Orchestrating a brighter world NEC

NEC Scalable Technology File System for AI (ScaTeFS for AI)
ユーザーズガイド

輸出する際の注意事項

本製品（ソフトウェアを含む）は、外国為替および外国貿易法で規定される規制貨物（または役務）に該当することがあります。

その場合、日本国外へ輸出する場合には日本国政府の輸出許可が必要です。

なお、輸出許可申請手続きにあたり資料等が必要な場合には、お買い上げの販売店またはお近くの当社営業拠点にご相談ください。

は　し　が　き

このマニュアルは、NEC Scalable Technology File System for AI 用のユーザーズガイドです。ここではマニュアルの目的、対象読者について説明します。

マニュアルの目的

このマニュアルでは、ScaTeFS の機能概要や運用管理の操作方法、コマンド、メンテナンスに関する説明を目的としています。

対象読者

このマニュアルは、次の方を対象読者として記述しています。

- システム管理者
- 一般利用者

備考

(1) Linux は Linus Torvalds氏 の日本およびその他の国における登録商標あるいは商標です。

(2) Red Hatは米国およびその他の国でのRed Hat, Inc. の登録商標もしくは商標です。

(3) CLUSTERPROは日本電気株式会社の登録商標です。

(4) Windowsは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。

(5) NVIDIAは、米国およびその他の国におけるNVIDIA Corporationの登録商標または商標です。

(6) その他記載の会社名、製品名は、それぞれの会社の商標、もしくは登録商標です。

本書の読み進め方

本書は、次の構成となっています。章ごとの対象読者の範囲は、表の一番右の列に示しています。

章	タイトル	内容	対象読者
1	機能概要	ScaTeFS for AIの主な機能やファイル分散の概要、設定について記載しています。	システム管理者 一般利用者
2	効率的に並列I/Oを行うための機能	効率的に並列I/Oを行うための機能として、ファイルのプリマップやフォーマット等について記載しています。	システム管理者 一般利用者
3	運用/操作方法	QUOTAなど運用管理に必要な機能や操作方法を記載しています。	システム管理者 一般利用者
4	ScaTeFSクライアント用とIOサーバ用のコマンドリファレンス一覧	ScaTeFSクライアントとIOサーバ上で利用できるScaTeFSコマンドの説明や実行書式、オプション等を記載しています。	システム管理者 一般利用者

関連説明書

- 『NEC Scalable Technology File System for AI (ScaTeFS for AI) システムガイド』
- 『NEC Scalable Technology File System for AI (ScaTeFS for AI) インストレーションガイド』
- 『NEC Scalable Technology File System for AI (ScaTeFS for AI) ソフトウェアライセンス管理説明書』

各種説明書は以下のNECサポートポータルのWebページから参照できます。
https://www.support.nec.co.jp/View.aspx?id=3170102881

用語定義・略語

用語	意味
GbE	「Gigabit Ethernet」の略称。
HDLM	「Hitachi Dynamic Link Manager」の略称。
IOサーバ	ScaTeFSを構成するサーバ。最低２台必要。
IOサーバデーモン	IOサーバ上で動作するデーモン。
NFS	「Network File System」の略称。
RHEL	「Red Hat Enterprise Linux」の略称。
ScaTeFS for AI	「NEC Scalable Technology File System for AI」の略称。
フェアシェアI/Oスケジューリング	ユーザ毎、またはノード毎にサーバ資源を公平に割り当てる機能のこと。
プリマップ	指定されたファイルサイズに相当する個数の実ファイルを各実ファイルシステム上に予め生成する機能のこと。同ファイルへの並列writeを行う場合、実ファイル生成のオーバーヘッドをプリマップにより低減することが目的。scatefs_premapを使用。
ルートIOサーバ	IOサーバの一種。mkfsを実行するサーバであり、クライアントがマウントする際のマウント先のサーバ。システム運用中においては、他のIOサーバと特に相違はなく同様な処理を行う。
仮想ファイル	仮想ファイルシステム上に作成されたファイル。ScaTeFS上のレギュラーファイル。
仮想ファイルシステム	複数のIOターゲットにより構成されるクライアント見えのファイルシステム。ScaTeFSそのもの。
実ファイル	複数のサーバに跨った仮想ファイルの断片。実際には、実ファイルシステム上のファイルのこと。
実ファイルシステム、IOターゲット	仮想ファイルシステムを構成する基本単位。各IOサーバ配下に作成される。実態は、Linuxで使用可能な通常のファイルシステム。
並列I/O	複数の計算ノードを使用して並列にデータを転送することにより、１つのファイルへの書き込み、読み込みを行うこと。巨大ファイルへのI/O効率を上げることが主たる目的。

目　次

表目次

図目次

第1章　　機能概要

NEC Scalable Technology File System(ScaTeFS: スケートエフエス)は、AIやHPCのシステムの大規模化、データの大容量化に対応できる分散・並列ファイルシステムです。ファイルシステム全体を管理するサーバは存在せず、データ、メタデータともに複数のIOサーバへ一様に分散 配置し、read/writeリクエストの処理、ファイル、ディレクトリの生成や属性の参照／更新などファイルシステムとしての基本的な機能すべてをリクエスト毎に各IOサーバへ分散して処理することで、負荷分散を行います。

本書では、このような分散の仕組みを活かし効率よく並列IOを行うための機能と運用管理で必要な機能の利用方法について説明します。

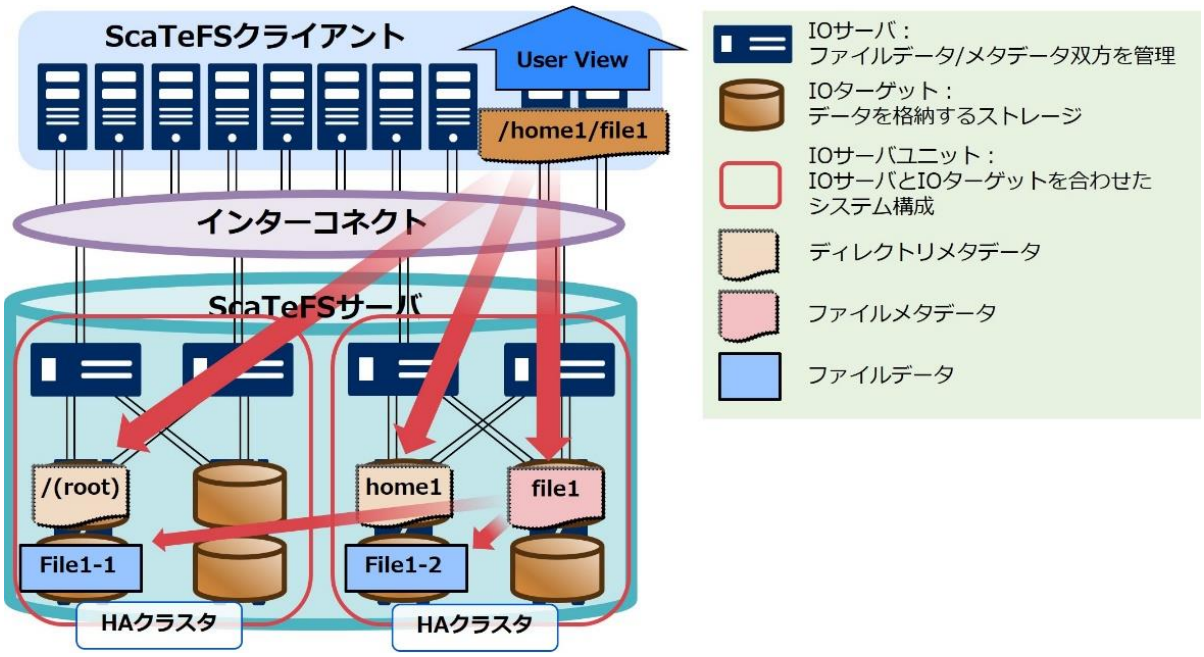

図 1-1　ScaTeFS の分散処理のイメージ

1.1　ScaTeFSでのファイル分散について

1.1.1 仮想ファイルシステムと実ファイルシステム

ScaTeFSは、複数のIOサーバにより構成されており、これを仮想的に１つのファイルシステムとしてScaTeFSクライアントに見せています。このため、これを「仮想ファイルシステム」と呼称します。

仮想ファイルシステムは、図 1-2のように各IOサーバ配下に接続されたストレージ上に作成される複数のLinuxのファイルシステムから成っています。これらを「実ファイルシステム」または「IOターゲット」と呼称します。実ファイルシステムは、各IOサーバ配下に最低１つ、一般に複数存在します。並列I/Oを効率よく実施するためには、仮想ファイルシステムが何台のIOサーバと実ファイ

ルシステムにより構成されているかを把握しておく必要があります。

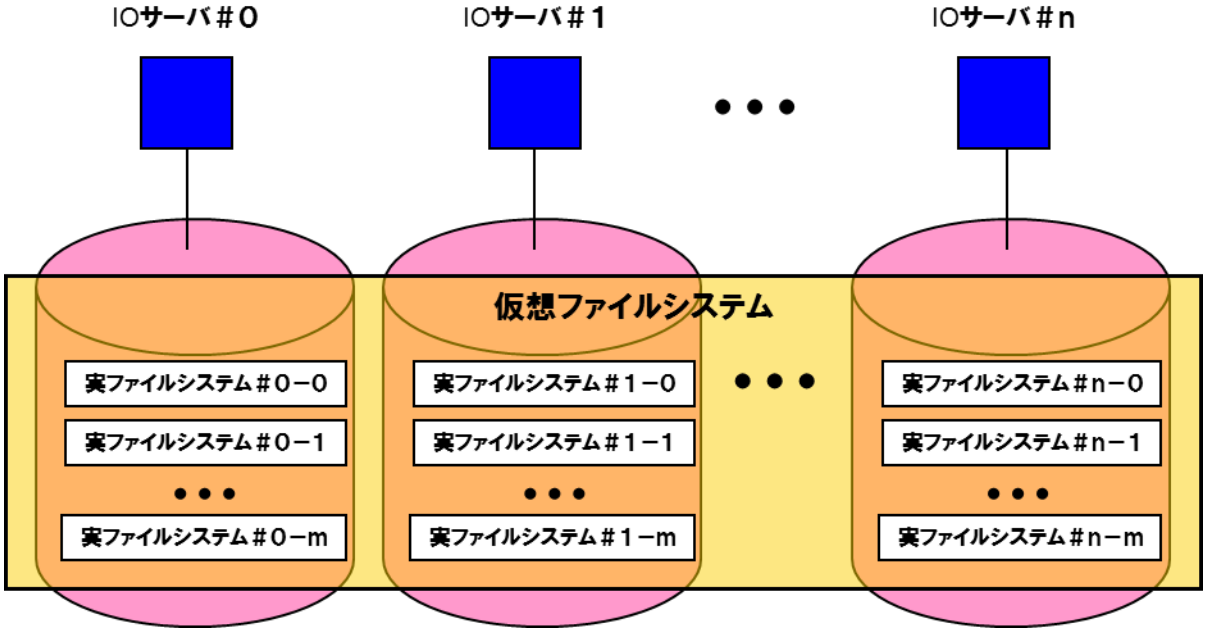

図 1-2　仮想ファイルシステムと実ファイルシステムの関係

図 1-2の例では、ファイルのデータは最大(n + 1)×(m + 1)個の実ファイルシステムに分散配置されることになります。

1.1.2 仮想ファイルと実ファイル

仮想ファイルの断片を各実ファイルシステムに分散配置します。この断片のことを実ファイルと呼称します。この断片と各実ファイルシステムへの配置の方法の違いにより、２種類のファイルフォーマットが選択できます。

形式１：ノンストライプフォーマット

形式２：ストライプフォーマット

デフォルトは、ノンストライプフォーマットです。

1.1.2.1　ノンストライプフォーマット（形式１）

図 1-3 の仮想ファイルのイメージのとおり、仮想ファイルは実ファイルを順に連結したものです。この連結の単位をチャンクサイズと呼称します。この値は、後述の scatefs_premap により設定可能であり、チャンクサイズの既定値は 256MB です。

図 1-3 は、ノンストライプフォーマット(形式１)の場合の仮想ファイルのイメージとこれを構成する実ファイルが各実ファイルシステムへどのように配置されるかを例示しています。

この場合、各 IO サーバ配下にそれぞれ２つの実ファイルシステム(ターゲット)を作成して１つの ScaTeFS を構成しています。図 1-3 の仮想ファイルは、チャンク番号#0～#10 で

構成されており、仮想ファイルの先頭であるチャンク番号#0 は、TID＝１に配置されています。以降は、これを起点として

TID＝(１，２，３，０，５，６，７，４，１・・・)

のように最初はチャンク番号#0 が配置されたターゲットと同列の各 IO サーバ配下のターゲットに配置され、IO サーバを一巡すると次のターゲットに配置されます。

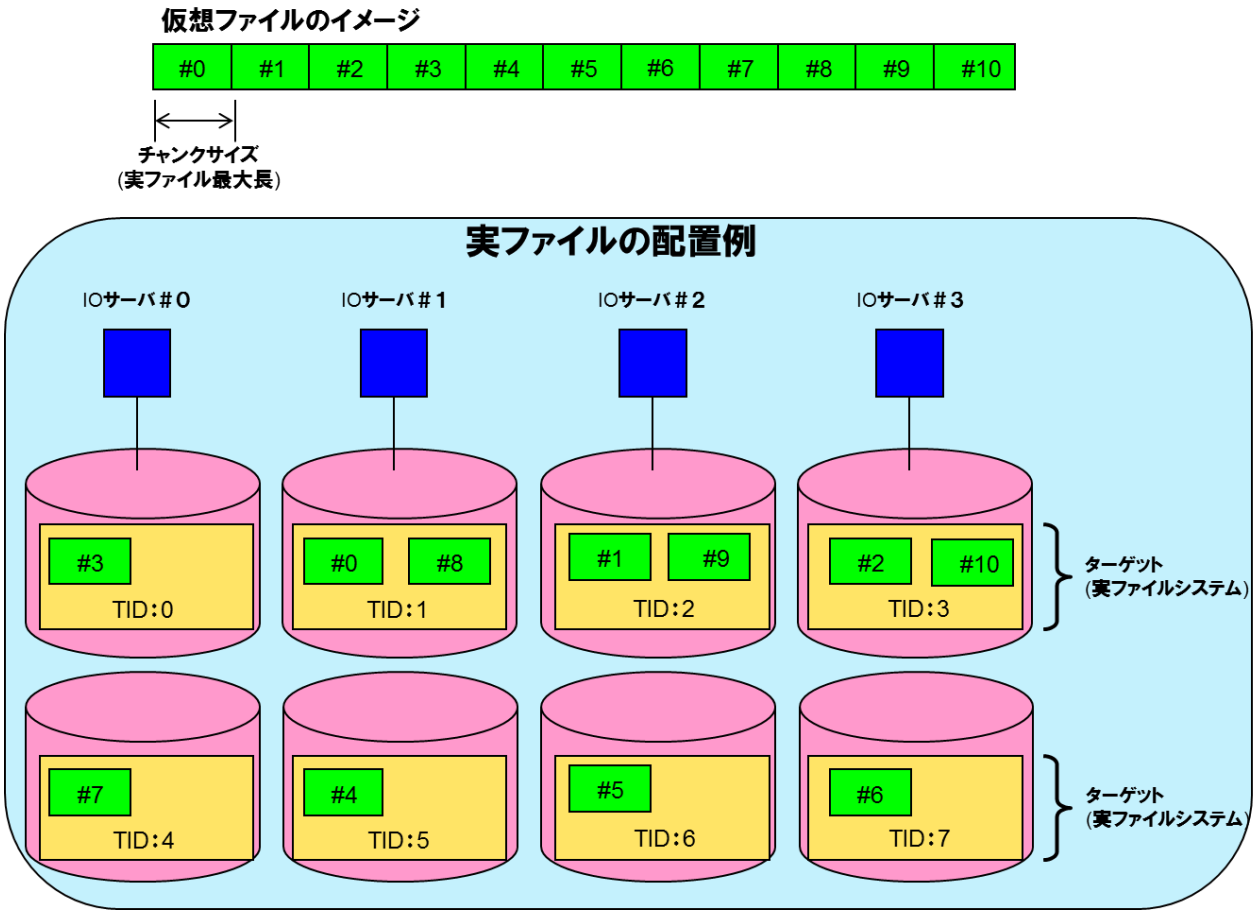

図 1-3　形式1の仮想ファイルと実ファイルの関係

1.1.2.2　ストライプフォーマット（形式２）

特定のノードから複数の IO サーバに同時にリクエストを発行することができるので、単体 I/O の処理を効率化したい時に有用です。たとえば、図 1-4 の例では、IO サーバが２台あり各 IO サーバ配下にターゲットが２つあるため、ストライプサイズの２倍または４倍の I/O サイズで read/write システムコールを呼び出した際に効果を期待できます。つまり、仮想ファイルの #0, #1 または #0, #1, #2, #3 に対してほぼ同時に read/write ができます。ただし、ScaTeFS が使用しているノード(クライアント)のネットワークインターフェースの持つ帯域に制限されることに注意してください。

なお、並列 I/O(後述)の場合は、I/O の発行の仕方によっては、ノード間で同一実ファイルの異なるオフセットを更新／参照することにより競合が発生することがあります。

図 1-4 のようにストライプサイズを、仮想ファイルを構成する基本単位とし、チャンクサイズはストライプサイズの倍数である必要があります。デフォルトのファイルフォーマットは形式 1 であるため、形式 2 を使用するためには後述の scatefs_premap により、明示的にストライプサイズとチャンクサイズを指定する必要があります。図 1-4 の例では、各 IO サーバ配下にそれぞれ 2 つの実ファイルシステム(ターゲット)を作成して 1 つの ScaTeFS を構成しています。図中の仮想ファイルは、チャンク番号#0～#20 で構成されており、仮想ファイルの先頭であるチャンク番号#0 は、TID= 3 に配置されています。以降は、これを起点とし、

TID=(3, 2, 1, 0, 3, 2, 1, 0・・・)

のように最初は#0 が配置されたターゲットと同列の各 IO サーバ配下のターゲットに配置され、IO サーバを一巡すると次のターゲットに配置されます。さらに、実ファイルのサイズがチャンクサイズに達すると、同一ターゲット内で新たに実ファイルを生成します。

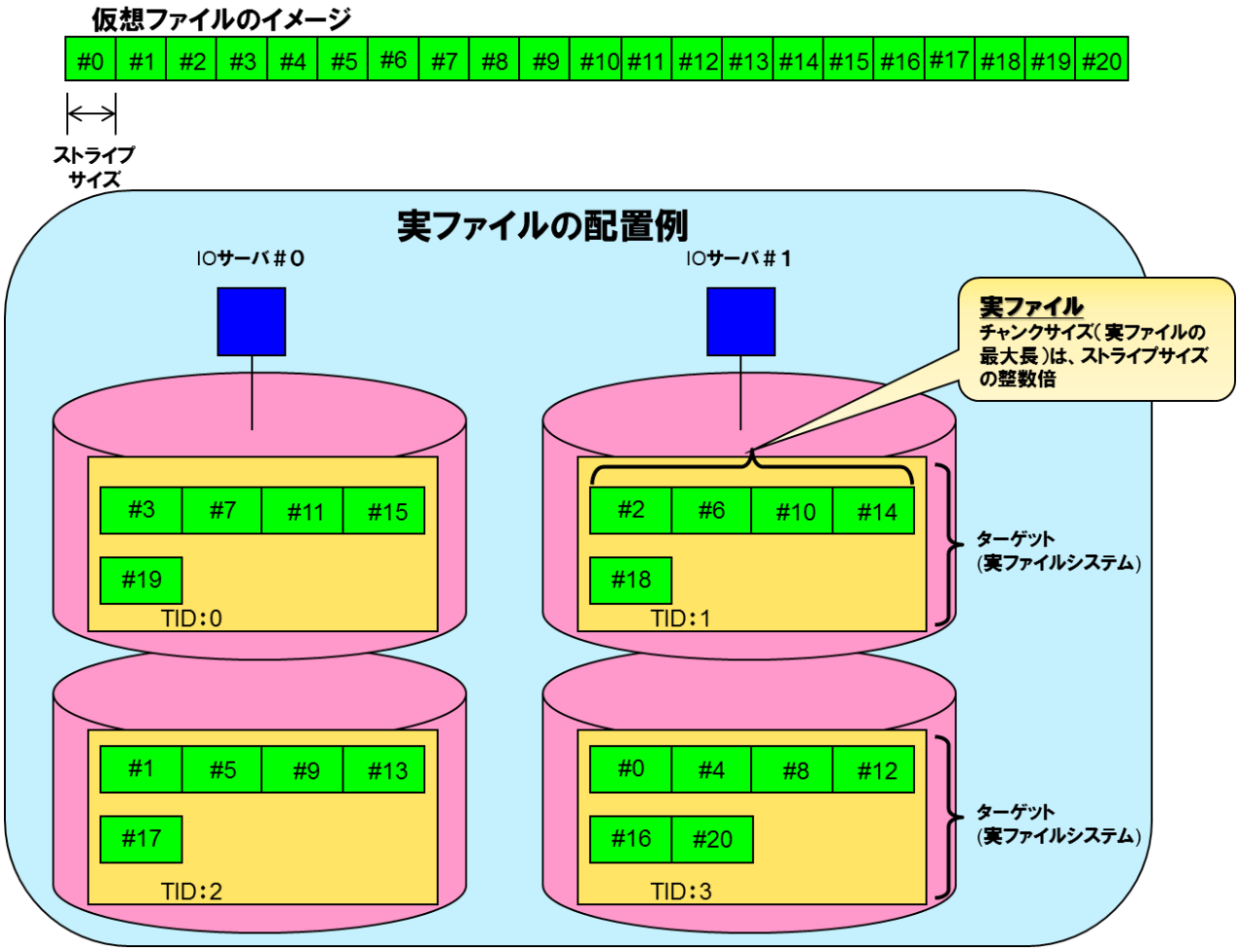

図 1-4　形式 2 の仮想ファイルと実ファイルの関係

1.2　ScaTeFS for AIの主な機能

- 並列I/Oを効率的に行うための機能
  - ファイルのプリマップ

    ファイルに write を行う前に予め IO サーバ上に実ファイルを作成しておく機能。
    多数の実ファイルで構成される巨大ファイルを write する際に、実ファイル作成のオーバーヘッドを軽減することができます。

