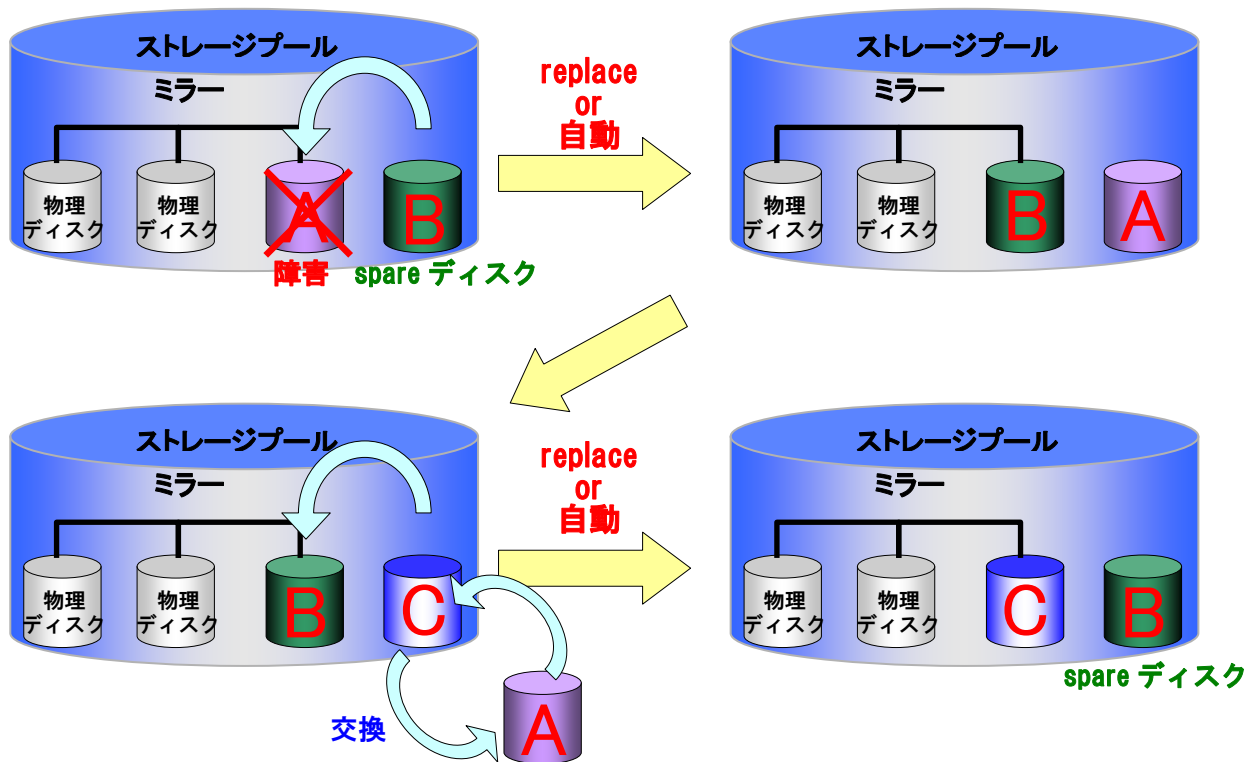
```
# zpool detach rpool c0t0d0
```

6) システムが新しいディスクから自動的に起動するように設定します。そのためには、eeprom コマンド または SPARC ブート PROM の setenv コマンドを使用します。

4-1 1.ホットスペア(spare)

ホットスペア機能を使って、ストレージプールで障害が発生したデバイスまたはエラー状態のデバイスを置き換えるために使用するディスクを指定できます。アクティブデバイスで障害が発生した場合には、そのデバイスがホットスペアに自動的に置き換えられます。また、ホットスペアデバイスは複数のプール間で共有できます。

プールの autoreplace プロパティ を on に設定した場合は、新しいデバイスが挿入されオンライン処理が完了すると、スペアは自動的に切り離されてスペアプールに戻されます。



ホットスペアを使用時には以下の点に注意してください。

- プール内でもっとも大きいディスクのサイズと等しいかそれ以上のディスクをスペアとして追加します。サイズの小さなディスクをスペアとしてプールに追加することも許可されていますがその場合、スペア

ディスクがアクティブになると、自動的または `zpool replace` コマンドによる置き換え操作がエラーで失敗します。

- ホットスペアをプール間で共有することができます。ただし、共有スペアがプールで使用されている場合は、`zpool export -f` (強制) オプションを使用しないとプールをエクスポートできません。プールに使用中の共有スペアがある場合に `-f` オプションを使用してプールをエクスポートするときは、別のプールがその使用中の共有スペアをアクティブにしようとするデータ破壊が発生する可能性があることに注意してください。

ホットスペアをアクティブにするには以下の方法があります。

- 手動で置き換える– `zpool replace` コマンドを使用して、ストレージプール内で障害の発生したデバイスをホットスペアで置き換えます。
- 自動的な置き換え– FMA エージェントは、エラー状態を受け取ると、プールを検査して使用可能なホットスペアがあるかどうかを確認し、ホットスペアがある場合は、障害の発生したデバイスを使用可能なスペアに置き換えます。

※FMA(Fault Management Architecture)–ハードに関する情報を、監視、解析し、デバイスの切り離しを自動で行うなどの可用性を高める機能。

1) デバイスをホットスペアとして指定します。

プール作成時にスペアを設定する場合

```
# zpool create testpool mirror c1t1d0 c2t1d0 spare c1t2d0 c2t2d0
```

プールを作成したあとにプールにホットスペアを追加する場合

```
# zpool add testpool spare c1t2d0 c2t2d0
```

2) `zpool list`、`zpool status` コマンドで確認をします。

```
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: 何も要求されませんでした
  構成:

      NAME                STATE  READ  WRITE  CKSUM
  testpool
    mirror
      c1t1d0              ONLINE  0      0      0
      c2t1d0              ONLINE  0      0      0
  スペア
      c1t2d0              AVAIL
      c2t2d0              AVAIL
```