  - ファイルフォーマットの設定

    プログラムが発行する I/O の仕方に適するファイルフォーマットを設定することで、効率的に並列 I/O が実行できます。

- 運用管理機能
  - 資源制限(QUOTA)

    ユーザ、グループ、ディレクトリに対して、ファイル数、ディスク使用量の制限を設定する機能。特定のユーザ、グループ、ディレクトリによるファイルシステム資源の浪費を防止することができます。

  - ファイルシステムの拡張

    既に使用している ScaTeFS に対して、IO サーバや IO ターゲットを追加することで、容量、帯域を拡張することができます。

  - フェアシェア

    ユーザ、クライアント毎に均等に IO を処理する機能。
    特定のユーザ、または特定のクライアントの処理負荷によるシステム全体のパフォーマンス低下を低減することができます。

  - リバランス

    既存ファイルのデータの配置を変更して、IO サーバ間、IO ターゲット間のストレージ使用量の不均衡を解消する機能。
    ファイルシステム拡張等で発生する、ストレージ使用量の不均衡によるアクセスの偏りを解消することができます。

- サブディレクトリマウント

  ScaTeFS のファイルシステムのうち、一部のディレクトリツリーだけを選択してクライアントからマウントする機能。

  一つのファイルシステムでありながら、用途に合わせて別々のディレクトリツリーをマウントして使用できるので、用途ごとにファイルシステムを作成する場合よりメンテナンスのコストが低くなります。

- 運用中の IO サーバのメンテナンス

  運用中に ScaTeFS IO サーバのパッケージをアップデートすることができます。

- ファイルシステム監視

  ファイルシステムの統計情報をリアルタイムに収集しモニタリングする機能。

  ファイルシステムの性能や使用量等の状況を GUI ベースでリアルタイムに確認できます。

- NFS や samba でのエクスポート

  NFS や samba を使用して、ScaTeFS クライアントがインストールされていない Linux や Windows のマシンで ScaTeFS が使用可能です。

第2章　　効率的に並列 I/O を行うための機能

本章においての並列I/Oとは、複数の計算ノードを使用して並列にデータを転送することにより、１つのファイルへの書き込み、読み込みを行うことを指しています。図 2-1は、並列I/Oの例です。

並列I/Oにより並列度に見合うI/O性能を得るためには、ScaTeFSを構成するIOサーバ数、IOターゲット数を考慮に入れた上で、並列I/Oの対象とする仮想ファイルのフォーマット(形式１／形式２)、チャンクサイズなどを決める必要があります。これは、IOサーバやストレージにおいて競合を発生させないようにするためです。

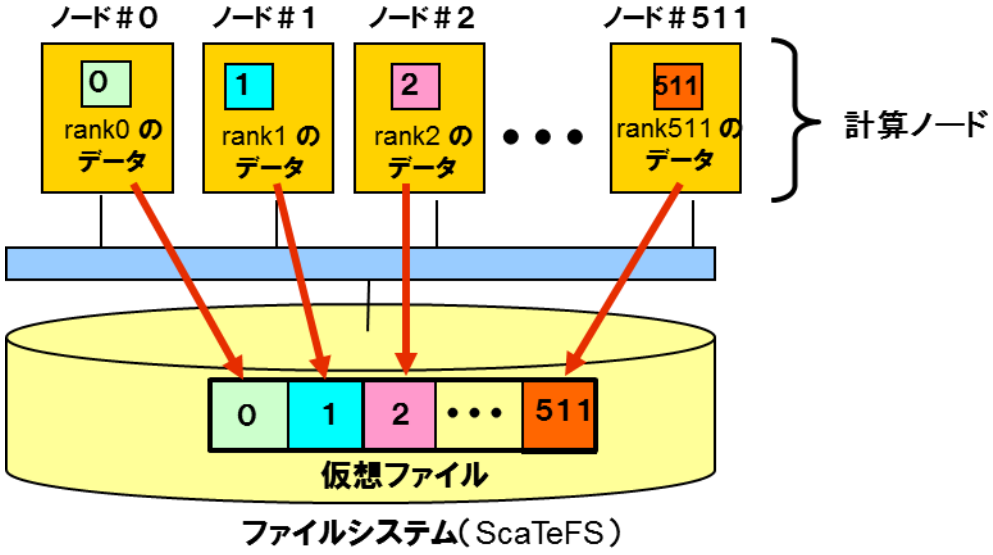

図 2-1　形式 1 を前提とした並列 I/O のイメージ

2.1　ファイルのプリマップ

図 2-1に示すように512ノードから一斉にwriteを行って、512個の実ファイルよりなる１つの仮想ファイルを作成するとします。この際、512個の実ファイルがほぼ同時に生成されることになり、仮想ファイルの管理情報の更新が若干オーバーヘッドとなる可能性が考えられます。これを軽減するために、writeに先立って予め必要数の実ファイルを生成するプリマップという機能があります。

後述のファイルフォーマットの指定で使用するscatefs_premapを実行する際に、ファイルサイズを指定することで、ファイルのプリマップができます。実行方法については、「2.2 ファイルフォーマットの設定と表示」を参照してください。

2.2　ファイルフォーマットの設定と表示

ファイルフォーマットの設定を行うには、対象がファイルである場合scatefs_premapを、対象がディレクトリである場合scatefs_setdirattrを使用します。ファイルフォーマットを確認するには、scatefs_getfinfoを使用します。以下に例を記載します。

2.2.1 ノンストライプフォーマット (形式 1) の設定

- ファイル

  scatefs_premapに -cオプションとファイルサイズを指定することで、形式1のファイルを作成します。例では、チャンクサイズ2G、ファイルのサイズ4Gでプリマップを行っています。

  (例)

```
$ scatefs_premap -c 2G 4G /mnt/scatefs/file000
```

  ファイルフォーマットのみを指定したファイルを作成する場合は、ファイルサイズを0に設定します。

  (例)

```
$ scatefs_premap -c 2G 0 /mnt/scatefs/file001
```

- ディレクトリ

  scatefs_setdirattrに-cオプションを指定することで、既存ディレクトリを形式1のフォーマットに変更します。変更完了後、ディレクトリ配下に新規作成されるファイル、ディレクトリに変更後の値が反映されます。既存ファイル、ディレクトリには反映されません。例では、チャンクサイズ4Gに設定を変更しています。

  (例)

```
$ scatefs_setdirattr -c 4G /mnt/scatefs/dir000
```

2.2.2 ストライプフォーマット (形式 2) の設定

- ファイル

  scatefs_premapに -s オプションを指定することで、形式2のファイルを作成します。 例では、ストライプサイズ4M、チャンクサイズ1G、ファイルサイズ1Gでプリマップを行っています。なお、既存ファイルを指定した場合、ファイルサイズが0の場合にのみプリマップを行うことが可能です。

(例)

```
$ scatefs_premap -s 4M -c 1G 1G /mnt/scatefs/file002
```

- ディレクトリ

  scatefs_setdirattrに -s オプションを指定することで、形式2にフォーマットを変更します。変更完了後、ディレクトリ配下に新規作成されるファイル、ディレクトリに変更後の値が反映されます。既存ファイル、ディレクトリには反映されません。例では、既存ディレクトリ性を、ストライプサイズ4M、チャンクサイズ1Gに変更しています。

  (例)

```
$ scatefs_setdirattr -s 4M -c 1G /mnt/scatefs/dir001
```

2.2.3 フォーマットの表示

scatefs_getfinfoでファイル/ディレクトリのフォーマット情報を表示することが可能です。

- ファイル

  (例)形式1

```
$ scatefs_getfinfo /mnt/scatefs/file001
format      : non stripe format
iot count   :                6
stripesize  :        268435456
chunksize   :        268435456
filesize    :       1610612736

format       ---   ファイルフォーマット
iot count    ---   使用しているIOターゲットの数
stripesize   ---   ストライプサイズ
chunksize    ---   チャンクサイズ
filesize     ---   ファイルサイズ
```

※形式1の場合、ストライプサイズ、チャンクサイズは同じ値になります。

(例)形式2

```
$ scatefs_getfinfo /mnt/scatefs/file002
format     :     stripe format
iot count  :                6
```

```
stripesize :          33554432
chunksize  :          67108864
filesize   :         268435456
```

-v オプションを指定することで、ファイルオフセットごとの実ファイル分布を表示することができます。形式１、形式２のファイルを例に表示情報を示します。

- ファイルの詳細表示

  (例)形式1

```
$ scatefs_getfinfo -hv /mnt/scatefs/file001
format     : non stripe format
iot count  :                 6
stripesize :            256.0M
chunksize  :            256.0M
filesize   :              1.5G

       offset                            no     ios     iot
-------------------------------------------------------------
              0 ...         268435455      0       0       0
      268435456 ...         536870911      1       1       3
      536870912 ...         805306367      2       0       1
      805306368 ...        1073741823      3       1       4
     1073741824 ...        1342177279      4       0       2
     1342177280 ...        1610612735      5       1       5

offset  ---  仮想ファイルのオフセットを示します。
no      ---  実ファイルのインデックスを示します。
ios     ---  実ファイルが格納されているIOサーバIDを示します。
iot     ---  実ファイルが格納されているIOターゲットIDを示します。
```

形式1の場合、オフセットと実ファイルのインデックスは一致します。以下に実ファイルの配置イメージを記載します。

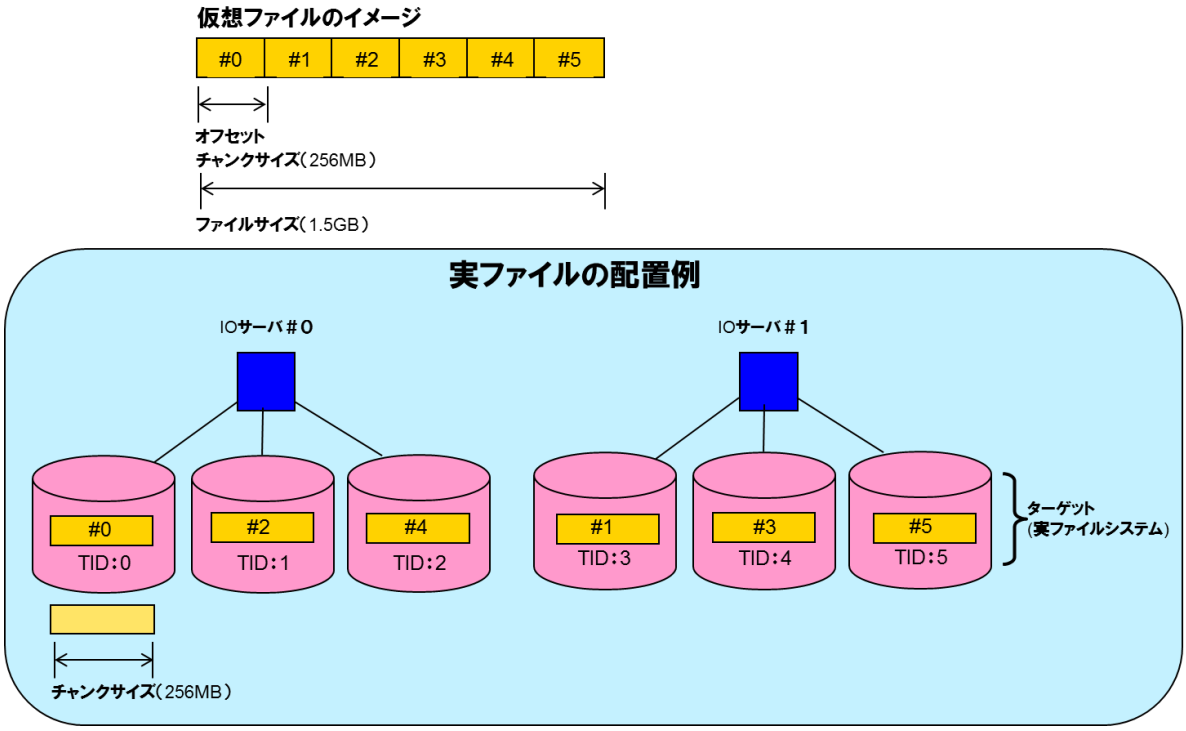

図 2-2　形式 1 における実ファイルの配置例

(例)形式2

```
$ scatefs_getfinfo -hv /mnt/scatefs/file002
format      :     stripe format
iot count   :                6
stripesize  :            32.0M
chunksize   :            64.0M
filesize    :           256.0M

        offset                          no      ios      iot
-------------------------------------------------------------
             0 ...         33554431      0        0        0
      33554432 ...         67108863      1        1        3
      67108864 ...        100663295      2        0        1
     100663296 ...        134217727      3        1        4
     134217728 ...        167772159      4        0        2
     167772160 ...        201326591      5        1        5
     201326592 ...        234881023      0        0        0
     234881024 ...        268435455      1        1        3
```

形式2の場合、ストライプサイズ単位で区切ったオフセットに対応する実ファイルのインデックスを表示します。以下に実ファイルの配置イメージを記載します。

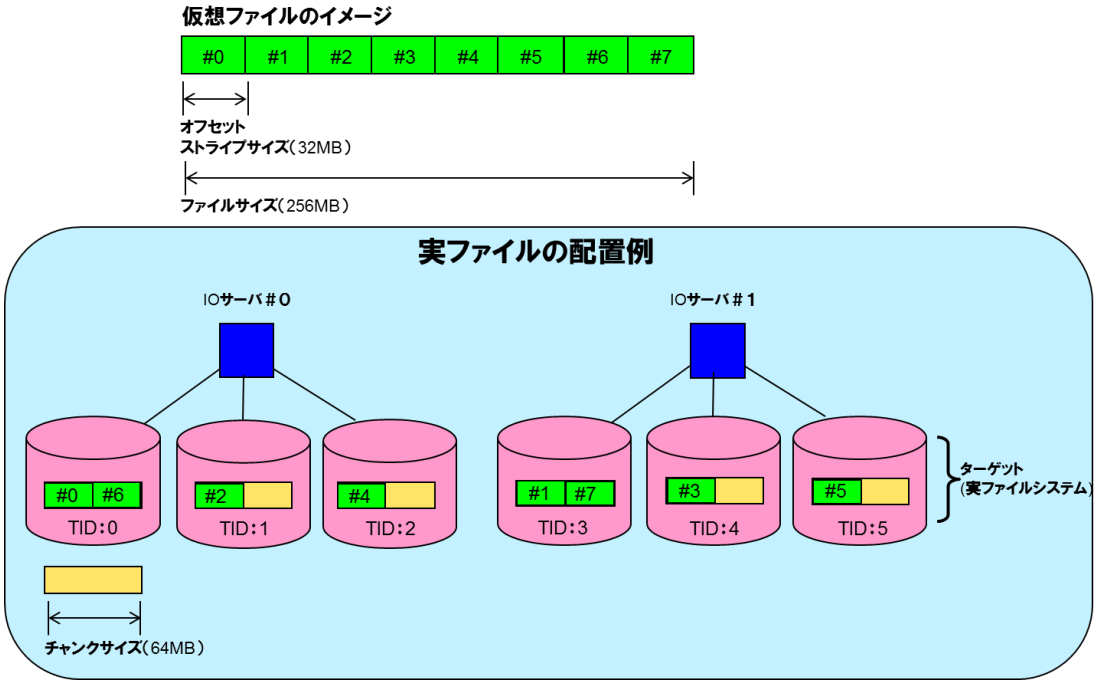

図 2-3　形式２における実ファイルの配置例

- ディレクトリ

  (例)形式1

```
$ scatefs_getfinfo -h /mnt/scatefs/dir001
format     : non stripe format
stripesize :            512.0M
chunksize  :            512.0M
```

※形式1の場合、ストライプサイズ、チャンクサイズは同じ値になります。

(例)形式2

```
$ scatefs_getfinfo -h /mnt/scatefs/dir002
format     :     stripe format
stripesize :             32.0M
chunksize  :              1.0G
```

※ディレクトリを対象とした詳細表示オプション(-v)は無効となります。

第3章　　運用/操作方法

3.1　マウント/アンマウント方法

3.1.1 マウント方法

ScaTeFSクライアントでmountコマンドを使ってファイルシステムをマウントします。

以下に、ルートIOサーバ"iosv00"のファイルシステム"scatefs00"を/mnt/scatefsにマウントする例を示します。

```
# mount -t scatefs -o rsize=4194304,wsize=4194304 iosv00:scatefs00 /mnt/scatefs
```

マウントオプションのrsizeとwsizeは、クライアントとIOサーバの間でファイルのデータを入出力する際の転送サイズを表します。ともに既定値は1MBですが、2MB、または4MBとした方が性能は向上します。

マウントオプションの詳細については、「4.1.1.1 scatefs」をご参照ください。

ファイルシステムに関する情報を/etc/fstabに記述し、Linuxマシンの起動時にファイルシステムを自動的にマウントする場合、マウントオプションに_netdevを記述してください。本オプションを記述しない場合、RHEL 8ではLinuxマシンの起動時にファイルシステムのマウントに失敗し、緊急モードのログインプロンプトがコンソールに表示されます。この場合、マウントオプションに_netdevを追加し再起動してください。以下に/etc/fstabの記述例を示します。

```
iosv00:scatefs00 /mnt/scatefs scatefs _netdev,rsize=4194304,wsize=4194304 0 0
```

マウントオプションとしてSELinuxのコンテキストが指定されなかった場合、既定値としてcontext="system_u:object_r:nfs_t:s0"が使用されます。他のコンテキストを使用したい場合は、マウントオプションでコンテキストを指定してください。

3.1.2 アンマウント方法

umountコマンドを使ってファイルシステムをアンマウントします。

以下に、/mnt/scatefsにマウントされているファイルシステムをアンマウントする例を示します。

```
# umount /mnt/scatefs
```

IOサーバとの通信が不通となった場合、-fオプションを使用することによりファイルシステムを強制的にアンマウントすることができます。以下に、/mnt/scatefsにマウントされているファイルシステムを強制的にアンマウントする例を示します。

```
# umount -f /mnt/scatefs
```

3.2　運用管理

3.2.1 IO サーバデーモンの操作

ファイルシステムのメンテナンス等において、IOサーバデーモンの停止が必要になる場合があります。

IOサーバデーモンの停止および起動はCLUSTERPROのコマンドを使用しますので、ペアのIOサーバのどちらかで以下のコマンドを実行します。

IOサーバデーモンの停止

```
# clprsc -t exec1
# clprsc -t exec2
```

IOサーバデーモンの起動

```
# clprsc -s exec1
# clprsc -s exec2
```

3.2.2 資源制限（QUOTA)

ファイルシステムごとに下記のQUOTA機能を提供します。

表 3-1　QUOTA 機能

制限対象	制限リソース	制限種別	
		ソフトリミット	ハードリミット
ユーザ	ファイル数	○	○
	ディスク容量	○	○
グループ	ファイル数	○	○
	ディスク容量	○	○
ディレクトリ	ファイル数	○	○
	ディスク容量	○	○

QUOTAは、ユーザやグループ、ディレクトリごとに設定が可能となります。制御リソースはファイル数とディスク容量であり、それぞれハードリミットとソフトリミットによる制限が可能です。

ハードリミットとは、その値に達した場合はそれ以上アロケートすることができない制限値です。ハードリミットに達した場合、書き込み要求に対しEDQUOTを返却します。

ソフトリミットとは、一時的に超えることができる制限値です。この値を超えた状態で設定され

た猶予期間を経過した場合、ハードリミットに達した場合と同様に扱います。猶予期間はデフォルトでは７日間ですが、ファイルシステムごとに1秒から(232-1) 秒の範囲で設定可能です。設定方法については、「4.2.1.7 scatefs_edquota」を参照してください。