エラー: 既知のデータエラーはありません

《参考》

スペアを削除する場合。

```
# zpool remove testpool c1t2d0
```

☞ ストレージプールが現在使用しているホットスペアは、削除できません。

zpool replace コマンドを使用して、ストレージプール内のホットスペアをアクティブにする/非アクティブにする場合

```
# zpool replace testpool c1t1d0 c2t1d0
# zpool status
  プール: testpool
  状態: ONLINE
  スクラブ: Tue Sep  8 13:48:59 2009
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c1t1d0	ONLINE	0	0	0
spare	ONLINE	0	0	0
c2t1d0	ONLINE	0	0	0
c1t2d0	ONLINE	0	0	0
スペア				
c1t2d0	INUSE	現在使用中です		
c2t2d0	AVAIL			

エラー: 既知のデータエラーはありません

非アクティブにする場合

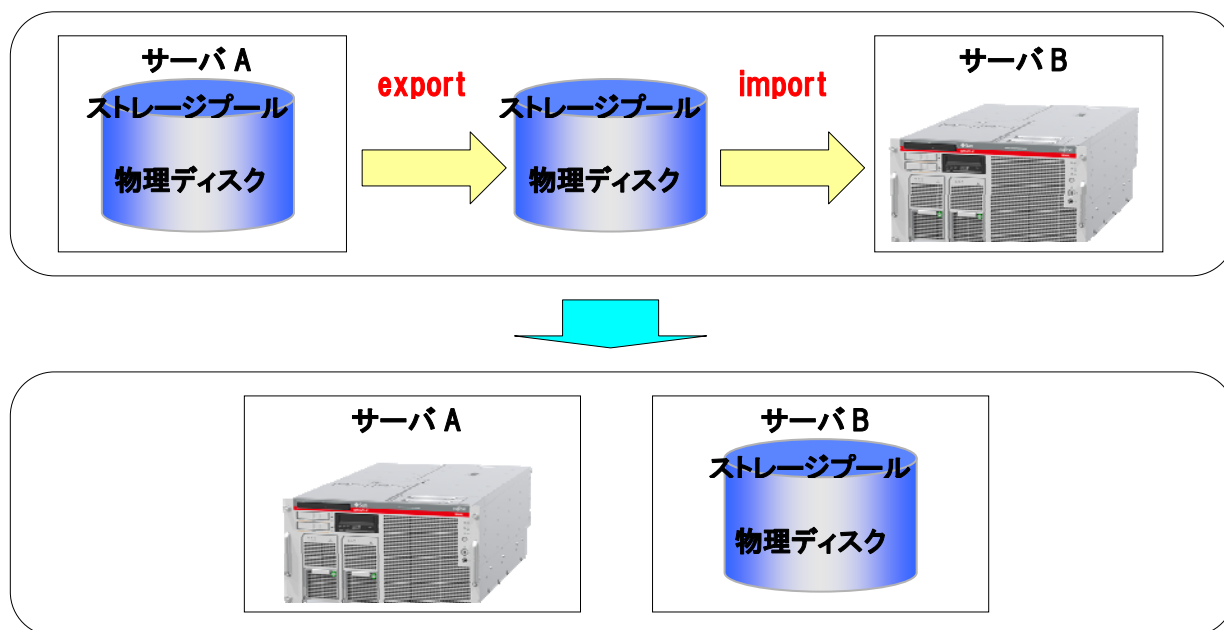
次の 3 つの方法を使って、ホットスペアを非アクティブにできます

- ホットスペアをストレージプールから取り除くことでホットスペアを取り消す (detach)
- 元のデバイスをホットスペアに置き換える (replace)
- ホットスペアを恒久的に交換する (元のデバイスを detach する)

4-1 2.ストレージプールの移行 (export/import)

4-1 2-1.エクスポート

ZFS はあるマシンのストレージプールをエクスポートして、別のマシンにインポートすることができます。この操作を行うことで、ストレージプールに関する全ての情報を移行することができます。



1) testpool をエクスポートします。

```
# zpool export testpool
```

```
# zpool list  
使用できるプールがありません
```

- export されたプールはシステムから認識されなくなります。
- プールの中にマウントされたファイルシステムがある場合はマウントの解除をおこなってから export を実行します。マウント解除に失敗した場合は、-f オプションを使用して強制的にマウントを解除できます。ただし、ZFS ボリュームがプール内で使用中である場合は、-f オプションを使用してもそのプールをエクスポートすることはできません。

4-1 2-2. インポート

プールをシステムから削除したら(エクスポートするか、デバイスを強制的に取り外す)、それらのデバイスをインポート先のシステムに接続します。ZFS では、一部のデバイスだけが使用できる状況に対応できることもあります。システム間でプールを移動するときは、必ずすべてのデバイスを移動してください。

デバイスは、同じデバイス名で接続されている必要はありません。デバイスを移動した場合またはデバイス名を変更した場合には、それらが自動的に検出され、構成がそれに合わせて調整されます。

各プールは、名前および一意の数値識別子を使って識別されます。インポートできるプールが複数あって、それらが同じ名前の場合でも、数値識別子を使ってそれらを識別することができます。

1) testpool をインポートします。

```
# zpool import testpool
```

《参考》

インポートできるプールを確認する場合。

```
# zpool import
  プール: testpool
  ID: 16722234661949228990
  状態: ONLINE
  アクション: プールの名前または数値識別子を使用してプールをインポートできます。
  構成:
```

testpool	ONLINE
mirror	ONLINE
c1t1d0	ONLINE
c2t1d0	ONLINE

インポートしたいプール名が既存のプール名と重複する場合は、別の名前でもインポートできます。

「testpool」を「examplepool」としてインポートする場合

```
# zpool import testpool examplepool
```

- プールを明示的にエクスポートしていない場合は、別のシステムでまだ使用されているプールを誤ってインポートすることを防ぐためにインポートできません。その場合は、-f オプションを使用する必要があります。

インポートしたいプールを ID で指定してインポートする場合

```
# zpool import 16722234661949228990
```

破棄 (destroy) された ZFS ストレージプールの回復

破棄されたプールを回復するには、zpool import コマンドに-Df オプションを使用して実行します。

```
# zpool import -Df testpool
```

 zpool import -D コマンドを使用すると破棄されたプールの情報が確認できます。

4-13.ZFS 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ボリュームの読み取り専用コピーのことで、スナップショット名は次のように指定します。

```
filesystem@snapname
```

```
volume@snapname
```

スナップショット名は、10 ページの「ZFS コンポーネントに名前を付けるときの規則」で定義されている規則に従って付ける必要があります。

ZFS スナップショットには次の特長があります。

- システムの再起動後も残ります。
- スナップショットの理論上の最大数は、 2^{64} (2 の 64 乗) です。
- スナップショットは、作成元のファイルシステムと同じストレージプールのディスク領域を直接消費します。

ボリュームのスナップショットに直接アクセスすることはできませんが、それらの複製、バックアップ、ロールバックなどを行うことはできます。

スナップショットには、変更できるプロパティはありません。また、データセットのプロパティをスナップショットに適用することもできません。

データセットのスナップショットが存在する場合、そのデータセットを破棄することはできません。次に例を示します。

4-13-1.スナップショットの作成

1) 「testpool/home」のスナップショットを「testpool/home@now」という名前をつけて作成します。