ハードリミットに到達またはソフトリミットに到達後に猶予期間が経過し、書き込みができない状態となった場合、ハードリミット、ソフトリミットを下回るまでファイルの削除を行うか、scatefs_edquotaコマンドにて、ハードリミット、ソフトリミットの上限値を変更することで解消されます。

ディスク容量の計算には、各IOターゲットに配置する実ファイルのファイルサイズを使用します。このため、実ファイルのホールサイズも使用量として計算されます。

QUOTA機能は、IOサーバ構築後は有効になっています。QUOTA機能が無効の場合には、ファイル数およびディスク容量の使用量をカウントしません。

ディレクトリQUOTA

ユーザ/グループQUOTAについてはLinux標準の機能であるためここでは説明を省略し、ディレクトリクォータについて説明します。ディレクトリQUOTAは、ユーザ/グループ単位のQUOTA制限とは別にディレクトリ単位でQUOTA制限を行う機能です。ディレクトリQUOTAとユーザ/グループQUOTAの使用量管理は同時に機能します。ディレクトリQUOTAを使用することで、より柔軟な資源管理が可能となります。

図 3-1は、ファイルシステム（FS1）のユーザ/グループのQUOTA制限とは別に、proj1/proj2ディレクトリそれぞれでQUOTA制限を行うイメージです。

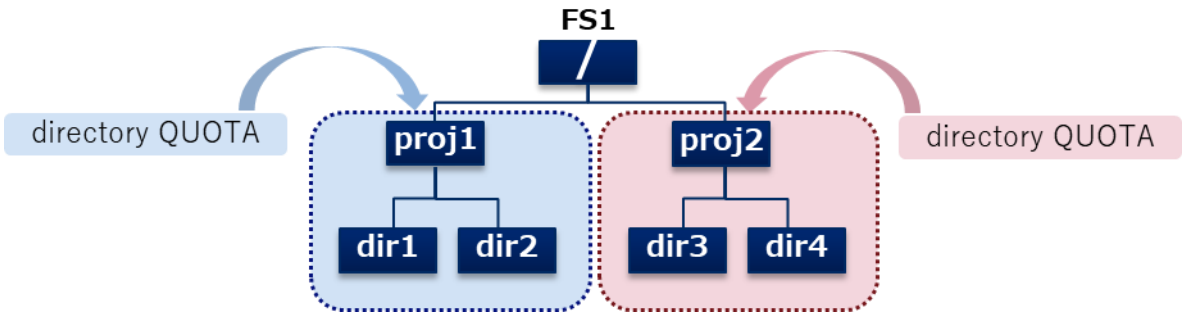

図 3-1　ディレクトリクォータイメージ図

ディレクトリクォータの運用は以下の手順で行います。

(1) QUOTA制御ディレクトリの作成

ディレクトリQUOTAを使用するためには、まず起点となるディレクトリを作成します。

この起点となるディレクトリのことをQUOTA制御ディレクトリと呼びます。図 3-1では、proj1/proj2がQUOTA制御ディレクトリに該当します。
QUOTA制御ディレクトリの作成には、scatefs_mkqdirコマンドを使用します。
scatefs_mkqdirコマンドについては、「4.2.1.16 scatefs_mkqdir」を参照してください。

(2) QUOTA情報の編集

ディレクトリQUOTA情報の編集には、scatefs_edquotaコマンドを使用します。
scatefs_edquotaコマンドについては、「4.2.1.7 scatefs_edquota」を参照してください。

(3) QUOTA設定の確認

QUOTA設定の確認には、scatefs_quota/scatefs_repquotaコマンドを使用します。
scatefs_quotaコマンドについては「4.2.1.2 scatefs_quota」を、scatefs_repquotaコマンドについては「4.2.1.20 scatefs_repquota」を参照してください。
QUOTA設定の確認には、scatefs_quota/scatefs_repquotaコマンドを使用します。
scatefs_quotaコマンドについては「4.2.1.2 scatefs_quota」を、scatefs_repquotaコマンドについては「4.2.1.20 scatefs_repquota」を参照してください。

- 使用（Used）　　：ディレクトリQUOTA内の使用量
- 使用可（Available）：ハードリミットまでの残り量（※）

※　ハードリミットよりも実際のファイルシステムの空き容量が少ない場合は、ファイルシステムの空き容量が使用可能量として表示されます。

（例1）

```
# mount -t scatefs HOST:FS1 /mnt/scatefs
# df /mnt/scatefs/proj1
Filesystem   1K-blocks   Used  Available  Use%  Mounted on
HOST:FS1        200704      0     200704    0%  /mnt/scatefs
```

QUOTA制御ディレクトリをサブディレクトリマウントした場合、dfコマンドの結果に当該ディレクトリのQUOTA情報が表示されます。サブディレクトリマウントについては「3.2.12 サブディレクトリマウント」を参照してください。

（例2）

```
mount -t scatefs HOST:FS1/proj1 /mnt/subdir
# df
```

```
Filesystem      1K-blocks  Used  Available Use%  Mounted on
 :
HOST:FS1/proj1     200704     0     200704   0%  /mnt/subdir
```

(4) QUOTA制御ディレクトリの削除

QUOTA制御ディレクトリの削除には、scatefs_rmqdirコマンドを使用します。

scatefs_rmqdirコマンドについては、「4.2.1.21 scatefs_rmqdir」を参照してください。

3.2.2.1　コマンド

QUOTA の設定は、以下の 2 つの方法があります。

- いずれかの IO サーバにログインし、ScaTeFS 用 QUOTA コマンドを実行する方法
- 事前に登録された Linux クライアントマシンからリモート CLI（scatefs_rcli）により ScaTeFS 用 QUOTA コマンドを実行する方法

関連するコマンドは各IOサーバデーモンが起動中かつQUOTA機能が有効な場合に限り実行可能となります。

以下、各コマンドの概要と代表的な実行イメージを記載します。詳細な利用方法については「4.1.1.3 scatefs_rcli」を参照してください。

コマンド	概要
scatefs_quotacheck	QUOTA 情報の再計算と quota ファイルの修復を行う
scatefs_edquota	ユーザ、グループおよびディレクトリの QUOTA を編集する
scatefs_quota	ディスクの使用状況と使用限度を表示する
scatefs_repquota	QUOTA 情報一覧を表示する
scatefs_mkqdir	QUOTA 制御ディレクトリを作成する
scatefs_rmqdir	QUOTA 制御ディレクトリを削除する

3.2.2.1.1　scatefs_quotacheck コマンド

scatefs_quotacheck コマンドでは、各ファイルシステムの QUOTA 情報の整合性を検証し、不具合があった場合に修正を行う機能を提供します。本コマンドは、IO サーバ上でのみ実行可能です。運用を停止してから scatefs_quotacheck コマンドを実行してください。

(例) IO サーバ上で、ファイルシステム scatefs00 のユーザやグループ、ディレクトリに対し、QUOTA 情報の整合性を検証する

```
# su - fsadmin
$ scatefs_quotacheck scatefs00
```

(例) IO サーバ上で、すべてのファイルシステム、ユーザやグループ、ディレクトリに対し、QUOTA 情報の整合性を検証する

```
# su - fsadmin
$ scatefs_quotacheck -a
```

(例) IO サーバ上で、ファイルシステム scatefs00 のグループに対し、設定されたハードリミット、ソフトリミットをクリアし、使用量情報を検証する

```
# su - fsadmin
$ scatefs_quotacheck -c -g scatefs00
```

3.2.2.1.2　scatefs_edquota コマンド

scatefs_edquota コマンドでは、ユーザやグループ、ディレクトリに QUOTA 設定を行う機能を提供します。本コマンドは、root ユーザのみが実行でき、リモート CLI コマンド(3.2.8 リモート CLI)経由にて使用することが可能です。

(例) IO サーバ上で、ファイルシステム scatefs00 のユーザ(UID 500) に対し、QUOTA を編集する(環境変数 EDITOR で設定したエディタを開きます)

```
# su - fsadmin
$ export EDITOR=/bin/vi
$ scatefs_edquota -u 500 scatefs00
```

(例) IO サーバ上で、ファイルシステム scatefs00 のユーザ(UID 500)に対し、ディスク容量のソフトリミットを 1000KB、ハードリミットを 2000KB に設定する

```
# su - fsadmin
$ scatefs_edquota -u 500 -b 1000:2000 scatefs00
```

(例) IO サーバ上で、ファイルシステム scatefs00 のディレクトリ"/dquota"に対し、ファイル数のソフトリミットを 5000 ファイル、ハードリミットを 10000 ファイルに設定する

```
# su - fsadmin
$ scatefs_edquota -d /dquota -i 5000:10000 scatefs00
```

(例) Linux クライアント上で、IO サーバ server00 のファイルシステム scatefs00 のグループ (GID 500)に対し、ハードリミット、ソフトリミットを設定する

```
$# scatefs_rcli server00 edquota -g 500 -b 1000:2000 -i 5000:10000 scatefs00
```

また、edquota コマンドでは、ソフトリミット超過にともない設定される猶予期間に関して、以下の設定を行う機能を提供します。

- 各ユーザやグループ、ディレクトリの残り猶予期間(grace time)
- 各ファイルシステムに属するすべてのユーザやグループ、ディレクトリがソフトリミット超過時に初期設定される猶予期間(period time)

(例) IO サーバ上でファイルシステム scatefs00 のユーザ(UID 500) に対し、残り猶予期間を編集する(環境変数 EDITOR で設定したエディタを開きます)

```
# su - fsadmin
$ export EDITOR=/bin/vi

Times to enforce softlimit for (user 0):
Time units may be: days, hours, minutes, or seconds
              Filesystem   block grace   inode grace
               scatefs00   3550seconds         unset
```

(例) IO サーバ上で、ファイルシステム scatefs00 のユーザ(UID 500)に対し、ディスク容量の残り猶予期間を 7 日(604800 秒)に設定する

```
$ scatefs_edquota -T -u 500 -b 604800 scatefs00
```

(例) Linux クライアント上で、IO サーバ server00 のファイルシステム scatefs00 のユーザ(UID 500) に対し、ファイル数の残り猶予期間を 1 時間(3600 秒)に設定する

```
$ scatefs_rcli server00 edquota -T -u 500 -i 3600 scatefs00
```

(例) IO サーバ上で、ファイルシステム scatefs00 のユーザに初期設定される猶予期間を編集する(環境変数 EDITOR で設定したエディタを開きます)

```
$ export EDITOR=/bin/vi
$ scatefs_edquota -t u scatefs00

Grace period before enforcing soft limits for users:
Time units may be: days, hours, minutes, or seconds
              Filesystem   block grace period    inode grace period
               scatefs00                7days           3600seconds
```

(例) IO サーバ上でファイルシステム scatefs00 のグループに初期設定されるディスク容量の猶予期間を 1 日(86400 秒)に設定する

```
$ scatefs_edquota -t g -b 86400 scatefs00
```

(例) Linux クライアント上で IO サーバ server00 のファイルシステム scatefs00 のディレクトリに初期設定されるファイル数の猶予期間を 10000 秒に設定する

```
$ scatefs_rcli server00 edquota -t d -i 10000 scatefs00
```

3.2.2.1.3　scatefs_quota コマンド

scatefs_ quota コマンドは、ファイルシステムの QUOTA 情報を表示する機能を提供します。本コマンドは、管理者および一般ユーザが実行でき、リモート CLI コマンド(3.2.8 リモート CLI)経由にて使用することが可能です。一般ユーザは、リモート CLI コマンドを使用して、自身または所属しているグループ、およびディレクトリの QUOTA 情報を確認することが可能です。

正確な情報の出力が必要な場合には、事前に scatefs_quotacheck コマンド(4.2.1.17 scatefs_quotacheck)を実行してください。

(例) IO サーバ上で、ファイルシステム scatefs00 のユーザ (UID 500)の QUOTA 情報を表示する

```
# su - fsadmin
$ scatefs_quota -s -u 500 scatefs00:sg000

ScaTeFS quotas for user  (uid 500)
        Filesystem:sgname         blocks          quota          limit  grace
files     quota     limit  grace
 ---------------------------------------------------------------------------
---------------------------------
           scatefs00:ROOT              0         488.2K          9.5M       -
0         10.0K     20.0K      -
```

(例) IO サーバ上で、ファイルシステム scatefs00 のディレクトリ“qdir”(DIRID 1000)の QUOTA 情報を出力する

```
# su - fsadmin
$ scatefs_quota -s -d qdir scatefs00

ScaTeFS quotas for directory /qdir (dirid 1000)
        Filesystem:sgname         blocks          quota          limit  grace
files     quota     limit  grace
 ---------------------------------------------------------------------------
---------------------------------
           scatefs00:ROOT              0         488.2K          9.5M       -
0         10.0K     20.0K      -
```

(例) Linux クライアント上で、IO サーバ server00 のファイルシステム scatefs00 のグループ”group500”(GID 500)を対象とした QUOTA 情報を出力する

```
$ scatefs_rcli server00 quota -s -g group500 scatefs00
ScaTeFS quotas for group group500 (gid 500)
        Filesystem:sgname        blocks         quota         limit  grace
files     quota     limit  grace
 ---------------------------------------------------------------------------
---------------------------------
           scatefs00:ROOT              0         7.63G         9.54G      -
```

```
0          10.0K      1.00M       -
```

3.2.2.1.4　scatefs_repquota コマンド

scatefs_repquota コマンドは、ファイルシステムの QUOTA 情報一覧を表示する機能を提供します。本コマンドは管理者のみが実行でき、リモート CLI コマンド(3.2.8 リモート CLI)経由にて使用することが可能です。表示される QUOTA 情報は、未使用のユーザやグループ、ディレクトリの QUOTA 情報は出力されません。

正確な情報の出力が必要な場合には、事前に scatefs_quotacheck コマンド(4.2.1.17 scatefs_quotacheck)を実行してください。

(例) IO サーバ上で、ファイルシステム scatefs00 のユーザの QUOTA 情報一覧を出力する

```
# su - fsadmin
$ scatefs_repquota -u scatefs00

*** Report for user quotas on scatefs00:ROOT
Block grace time: 7days; Inode grace time: 7days
            Block limits                                        File limits
   user(id)            used            soft            hard   grace        used
soft      hard  grace
 ----------------------------------------------------------------------------
--------------------
           0               0           32768           65536       -           0
10000      10000       -
         512               0           32768           65536       -           0
20000      30000       -
        1024               0           32768           65536       -           0
50000      60000       -
        2048          225416          524288         1048576       -         729
512         1024   6days
```

(例) IO サーバ上で、ファイルシステム scatefs00 のディレクトリの QUOTA 情報一覧を出力する

```
# su - fsadmin
```

```
$ scatefs_repquota -d scatefs00

*** Report for directory quotas on scatefs00:ROOT
Block grace time: 7days; Inode grace time: 7days
                              Block                                    limits
File limits
   directory(name)     used           soft           hard    grace        used
soft      hard   grace
 ------------------------------------------------------------------------------
------------------------
   qdir00               32768        2097152        4194304        -         750
500       1000    6days
   qdir01               65536        2097152        4194304        -         256
500       1000       -
   qdir03              524288        2097152        4194304        -         300
500          0       -
```

(例) Linux クライアント上で、IO サーバ server00 のファイルシステム scatefs00 のグループを対象とした QUOTA 情報一覧を表示する

```
# scatefs_rcli server00 repquota -g scatefs00
（出力イメージ省略）
```

また、scatefs_repquota コマンドでは、設定されているハードリミット、ソフトリミットを再設定が可能な形式でバックアップする機能を提供します。バックアップは、標準出力表示およびファイル作成にて行います。バックアップ機能は IO サーバ上でのみ実行可能です。

(例) IO サーバ上で、ファイルシステム scatefs01 のユーザのバックアップ内容を一覧で出力した後、同情報のバックアップを出力する

```
# su - fsadmin
$ scatefs_repquota -u -b scatefs01
/opt/scatefs/bin/scatefs_edquota  -t  u       -b  604800             -i  604800
scatefs01:SG1 || echo "error: user grace scatefs01"
/opt/scatefs/bin/scatefs_edquota   -u   1024   -b   102400:204800   -i   128:256
scatefs01:SG1 || echo "error: uid 1024  scatefs01"
/opt/scatefs/bin/scatefs_edquota   -u   2048   -b   102400:204800   -i   128:256
scatefs01:SG1 || echo "error: uid 2048  scatefs01"
```

```
/opt/scatefs/bin/scatefs_edquota  -u  3072  -b  102400:204800  -i  128:256
scatefs01:SG1 || echo "error: uid 3072  scatefs01"
$ ls -l
-rw-rw-r--    1    root    fsadmin    630        9  月       18    16:58    2014
scatefs_quota.fsid1.sgid1.user
```

(例) IO サーバ上で、ファイルシステム scatefs01 のユーザのソフトリミット、ハードリミットをバックアップファイルからリストアする

```
# su - fsadmin
$ ls -l
-rw-rw-r--    1    root    fsadmin    630        9  月       18    16:58    2014
scatefs_quota.fsid1.sgid1.user
$ sh ./scatefs_quota.fsid1.sgid1.user
```

3.2.2.1.5　scatefs_mkqdir コマンド

scatefs_mkqdir コマンドは、ファイルシステムの QUOTA 設定が可能なディレクトリを作成する機能を提供します。本コマンドは、IO サーバ上でのみ実行可能です。QUOTA 情報は作成したディレクトリ毎に管理し、ディレクトリおよび配下の使用量のカウントと、ハードリミット、ソフトリミット、残り猶予時間の設定に対応します。このコマンドで作成したディレクトリを削除する場合は、scatefs_rmqdir コマンド(4.2.1.21 scatefs_rmqdir)を使用する必要があります。

(例) IO サーバ上で、ファイルシステム scatefs00 のルートディレクトリ配下に QUOTA 制御ディレクトリ"dquota00"を作成する

```
# su - fsadmin
$ scatefs_mkqdir scatefs00 /dquota00
```

(例) IO サーバ上で、ファイルシステム ID 1 のディレクトリ"work"配下に QUOTA の設定を行うディレクトリ"dquota01"を作成する

```
# su - fsadmin
$ scatefs_mkqdir 1 /work/dquota01
```

3.2.2.1.6　scatefs_rmqdir コマンド

scatefs_rmqdir コマンドでは、ファイルシステムの QUOTA 設定が可能なディレクトリを削除する機能を提供します。本コマンドは、IO サーバ上でのみ実行可能です。

(例) IO サーバ上で、ファイルシステム scatefs00 の QUOTA 制御ディレクトリ”dquota00”を削除する

```
# su - fsadmin
$ scatefs_rmqdir scatefs00 /dquota00
```

(例) IO サーバ上で、ファイルシステム ID 1 の QUOTA 制御ディレクトリ”/dquota /dquota01”を削除する

```
# su - fsadmin
$ scatefs_rmqdir 1 /work/dquota01
```

3.2.3 レコードロック強制解除

ScaTeFSではPOSIX.1で定義されている標準的なレコードロックを行う機構を提供しています。通常は、特定の計算ノードが資源を排他利用する場合にレコードロックを行い、利用終了と同時にレコードロックを解除します。しかし、レコードロック中の計算ノードに障害が発生した場合、運用によっては当該ノードからレコードロックの解除が長期にわたり実施できない場合があります。このため、特定の計算ノードのレコードロック情報をすべて強制解除する機能をscatefs_lockreleaseコマンドとして提供します。

(例) クライアントID「XX.XX.XX.XX」に該当するレコードロック情報を強制解除します。

```
# su - fsadmin
$ scatefs_lockrelease -r @XX.XX.XX.XX
```

コマンドの詳細については、「4.2.1.12 scatefs_lockrelease」 をご参照ください。

3.2.4 ファイルシステムの拡張

IOサーバやIOターゲットを追加することで、ファイルシステムを拡張することが可能です。拡張時はファイルシステムの運用を停止する必要があります。ファイルシステムを拡張する場合は、サ

ポート部門までお問い合わせください。

3.2.5 フェアシェア

IOサーバでは、フェアシェアIOスケジューリング機能を提供します。この機能は、従来のジョブスケジューリングではなく、IOサーバ上のIOリソースのフェアシェアを実現します。