```
# zfs snapshot testpool/home@now
```

《参考》

「testpool/home」のすべての子孫を含めて、「testpool/home@now」という名前をつけてスナップショットを作成する場合。

```
# zfs snapshot -r testpool/home@now
```

☛ -r オプションを使用すると、すべての子孫のスナップショットを作成することができます。

作成されたスナップショットのみを確認する場合。

```
# zfs list -t snapshot
```

☛ -t オプションで表示するファイルタイプ「snapshot」を指定することで、作成されたスナップショットのみを表示できます。

snapshot を削除する場合。

```
# zfs destroy testpool/home@now
```

☛ -r オプションを使用すると子孫を含めたスナップショットを削除することができます。

スナップショットの名前を testpool/home@now から testpool/home@today に変更する場合。

```
# zfs rename testpool/home@now testpool/home@today  
または  
# zfs rename testpool/home@now today
```

☛ -r オプションを使用すると子孫を含めたスナップショットの名前を変更します。

☛ スナップショットの変更名が変更元のプールやファイルシステムと異なる場合はサポートされていません。

ファイルシステムのスナップショットには、ファイルシステムを含むルートの.zfs/snapshot ディレクトリからアクセスできます。たとえば、testpool/home/user1 が/testpool/home/user1 にマウントされている場合は、testpool/home/user1@now スナップショットのデータには、/testpool/home/user1/.zfs/snapshot/now ディレクトリからアクセスできます。

```
# ls /testpool/home/user1/.zfs/snapshot/  
now
```

☛ データは読み取り専用でアクセスできます。

4-1 3-2. ロールバック(rollback)

ロールバックを行うと、そのスナップショットより後に加えられたすべての変更を破棄して、ファイルシステムはそのスナップショットが作成されたときの状態に戻ります。

ロールバックを行うには以下の条件を満たす必要があります。

- 最新のスナップショット以外のスナップショットにロールバックすることはできません。それより前のスナップショットにロールバックするには、中間にあるスナップショットをすべて破棄する必要があります。
- 中間にあるスナップショットのクローンが存在する場合は、-R オプションを指定してクローンも破棄する必要があります。

1) testpool/home/user1@now を rollback します

```
# zfs rollback testpool/home/user1@now
```

- 👉 -r オプションを使用すると、testpool/home/user1@now 以降にとったスナップショットがあった場合、そのスナップショットは削除されます。
- 👉 rollback されるスナップショットは指定したファイルシステムのみ戻り、その配下に作成したファイルシステムは戻らないことに注意してください。

4-1 3-3. クローン(clone)

「クローン」とは、書き込み可能なボリュームまたはファイルシステムのことで、最初の内容は作成元のデータセットと同じです。スナップショットの場合と同様に、クローンは瞬間的に作成され、最初は追加のディスク領域を消費しません。また、クローンのスナップショットを作成することもできます。

クローンはデータセット階層内の別の場所に作成されますが、クローンが存在する間は元のスナップショットを破棄することはできません。また、クローンには、作成元のデータセットのプロパティ mountpoint 以外は継承されません。クローンのすべてのディスク領域は、最初は元のスナップショットと共有されます。

1) クローンを作成します。

testpool/home/user1@now のクローンを testpool/home/user2 という名前で作成します。

```
# zfs clone testpool/home/user1@now testpool/home/user2
```

《参考》

クローンのすべてのディスク領域は、最初は元のスナップショットと共有されるため、used プロパティの初期値はゼロになります。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
testpool             296K    29.3G    21K      /testpool
testpool/home        76.5K    29.3G    22K      /testpool/home
testpool/home/user1  18.5K    29.3G    18.5K    /testpool/home/user1
testpool/home/user1@now  0        -        18.5K    -
testpool/home/user2  0        29.3G    18.5K    /testpool/home/user2
```

クローンにプロパティを設定することも可能です。

作成したクローン testpool/home/user2 に 5G バイトの quota のプロパティを設定する場合。

```
# zfs set quota=5g testpool/home/user2
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
testpool             294K    29.3G    21K      /testpool
testpool/home        76.5K    29.3G    22K      /testpool/home
testpool/home/user1  18.5K    29.3G    18.5K    /testpool/home/user1
testpool/home/user1@now  0        -        18.5K    -
testpool/home/user2  0        5G       18.5K    /testpool/home/user2
```

クローンの破棄

クローン「testpool/home/user2」を破棄します。

```
# zfs destroy testpool/home/user2
```

クローンの置き換え

アクティブな ZFS ファイルシステムをそのファイルシステムのクローンで置き換えることができます。置き換え後は、「作成元」のファイルシステムが、指定されたファイルシステムのクローンになります。

クローン「testpool/home/user2」を置き換えます。

```
# zfs promote testpool/home/user2
```

この zfs list 出力から、元の testpool/home/user1 ファイルシステムの領域計上^g、testpool/home/user2 ファイルシステムの領域計上で置換されていることがわかります。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
testpool             296K    29.3G    21K      /testpool
testpool/home        58K     29.3G    22K      /testpool/home
testpool/home/user1  0        29.3G    18.5K    /testpool/home/user1
testpool/home/user2  18.5K    29.3G    18.5K    /testpool/home/user2
testpool/home/user2@now  0        -        18.5K    -
```

クローンの置き換えをすることで、クローン作成元のファイルシステムの削除が可能になります。

```
# zfs destroy testpool/home/user1
```

ファイルシステムの名前を変更することで、クローンの置換をすることも可能です。

```
# zfs rename testpool/home/user1 testpool/home/user1.old ←クローン作成元ファイルシステムの名前変更  
# zfs rename testpool/home/user2 testpool/home/user1 ←クローンの名前変更
```

4 - 1 4 .ZFS データの送信/受信 (send/receive)

zfs send コマンドを実行すると、スナップショットのストリーム表現が作成され、標準出力に書き込まれます。この出力は、ファイルまたは別のシステムにリダイレクトできます。

receive コマンドを実行すると、ストリームに内容が指定されているスナップショットが作成され、標準入力に渡されます。ストリーム全体を受信する場合は、新しいファイルシステムも作成されます。

4 - 1 4 - 1 .スナップショットの送信

1) testpool/home/user1@snap1 をテープデバイスにバックアップします。

```
# zfs send testpool/home/user1@snap1 > /dev/rmt/0
```

《参考》

zfs send -i オプションを使用すれば、増分データを送信できます。

testpool/home/user1@snap1 の増分データ snap2 をテープデバイスにバックアップします。

```
# zfs send -i testpool/home/user1@snap1 testpool/home/user1@snap2 > /dev/rmt/0
```

👉 増分ソース snap1 は、スナップショット名の「@」以降の構成要素「snap1」だけでの指定も可能です。

同じシステム上の別のストレージプールに送信する場合。

testpool/home/user1@snap1 を backuppool/data に snap1 と名前を付けて送信します。

```
# zfs send testpool/home/user1@snap1 | zfs receive backuppool/data/snap1
```

👉 zfs receive コマンドは zfs recv と入力しても実行可能です。

4-1 4-2.ZFS スナップショットの受信

ファイルシステムのスナップショットを受信するときは、以下の点に留意してください。

- スナップショットおよびファイルシステムが受信されます。
- ファイルシステムとその子孫のすべてのファイルシステムがマウント解除されます。
- ファイルシステムが受信されている間は、それらにアクセスできません。
- 受信される元のファイルシステムは、その転送中に存在してはいけません。
- ファイルシステム名が重複している場合には、`zfs rename` を使用してそのファイルシステムの名前を変更できます。

1) 現在の `zfs` ファイルシステムの状態を確認します。

```
# zfs list
```

2) テープデバイスからスナップショットを受信します。

テープデバイスから `testpool/home` 配下に `user1_data` という名前でスナップショットを受信します。

```
# zfs receive testpool/home/user1_data < /dev/rmt/0
```

《参考》

対象のファイルシステムに変更を加え、新たに増分スナップショットを送信する場合は、まず受信側のファイルシステムをロールバックする必要があります。`zfs receive` コマンドで `-F` オプションを使用すればロールバック手順をする必要がありません。

変更を加えた「`/backuppool/home/user1`」をロールバックして、増分スナップショット「`testpool/home/user1@snap2`」を受信する場合

```
# zfs send -I snap1 testpool/home/user1@snap2 testpool/home/user1@snap3 > zfs recv-F  
/backuppool/home/user1_snap
```

- 👉 最新のスナップショットが増分ソースと一致しない場合は、ロールバックも受信も完了せず、エラーメッセージが返されます。
- 👉 `zfs receive` コマンドに、増分ソースと一致しない異なるファイルシステムの名前を間違えて指定した場合は、ロールバックも受信も完了せず、エラーメッセージが返されます。

4-15 ZFS データの完全性をチェックする (scrub)

ZFS には fsck に相当するユーティリティーは存在しません。代わりに「スクラブ」という機能が用意されています。この機能はプールに含まれるすべてのデータを 1 回たどってみて、すべてのブロックが読み取り可能であることを確認します。

1) testpool をスクラブします。


```
# zpool scrub testpool
```

2) 現在のスクラブの状態を表示するには zpool status コマンドを実行します。

```
# zpool status -v testpool
プール: testpool
状態: ONLINE
スクラブ: Mon Sep 14 09:31:59 2009
上で 0 エラーが発生した 0h0m のあとの scrub completed 構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool	ONLINE	0	0	0
mirror	ONLINE	0	0	0
c2t10d0s0	ONLINE	0	0	0
c2t10d0s1	ONLINE	0	0	0
スペア				
c2t10d0s4	AVAIL			

エラー: 既知のデータエラーはありません

 -v オプションを使用すると、詳細なデータエラー情報を表示し、前回のスクラブが完了して以降のデータエラーすべての完全なリストを出力します。

《参考》

現在 testpool で進行中のスクラブを中止する場合。

```
# zfs scrub -s testpool
```

 -s オプションを使用すると、進行中のスクラブを中止することができます。

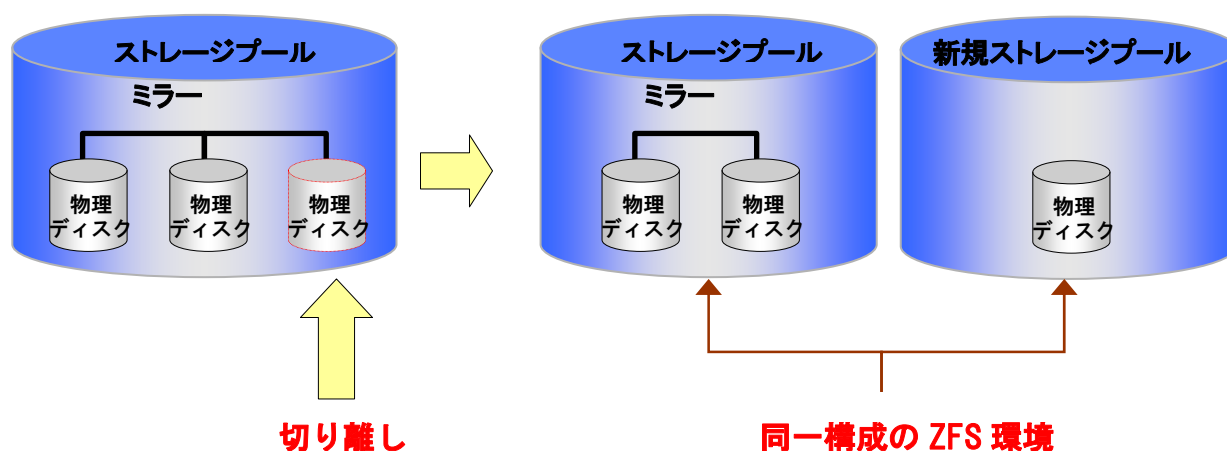
4-16. Oracle Solaris 10 OS 9/10 (U9)からの新機能

4-16-1. ミラーディスクの切り離し(split)

ミラー構成の ZFS ストレージプールからミラー構成のディスクを切り離して、切り離し元と同じ構成を持った新規ストレージプールを作成することが可能です。

切り離したディスクをインポートすることで、元の ZFS ストレージプールと同じ構成で新規ストレージプールを作成することができるのでバックアップとして利用することが可能です。

※データの冗長性を保つため、3 本以上のミラーディスクに対して実施することを推奨します。



1) testpool のデバイス状態を確認します。

```
# zpool status testpool
  プール: testpool
  状態: ONLINE
  スクラブ: Wed Jan  5 13:56:33 2011
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:

    NAME                STATE          READ    WRITE    CKSUM
  testpool              ONLINE         0        0         0
    mirror-0  ONLINE            0        0         0
    c0t1d0    ONLINE            0        0         0
    c0t2d0    ONLINE            0        0         0

エラー: 既知のデータエラーはありません
```

📌 Solaris 10 OS 9/10 (U9)の場合、ミラー構成は mirror-*で表示されます

2) ファイルシステムを確認します。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
testpool            140K    4.89G    23K      /testpool
testpool/home       42K     4.89G    21K      /testpool/home
testpool/home/user1 21K     4.89G    21K      /testpool/home/user1
```

3) testpool へミラー用ディスクを追加します。