この機能を利用することによって効率的な負荷分散が行われ、特定のユーザ、または特定の計算ノードの処理負荷によるシステム全体のパフォーマンス低下を低減します。

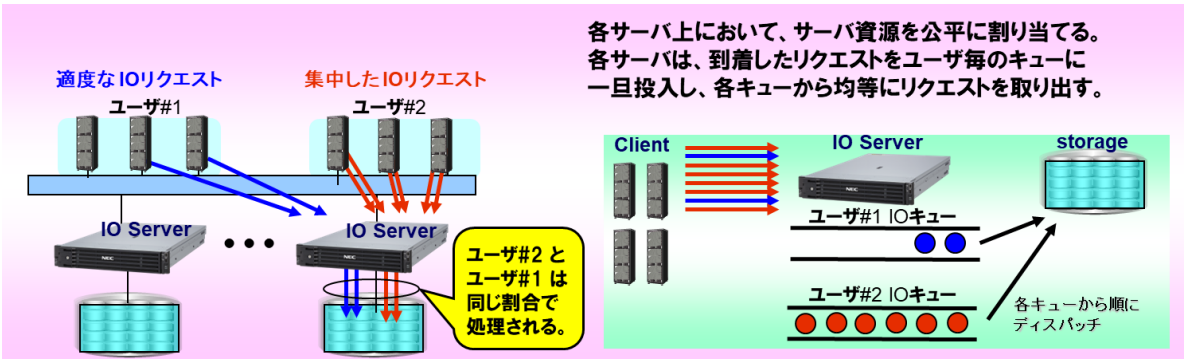

図 3-2　フェアシェアのイメージ図

IOサーバのコンフィグファイルに所定の情報を登録することで利用が可能となります。ただし、運用中の動的な変更はサポートしていません。

3.2.5.1　ポリシーの種類

IO スケジューリング機能は以下の 3 つのポリシーから選択可能とします。

- フェアシェアなし(デフォルト)
- ユーザ(UID)ごとの均等化
- ClientID(クライアント毎にユニークな ID)ごとの均等化

ポリシーは全 IO サーバで同一のものとする必要があり、ポリシーを変更後は IO サーバの再起動が必要となります。

3.2.5.2　ポリシーの変更方法

ポリシーを変更する際の手順は下記になります。

(1) コンフィグファイル scatefssrv.conf の FAIRPOLICY を変更します。

設定可能な値は以下となります。

0：フェアシェアなし(デフォルト)

1：UID ごとの均等化

2：ClientID ごとの均等化

(2) scatefs_admin コマンドを使用して、修正した scatefssrv.conf を全 IO サーバに配布します。

(3) 各 IO サーバを再起動します。

3.2.6 容量管理

ScaTeFSでは、容量がしきい値を超えるIOターゲットへの書き込み要求を受けた場合、容量に十分空きがある他のIOターゲットを選定し利用することでIOを継続します。しかし、本機能は通常処理コストが高いため動作しないことが望ましい状態と言えます。このような状態となった場合、システム負荷が低い状態において、ファイルシステムのリバランス実施の検討が必要です。

3.2.7 リバランス

ファイルシステムを拡張すると、既存ファイルと新規ファイルへのアクセスに偏りが発生することがあります。ScaTeFSでは、この偏りを解消し、全IOサーバ分の帯域を活用する、リバランス機能を提供します。リバランス機能は、ファイルシステムの運用を停止することなく実施できます。

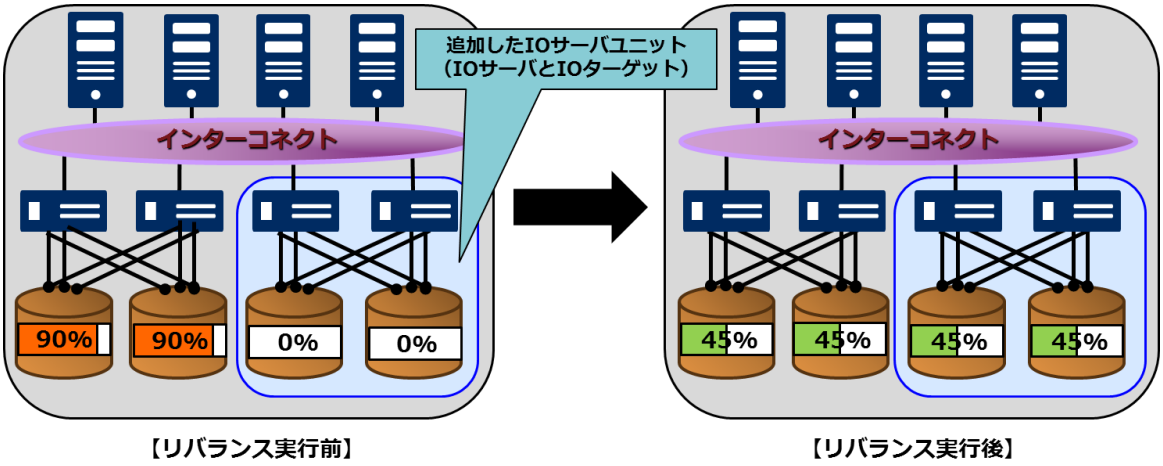

図 3-3　IO サーバユニットを追加した時のリバランスの実行例

リバランスは、以下の手順で実施します。

(1) リバランス対象ファイルの抽出
(2) リバランス対象ファイルのマイグレーション
(3) 抽出結果のクリア
(4) マイグレーション情報のクリア（メンテナンス時に実施）

(1) リバランス対象ファイルの抽出

IO サーバで scatefs_rebalance コマンドを使用し、リバランス対象ファイルを抽出します。
抽出の完了は、レポート機能でも確認できます。

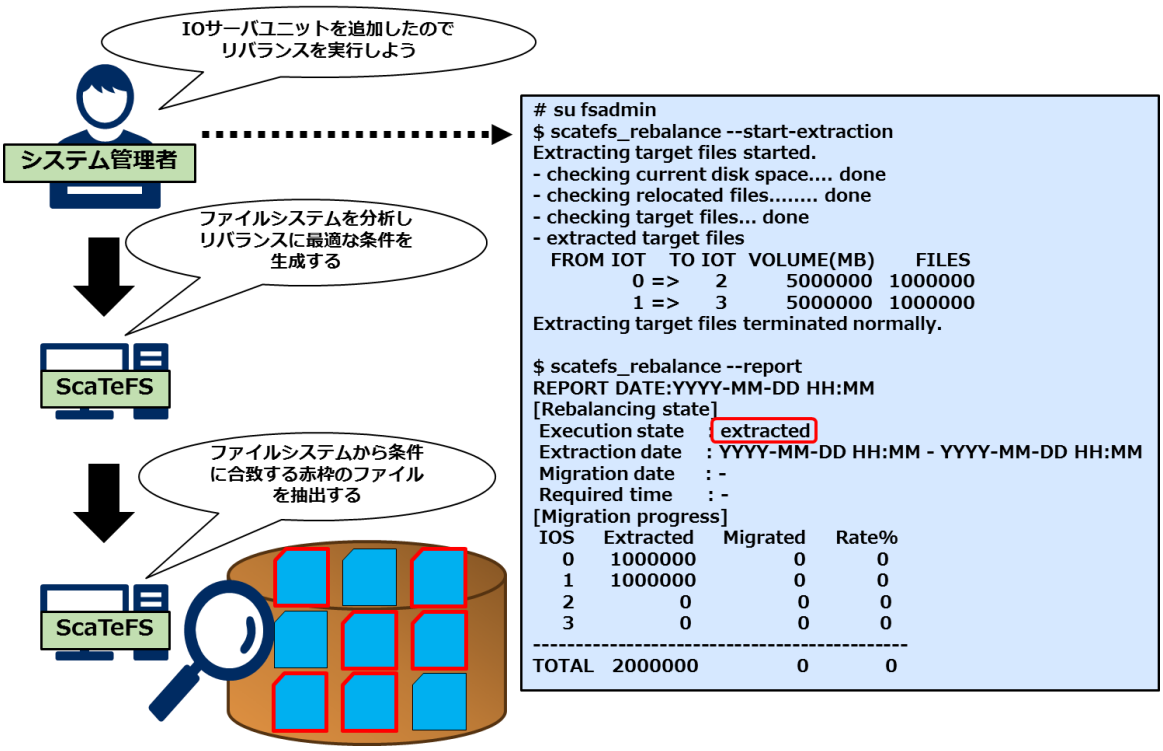

図 3-4　リバランス対象ファイル抽出の実行例

抽出し直す場合は、抽出結果をクリアしてから再度抽出を実施します。

```
# su fsadmin
$ scatefs_rebalance --clear
scatefs_rebalance: rebalance information was cleared.
$ scatefs_rebalance --start-extraction
Extracting target files started.
…
```

また、Linux クライアントで scatefs_rebalance_import コマンドを使用して、
リバランス対象ファイルを指定することもできます。

(2) リバランス対象ファイルのマイグレーション

リバランス対象ファイルの抽出が完了したら、IO サーバで scatefs_rebalance コマンドを使用し、マイグレーションサービスを開始します。これにより、対象ファイルがマイグレーションされます。マイグレーションの状況は、レポート機能で確認します。

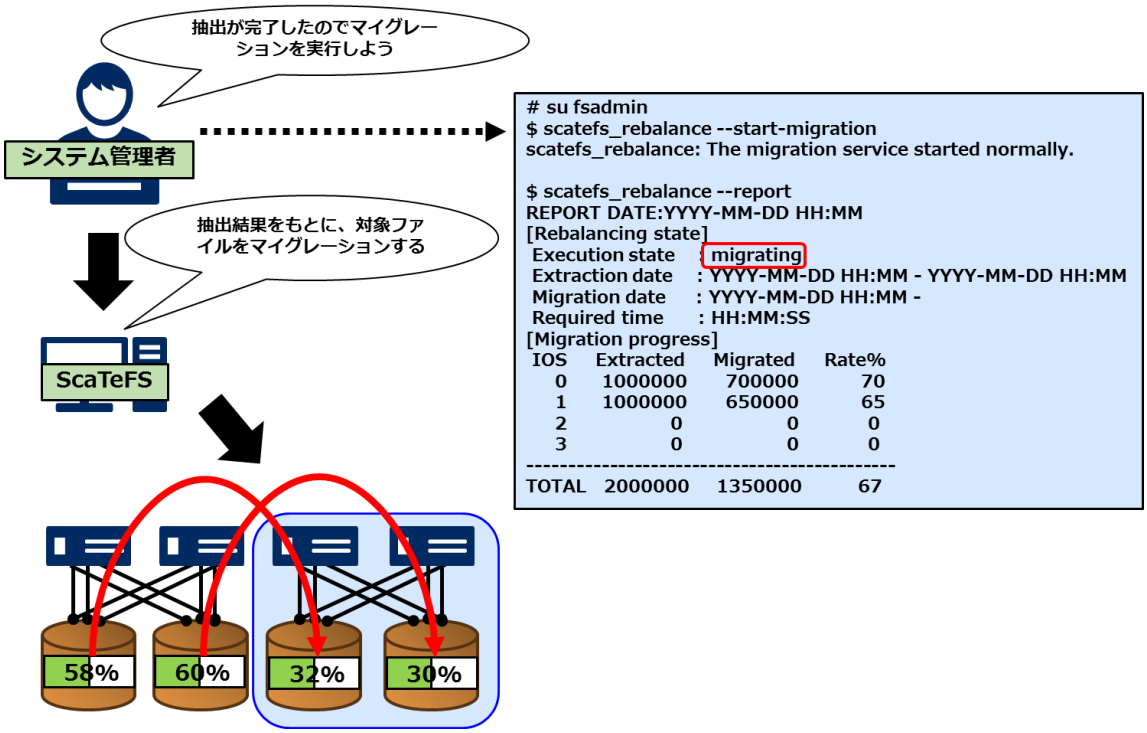

図 3-5　リバランス対象ファイルのマイグレーション実行例

マイグレーションが完了したら、マイグレーションサービスを停止します。

```
# su fsadmin
$ scatefs_rebalance --report
REPORT DATE:YYYY-MM-DD HH:MM
[Rebalancing state]
 Execution state    : migrated
 Extraction date    : YYYY-MM-DD HH:MM - YYYY-MM-DD HH:MM
 Migration date     : YYYY-MM-DD HH:MM - YYYY-MM-DD HH:MM
 Required time      : HH:MM:SS
[Migration progress]
 IOS    Extracted    Migrated    Rate%
   0    1000000      1000000       100
   1    1000000      1000000       100
   2          0            0         0
   3          0            0         0
-------------------------------------------------
TOTAL   2000000      2000000       100

$ scatefs_rebalance --stop-migration
scatefs_rebalance: The migration service stopped normally.
```

マイグレーション中でも、必要に応じマイグレーションサービスを一時停止と再開ができます。

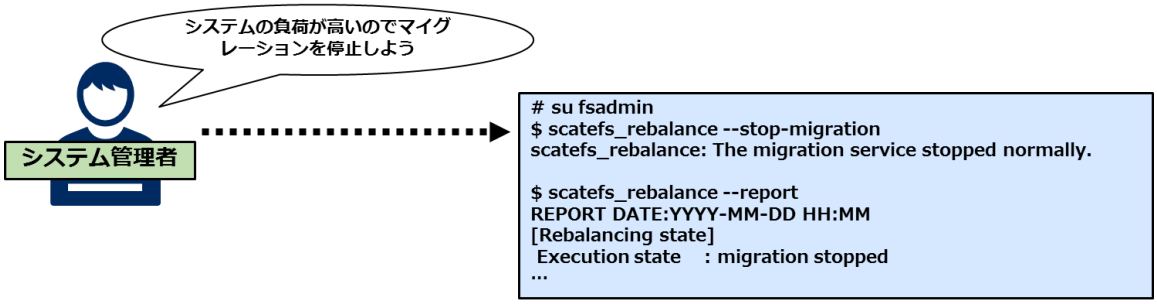

図 3-6　マイグレーションサービスの一時停止の実行例

なお、マイグレーションサービスの停止は一時的に受け付けられない場合があります。その場合は、再度コマンドを実行してください。

(3) 抽出結果のクリア

マイグレーションが完了したら、IO サーバで scatefs_rebalance コマンドを使用し、抽出した情報をクリアします。

```
# su fsadmin
$ scatefs_rebalance --clear
scatefs_rebalance: rebalance information was cleared.
```

以上でリバランス作業は完了です。

(4) マイグレーション情報のクリア（メンテナンス時に実施）

マイグレーションが終わり、すべてのクライアントのアンマウント（※）を確認したら、IO サーバで scatefs_migrate コマンドを使用し、マイグレーション情報をクリアします。マウント中のクライアントが存在する状況下では実施しないでください。

```
# su fsadmin
$ scatefs_migrate --clear
scatefs_migrate: The migration information was cleared.
```

（※）ScaTeFSクライアント上でScaTeFSのファイルシステムをNFSでエクスポートしている場合、
そのファイルシステムをマウントしているすべてのNFSクライアントからそのファイルシステムを
アンマウントしてください。次に、ScaTeFSクライアントでnfsサービスを停止してください。

マウント中のクライアントが存在しない状況でクリアできない場合は、フォースを指定します。

```
# su fsadmin
$ scatefs_migrate --clear
scatefs_migrate: cannot clear.
$ scatefs_migrate --clear --force
scatefs_migrate: The migration information was cleared forcibly.
```

3.2.8 リモート CLI

IOサーバ上に配置された一部のコマンドをクライアントから実行する仕組みとして、リモートCLI(scatefs_rcli)を提供します。scatefs_rcliで実行可能なサブコマンドは以下のとおりです。

表 3-2　リモート CLI のサブコマンド

サブコマンド名	概要	実行ユーザ制限
df	ScaTeFSの使用状況表示	なし
detail	ScaTeFSの構成情報表示	特権ユーザのみ実行可能
logcollect	IOサーバのログ表示	特権ユーザのみ実行可能
quota	ScaTeFSのquota情報表示	なし
repquota	ScaTeFSのquota情報の一覧表示	特権ユーザのみ実行可能
edquota	ScaTeFSのユーザおよびグループquotaの編集	特権ユーザのみ実行可能
ifstat	IOサーバのインターフェース状態表示	特権ユーザのみ実行可能
mkqdir	ScaTeFSのQUOTA対応ディレクトリの作成	特権ユーザのみ実行可能
rmqdir	ScaTeFSのQUOTA対応ディレクトリの削除	特権ユーザのみ実行可能

3.2.8.1　特権ユーザ

リモート CLI は一部のサブコマンドを除き、実行するには特別な権限が必要となります。この権限を持ったユーザを特権ユーザと呼びます。root 以外の特定のユーザを特権ユーザとするには、fsadmin グループに所属させてください。fsadmin グループに所属するユーザは、リモート CLI を実行する上での特権ユーザとなります。

(例)

```
・fsadminグループを追加
# groupadd fsadmin

・fooユーザが所属するグループにfsadminを追加
# usermod foo -G xxx,yyy,fsadmin
  ※xxx,yyy は既に所属しているグループ
```

3.2.8.2　リモート CLI ユーザの登録

クライアントから scatefs_rcli を使用するためには、IO サーバでの登録が必要になります。ユーザの登録は scatefs_rcliadm コマンドで行います。登録後、「4.2.1.18 scatefs_rcliadm」の例を参照し、動作確認を実施してください。

(例)

- clientA の foo ユーザを登録

```
$ scatefs_rcliadm add clientA foo
```

- 確認

```
$ scatefs_rcliadm info
clientA foo
```

- clientA の foo ユーザを削除

```
$ scatefs_rcliadm delete clientA foo
```

3.2.8.3　リモート CLI の実行

scatefs_rcliadm で登録されたユーザは、scatefs_rcli コマンドを実行することができます。

(例)

- clientA の foo ユーザが serverB の FSID#0 を指定し df サブコマンドを実行

```
$ scatefs_rcli serverB df 0
 IOT IOS SGID   1K-blocks          Used   Available  Use%   Mounted on
   0   0    0    11867221        305180    10974464    3%   /mnt/iot/0
   2   1    0    11867221        305180    10974464    3%   /mnt/iot/2
   1   0    0    11867213        305180    10974457    3%   /mnt/iot/1
   3   1    0    11867213        305180    10974457    3%   /mnt/iot/3
 ------------------------------------------------------------------------
   TOTAL         47468868       1220720    43897842    3%
```

- 登録されていないユーザで実行した場合

```
$ scatefs_rcli serverB df scatefs
Permission denied.
scatefs_rcli: df to serverB failed
```

3.2.9 情報表示

システムを構成する様々な情報を取得するインターフェースをIOサーバ上に配置されたコマンドとして提供します。

- scatefs_df

  ファイルシステム使用状況表示

  (例) ファイルシステムのディスク使用状況

```
$ scatefs_df scatefs00
 IOT IOS SGID      1K-blocks          Used      Available  Use%  Mounted on
   0   0    0    14276233588    8482274292     5492881741   60%  /mnt/iot/0
   3   1    0    14276233588    8488604028     5486868491   60%  /mnt/iot/3
   1   0    0    14276233588    8471883748     5502752757   60%  /mnt/iot/1
   4   1    0    14276233588    8461444560     5512669986   60%  /mnt/iot/4
   2   0    0    14276233588    8471705888     5502921724   60%  /mnt/iot/2
   5   1    0    14276233588    8471343548     5503265947   60%  /mnt/iot/5
 ------------------------------------------------------------------------------
 TOTAL           85657401528   50847256064    33001360646   60%
```

  (例) ファイルシステムの inode 使用状況

```
$ scatefs_df scatefs00 -i
 IOT IOS SGID       Inodes        IUsed        IFree  IUse%   Mounted on
   0   0    0     32527525       816564     31710961     3%   /mnt/iot/0
   3   1    0     32527531       816625     31710906     3%   /mnt/iot/3
   1   0    0     32527531       816671     31710860     3%   /mnt/iot/1
   4   1    0     32527531       816755     31710776     3%   /mnt/iot/4
   2   0    0     32527531       816573     31710958     3%   /mnt/iot/2
   5   1    0     32527531       816734     31710797     3%   /mnt/iot/5
 ------------------------------------------------------------------------------
 TOTAL          195165180      4899922    190265258     3%
```

- scatefs_detail

  ファイルシステムの構成情報表示

  (例) ファイルシステム全体

```
$ scatefs_detail -f 0
display detail FS#0
 FS Name        =>      scatefs00
```

```
 Root IOS       =>      IOS#0(IOT#0)
  IP            =>      10.0.0.1
  FIP           =>      10.0.1.1 10.0.2.1
PCI-ID@PORT     =>      0000:83:00.1@1
  INIP          =>      10.0.3.1
 Number of IOS  =>      2
 Number of IOT  =>      6 / 1024
 Number of SG   =>      1 / 8
 Data FS type   =>      ext4
 Ctrl FS type   =>      ext4
 Version        =>      0x00010000
 IOTs           =>      0 3 1 4 2 5
 SG             =>      ROOT
```

(例) IOS 単位で表示

```
-bash-4.1$ scatefs_detail -s 0
display detail IOS#0
 IP ADDRESS             =>      10.0.0.1
 Floating IP ADDRESS    =>      10.0.1.1 10.0.2.1
PCI-ID@PORT             =>      0000:83:00.1@1
 Inner IP ADDRESS       =>      10.0.3.1
 PORT for Client        =>      50000
 PORT for Server        =>      50001
 PORT for Client Data   =>      50002
 Defined IOTs           =>      0 1 2
 Defined FS             =>      0
```

(例) IOT 単位で表示

```
$ scatefs_detail -t 0
display detail IOT#0
 defined server =>      IOS#0
 filesystem     =>      scatefs00
 storagegroup   =>      ROOT
 data device    =>      /dev/vg_data01/lv_data01
 ctrl device    =>      /dev/vg_ctrl01/lv_ctrl01
```

- scatefs_statcollect

  IO サーバの統計情報の表示

  (例) 全 IOS の統計情報を表示

```
$ scatefs_statcollect -a
[IOS#0]
 :
[IOS#1]
 :
```

  (例) IO サーバ ID＃0 のプロシージャの統計情報を表示

```
$ scatefs_statcollect -n 0 -p
[IOS#0]
 :
```