```
# zpool attach testpool c0t1d0 c0t3d0
```

4) attach 後に testpool デバイスの状態を確認します。

```
# zpool status testpool
  プール: testpool
  状態: ONLINE
  スクラブ: Fri Mar 25 13:10:33 2011
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:

    NAME                STATE  READ  WRITE  CKSUM
    testpool            ONLINE  0     0     0
      mirror-0          ONLINE  0     0     0
        c0t1d0           ONLINE  0     0     0
        c0t2d0           ONLINE  0     0     0
        c0t3d0           ONLINE  0     0     0          73.5K resilvered
```

エラー: 既知のデータエラーはありません

🟢 testpool にミラー用ディスク(c0t3d0)が追加され、3面ミラー構成になっていることを確認します。

5) ミラーディスクを切り離します。

```
# zpool split testpool testpool2
```

🟢 testpool からディスクを切り離して新規ストレージプール(testpool2)を作成します。

6) split 後に testpool のデバイス状態を確認します。

```
# zpool status testpool
  プール: testpool
  状態: ONLINE
  スクラブ: Fri Mar 25 13:10:33 2011
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:

    NAME                STATE  READ  WRITE  CKSUM
    testpool            ONLINE  0     0     0
      mirror-0          ONLINE  0     0     0
        c0t1d0           ONLINE  0     0     0
        c0t2d0           ONLINE  0     0     0
```

エラー: 既知のデータエラーはありません

🟢 testpool のディスク (c0t3d0) が切り離されていることを確認します。

7) 切り離して作成した新規ストレージプールを確認します。

```
# zpool import
```

```
  プール: testpool2
```

```
  状態: ONLINE
```

```
  スクラブ: 何も要求されませんでした
```

```
  構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool2	ONLINE	0	0	0
c0t3d0	ONLINE	0	0	0

エラー: 既知のデータエラーはありません

🟢 import 可能なストレージプールが表示されます。

8) 切り離して作成した新規ストレージプール testpool2 をインポートします。

```
# zpool import testpool2
```

9) testpool2 のデバイス構成を確認します。

```
# zpool status testpool2
```

```
  プール: testpool2
```

```
  状態: ONLINE
```

```
  スクラブ: 何も要求されませんでした
```

```
  構成:
```

NAME	STATE	READ	WRITE	CKSUM
testpool2	ONLINE	0	0	0
c0t3d0	ONLINE	0	0	0

エラー: 既知のデータエラーはありません

🟢 testpool2 がインポートされていることを確認します。

10) testpool2 ヘミラー用ディスクを追加します。

```
# zpool attach testpool2 c0t3d0 c0t4d0
```

1 1) testpool2 のデバイス構成を確認します。

```
# zpool status testpool2
  プール: testpool2
  状態: ONLINE
  スクラブ: Wed Jan  5 14:14:19 2011
  上で 0 エラーが発生した 0h0m のあとの resilver completed 構成:
```

NAME	STAT	READ	WRITE	CKSUM
testpool2	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
c0t3d0	ONLINE	0	0	0
c0t4d0	ONLINE	0	0	0 146K resilvered

エラー: 既知のデータエラーはありません

👉 testpool2 にミラー用ディスク(c0t4d0)が追加され、2面ミラー構成になっていることを確認します。

1 2) testpool2 のファイルシステムを確認します。

```
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
testpool	173K	9.78G	23K	/testpool
testpool/home	42K	9.78G	21K	/testpool/home
testpool/home/user1	21K	9.78G	21K	/testpool/home/user1
testpool2	175K	9.78G	23K	/testpool2
testpool2/home	44K	9.78G	23K	/testpool2/home
testpool2/home/user1	21K	9.78G	21K	/testpool2/home/user1

👉 testpool と testpool2 のファイルシステム構成とサイズが同一であることを確認します。

4-16-2.ZFS スナップショットの保護(hold)

スナップショットが不用意に削除されないように、スナップショットを保護することが可能です。

この機能と `zfs destroy -d` コマンドを使用することでクローンが存在するスナップショットを遅延削除することが可能です。この場合、保護解除とクローン削除を実行したタイミングでスナップショットが削除されます。

1) スナップショットを作成します。

```
# zfs snapshot testpool/home/user1@now
```

- 2) ファイルシステムを確認します。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
Testpool            140K    9.78G    23K      /testpool
testpool/home       44K     9.78G    23K      /testpool/home
testpool/home/user1 21K     9.78G    21K      /testpool/home/user1
testpool/home/user1@now 0       -        21K      -
```

- 3) 作成したスナップショットを保護します。

```
# zfs hold mytag testpool/home/user1@now
```

 testpool/home/user1@now に保護タグ mytag を付与します。

- 4) 保護したスナップショットを表示します。

```
# zfs holds testpool/home/user1@now
NAME                TAG      TIMESTAMP
testpool/home/user1@now  mytag   Wed Feb 16 18:43:24 2000
```

保護されたスナップショットを削除しようとした場合。

```
# zfs destroy testpool/home/user1@now
cannot destroy 'testpool/home/user1@now': dataset is busy
```

 保護しているスナップショットは削除できません。削除を行う場合は保護を解除する必要があります。

- 5) スナップショットの保護を解除します。

```
# zfs release mytag testpool/home/user1@now
```

- 6) 保護の解除後、スナップショットを削除します。

```
# zfs destroy testpool/home/user1@now
```

- 7) ファイルシステムを確認します。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
Testpool            138K    9.78G    23K      /testpool
testpool/home       44K     9.78G    23K      /testpool/home
testpool/home/user1 21K     9.78G    21K      /testpool/home/user1
```

 スナップショットが削除されていることを確認します。

《参考》クローンが存在するスナップショットの削除

スナップショットの保護機能と `zfs destroy -d` コマンドを使用することでクローンが存在するスナップショットを削除することが可能です。

clone を作成します。

`testpool/home/user1@now` のクローンを `testpool/home/user1_clone` という名前で作成します。

```
# zfs clone testpool/home/user1@now testpool/home/user1_clone
```

作成した clone を確認します。

```
# zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
testpool            146K    9.78G    23K      /testpool
testpool/home       44K     9.78G    23K      /testpool/home
testpool/home/user1 21K     9.78G    21K      /testpool/home/user1
testpool/home/user1@now 0       -        21K     -
testpool/home/user1_clone 0       9.78G    21K     /testpool/home/user1_clone
```

スナップショット `testpool/home/user1@now` を保護します。

```
# zfs hold mytag testpool/home/user1@now
```

保護したスナップショットを確認します。

```
# zfs holds testpool/home/user1@now
NAME                TAG          TIMESTAMP
testpool/home/user1@now mytag       Sun Feb 20 13:32:51 2000
```

保護したスナップショットを削除します。

※保護の解除とクローンの削除が実行されるまで削除されません。