  (例) IO サーバ ID＃1 の関数の統計情報を表示

```
$ scatefs_statcollect -n 1 -f
[IOS#1]
 :
```

- scatefs_logcollect

  IO サーバのログ表示

  ※ログをファイルに保存する場合は、リダイレクトしてください。

  (例) 全 IO サーバのログを表示

```
$ scatefs_logcollect -a
  ※結果を保存する場合
$ scatefs_logcollect -a > ioserver.log
```

  (例) IO サーバの全ログを表示 (ローテートされたファイル、gz 形式で圧縮されたファイルを含める)

```
$ scatefs_logcollect -a -m
```

(例) IO サーバ ID＃0 のログを表示

```
$ scatefs_logcollect -n 0
```

(例) IO サーバ ID＃1 と＃2 のログを表示

```
$ scatefs_logcollect -n 1,2
```

3.2.10 システムファイルの管理

IOサーバのシステムファイルを管理するコマンドとして、scatefs_adminを提供します。scatefs_adminでは/etc/scatefs配下の各システムファイルをIOサーバ間で一致しているかのチェック、指定したIOサーバへの転送/ロールバック、チューニングパラメータファイルの作成などが可能です。コマンドの詳細は「4.2.1.5 scatefs_admin」を参照してください。

(例) ScaTeFSの情報ファイル(system.info)がIOS間で一致しているか確認

```
$ scatefs_admin --check all system
```

(例) IOサーバデーモンのチューニングパラメータ設定ファイル(scatefssrv.conf)のデフォルトを作成

```
$ scatefs_admin --create tune
```

(例) IOサーバデーモンのチューニングパラメータ設定ファイル(scatefssrv.conf)を全IOサーバに転送

```
$ scatefs_admin --trans all tune
```

3.2.11 ファイルシステムの監視

ファイルシステムの統計情報をリアルタイムに収集しモニタリングする機能を提供します。必要となるソフトウェアをインストールおよび設定し、パッケージに同梱されているテンプレートをインポートすることにより、GUIベースでのファイルシステムのモニタリングが可能となります。

以下の統計情報をサポートします。

表 3-3　統計情報

ソース	統計情報
IOサーバ	ファイルシステム、IOサーバ、ユーザIDごとのread/writeのスループットやメタデータオペレーション性能データ IOサーバごとのネットワークトラフィックやCPU情報 ファイルシステムごとの使用量 ファイルシステムごとのプロファイル情報（ディレクト内のファイル数やファイルサイズごとの分布など）※
IOターゲット	IOターゲットごとのread/write数 IOターゲットごとの使用量

※scatefs_profstatコマンド（引数なし）を実行することにより、ファイルシステムごとのプロファイル情報を収集します。監視間隔に合わせてコマンドを実行してください。なお、収集に要する時間はご利用になる環境により異なります。

本機能を構成するソフトウェアとソフトウェアの要件を記載します。

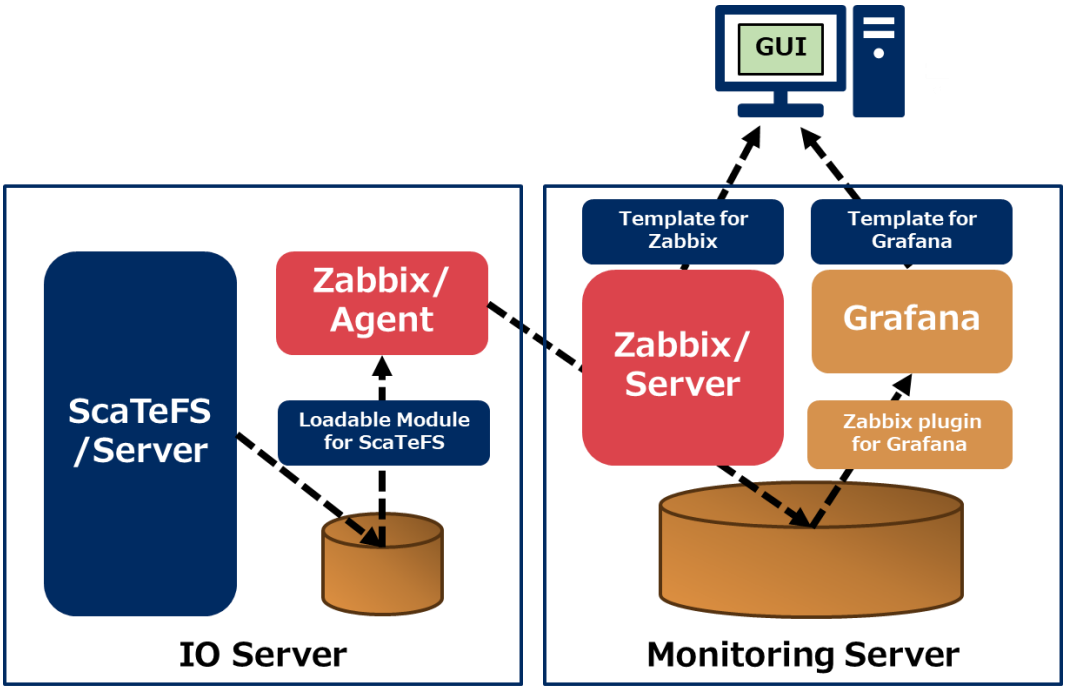

図 3-7　構成図

表 3-4　ソフトウェア

ソフトウェア	バージョン
ScaTeFS/Server	scatefs-ai-srv 1.0以降 scatefs-ai-mon 1.0以降 scatefs-ai-monパッケージには下記が同梱されています。 Loadable Module for ScaTeFS、Template for Zabbix、Template for Grafana
Zabbix/Server	zabbix-server 6.0 LTS以降 動作確認済み：6.0.25-release1.el8
Zabbix/Agent	zabbix-agent 6.0 LTS 動作確認済み：6.0.25-release1.el8
Grafana	grafana-10.2以降 動作確認済み：grafana-10.2.2-1
Zabbix plugin for Grafana	v4.4.4以降 動作確認済み：alexanderzobnin-zabbix-app-4.4.4.linux_amd64.zip

ScaTeFS/Serverをインストール後、本機能を使用するために必要な設定方法について説明します。なお、ZabbixやGrafanaを使用する上での基本的な設定はコミュニティが提供するドキュメントをご参照ください。

- Loadable Module for ScaTeFS
  scatefs-ai-srv パッケージの入手方法と同様に、scatefs-ai-mon パッケージを入手しインストールします。
- Zabbix/Agent
  Zabbix コミュニティからソフトウェアを入手しインストールします。
  Loadable Module for ScaTeFS を使用するために、zabbix_agentd.conf に以下の設定を追加してください。

```
LoadModulePath=/opt/scatefs/lib/
LoadModule=libscatefszbx.so
UserParameter=scatefs.alive.daemon, pgrep scatefs_server > /dev/null 2>&1; echo $?
```

- Zabbix/Server
  Zabbix コミュニティからソフトウェアを入手しインストールします。テンプレートを使用するために、以下の設定をしてください。
  (1) scatefs-ai-mon パッケージでインストールされた Zabbix 用テンプレートをインポートします。
  (2) ファイルシステムを構成する IO サーバを監視対象ホストに登録します。これらの IO サーバは同じホストグループに属するように設定します。
  (3) 追加した監視対象ホストに(1)でインストールしたテンプレートを追加します。
  (4) 追加した監視対象ホストのマクロに以下を追加します。
  マクロ名：{$SCATEFS_HOSTGROUPNAME}
  値：(2)で設定したホストグループ名
  (5) 「/etc/zabbix/zabbix_server.conf」に以下の設定を追加してください。
  IO サーバ 1 セット (2 台) につき、CacheSize は 16MB、TrendCacheSize は 8MB を指定してください。

```
CacheSize=16MB
TrendCacheSize=8MB
```

- Grafana と Zabbix plugin for Grafana

  Grafana コミュニティからソフトウェアを入手しインストールします。テンプレートを使用するために、以下の設定をしてください。

  (1) Zabbix plugin for Grafana を有効にして、データソースを追加します。

  (2) scatefs-ai-mon パッケージに同梱されている Grafana 用テンプレートをインポートします。

テンプレートの内容について説明します。

- Zabbix テンプレート

  モニタリングに必要となる監視アイテムを定義しています。また、以下の障害監視トリガを定義しています。

  - ScaTeFS/Server デーモンの死活監視

    ScaTeFS/Server デーモンプロセスの有無を監視します。

  - ScaTeFS ファイルシステムの使用量監視

    使用量を 3 つのレベルで監視します。

- Grafana テンプレート

  3 つのスクリーンを定義しています。

  - Data screen of ScaTeFS

    ファイルシステムや IO サーバごとの、read/write オペレーションに関する各種統計情報を表示します。

  - Metadata screen of ScaTeFS

    ファイルシステムや IO サーバごとの、メタデータオペレーションに関する各種統計情報を表示します。

  - ScaTeFS throughput/IO size per UID

    ユーザ ID ごとの、read/wiite やメタデータオペレーションに関する各種統計情報を表示します。

3.2.12 サブディレクトリマウント

ScaTeFSのファイルシステムのうち、一部のディレクトリツリーだけを選択してクライアントからマウントする機能を提供します。本機能でマウント可能なサブディレクトリとは、ScaTeFSファイルシステムのディレクトリ階層にある任意のディレクトリです。サブディレクトリマウントを利用することで、ファイルシステムの一部をアクセス対象とした運用が可能となります。

図 3-8にて、サブディレクトリマウントの運用イメージを記載します。

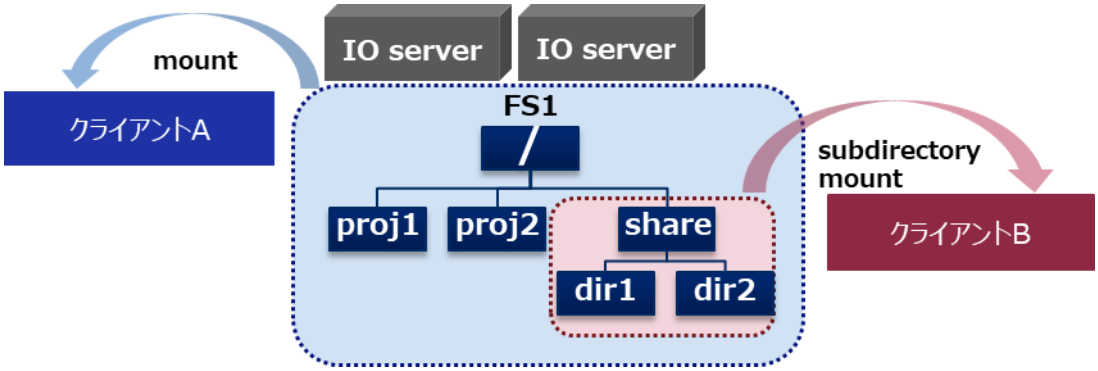

図 3-8　サブディレクトリマウントの運用イメージ

この図では、2台のIOサーバで構成されたScaTeFSファイルシステム（FS1）を2つのクライアント（クライアントA/B）でマウントしています。AはFS1全体をマウントしており、FS1のすべてのディレクトリにアクセス可能です。BはFS1の/shareディレクトリを部分的にマウントしており、/share/dir1、/share/dir2はアクセス可能ですが、/proj1、/proj2にはアクセスできません。

FS1に対するクライアントA/BのFS1へのアクセス状態（〇：アクセス可能、×：アクセス不可）は以下となります。

ディレクトリ	クライアント A	クライアント B
/ (FS1 全体)	〇	×
/proj1	〇	×
/proj2	〇	×
/share	〇	〇
/share/dir1	〇	〇
/share/dir2	〇	〇

3.2.12.1 マウント方法

サブディレクトリのマウントは、マウントするターゲットにサブディレクトリのパス名”/SUBDIR”を付加し、”HOST:FSNAME/SUBDIR”の形式で行います。

以下は、HOST:FS1/share を/mnt/subdir にマウントする例です。

```
# mount -t scatefs HOST:FS1/share /mnt/subdir
```

3.2.12.2 アンマウント方法

umount コマンドを使い、従来と同じようにファイルシステムをアンマウントします。/mnt/subdirにマウントされているファイルシステムの一部 (HOST:FS1/share) をアンマウントするには、次のいずれかを実行します。

(例 1)

```
# umount /mnt/subdir
```

(例 2)

```
# umount HOST:FS1/share
```

3.3　メンテナンス

3.3.1 ScaTeFS のシステム起動と停止

ScaTeFSのシステムの起動および停止の順序を記載します。

【システム起動の順序】

(1) ストレージの起動

IO サーバに接続しているストレージを起動します。詳細はストレージのマニュアルを参照してください。

(2) IO サーバの起動

2 台の IO サーバの電源ボタンを続けて押し、IO サーバを起動します。

※2 台の IO サーバの起動間隔を空けないでください。

IO サーバが起動後に状態を確認します。詳細は「3.3.2 IO サーバの起動と停止」の起動確認を参照してください。

(3) Linux クライアントの起動

電源ボタンを押して Linux クライアントを起動します。

(4) ScaTeFS のマウント

Linux クライアントが起動後に ScaTeFS をマウントできることを確認します。

詳細は「3.1.1 マウント方法」のマウント方法を参照してください。

【システム停止の順序】

(1) ScaTeFS のアンマウント

すべての Linux クライアントで ScaTeFS をアンマウントします。

詳細は「3.1.2 アンマウント方法」のアンマウント方法を参照してください。

(2) Linux クライアントの停止

すべての Linux クライアントを停止します。

(3) IO サーバの停止

IO サーバを停止します。詳細は「3.3.2 IO サーバの起動と停止」の停止を参照してください。

(4) ストレージの停止

IO サーバに接続しているストレージを停止します。詳細はストレージのマニュアルを参照してください。

3.3.2 IO サーバの起動と停止

クラスタ構成のIOサーバの起動および停止方法を記載します。

- 起動

  2 台の IO サーバの電源ボタンを続けて押し、IO サーバを起動します。

  ※2 台の IO サーバの起動間隔を空けないでください。

- 停止

  どちらかの IO サーバへログインして clpstdn コマンドを実行します。

  2 台の IO サーバが停止します。

```
# clpstdn
```

- 再起動

  どちらかの IO サーバへログインして clpstdn -r コマンドを実行します。

  2 台の IO サーバが再起動します。

```
# clpstdn -r
```

- 起動確認

  どちらかの IO サーバへログインして clpstat コマンドを実行します。クラスタ状態が表示されますので下記を確認します。

  a) すべてのリソースが Online もしくは Normal であること

  b) <group>タグの current には当該グループのサーバ名が表示されていること

     failover1 グループの current に iosv00、failover2 グループの current に iosv01 が表示されていることが正しいクラスタ状態です。

     もしフェイルオーバしている場合は、<group>タグの current に同じサーバ名が表示されます。問題がある場合は該当リソースの障害を解消してください。

  以下に実行例を記載します。

```
# clpstat
 ======================  CLUSTER STATUS  ==========================
  Cluster : cluster
  <server>
   *iosv00 ..........: Online
      lankhb1        : Normal          Kernel Mode LAN Heartbeat
      diskhb1        : Normal          DISK Heartbeat
    iosv01 ..........: Online
      lankhb1        : Normal          Kernel Mode LAN Heartbeat
      diskhb1        : Normal          DISK Heartbeat
```

```
<group>
  failover1 .......: Online
    current        : iosv00
    disk_c_01      : Online
    disk_c_02      : Online
    disk_c_03      : Online
    disk_d_01      : Online
    disk_d_02      : Online
    disk_d_03      : Online
    exec1          : Online
    exec_route1    : Online
    fip_ib1        : Online
    volmgr_c_01    : Online
    volmgr_c_02    : Online
    volmgr_c_03    : Online
    volmgr_d_01    : Online
    volmgr_d_02    : Online
    volmgr_d_03    : Online
  failover2 .......: Online
    current        : iosv01
    disk_c_04      : Online
    disk_c_05      : Online
    disk_c_06      : Online
    disk_d_04      : Online
    disk_d_05      : Online
    disk_d_06      : Online
    exec2          : Online
    exec_route2    : Online
    fip_ib2        : Online
    volmgr_c_04    : Online
    volmgr_c_05    : Online
    volmgr_c_06    : Online
    volmgr_d_04    : Online
    volmgr_d_05    : Online
    volmgr_d_06    : Online
<monitor>
  diskw_c_01       : Normal
  diskw_c_04       : Normal
  fipw1            : Normal
  fipw2            : Normal
```

```
    genw1             : Normal
    genw2             : Normal
    userw             : Normal
    volmgrw1          : Normal
    volmgrw10         : Normal
    volmgrw11         : Normal
    volmgrw12         : Normal
    volmgrw2          : Normal
    volmgrw3          : Normal
    volmgrw4          : Normal
    volmgrw5          : Normal
    volmgrw6          : Normal
    volmgrw7          : Normal
    volmgrw8          : Normal
    volmgrw9          : Normal
 ================================================================
#
```

3.3.3 メンテナンスまたはチューニング時に活用できるコマンド

メンテナンスまたはチューニング時に活用できるScaTeFS for AI用のコマンドの詳細については、「第4章 ScaTeFSクライアント用とIOサーバ用のコマンドリファレンス一覧」をご参照ください。

3.3.4 運用中サーバのメンテナンス

IOサーバのメンテナンス作業に関して説明します。

3.3.4.1　バックアップ

ScaTeFS システムとして特別なバックアップ機能はサポートしていません。このため、バックアップサーバ上でファイルシステムをマウントし、仮想ファイル単位でバックアップを実施してください。

3.3.4.2　ScaTeFS for AI パッケージの無停止アップデート

ScaTeFS for AI パッケージの無停止アップデート手順については、『NEC Scalable Technology File System for AI (ScaTeFS for AI) インストレーションガイド』の「2.5.2 システム運用を継続して行うアップデート(無停止アップデート)」をご参照ください。

なお、scatefs-ai-srv パッケージは、ファイルシステムの運用を継続した状態でアップデート(無停止アップデート)が可能です。ただし、ファイルシステムを構成する全 IO サーバ

で同期が必要な場合は、無停止アップデート対象外です。パッケージが無停止アップデート可能か否かは、パッケージの指示書に記載されていますので確認してください。

3.3.5 運用を停止する必要のある事項

システム運用中に実施できないメンテナンス作業があります。これらメンテナンスを実施する場合、システムの運用を停止する必要があります。

- scatefs_extendfs によるファイルシステムの拡張。
- fsck による修復(ローカルファイルシステムおよび ScaTeFS ファイルシステム)。
- scatefs_quotacheck による ScaTeFS QUOTA 情報の整合性チェックと修復。

3.3.6 ファイルシステムの整合性チェックと修復

ScaTeFSファイルシステム専用のファイルシステムの整合性チェックと修復機能を提供します。修復ではScaTeFSファイルシステムの運用停止が必要です。修復には、以下の2通りの手順があります。

■ 通常の修復手順（推奨手順）

一度の停止期間内にすべてのメンテナンスを実施します。

① ScaTeFS ファイルシステムの運用を停止
② ローカルファイルシステム提供の fsck を実施 (必要な場合)
③ ScaTeFS ファイルシステムの整合性チェックと修復を実施
④ QUOTA 情報の整合性チェックと修復を実施 (実施を推奨)
⑤ ScaTeFS ファイルシステムの運用を再開

■ より停止時間を短くする修復手順

① ディスク障害などローカルファイルシステムの修復が必要な場合
- ScaTeFS ファイルシステムの運用を停止
- 障害の要因を取り除きローカルファイルシステム提供の fsck を実施
- ScaTeFS ファイルシステムの運用を再開

② ScaTeFS ファイルシステムの整合性チェックのみを実施し実行結果を任意のファイルへ退避 (チェックのみの場合運用中に実施可能)
③ ScaTeFS ファイルシステムの運用を停止
④ ローカルファイルシステム提供の fsck を実施 (必要な場合)
⑤ ②の整合性チェック結果を入力として ScaTeFS ファイルシステムの修復を実施
⑥ QUOTA 情報の整合性チェックと修復を実施 (実施を推奨)
⑦ ScaTeFS ファイルシステムの運用を再開

なお、ディスク障害などローカルファイルシステムの修復が必要な場合、②～③の運用中に、特定のディレクトリやファイルにアクセスできないなどの事象が発生する場合があります。これらは⑤を実施することにより解消されます。

各コマンドの使用方法を以下に記載します。

- 整合性チェック

  実施対象のファイルシステム ID を指定し、ファイルシステムの整合性チェックを行います。

  (例)

```
$ scatefs_fsck -n fsid
```

- 整合性チェックと修復

  実施対象のファイルシステムIDを指定し、ファイルシステムの修復を行います。ファイルシステムの修復前にすべてのIOサーバでIOサーバデーモンの停止を行います。

  ※修復が完了しましたら正しく修復していることの検証のため、再度修復を実施してください。

  (例)

```
$ scatefs_fsck fsid
```

- 整合性チェック結果をもとに ScaTeFS ファイルシステムの修復

  整合性チェック結果ファイルを指定し、ファイルシステムの修復を行います。整合性チェック結果で修復対象が絞り込まれているため、高速に修復が可能になります。ファイルシステムの修復前にすべての IO サーバで IO サーバデーモンの停止を行います。

  ※修復が完了しましたら正しく修復していることの検証のため、再度修復を実施してください。

  (例)

```
$ scatefs_f2fsck infile
```

- QUOTA 情報の整合性チェックと修復

  実施対象のファイルシステム名を指定し、QUOTA 情報の整合性チェックと修復を行います。QUOTA 情報の整合性チェックと修復前にすべての IO サーバで IO サーバデーモンの起動を行います。

  (例)

```
$ scatefs_quotacheck fsname
```

3.3.7 ネットワークの経路障害とパス切り替え

ScaTeFSクライアントは複数の経路を使ってIOサーバと通信を行います。

ScaTeFSクライアントとIOサーバ間の一部の経路にネットワーク障害が発生した場合、ScaTeFSクライアントは利用できる経路に切り替えて通信を継続します(パス切り替え)。ネットワーク障害が発生した経路はScaTeFS経路監視デーモンが監視し、復旧を検知すると自動的に経路の利用が再開されます。そのため、ネットワーク障害の復旧後に必要な処置は特にありません。

3.3.8 syslog メッセージ

3.3.8.1　Linux クライアント

ファイルシステムオペレーション機能

ScaTeFS:400100 commit error after file close. filesystem name=<*filesystem name*> dev=<*device number*> code=<*code*> data=<*internal data*>

[種別] ERROR

[説明] ファイルシステムで、ファイルクローズ後に、そのファイルに書き出されたデータの IO サーバのストレージへの同期処理でエラーが発生しました。

filesystem name: ファイルシステム名

device number: ファイルシステムのデバイス番号

code: エラーを表すコード(errno と同じ値)

internal data: 内部データ

同じファイルシステムで継続してエラーが発生した場合、1 時間間隔で本メッセージが出力されます。

[対処] 障害原因を取り除いた後に、障害にあったファイルを、ファイルを作成するジョブの再実行等により復旧してください。

障害にあったファイルは、障害発生日時(本メッセージの出力日時)、本メッセージ中のファイルシステム情報、後述の ScaTeFS:400101 のメッセージ中のファイル情報、アプリケーションによるファイルのアクセス状況等から特定することになります。

ScaTeFS:400101 commit error after file close. dev=<*device number*> ino=<*inode number*> uid=<*user id*> gid=<*group id*> code=<*code*> data=<*internal data*>

[種別] ERROR

[説明] ファイルで、ファイルクローズ後に、そのファイルに書き出されたデータの IO サ

ーバのストレージへの同期処理でエラーが発生しました。

device number: ファイルシステムのデバイス番号

inode number: ファイルの inode 番号

user id: ファイルのユーザ ID

group id: ファイルのグループ ID

code: エラーを表すコード(errno と同じ値)

internal data: 内部データ

本メッセージの出力数が 5 秒間で 200 個を超えた場合、それ以降の 5 秒間は、本メッセージの出力は抑止されます。抑止された場合、後述の ScaTeFS:400102 のメッセージが出力されます。

ScaTeFS:400102 のメッセージが出力された場合、本メッセージから、障害にあったすべてのファイルを特定することはできません。障害にあったが本メッセージが出力されなかったファイルが存在します。

[対処] 障害原因を取り除いた後に、障害にあったファイルを、ファイルを作成するジョブの再実行等により復旧してください。

障害にあったファイルは、障害発生日時(本メッセージの出力日時)、前述の ScaTeFS:400100 のメッセージ中のファイルシステム情報、本メッセージのファイル情報、アプリケーションによるファイルのアクセス状況等から特定することになります。

ScaTeFS:400102 のメッセージが出力された場合、本メッセージから、障害にあったすべてのファイルを特定することはできません。障害発生直前のアプリケーションによるファイルのアクセス状況等から、障害にあったファイルを特定する必要があります。

ScaTeFS:400102 drop commit error messages due to rate-limiting. data=<*internal data*>

[種別] ERROR

[説明] ファイルで、ファイルクローズ後に、そのファイルに書き出されたデータの IO サーバのストレージへの同期処理でエラーが発生したこと表すメッセージ(ScaTeFS:400101)の出力が抑止されました。

internal data: 内部データ

[対処] 不要です。

データ転送機能(TCP)

ScaTeFS:RPC: all connections related to 〈*ServerAddress*〉:〈*Port*〉 are failed, still trying

[種別] WARNING

[説明] IOサーバとの通信が失敗しました。全パス障害が発生しています。

[対処] ネットワーク経路に異常がないか確認してください。IOサーバの状態を確認してください。

ScaTeFS:RPC: all connections related to 〈*ServerAddress*〉:〈*Port*〉 are failed, timed out

[種別] WARNING

[説明] IOサーバとの通信が失敗しました。全パス障害が発生しています。ソフトマウントのため、ファイル操作はエラーとなります。

[対処] ネットワーク経路に異常がないか確認してください。IOサーバの状態を確認してください。

ScaTeFS:RPC: retry to server 〈*ServerAddress*〉:〈*Port*〉 has been cancelled by signal.

[種別] NOTICE

[説明] 再送を行っていましたが、要求がシグナルにより中断されました。

[対処] 不要です。

ScaTeFS:RPC: server 〈*ServerAddress*〉:〈*Port*〉 OK

[種別] NOTICE

[説明] 再送が発生していましたが、IOサーバと通信ができました。

[対処] 不要です。

ScaTeFS:RPC: server <*ServerAddress*>:<*Port*> is unavailable. Using alternative connection path

[種別] WARNING

[説明] 再送オーバーが発生したため、パス切り替え処理を開始しています。

[対処] ネットワーク経路に異常がないか確認してください。IOサーバの状態を確認してください。

ScaTeFS:RPC: server <*ServerAddress*>:<*Port*> not responding, still trying

[種別] NOTICE

[説明] IOサーバとの通信がタイムアウトしたため、再送を行っています。

[対処] 頻発する場合、ネットワーク経路に異常がないか確認してください。また、IOサーバの状態を確認してください。

ScaTeFS:RPC: server <*ServerAddress*>:<*Port*> not responding, timed out. (pid=<*PID*>, proc=<*ProcedureNumber*>)

[種別] NOTICE

[説明] RPCの応答がありません。ソフトマウントのため、RPC要求はエラーとなりました。

[対処] IOサーバの状態を確認してください。また、ネットワーク経路に異常がないか確認してください。

ScaTeFS:pmond: connect to server <*ServerAddress*>:<*Port*> ok

[種別] NOTICE

[説明] 障害状態の経路が復旧しました。

[対処] 不要です。

3.3.8.2　IO サーバ

syslog を使用した IO サーバの障害監視方法を記載します。

(***は任意の文字列を示す)

ストレージ関連メッセージ

```
lpfc***Down
または
lpfc***Reset
```

[種別] ERROR

[説明] サーバ側FCポートに障害が検出されました。

[対処] ストレージとIOサーバの経路上に障害が発生した可能性があります。

サポート部門に連絡してください。

```
KAPLnnnnn-E ***
  ※nnnnn：メッセージID
```

[種別] ERROR

[説明] HDLMがエラーを検出しました。

[対処] HDLMユーザーズガイドを確認の上、サポート部門に連絡してください。

CLUSTERPRO 関連メッセージ

```
There was a request to restart resource(***) from the clprm process
```

[種別] WARNING

[説明] CLUSTERPROがリソースの異常を検出し当該リソースを再起動しました。

CLUSTERPROによりフェイルオーバが実行される可能性があります。

[対処] ScaTeFSの状態確認(*1)を実施し、リソース異常の原因を取り除いてください。

メッセージ詳細は、CLUSTERPRO関連マニュアルを確認してください。

```
Detected an error in monitoring ***
```

[種別] ERROR

[説明] CLUSTERPROがモニタリソースの監視で異常を検出しました。

CLUSTERPROによりフェイルオーバが実行される可能性があります。

[対処] ScaTeFSの状態確認(*1)を実施し、リソース異常の原因を取り除いてください。
メッセージ詳細は、CLUSTERPRO関連マニュアルを確認してください。

Resource *** of server *** has stopped

[種別] ERROR
[説明] IOサーバの特定のリソースが停止しました。
CLUSTERPROによりフェイルオーバが実行されます。
[対処] フェイルオーバした状態で運用は継続可能です。ScaTeFSの状態確認(*1)を実施し、リソース異常の原因を取り除いてください。ただし、2セット以上のIOサーバでフェイルオーバが発生し、かつリソース異常の原因が不明の場合は、障害拡大の可能性があるため、直ちに運用を停止してください。
メッセージ詳細は、CLUSTERPRO関連マニュアルを確認してください。

ScaTeFS 関連メッセージ

IOS*** server started (secondary mode)

[種別] ERROR
[説明] ScaTeFSサーバ機能がフェイルオーバしました。
[対処] フェイルオーバした状態で運用は継続可能です。
ScaTeFSの状態確認(*1)を実施し、リソース異常の原因を取り除いてください。
ただし、2セット以上のIOサーバでフェイルオーバが発生し、かつリソース異常の原因が不明の場合は、障害拡大の可能性があるため、直ちに運用を停止してください。

async event(IBV_EVENT_LID_CHANGE) at hca(***). stop the daemon. または async event(IBV_EVENT_CLIENT_REREGISTER) at hca(***). stop the daemon.

[種別] ERROR
[説明] サブネットマネージャの再起動等により、IOサーバデーモンが再起動しました。
[対処] サブネットマネージャの状態に問題がないか確認してください。
また、メンテナンスによるサブネットマネージャの再起動はScaTeFSの運用を停止してから実施するようにしてください。

async event(IBV_EVENT_SM_CHANGE) at hca(***). stop the daemon.

[種別] ERROR

[説明] サブネットマネージャが予備のサブネットマネージャに切り替わり、IOサーバデーモンが再起動しました。

[対処] サブネットマネージャの状態に問題がないか確認してください。

InfiniBand timeout happened on HCA#<*N*> (PID=*** CLIENTID=***)

[種別] WARNING

[説明] InfiniBandによる通信でタイムアウトが発生しました。

N: IOサーバのHCAを識別する番号。scatefs_addiosコマンドに指定する定義ファイルにおける、pciid@hcaportの項目にN番目に指定したHCAに該当します。(0オリジン)

[対処] ネットワーク経路に異常がないか確認してください。

NET: hca(***:<*hca-id1*>:<*hca-port1*>) is replaced with hca(***:<*hca-id2*>:<*hca-port2*>)

[種別] WARNING

[説明] IOサーバデーモン起動時に非ACTIVE状態のHCAを検出したので、ACTIVE状態のHCAで代替して起動しました。<hca-idX>:<hca-portX>はscatefs_addiosコマンドに指定する定義ファイル中のpciid@hcaportに指定したHCAのIDとポート番号です。

[対処] IOサーバのHCAに異常がないか確認してください。

【注釈】

(*1)「ScaTeFS の状態確認」とは下記を指します。

- clpstat コマンドでクラスタ状態を表示し、以下を確認してください。
  異なる場合、当該リソースに何らかの問題が発生しています。
  - 全てのリソースが Online もしくは Normal であること
  - <group>タグの current には当該グループのサーバ名が表示されていること
    (フェイルオーバしている場合は、2 つの<group>タグの current に同じサーバ名が表示されます)
- クライアントから正常にアクセスできていることを確認してください。
  - 3.1.1 マウント方法 に記載されているマウント後の IO 確認を実施する

3.4　Linuxクライアントのオプション設定

3.4.1 ファイルクローズ時の同期遅延

マウントオプションでのsync_on_close(既定値)またはno_sync_on_closeの指定により、ファイルクローズ時にクライアントがファイルデータをIOサーバのスレージと同期させるかどうかを指定することができます。

sync_on_close(既定値)の場合、ファイルクローズ時に、クライアントはファイルに書き出されたデータをIOサーバへ送信し、送信したデータとIOサーバのストレージの同期を行います。データ保全性が最も高いモードです。

no_sync_on_closeの場合、ファイルクローズ時に、クライアントはファイルに書き出されたデータをIOサーバへ送信しますが、送信したデータとIOサーバのストレージの同期は行いません。送信したデータとIOサーバのストレージの同期は、ファイルクローズ後に非同期で行われます。sync_on_closeに比べデータ保全性は低下しますが、数十KB以下の小さなファイルの作成時間を短縮することができます。

no_sync_on_closeを指定した場合、ファイルに書き出されたデータをIOサーバのストレージと同期させる処理が、ファイルクローズ後に遅延されることにより、ファイルクローズの処理時間が短縮されます。これにより、数十KB以下の小さなファイルを多数作成するtarコマンドやcpコマンド等の処理時間を短縮することができます。

ただし、次の場合では、no_sync_on_closeを指定してもアプリケーションの処理時間の短縮効果は小さい、または、短縮効果はありません。

- ファイルに書き込んだデータのサイズが大きい場合(数十 KB 以上)
- アプリケーションで明示的に書き込んだデータの同期(fsync(2),msync(2)等)を行っている場合
- アプリケーションでレコードロックやファイルロックを行っている場合
- アプリケーションで同一ファイルに対してオープンとクローズを繰り返し行っている場合

no_sync_on_closeを指定した場合、次の注意事項があります。

- ファイルクローズ後にクライアントと IO サーバの両方が同時にダウンした場合、IO サーバのストレージへの同期がまだ完了していない更新データは失われます。アプリケーションがデータを書き出してから IO サーバのストレージへの同期が完了するまでの時間は、クライアントでのダーティデータが書き出されるまでの時間の設定に依存しますが、概ね 2 分以内です。
- ファイルクローズ後に、更新データの IO サーバのストレージへの遅延同期処理でエラーが

発生した場合、ファイルをクローズしたアプリケーションでそのエラーを検出することはできません。ここでのエラーとしては、たとえばストレージ障害による IO エラーがあります。IO サーバダウンは含みません。

このとき、クライアントの syslog にエラーメッセージ(ScaTeFS:400100, ScaTeFS:400101)が出力されます。障害にあったファイルは、障害発生日時(前述のメッセージの出力日時)、メッセージ中のファイルシステム情報やファイル情報、アプリケーションによるファイルのアクセス状況等から特定することになります。

3.4.2 注意事項

3.4.2.1　オープンしているファイルの削除について

1 つのクライアント上で、あるプロセスがオープンしているファイルを削除すると、オープン中のファイルはすぐには削除されず、いったん次の形式のファイル名に自動的にリネームされます。

形式: .scatefsXXX...X(X:英数字)

例: .scatefs00000000010107640000000ab

このファイルは、オープンしていたプロセスからクローズされると自動的に削除されます。自動的に削除される前に、このファイルを手動で削除しようとすると、"Device or resource busy"のエラーになります。

3.4.2.2　二重マウント時の注意事項（RHEL 8)

ScaTeFS のファイルシステムをマウントした状態で、同じマウントポイントに他のファイルシステムをマウントした場合、はじめに umount コマンドを使用して他のファイルシステムをアンマウントし、次に ScaTeFS のファイルシステムを/sbin/umount.scatefs コマンドを使用してアンマウントしてください。2 番目の ScaTeFS のファイルシステムのアンマウントを、umount コマンドを使用して行った場合、エラーとなりアンマウントに失敗します。

```
# umount /mnt/scatefs
# /sbin/umount.scatefs /mnt/scatefs
```

3.4.2.3　mlocate パッケージを使用する場合の注意事項

mlocate パッケージをインストールしている場合、既定値では updatedb が ScaTeFS のパスを毎日チェックします。各クライアントでこれが行われるとシステムへの大きな負荷になります。以下のように /etc/updatedb.conf ファイルの PRUNEFS に scatefs を追加してチェック対象外としてください。

```
# rpm -q mlocate
mlocate-XXX.x86_64
# grep PRUNEFS /etc/updatedb.conf
PRUNEFS = "9p afs anon_inodefs auto autofs bdev binfmt_misc cgroup cifs coda
configfs
cpuset debugfs devpts ecryptfs exofs fuse fuse.sshfs fusectl gfs gfs2 gpfs
hugetlbfs
inotifyfs iso9660 jffs2 lustre mqueue ncpfs nfs nfs4 nfsd pipefs proc ramfs
rootfs
rpc_pipefs securityfs selinuxfs sfs sockfs sysfs tmpfs ubifs udf usbfs ceph
fuse.ceph
scatefs"
```

3.5　NFSサーバを使ってエクスポートする方法

Linuxクライアント上のNFSサーバを使って、ファイルシステムをNFSクライアントへエクスポートすることができます。

ファイルシステムをエクスポートする場合、/etc/exportsにfsidオプションを使ってファイルシステムを識別する整数を記述する必要があります。以下に/etc/exportsの記述例を示します。

```
/mnt/scatefs *(rw,no_root_squash,mp,fsid=1)
```

また、以下の注意事項があります。

- サポートするNFSバージョンは3のみです。また、サポートするプロトコルはTCPのみです。
- サポートするNFSクライアントはLinuxのみです。
- NFSクライアントがLinuxの場合、NFSクライアントでのマウント時にNFSバージョンとして3を明示してください。Linuxのディストリビューションによっては、NFSバージョンを明示しないとNFSバージョンとしてサポート対象外の4が使用されます。

NFSバージョン4の使用を防止することもできます。設定の詳細についてはお使いのRHELの「ストレージ管理ガイド」を参照してください。

- NFSクライアントがファイルをロックした場合、そのロックの影響を受けるのは、同じNFSサーバの他のNFSクライアントと、そのNFSサーバが存在するScaTeFSのクライアントのみです。NFSを介さずにファイルシステムを直接アクセスするクライアントは、NFSクライアントによるロックの影響を受けません。

第4章　ScaTeFS クライアント用と IO サーバ用のコマンドリファレンス一覧

4.1　ScaTeFSクライアント

ScaTeFSクライアント上で使用可能なコマンドを以下に記載します。

4.1.1 管理者向け

4.1.1.1　scatefs

名前

scatefs – ScaTeFS のマウントとアンマウント

書式

mount -t scatefs [-f] [-o options] server:fsname[/subdir] mountpoint

umount [-f] mountpoint | server:fsname[/subdir]

説明

ScaTeFS のマウント

NEC Scalable Technology File System(ScaTeFS)をマウントする場合、mount コマンドの -t オプション(ファイルシステムタイプ)に scatefs を指定します。server はルート IO サーバの IP アドレスまたはホスト名、fsname は ScaTeFS のファイルシステム名、subdir はサブディレクトリ名、mountpoint はマウントポイントをそれぞれ指定します。subdir を指定しない場合、ファイルシステム全体をマウントします。