```
# zfs destroy -d testpool/home/user1@now
```

削除したスナップショットの `defer_destroy` プロパティを確認します。

```
# zfs get defer_destroy testpool/home/user1@now
NAME                PROPERTY          VALUE    SOURCE
testpool/home/user1@now defer_destroy     on       -
```

 遅延削除を実行すると `defer_destroy` プロパティが `on` になります。

4-17.トラブルシューティング

4-17-1. 損傷した ZFS 構成の修復

ZFS では、アクティブなプールとその構成のキャッシュをルートファイルシステム上で管理しています。このファイルが破壊した場合、またはこのファイルがなんらかの形でディスクに保管されている内容と同期しなくなった場合には、そのプールを開くことができなくなります。なんらかの方法でプールを見ることができる場合には、プールをエクスポートして再度インポートする方法で、構成を回復することができます。

プールのインポートとエクスポートの詳細については、本手順書の「ZFS ストレージプールの移行」を参照してください。

4-17-2. 見つからないデバイスを使用可能にする

デバイスを開けない場合には、zpool status の出力に UNAVAILABLE と表示されます。この状態は、プールにはじめてアクセスしたときにデバイスを開けなかったか、またはそのデバイスがそれ以降使用できない状態であることを示しています。このデバイスが原因で、最上位レベルの仮想デバイスが使用できない場合、そのプールの内容にはアクセスできません。この場合、通常の動作に戻すために必要な操作は、そのデバイスをシステムに再接続することだけです。

修復の基本的な流れは以下のようになります。

1. デバイスを物理的に再接続する
2. デバイスをシステムに再接続しても、デバイスが使用できるようになったことが自動的に検出されないこともあります。その場合は、zpool online コマンドを使ってZFSに通知します。

4-17-3. 損傷したデバイスを修復する

➤ 一時的なエラーを消去

デバイスエラーが一時的で今後の使用に影響しないと考えられる場合は、デバイスエラーを安全に解消することで、致命的なエラーが発生していないことを指定することができます。

エラーを消去するには、zpool clear コマンドを使用します。

➤ ZFS ストレージプール内のデバイスを置き換える

デバイスの損傷が永続的である場合、または永続的な損傷が今後発生する可能性がある場合には、そのデバイスを置き換える必要があります。デバイスを置き換えられるかどうかは、構成によって異なります。

置き換えの可否	状態	対処
置き換え可能	<ul style="list-style-type: none"> ・プールがONLINE 状態である かつ ・正常なデータを取得するための十分な複製が存在している 	デバイスの置き換え
置き換え不可	<ul style="list-style-type: none"> ・プールが障害状態 または ・非冗長な構成でデバイスに大量のデータエラーが含まれている 	プールの破棄 → プールの再構成 →データの復元

置き換えられるデバイスであることを確認したら、zpool replace コマンドを使ってデバイスを置き換えます。デバイスの置き換えの詳細については、本手順書の「デバイスの置き換え」を参照してください。

4-17-4. 損傷したデータを修復する

➤ 破壊されたファイルまたはディレクトリを修復する

ファイルまたはディレクトリに損傷が発生している場合、データの正常なコピーがシステムのどこにも存在しなければ、事実上それらはバックアップからデータを復元しなければいけませんが、プール全体を復元しなくても損傷から回復できる場合があります。

ファイルデータブロックの中で損傷が発生した場合は、ファイルを安全に削除することができるため、システムのエラーを解消できます。永続的なエラーが発生しているファイル名のリストを表示するには、zpool status -v コマンドを使用します。次に例を示します。