-f　/etc/mtab へエントリの追加は行いますが、マウント自体は行いません。

-o に指定可能なオプション:

ro | rw　ファイルシステムを[読み取り専用(ro)/読み書き可能(rw)]に設定します。デフォルトは rw です。

fg | bg　最初のマウント試行がタイムアウトになった場合、再試行を[フォアグラウンド(fg)//バックグラウンド(bg)]で行います。デフォルトは fg です。

soft | hard	IO サーバとの通信が全パス障害となった際の動作を指定します soft の場合は呼び出したプログラムに対して I/O エラーを返します。hard の場合はリトライし続けます。デフォルトは hard です。
sync | async	ファイルシステムに対するすべてのデータの書き込みを[同期(sync)/非同期(async)]で行います。デフォルトは async です。
cto | nocto	ファイルをオープンする際、新たな属性の取得を[有効(cto)/無効(nocto)]に設定します。デフォルトは cto です。
ac | noac	属性のキャッシングを[有効(ac)/無効(noac)]に設定します。noac を指定した場合、ファイルシステムに対するすべてのデータの書き込みは同期になります。デフォルトは ac です。
exec | noexec	実行形式ファイルの実行を[許可(exec)/非許可(noexec)]に設定します。デフォルトは exec です。
suid | nosuid	ファイルシステム上で動作する実行形式ファイルのセットユーザ ID、セットグループ ID を[有効(suid)/無効(nosuid)]に設定します。デフォルトは suid です。
sync_on_close | no_sync_on_close	ScaTeFS クライアントが、ファイルクローズ時にファイルデータを IO サーバのストレージと同期させるかどうかを指定します。sync_on_close の場合、ファイルクローズ時に、ファイルに書き込まれたデータを IO サーバへ送信し、送信したデータと IO サーバのストレージの同期を行います。no_sync_on_close の場合、ファイルクローズ時に、ファイルに書き込まれたデータを IO サーバへ送信しますが、送信したデータと IO サーバのストレージの同期は行いません。送信したデータと IO サーバのストレージの同期は、ファイルクローズ後に非同期で行われます。デフォルトは sync_on_close です。

rsize=n　サーバからファイルを読み込む際に用いるバッファのバイト数を指定します。デフォルト値は 1048576 バイトです。単位には k/K(キロバイト), m/M(メガバイト)が指定可能です。
たとえば、rsize=4194304 と rsize=4M と rsize=4m はいずれも同じ指定になります。

wsize=n　サーバにファイルを書き込む際に用いるバッファのバイト数を指定します。デフォルト値は 1048576 バイトです。
単位には k/K(キロバイト), m/M(メガバイト) が指定可能です。
たとえば、wsize=4194304 と wsize=4M と wsize=4m はいずれも同じ指定になります。

retry=n　フォアグラウンド(fg)、またはバックグラウンド(bg)でのマウントオペレーションが、リトライを放棄するまでの時間(分単位)を指定します。fg のデフォルトは 2 分、bg のデフォルトは 10000 分です。

timeo=n　RPC のタイムアウト時間を 1/10 秒単位で指定します。デフォルトは 600 です（ 60 秒)。 タイムアウト時間を経過すると再送を行います。

retrans=n　メジャータイムアウトになるまでの再送回数を指定します。メジャータイムアウトになるとそのパスを復旧監視状態にして、他のパスで通信を行います(パス切り替え)。障害監視状態のパスは、復旧すると自動的に利用を再開します。TCP を使う場合のデフォルトは 5 回です。

acregmin=n　レギュラーファイルの属性がキャッシュされる最小の時間を、秒単位で指定します。この時間内では、サーバへの新たな情報の問い合わせは行いません。デフォルトは 3 秒です。

acregmax=n　レギュラーファイルの属性がキャッシュされる最大の時間を、秒単位で指定します。この時間を越えると、サーバへ新たな情報の問い合わせを行います。デフォルトは 60 秒です。

acdirmin=n　ディレクトリの属性がキャッシュされる最小の時間を、秒単位で指定します。この時間内では、サーバへの新たな情報の問い合わせは行いません。デフォルトは 30 秒です。

acdirmax=n　ディレクトリの属性がキャッシュされる最大の時間を、秒単位で指定します。この時間を越えると、サーバへ新たな情報の問い合わせを行います。デフォルトは 60 秒です。

actimeo=n　acregmin, acregmax, acdirmin, acdirmax をすべて同じ値に設定します。デフォルト値はありません。

lookupcache=mode　ファイルやディレクトリ等のエントリが IO サーバに存在するかどうかを検索した結果のキャッシュ方法を指定します。

mode として、all、pos または positive、none のいずれかを指定できます。デフォルトは all です。

検索したエントリが IO サーバに存在した結果を positive と表し、検索したエントリが IO サーバに存在しなかった結果を negative と表します。 positive または negative でキャッシュされたエントリに対してアクセスが行われた場合、IO サーバに対してエントリが存在するかどうかの検索は行われません。 キャッシュ時間は、エントリの親ディレクトリの属性キャッシュの時間と同じです。

all の場合、エントリが存在したことと、エントリが存在しなかったことの両方をキャッシュします。

pos または positive の場合、エントリが存在したことのみをキャッシュします。過去に存在しなかったエントリに対しては、常に IO サーバへエントリが存在するかどうかの検索が行われます。

none の場合、エントリが存在したことと、エントリが存在しなかったことのいずれもキャッシュしません。 常に IO サーバへエントリが存在するかどうかの検索が行われます。 他のクライアントで作成されたまたは削除されたエントリを素早く検出することができますが、アプリケーションや IO サーバのパフォーマンスに影響を与えます。

ScaTeFS のアンマウント

ScaTeFS をアンマウントする場合、umount コマンドにマウントポイント、もしくはサーバ名:ファイルシステム名[/サブディレクトリ名]を指定します。下記のオプションが有効となります。

-f　　強制的にアンマウントします。

ファイル

/etc/fstab　　ファイルシステム一覧

/etc/mtab　　マウントされたファイルシステム一覧

関連項目

fstab , mount , umount

4.1.1.2　scatefs_stat

名前

scatefs_stat - ScaTeFS 統計情報の表示

書式

scatefs_stat [-h {hostname|address}|-s serverid -p [-t time]]

説明

scatefs_stat は、NEC Scalable Technology File System(ScaTeFS)および リモートプロシジャコール(RPC)の統計情報を表示します。オプションを省略すると、すべてのサーバの統計情報を表示します。

-h {hostname|address}　　指定したホスト名または IP アドレスを持つサーバの統計情報を表示します。

-s serverid　　指定したサーバ ID を持つサーバの統計情報を表示します。

-p　　プロシージャの統計情報を表示します。

-t time　　time で指定した時間(秒単位)に発行したプロシージャの統計情報を表示します。
-p オプションを必ず指定してください。
time は 5 から INT_MAX の範囲で指定してください。

注意

本コマンドは TCP のプロトコルのみ対応しています。

関連ファイル

/proc/net/scaterpc/scatefs

関連項目

scatefs_premap , scatefs_setdirattr , scatefs_getfinfo , scatefs_rcli

診断

- 処理が正常終了すると、0 を返します
- 異常終了した場合は、0 以外の値を返します。

4.1.1.3　scatefs_rcli

名前

scatefs_rcli – IO サーバの ScaTeFS コマンドのリモート実行

書式

scatefs_rcli server command [args..]

server　　リモートコマンドを実行する IO サーバを指定します。

command　サブコマンドを指定します。

args　　各サブコマンドに対応した引数を指定します。

[サブコマンド一覧]

1. df (ScaTeFS の使用状況表示)

   df サブコマンドには以下の引数を指定することが可能です。

   df [-i|-D] [-h|-H]

   -i　　inode の使用状況を表示します。

   -D　　ディレクトリとして作成可能な inode の使用状況を表示します。

   -h　　それぞれのサイズに、たとえばメガバイトなら M のようなサイズ文字を付加します。
   サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)があります。
   10 の累乗ではなく 2 の累乗を用いるので、M は 1,048,576 バイトを表します。

-H　　それぞれのサイズについて、たとえばメガバイトなら M といったサイズ文字を付加します。
サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)があります。
2 の累乗ではなく 10 の累乗を用いるので、M は 1,000,000 バイトを表します。

2. detail (ScaTeFS の構成情報表示)

detail サブコマンドには以下の引数を指定することが可能です。

detail {-f|-s|-t} [id]

detail -f id

-f　　ファイルシステムの情報を表示します。
id を省略した場合はシステムに属するファイルシステム全体の情報を表示します。

-s　　サーバの情報を表示します。
id を省略した場合はシステムに属する IO サーバ全体の情報を表示します。

-t　　IO ターゲットの情報を表示します。
id を省略した場合はシステムに属する IO ターゲット全体の情報を表示します。

3. logcollect (IO サーバのログ表示)

logcollect サブコマンドには以下の引数を指定することが可能です。

logcollect {-n node[,node..]|-a} [-m]

-n node[,node..]　　指定した IO サーバ ID のログを表示します。
カンマ区切りで複数の ID を指定することが可能です。

-a　　すべての IO サーバを対象とします。

-m　　すべてのログを出力します。

4. quota (ScaTeFS の QUOTA 情報の表示)

デフォルトでは容量(ブロック数)の使用量とリミットをキロバイト(1024 バイト)単位で表示します。

quota サブコマンドには以下の引数を指定することが可能です。

quota [-sq] [-u uid|username] [-g gid|groupname] [-d dirid] [fsname]

quota [-sq] -d dirname [fsname]

-s	それぞれのサイズにサイズ文字を付加します。 サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)、P(ペタ)、E(エクサ)があります。 ブロック数は 2 の累乗を用いるので、たとえば M は 1,048,576 バイトを表します。 ファイル数は 10 の累乗を用いるので、たとえば M は 1,000,000 ファイルを表します。
-q	より簡潔なメッセージ(使用量が QUOTA 制限を超過しているファイルシステムの情報だけ)を表示します。
-u uid|username	指定したユーザ ID またはユーザ名の QUOTA 情報を表示します。 -u および-g および-d オプションを省略した場合は実行ユーザの quota を取得します。 一般ユーザの場合は、自分のユーザ ID、ユーザ名が指定可能です。

-g gid|groupname	指定したグループ ID またはグループ名の QUOTA 情報を表示します。 一般ユーザの場合は、自分が所属するグループ ID、グループ名が指定可能です。
-d dirid|dirname	指定したディレクトリ ID またはディレクトリ名の QUOTA 情報を表示します。 指定するディレクトリ名は ScaTeFS のトップディレクトリから始まるパス名とします。 数字を含むディレクトリ名には、先頭に「/」を追加します。 表示するディレクトリ名は、scatefs_mkqdir で作成または変更したパス名とします。
fsname	ファイルシステムの QUOTA 情報を表示します。 省略した場合は、すべてのファイルシステムの QUOTA 情報を表示します。

5. repquota (ScaTeFS の QUOTA 情報の一覧表示)

各ユーザーの現在のファイル数と使用容量(キロバイト(1024 バイト)単位)を、edquota で設定した値とともに表示します。

repquota サブコマンドには以下の引数を指定することが可能です。

repquota [-sbug] [-d [-n|v] [-t N]] [fsname]

-s	それぞれのサイズにサイズ文字を付加します。 サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)、P(ペタ)、E(エクサ)があります。 ブロック数は 2 の累乗を用いるので、たとえば M は 1,048,576 バイトを表します。 ファイル数は 10 の累乗を用いるので、たとえば M は 1,000,000 ファイルを表します。 -b が指定された場合はサイズ文字の付加は無視します。

-b　設定済み QUOTA 情報のバックアップファイルを作成します。
バックアップファイルは、ハードリミット、ソフトリミット、猶予時間情報を格納します。
バックアップファイル名には fsid と sgid を含みます。

-u　ユーザの QUOTA 情報を表示します。
-u,-g,-d が指定されない場合はユーザの QUOTA 情報を表示します。

-g　グループの QUOTA 情報を表示します。

-d　ディレクトリの QUOTA 情報を表示します。

-n　情報としてディレクトリ ID を表示します。

-v　情報としてディレクトリ ID とディレクトリ名を表示します。
-n、-v が指定されない場合はディレクトリ名を表示します。
ディレクトリ名は、scatefs_mkqdir で作成または変更したパス名を ScaTeFS のルートディレクトリを表す「/」を含まずに表示します。

-t N　ディレクトリ名の表示領域をシングルバイト文字で N 文字数分とします。
省略した場合はシングルバイト文字で 16 文字数分とします。
指定内容に関わらず、ディレクトリ名をフルパスで表示します。
-n が指定された場合は無効です。

fsname　ファイルシステムの QUOTA 情報を表示します。
省略した場合は、すべてのファイルシステムの QUOTA 情報を表示します。

ID 変換使用例

scatefs_repqname を使用することで、repquota の uid/gid をユーザ名/グループ名に変換することが可能です。
以下は、最もシンプルな例です。scatefs_repqname は内部的に repquota (-u,-g,-s オプション)を呼び出し、その出力を username/groupname に変換し

ます。

scatefs_repqname server

以下は、repquota の出力を scatefs_repqname で変換する例です。

scatefs_rcli server repquota | scatefs_repqname

6. edquota (ScaTeFS の QUOTA 情報の編集)

edquota サブコマンドには以下の引数を指定することが可能です。

edquota {-u uid|-g gid|-d dirid} {-b [SOFTLIMIT:]HARDLIMIT|-i [SOFTLIMIT:]HARDLIMIT} fsname

edquota -d dirname {-b [SOFTLIMIT:]HARDLIMIT|-i [SOFTLIMIT:]HARDLIMIT} fsname

edquota -T {-u uid|-g gid |-d dirid} {-b BLOCKTIME|-i INODETIME} fsname

edquota -T -d dirname {-b BLOCKTIME|-i INODETIME} fsname

edquota -t {u|g|d} {-b BLOCKPERIOD|-i INODEPERIOD} fsname

-u uid	指定したユーザ ID の QUOTA 情報を編集します。
-g gid	指定したグループ ID の QUOTA 情報を編集します。
-d dirid	指定したディレクトリ ID の QUOTA 情報を編集します。
-d dirname	指定したディレクトリ名の QUOTA 情報を編集します。 ディレクトリ名はScaTeFSのトップディレクトリから始まるパス名で指定します。 数字を含むディレクトリ名には、先頭に「/」を追加します。

-b [SOFTLIMIT:]HARDLIMIT	-T, -t オプションが省略された場合に、ディスク容量のソフトリミット/ハードリミット(バイト)を設定します。 SOFTLIMIT を省略した場合は、HARDLIMIT をソフトリミットとハードリミットに設定します
-i [SOFTLIMIT:]HARDLIMIT	-T, -t オプションが省略された場合に、inode 数のソフトリミット/ハードリミットを設定します。 SOFTLIMIT を省略した場合は、HARDLIMIT をソフトリミットとハードリミットに設定します。
-T	ディスク容量、inode 数のソフトリミット超過で設定された猶予時間を個々に変更します。
-b BLOCKTIME	ディスク容量の猶予時間 BLOCKTIME を秒単位で設定します。
-i INODETIME	inode 数の猶予時間 INODETIME を秒単位で設定します。
-t {u|g|d}	ディスク容量、inode 数のユーザ(u)、グループ(g)、ディレクトリ(d)に対する猶予時間を設定します。
-b BLOCKPERIOD	ディスク容量の猶予時間 BLOCKPERIOD を秒単位で設定します。
-i INODEPERIOD	inode 数の猶予時間 INODEPERIOD を秒単位で設定します。
fsname	指定したファイルシステムの設定を変更します。

7. ifstat (IO サーバのインターフェース状態の表示)

   デフォルトではすべてのインターフェース状態を表示します。

   IO サーバデーモンが 1 週間以内に core ファイルを出力していた場合、警告を表示します。

   ifstat サブコマンドには以下の引数を指定することが可能です。

   ifstat {-n node[,node...]|-a} [-if]

-n node[,node...]	指定した IO サーバ ID の情報を表示します。 カンマ区切りで複数の IO サーバ ID を指定することが可能です。
-a	すべての IO サーバを対象とします。
-i	フローティング IP アドレスの状態を表示します。
-f	FC パスの状態を表示します。

8. mkqdir (ScaTeFS の QUOTA 対応ディレクトリの作成)

mkqdir サブコマンドには以下の引数を指定することが可能です。

mkqdir [--dirid id] [--mode nnn] [--uid id] [--gid id] [--chunksize size] [--stripesize size] fs dirname

mkqdir -c fs dirname

fs	ScaTeFS のファイルシステム名もしくはファイルシステム ID を指定します。
dirname	作成するディレクトリ名を ScaTeFS のトップディレクトリから始まるパス名で指定します。 -c オプションを指定してパス情報を修正する場合は、現状のパス名を指定します。
-c	dirname で指定した作成済みのディレクトリ QUOTA に対応するディレクトリについて、scatefs_quota,scatefs_repquota で表示するパス名を修正します。

--dirid id　作成するディレクトリに対応するディレクトリ ID を指定します。
id には 1 から 4,294,967,295 の整数を指定します。
--dirid の指定が無い場合は自動でユニークな値を設定します。

--mode nnn　パーミッションを 3 桁の 8 進数で指定します。

--uid id　クライアントのユーザ ID を指定します。

--gid id　クライアントのグループ ID を指定します。

--chunksize size　チャンクサイズを指定します。
省略した場合はデフォルト値を設定します。
デフォルト値は、ノンストライプフォーマット場合は 256 メガバイト、ストライプフォーマットの場合は 1 ギガバイトです。

チャンクサイズは 4K(4096) の倍数の値で指定する必要があります。
サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト) です。
たとえば、--chunksize 268435456 と --chunksize 256m と --chunksize 256M はいずれも同じ指定となります。

ストライプフォーマットの場合、chunksize オプションと stripesize オプションの両方を指定します。
ノンストライプフォーマットの場合、chunksize オプションのみを指定します。
ノンストライプフォーマットの場合、ストライプサイズとチャンクサイズは同じ値になります。

--stripesize　ストライプサイズを指定します。

size

ストライプサイズは 4K(4096) の倍数の値で指定する必要があります。
サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト) です。
たとえば、--stripesize 4194304 と --stripesize 4m と --stripesize 4M はいずれも同じ指定となります。

ストライプフォーマットの場合、チャンクサイズはストライプサイズの倍数であり、かつストライプサイズより大きい値である必要があります。

9. rmqdir (ScaTeFS の QUOTA 対応ディレクトリの削除)

rmqdir サブコマンドには以下の引数を指定することが可能です。

rmqdir [-f] fs dirname

fs	ScaTeFS のファイルシステム名もしくはファイルシステム ID を指定します。
dirname	削除するディレクトリを ScaTeFS のトップディレクトリから始まるパス名で指定します。
-f	QUOTA に対応するディレクトリとして登録されていない場合でも強制で削除します。

説明

1. df
   ScaTeFS の使用状況を表示します。
2. detail
   ScaTeFS の構成情報を表示します。特権ユーザのみが実行できます。
3. logcollect
   IO サーバのログを表示します。特権ユーザのみが実行できます。
4. quota
   ScaTeFS の QUOTA 情報を表示します。
5. repquota

ScaTeFS の QUOTA 情報一覧を表示します。特権ユーザのみが実行できます。

6. edquota

   ScaTeFS の QUOTA 情報を編集します。特権ユーザのみが実行できます。

7. ifstat

   IO サーバのインターフェース状態を表示します。特権ユーザのみが実行できます。

8. mkqdir

   ScaTeFS の QUOTA 対応ディレクトリを作成します。特権ユーザのみが実行できます。

9. rmqdir

   ScaTeFS の QUOTA 対応ディレクトリを削除します。特権ユーザのみが実行できます。

関連項目

scatefs_rcliadm , scatefs_df , scatefs_detail , scatefs_logcollect , scatefs_quota , scatefs_repquota , scatefs_edquota , scatefs_ifstat , scatefs_mkqdir , scatefs_rmqdir

注意事項

- scatefs_rcli を使用するためには、IO サーバのコマンド scatefs_rcliadm で scatefs_rcli の実行を許可するクライアント、ユーザの登録が必要になります。
- scatefs_rcli を連続実行すると、エラーになる場合があります。 その場合、1 分程度待ってから再実行して下さい。

4.1.1.4　scatefs_rebalance_import

名前

scatefs_rebalance_import - リバランス対象ファイルのインポート

書式

scatefs_rebalance_import host fsid infile

説明

scatefs_rebalance_import は、リバランス対象ファイルを IO サーバにインポートします。リバランス機能の詳細は「3.2.7 リバランス」を参照してください。

オプション

host	ルート IO サーバの IP アドレスを指定します。
fsid	対象となるファイルシステム ID を指定します。

infile　　リバランス対象ファイル群のフルパスが記述されたファイルを指定します。

[infile の例]

```
$ cat infile
/mnt/scatefs/file1
/mnt/scatefs/file2
/mnt/scatefs/file3
/mnt/scatefs/file4
```

関連項目

scatefs_rcli

注意事項

scatefs_rebalance_import を使用するためには、IO サーバのコマンド scatefs_rcliadm で scatefs_rcli の実行を許可するクライアント、ユーザの登録が必要になります。

4.1.1.5　scatefs_check

名前

scatefs_check - IO サーバまでのネットワーク状態の診断

書式

scatefs_check [-t count] [-o options] server:fsname

説明

scatefs_check は、NEC Scalable Technology File System (ScaTeFS) に対応したリモートプロシジャコール(RPC)を IO サーバに送信し、送信時間を表示します。ScaTeFS をマウントしていない状態で RPC を送信することが可能です。server:fsname には IO サーバとファイルシステム名を指定します。

オプション

-t count　　RPC 発行回数を指定します。送信時間の平均を表示します。
-t オプションを指定しない場合、RPC 発行回数のデフォルトは 1 回です。

-o options　rsize=n　サーバからデータを読み込む際に用いるバッファのバイト数を指定します。
デフォルト値は 1048576 バイトです。
単位には k/K(キロバイト), m/M(メガバイト) が指定可能です。
たとえば、rsize=4194304 と rsize=4M と rsize=4m はいずれも同じ指定になります。

wsize=n　サーバにデータを書き込む際に用いるバッファのバイト数を指定します。
デフォルト値は 1048576 バイトです。
単位には k/K(キロバイト), m/M(メガバイト) が指定可能です。
たとえば、wsize=4194304 と wsize=4M と wsize=4m はいずれも同じ指定になります。

関連項目

scatefs

注意事項

scatefs_check は root アカウントで実行して下さい。

4.1.2 一般利用者向け

以下、一般利用者向けのコマンドです。

4.1.2.1 scatefs_setdirattr

名前

scatefs_setdirattr - ScaTeFS ディレクトリの属性設定

書式

1. stripe format
   scatefs_setdirattr -s size [-c size] dirpath
2. non stripe format
   scatefs_setdirattr -c size dirpath