```
# zpool status -v
pool: testpool
state: ONLINE
status: One or more devices has experienced an error resulting in data
corruption. Applications may be affected.
action: Restore the file in question if possible. Otherwise restore the
entire pool from backup.
see: http://www.sun.com/msg/ZFS-8000-8A
scrub: none requested
config:
NAME      STATE  READ  WRITE  CKSUM
testpool  ONLINE  0     0     0
c1t1d0s6  ONLINE  0     0     0
c1t1d0s7  ONLINE  0     0     0
errors: Permanent errors have been detected in the following files:
/testpool/a.txt
/testpool/user1/b.txt
/testpool/user2/dir/d.txt
/testpool/user3/e.txt
/testpool/user3/dir1/f.txt
```


コマンド結果の説明

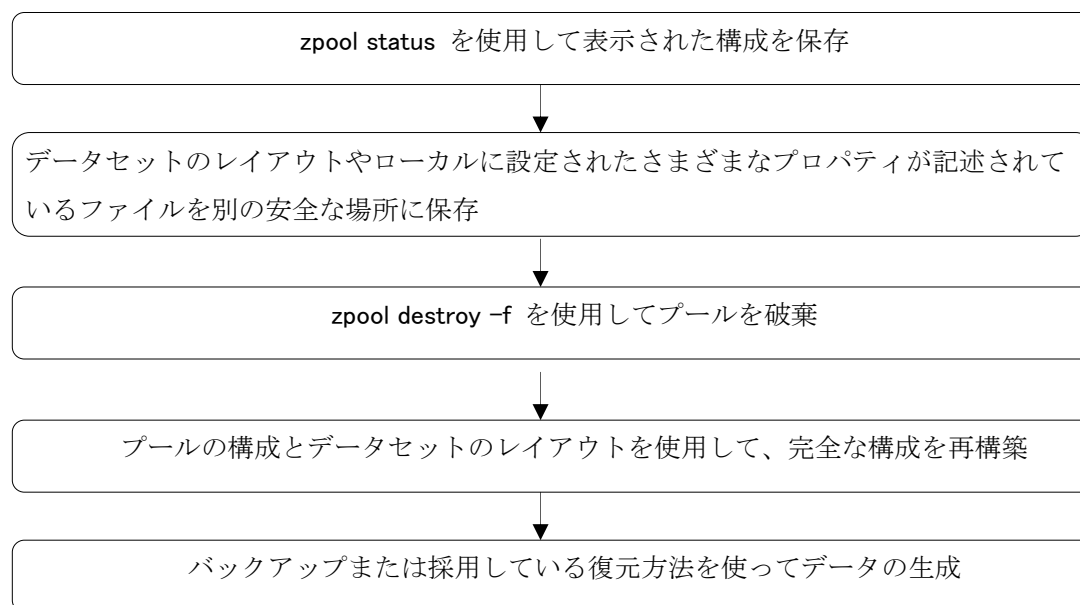
表示例	状態
/testpool/a.txt	ファイルへの完全なパスが見つかり、データセットがマウントされている場合。ファイルへの完全なパスが表示されます。
testpool/user3/e.txt	ファイルへの完全なパスは見つかったが、データセットがマウントされていない場合。前にスラッシュ(/) が付かず、後ろにファイルへのデータセット内のパスが付いたデータセット名が表示されます。
testpool/dnode:<0x0>	エラーにより、またはdnode_t の場合のようにオブジェクトに実際のファイルパスが関連付けられていないことにより、ファイルパスに対するオブジェクト番号を正常に変換できない場合。後ろにオブジェクト番号の付いたデータセット名が表示されます。
<metadata>	メタオブジェクトセット(MOS) のオブジェクトが破壊された場合は、後ろにオブジェクト番号の付いた<metadata> という特別なタグが表示されます。

ディレクトリまたはファイルのメタデータの中で破壊が発生している場合には、そのファイルを別の場所に移動するしかありません。任意のファイルまたはディレクトリを別の場所に安全に移動することができ、そこで元のオブジェクトを復元することができます。

➤ ZFS ストレージプール全体の損傷を修復する

プールメタデータで損傷が発生していて、プールを開くことができない場合は、プールとそのすべてのデータをバックアップから復元する必要があります。

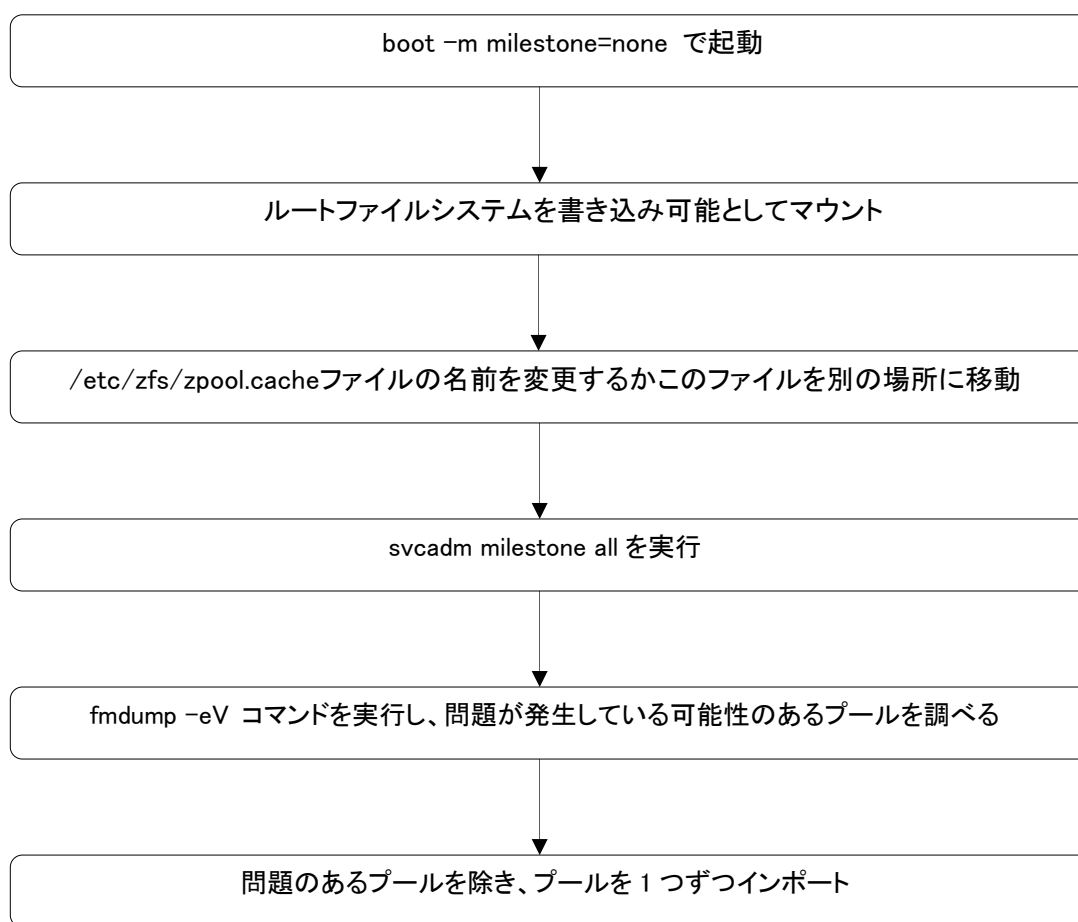
その流れは以下の通りです。



4-18. 起動できないシステムの修復

ZFS は、エラーが発生した場合でも、堅牢で安定した状態であるように設計されていますが、予期しない異常な操作のためにプールにアクセスするときにパニックが発生することがあります。この状況から回復するには起動時にどのプールも探さないように ZFS を設定する必要があります。

以下にその手順を記述します。



付録1. プロパティの紹介

次の表には、ZFS ファイルシステムの読み取り専用のネイティブプロパティと設定可能なネイティブプロパティの両方を示しています。読み取り専用のネイティブプロパティには、その旨を記載しています。それ以外のプロパティは、すべて設定可能なプロパティです。

プロパティ名	種類	デフォルト値	説明
aclinherit	文字列	secure	ファイルとディレクトリが作成されるときに ACL エントリをどのように継承するかを制御します。値は、discard、noallow、secure、および passthrough です。
aclmode	文字列	groupmask	chmod を実行するときに ACL エントリをどのように変更するかを制御します。値は、discard、groupmask、および passthrough です。
atime	ブール型	on	ファイルを読み取るときにファイルのアクセス時刻を更新するかどうかを制御します。このプロパティをオフに設定すると、ファイルを読み取る時に書き込みトラフィックが生成されなくなるため、パフォーマンスが大幅に向上する可能性があります。ただし、メールプログラムなどのユーティリティーが予期しない動作をすることがあります。
available	数値	なし	読み取り専用プロパティ。データセットおよびそのすべての子で利用できる容量を調べます。プール内でほかのアクティビティーが実行されていないことを前提とします。容量はプール内で共有されるため、プールの物理サイズ、割り当て制限、予約、プール内のほかのデータセットなどのさまざまな要因によって、利用できる容量が制限されることがあります。 このプロパティは、列の短縮名avail を使用して参照することもできます

プロパティ名	種類	デフォルト値	説明
canmount	ブール型	on	<p>指定のファイルシステムがzfs mount コマンドを使ってマウントできるかどうかを制御します。このプロパティはどのファイルシステムにも設定可能で、プロパティ自体は継承可能ではありません。ただし、このプロパティをoff に設定するとマウントポイントを子孫のファイルシステムに継承できますが、ファイルシステム自体がマウントされることはありません。</p> <p>noauto オプションを設定すると、データセットのマウントおよびマウント解除は明示的に実行することが必要になります。データセットの作成時やインポート時に、データセットが自動的にマウントされることはありません。また、zfs mount-a コマンドでマウントされることや、zfs unmount-a コマンドでマウント解除されることもありません。</p>
compression	文字列	off	<p>このデータセットに対する圧縮を有効または無効にします。値はon、off、lzjb、gzip、またはgzip-N です。このプロパティをlzjb、gzip、またはgzip-N に設定することは、このプロパティをonに設定することと同じ効果を持ちます。デフォルト値はoff です。既存のデータを持つファイルシステムでcompression を有効にした場合は、新しいデータのみが圧縮されます。既存のデータは圧縮されないまま残されます。</p> <p>このプロパティは、列の短縮名compress を使用して参照することもできます。</p>
compressratio	数値	なし	<p>読み取り専用プロパティ。このデータセットに適用された圧縮率を調べます。乗数で表現されます。zfsset compression=on dataset を実行すると、圧縮を有効にできます。</p> <p>すべてのファイルの論理サイズおよび参照する物理データの量から計算されます。compression プロパティを使用して明示的に圧縮されたデータセットも含まれます。</p>
copies	数値	1	<p>ファイルシステムごとのユーザーデータのコピー数を設定します。使用できる値は1、2、または3 です。これらのコピーは、プールレベルの冗長性を補うものです。ユーザーデータの複数のコピーで使用される領域は、対応するファイルとデータセットから取られるため、割り当て制限と予約にとって不利に働きます。また、複数のコピーを有効にすると、used プロパティが更新されます。既存のファイルシステムでこのプロパティを変更しても、新たに書き出されるデータが影響を受けるだけなので、ファイルシステムの作成時にこのプロパティの設定を検討してください。</p>

プロパティ名	種類	デフォルト値	説明
creation	文字列	なし	読み取り専用プロパティ。このデータセットが作成された日時を調べます。
devices	ブール型	on	ファイルシステム内のデバイスファイルを開くことができるかどうかを制御します。
exec	ブール型	on	このファイルシステムに含まれるプログラムの実行を許可するかどうかを制御します。off に設定した場合は、PROT_EXEC による mmap(2) 呼び出しも許可されません。
mounted	boolean	なし	読み取り専用のプロパティ。このファイルシステム、クローン、またはスナップショットが現在マウントされているかどうかを調べます。このプロパティは、ボリュームには適用されません。値にはyesまたはno を指定できます。
origin	文字列	なし	複製されたファイルシステムまたはボリュームのための読み取り専用プロパティ。どのスナップショットからクローンが作成されたかを調べます。クローンが存在する場合には、-r や-f オプションを使用しても、作成元は破棄できません。複製されていないファイルシステムには、作成元は存在しません。
quota	数値(または none)	none	データセットおよびその子孫が消費できる容量を制限します。このプロパティは、使用される容量に強い制限値を適用します。容量には、子孫(ファイルシステムやスナップショットを含む) が消費するすべての容量も含まれます。割り当て制限がすでに設定されているデータセットの子孫に割り当て制限を設定した場合は、祖先の割り当て制限は上書きされずに、制限が追加されます。ボリュームには割り当て制限を設定できません。volsize プロパティが暗黙的な割り当て制限として機能します。
readonly	ブール型	off	このデータを変更できるかどうかを制御します。on に設定すると、そのデータセットを変更できなくなります。 このプロパティは、列の短縮名rdonly を使用して参照することもできます。
recordsize	数値	128K	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、列の短縮名recsize を使用して参照することもできます。

プロパティ名	種類	デフォルト値	説明
referenced	数値	なし	なし読み取り専用プロパティ。このデータセットからアクセスできるデータの量を調べます。プール内のほかのデータセットで共有されるデータも含まれることがあります。 スナップショットまたはクローンを作成したときには、それらの作成元のファイルシステムまたはスナップショットと同じ領域を最初は参照しています。内容が同じであるためです。 このプロパティは、列の短縮名refer を使用して参照することもできます
refquota	数値(またはnone)	none	1 つのデータセットが消費できる容量を設定します。このプロパティにより、使用される容量に対して強い制限値が設定されます。この強い制限値には、スナップショットやクローンなどの子孫で使用される容量は含まれません。
reservation	数値(またはnone)	none	データセットおよびその子孫に保証される最小容量。使用している容量がこの値を下回っているデータセットは、予約に指定された容量を使用していると見なされます。予約は、親データセットが使用する容量に計上されるので、親データセットの割り当て制限と予約を減らすこととなります。 このプロパティは、列の短縮名reserv を使用して参照することもできます。
setuid	ブール型	on	そのファイルシステムでsetuid ビットを考慮するかどうかを制御します。
sharenfs	文字列	off	ファイルシステムをNFS 経由で使用できるかどうか、およびどのオプションを使用するかを制御します。on に設定した場合は、zfs share コマンドがオプションなしで呼び出されます。または、このプロパティの内容に対応するオプションを使って、zfs share コマンドが呼び出されます。off に設定した場合は、従来のshare とunshare コマンド、およびdfstabファイルを使用してファイルシステムが管理されます。
snapdir	文字列	hidden	ファイルシステムのルートから.zfs ディレクトリを見えるようにするかどうかを制御します。
type	文字列	なし	読み取り専用プロパティ。データセットの種類を調べます。filesystem (ファイルシステムまたはクローン)、volume、またはsnapshot のいずれかになります。
used	数値	なし	読み取り専用プロパティ。データセットおよびそのすべての子孫が消費する容量を調べます。

プロパティ名	種類	デフォルト値	説明
volsize	数値	なし	ボリュームの場合に、ボリュームの論理サイズを指定します。
volblocksize	数値	8K バイト	ボリュームの場合に、ボリュームのブロックサイズを指定します。ボリュームが書き込まれたあとに、ブロックサイズを変更することはできません。ブロックサイズはボリュームを作成するときに設定してください。ボリュームのデフォルトブロックサイズは、8K バイトです。512 バイトから128K バイトの範囲で、任意の2 の累乗を指定できます。 このプロパティは、列の短縮名 volblock を使用して参照することもできます。
zoned	ブール型	なし	このデータセットが非大域ゾーンに追加されているかどうかを指定します。このプロパティが設定されている場合、そのマウントポイントは大域ゾーンで考慮されません。ZFS では、このようなファイルシステムを要求されても、マウントすることはできません。ゾーンを最初にインストールしたときには、追加されたすべてのファイルシステムにこのプロパティが設定されます。
xattr	ブール型	on	このファイルシステムで拡張属性を有効にするか無効にするかを指定します。デフォルト値はon です。

FUJITSU

shaping tomorrow with you