-s size　ストライプサイズを指定

-c size　チャンクサイズを指定

dirpath　ディレクトリパスを指定

説明

scatefs_setdirattr は、dirpath に指定されたディレクトリに対し属性(ストライプサイズ/チャンクサイズ)を設定します。

設定後、dirpath 配下に新規作成されるディレクトリ/ファイルに属性が反映されます。

dirpath 配下の既存ディレクトリ/ファイルには反映されません。

1. stripe format

   ストライプサイズ(-s オプション)の指定が必要です。

   チャンクサイズ(-c オプション)の指定も可能であり、指定が無い場合はデフォルト値(1G バイト)に設定します。

   チャンクサイズ、ストライプサイズは 4K(4096)の倍数の値を指定する必要があります。

   チャンクサイズはストライプサイズの倍数であり、かつストライプサイズより大きい値である必要があります。

   サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト), t/T(テラバイト) です。

   たとえば、-s 4194304 と -s 4m と -s 4M はいずれも同じ指定となります。

2. non stripe format

   チャンクサイズ(-c オプション)の指定が必要です。

   チャンクサイズは 4K(4096)の倍数の値を指定する必要があります。

   サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト), t/T(テラバイト) です。

   たとえば、-c 268435456 と -c 256m と -c 256M はいずれも同じ指定となります。

関連項目

scatefs_premap , scatefs_getfinfo , scatefs_rcli , scatefs_stat

注意事項

本コマンドは指定されたディレクトリの作成は行わないため、dirpath には存在するディレクトリを指定する必要があります。

4.1.2.2　scatefs_premap

名前

scatefs_premap - ScaTeFS ファイルのプリマップ

書式

1. stripe format

   scatefs_premap -s size [-c size] filesize filepath
2. non stripe format

   scatefs_premap -c size filesize filepath
3. default

   scatefs_premap filesize filepath

-s size	ストライプサイズを指定
-c size	チャンクサイズを指定
filesize	プリマップするファイルサイズを指定
filepath	ファイルパスを指定

説明

scatefs_premap は、filepath に指定されたファイルに対し、filesize で指定されたサイズ分、ファイルのプリマップを行います。 本コマンドでファイルを作成することにより、大規模な ScaTeFS(多数ノード)での並列 I/O を実行するような場合、ファイルへの書き込み効率化が期待できます。

オプションには、以下 3 種の指定方法があります。

1. stripe format

   ストライプサイズ(-s オプション)の指定が必要です。

   チャンクサイズ(-c オプション)の指定も可能であり、指定が無い場合はデフォルト値(1G バイト)に設定します。

   チャンクサイズ、ストライプサイズは 4K(4096)の倍数の値を指定する必要があります。

   チャンクサイズはストライプサイズの倍数であり、かつストライプサイズより大きい値である必要があります。

   サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト), t/T(テラバイト) です。

   たとえば、-s 4194304 と -s 4m と -s 4M はいずれも同じ指定となります。
2. non stripe format

   チャンクサイズ(-c オプション)の指定が必要です。

   チャンクサイズは 4K(4096)の倍数の値を指定する必要があります。

   サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト), t/T(テラバイト) です。

   たとえば、-c 268435456 と -c 256m と -c 256M はいずれも同じ指定となりま

す。

3. default
   ストライプサイズ(-s　オプション)、チャンクサイズ(-c　オプション)の　指定が無い場合、ファイルの親ディレクトリが持つチャンクサイズ、ストライプサイズの値を引き継ぎます。

関連項目

scatefs_setdirattr , scatefs_getfinfo , scatefs_rcli , scatefs_stat

注意事項

- 本コマンドはレギュラーファイルのみを対象としています。
- 指定されたファイルが存在しない場合は作成し、既に存在する場合はファイルサイズが 0 の状態に限り実行可能です。

4.1.2.3　scatefs_getfinfo

名前

scatefs_getfinfo - ScaTeFS ファイル/ディレクトリの情報取得

書式

scatefs_getfinfo [-h|-H] [-v] path

-h	それぞれのサイズについて、たとえばメガバイトなら M のようなサイズ文字を付加します。 サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)、P(ペタ)があります。 10 の累乗ではなく 2 の累乗を用いるので、 M は 1,048,576 バイトを表します。
-H	それぞれのサイズについて、たとえばメガバイトなら M といったサイズ文字を付加します。 サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)、P(ペタ)があります。 2 の累乗ではなく 10 の累乗を用いるので、M は 1,000,000 バイトを表します。
-v	ファイルの構成情報をオフセットごとに表示します。
path	ファイル/ディレクトリのパスを指定します。

説明

scatefs_getfinfo は、path に指定された ファイル/ディレクトリの情報を取得します。

ファイルを指定した場合、ファイルフォーマット、IO ターゲット数、チャンクサイズ、ストライプサイズ、ファイルサイズを出力します。ディレクトリを指定した場合、ファイルフォーマット、チャンクサイズ、ストライプサイズを出力します。

-v オプションはファイルにのみ対応しています。

関連項目

scatefs_premap , scatefs_setdirattr , scatefs_rcli , scatefs_stat

注意事項

本コマンドはレギュラーファイル/ディレクトリのみを対象としています。

4.2　IOサーバ

IOサーバ上で使用可能なコマンドを記載する。

4.2.1 管理者向け

IOサーバ上で使用可能な管理者向けのコマンドの詳細を以下に記載する。

4.2.1.1　scatefs_df

名前

scatefs_df - ScaTeFS の使用状況表示

書式

scatefs_df [-i|-D] [-h|-H] {fsid|fsname}

説明

scatefs_df は NEC Scalable Technology File System(ScaTeFS)の使用状況を表示します。

オプション

-i	node の使用状況を表示します。
-D	ディレクトリとして作成可能な inode の使用状況を表示します。
-h	それぞれのサイズに、たとえばメガバイトなら M のようなサイズ文字を付加します。 サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)があります。10 の累乗ではなく 2 の累乗を用いるので、M は 1,048,576 バイトを表します。

-H　　それぞれのサイズについて、たとえばメガバイトなら M といったサイズ文字を付加します。
サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)があります。
2 の累乗ではなく 10 の累乗を用いるので、M は 1,000,000 バイトを表します。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_detail , scatefs_statcollect , scatefs_logcollect

4.2.1.2　scatefs_quota

名前

scatefs_quota - ScaTeFS の QUOTA 情報の表示

書式

scatefs_quota [-sq] {-u uid|-g gid|-d dirid} [fsname..]

scatefs_quota [-sq] -d dirname [fsname]

説明

scatefs_quota は NEC Scalable Technology File System(ScaTeFS)の QUOTA 情報を表示します。デフォルトでは容量(ブロック数)の使用量とリミットをキロバイト(1024 バイト)単位で表示します。

オプション

-s　　それぞれのサイズにサイズ文字を付加します。
サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)、P(ペタ)、E(エクサ)があります。
ブロック数は 2 の累乗を用いるので、たとえば M は 1,048,576 バイトを表します。
ファイル数は 10 の累乗を用いるので、たとえば M は 1,000,000 ファイルを表します。

-q　　より簡潔なメッセージ(使用量が QUOTA 制限を超過しているファイルシステムの情報だけ)を表示します。

-s	それぞれのサイズにサイズ文字を付加します。 サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)、P(ペタ)、E(エクサ)があります。 ブロック数は 2 の累乗を用いるので、たとえば M は 1,048,576 バイトを表します。 ファイル数は 10 の累乗を用いるので、たとえば M は 1,000,000 ファイルを表します。
-u uid	指定したユーザ ID の QUOTA 情報を表示します。
-g gid	指定したグループ ID の QUOTA 情報を表示します。
-d dirid	指定したディレクトリ ID の QUOTA 情報を表示します。
-d dirname	指定したディレクトリ名の QUOTA 情報を表示します。 ディレクトリ名は ScaTeFS のトップディレクトリから始まるパス名とします。 数字を含むディレクトリ名には、先頭に「/」を追加します。 表示するディレクトリ名は、scatefs_mkqdir で作成または変更したパス名とします。
fsname	指定したファイルシステムの QUOTA 情報を表示します。 省略した場合は、すべてのファイルシステムの QUOTA 情報を表示します。

ファイル

/etc/scatefs/system.info	ScaTeFS の情報ファイル
/etc/scatefs/diridtab	ディレクトリ QUOTA の情報ファイル

関連項目

scatefs_repquota , scatefs_edquota , scatefs_quotacheck ,
scatefs_mkqdir , scatefs_rmqdir

4.2.1.3　scatefs_addios

名前

scatefs_addios - IO サーバの登録

書式

scatefs_addios -f datafile

説明

scatefs_addios はノードを NEC Scalable Technology File System(ScaTeFS) の IO サーバとしてシステムに登録します。

オプション

-f datafile　　IO サーバの情報を記載した datafile を指定します。

データファイル

datafile には以下のエントリを記載します。

ipaddr	運用・管理ポートの IP アドレスを指定します。 ipaddr は始めのエントリとして記載する必要があります。
fipaddr	ファイルシステムポート(10GbE)の IP アドレスを指定します。 複数登録する場合はスペース区切りで記載します。
iftypes	fipaddr のインターフェースタイプを指定します。 10GbE の場合は 1 を指定します。 iftypes は fipaddr と同じ数だけスペース区切りで記載します。 iftypes は省略可能であり、省略した場合のデフォルトは 10GbE になります。
inipaddr	IO サーバ間インタコネクト用ポートの IP アドレスを指定します。 inipaddr を使用しない場合は省略可能です。
cport	クライアント接続ポート番号を指定します。 cport は省略可能であり、省略した場合のデフォルトは 50000 になります。
sport	サーバ間通信接続ポート番号を指定します。 sport は省略可能であり、省略した場合のデフォルトは 50001 になります。
cdport	データ転送用クライアント接続ポート番号を指定します。 cdport は省略可能であり、省略した場合のデフォルトは 50002 になります。

以下に、2 つの IO サーバを追加する場合の datafile の例を記載します。

[例 1] inipaddr, cport, sport, cdport を省略した場合

```
ipaddr      192.168.x.1
fipaddr     172.10.y.0 172.10.y.1

ipaddr      192.168.x.2
fipaddr     172.10.y.2 172.10.y.3
```

[例 2] inipaddr, cport, sport, cdport を省略しない場合

```
ipaddr      192.168.x.1
fipaddr     172.10.y.0 172.10.y.1
inipaddr    10.2.z.0
cport       50000
sport       50001
cdport      50002

ipaddr      192.168.x.2
fipaddr     172.10.y.2 172.10.y.3
inipaddr    10.2.z.1
cport       50000
sport       50001
cdport      50002
```

ファイル

/etc/scatefs/system.info	ScaTeFS の情報ファイル
/etc/scatefs/server.info	IO サーバの設定ファイル

関連項目

scatefs_addiot , scatefs_mkfs , scatefs_extendfs , scatefs_detail

注意事項

- scatefs_addios は fsadmin アカウントで実行して下さい。
- scatefs_addios は 1 台の IO サーバで実行して下さい。
  1 回の実行によりすべての IO サーバに情報が設定されます。
- scatefs_detail の -s オプションで登録した IO サーバを確認できます。

4.2.1.4　scatefs_addiot

名前

scatefs_addiot - IO ターゲットの登録

書式

scatefs_addiot -f datafile

説明

scatefs_addiot はデータストアを NEC Scalable Technology File System(ScaTeFS) の IO ターゲットとしてシステムに登録します。

scatefs_addiot を実行する前提として、IO ターゲットを持つ IO サーバの情報を scatefs_addios で登録する必要があります。

オプション

-f datafile	IO ターゲットの情報を記載した datafile を指定します。

データファイル

datafile には以下のエントリを記載します。

iosid	IO サーバ ID を指定します。 iosid は始めのエントリとして記載する必要があります。 IO サーバ ID は、scatefs_detail -s で確認できます。
data	データ領域に使用するデバイス名を指定します。
ctrl	メタデータ領域に使用するデバイス名を指定します。

以下に、8 つの IO ターゲットを追加する場合の datafile の例を記載します。

[例]

```
iosid       0
data        /dev/vg_data00/lv_data00
ctrl        /dev/vg_ctrl00/lv_ctrl00
data        /dev/vg_data01/lv_data01
ctrl        /dev/vg_ctrl01/lv_ctrl01

iosid       1
data        /dev/vg_data02/lv_data02
ctrl        /dev/vg_ctrl02/lv_ctrl02
data        /dev/vg_data03/lv_data03
ctrl        /dev/vg_ctrl03/lv_ctrl03

iosid       2
data        /dev/vg_data04/lv_data04
ctrl        /dev/vg_ctrl04/lv_ctrl04
data        /dev/vg_data05/lv_data05
ctrl        /dev/vg_ctrl05/lv_ctrl05

iosid       3
data        /dev/vg_data06/lv_data06
ctrl        /dev/vg_ctrl06/lv_ctrl06
data        /dev/vg_data07/lv_data07
ctrl        /dev/vg_ctrl07/lv_ctrl07
```

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_addios , scatefs_mkfs , scatefs_extendfs , scatefs_detail

注意事項

- scatefs_addiot は fsadmin アカウントで実行して下さい。
- scatefs_addiot は 1 台の IO サーバで実行して下さい。
  1 回の実行によりすべての IO サーバに情報が設定されます。
- scatefs_detail の -t オプションで登録した IO ターゲットを確認できます。

4.2.1.5　scatefs_admin

名前

scatefs_admin - ScaTeFS のシステムファイルの管理

書式

scatefs_admin {--trans {iosid|all} | --check | --rollback {iosid|all}} {system|tune|diridtab|all}

scatefs_admin –serverinfo

scatefs_admin --create tune

説明

scatefs_admin は NEC Scalable Technology File System(ScaTeFS)のシステムファイルの管理を行います。

オプション

--trans {iosid|all}　指定した IO サーバ、またはすべての IO サーバ(all)にシステムファイルを転送します。
system を指定した場合は ScaTeFS の情報ファイル(system.info)を転送します。
tune を指定した場合は IO サーバデーモンのチューニングパラメータ設定ファイル(scatefssrv.conf)を転送します。
diridtab を指定した場合はディレクトリ ID とディレクトリの対応一覧(diridtab)を転送します。
all を指定した場合は 4 つのファイルを転送します。

--check　すべての IO サーバのシステムファイルを確認します。
system を指定した場合は system.info を確認します。
tune を指定した場合は scatefssrv.conf を確認します。
diridtab を指定した場合は diridtab を確認します。
all を指定した場合は 4 つのファイルを確認します。

--rollback {iosid|all}	指定した IO サーバ、またはすべての IO サーバ(all)のシステムファイルを 1 世代前のものにロールバックします。 system を指定した場合は system.info をロールバックします。 tune を指定した場合は scatefssrv.conf をロールバックします。 diridtab を指定した場合は diridtab をロールバックします。 all を指定した場合は 4 つのファイルをロールバックします。
--serverinfo	すべての IO サーバの設定ファイル(server.info)を更新します。
--create tune	tune を指定することで、scatefssrv.conf を作成します。 scatefssrv.conf にはデフォルト値を設定します。

ファイル

/etc/scatefs/system.info	ScaTeFS の情報ファイル
/etc/scatefs/server.info	IO サーバの設定ファイル
/etc/scatefs/scatefssrv.conf	IO サーバデーモンのチューニングパラメータ設定ファイル
/etc/scatefs/diridtab	ディレクトリ QUOTA の情報ファイル

関連項目

scatefs_detail , scatefs_rcliadm

注意事項

- system.info を対象に転送またはロールバックを実行する場合、IO サーバデーモンを停止して下さい。
- scatefs_admin は fsadmin アカウントで実行して下さい。
- scatefs_admin を連続実行すると、エラーになる場合があります。
  その場合、1 分程度待ってから再実行して下さい。

4.2.1.6　scatefs_detail

名前

scatefs_detail - ScaTeFS の構成情報表示

書式

scatefs_detail {-f|-s|-t} [id]

説明

scatefs_detail は NEC Scalable Technology File System(ScaTeFS)の構成情報を表示します。

オプション

-f	ファイルシステムの情報を表示します。 id を省略した場合は、システムに属するファイルシステム全体の情報を表示します。
-s	IO サーバの情報を表示します。 id を省略した場合は、システムに属する IO サーバ全体の情報を表示します。
-t	IO ターゲットの情報を表示します。 id を省略した場合はシステムに属する IO ターゲット全体の情報を表示します。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_df , scatefs_logcollect , scatefs_statcollect , scatefs_admin

4.2.1.7　scatefs_edquota

名前

scatefs_edquota - ScaTeFS の QUOTA 情報の編集

書式

scatefs_edquota {-u uid|-g gid|-d dirid} [-b [SOFTLIMIT:]HARDLIMIT] [-i [SOFTLIMIT:]HARDLIMIT] fsname

scatefs_edquota -d dirname [-b [SOFTLIMIT:]HARDLIMIT] [-i [SOFTLIMIT:]HARDLIMIT] fsname

scatefs_edquota -T {-u uid|-g gid|-d dirid} [-b BLOCKTIME] [-i INODETIME] fsname

scatefs_edquota -T -d dirname [-b BLOCKTIME] [-i INODETIME] fsname

scatefs_edquota -t {u|g|d} [-b BLOCKPERIOD] [-i INODEPERIOD] fsname

説明

scatefs_edquota は NEC Scalable Technology File System(ScaTeFS)の QUOTA 情報を編集します。以下の設定が可能です。

- ディスク容量、inode 数のソフトリミット/ハードリミット。
- ディスク容量、inode 数のソフトリミット超過で設定される猶予時間。
- ユーザ(u)、グループ(g)、ディレクトリ(d)ごとにディスク容量、inode 数のソフトリミット超過で設定される猶予時間。

-b および-i オプションを指定しない場合、環境変数 EDITOR または VISUAL で指定されたエディタを起動します。

エディタを終了すると編集内容を QUOTA 情報に反映します。

オプション

-u uid	指定したユーザ ID の QUOTA 情報を編集します。
-g gid	指定したグループ ID の QUOTA 情報を編集します。
-d dirid	指定したディレクトリ ID の QUOTA 情報を編集します。
-d dirname	指定したディレクトリ名の QUOTA 情報を編集します。 ディレクトリ名は ScaTeFS のトップディレクトリから始まるパス名で指定します。 数字を含むディレクトリ名には、先頭に「/」を追加します。 エディタ起動時には、scatefs_mkqdir で作成または変更したディレクトリ名を含むディレクトリの QUOTA 情報を表示します。
-b [SOFTLIMIT:]HARDLIMIT	-T,-t オプションが省略された場合、ディスク容量のソフトリミット/ハードリミット(バイト)を設定します。 SOFTLIMIT を省略した場合は、HARDLIMIT をソフトリミットとハードリミットに設定します。

-i [SOFTLIMIT:]HARDLIMIT	-T,-t オプションが省略された場合に、inode 数のソフトリミット/ハードリミットを設定します。SOFTLIMIT を省略した場合は、HARDLIMIT をソフトリミットとハードリミットに設定します。
-T	ディスク容量､inode 数のソフトリミット超過で設定された猶予時間を個々に変更します。
-b BLOCKTIME	ディスク容量の猶予時間 BLOCKTIME を秒単位で設定します。
-i INODETIME	inode 数の猶予時間 INODETIME を秒単位で設定します。
-t {u|g|d}	ディスク容量、inode 数のユーザ(u)、グループ(g)、ディレクトリ(d)に対する猶予時間を設定します。
-b BLOCKPERIOD	ディスク容量の猶予時間 BLOCKPERIOD を秒単位で設定します。
-i INODEPERIOD	inode 数の猶予時間 INODEPERIOD を秒単位で設定します。
fsname	指定したファイルシステムの設定を変更します。

ファイル

/etc/scatefs/system.info	ScaTeFS の情報ファイル
/etc/scatefs/diridtab	ディレクトリ QUOTA の情報ファイル

関連項目

scatefs_quota , scatefs_repquota , scatefs_quotacheck ,
scatefs_mkqdir , scatefs_rmqdir

4.2.1.8　scatefs_extendfs

名前

scatefs_extendfs - ScaTeFS の拡張

書式

scatefs_extendfs -f datafile [--force]

説明

scatefs_extendfs は NEC Scalable Technology File System(ScaTeFS)の拡張を行います。

オプション

-f datafile　　追加する IO ターゲットの情報を記載した datafile を指定します。

--force　　強制実行します。

データファイル

オプション指定により、datafile に記載する情報が異なります。

1. ファイルシステムの拡張(--addsg, --addiot の指定無し)

fsid　　拡張対象のファイルシステム ID を指定します。
datafile に指定できるファイルシステム ID は 1 つです。

addiotid　　追加する IO ターゲット ID をスペース区切りで指定します。
IO ターゲット ID は、scatefs_detail -t で確認できます。

[例] ファイルシステム ID 0 を拡張

```
fsid        0
addiotid    3 4
```

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_addios , scatefs_addiot , scatefs_mkfs , scatefs_admin

注意事項

- scatefs_extendfs は IO サーバデーモンを停止してから実行して下さい。
- scatefs_extendfs は fsadmin アカウントで実行して下さい。
- scatefs_extendfs は 1 台の IO サーバで実行して下さい。
  1 回の実行によりすべての IO サーバに情報が設定されます。
- scatefs_detail で、拡張したファイルシステムの情報を確認できます。

4.2.1.9　scatefs_f2fsck

名前

scatefs_f2fsck - ScaTeFS のチェックと修復（ファイル指定）

書式

scatefs_f2fsck [-n] infile

説明

scatefs_f2fsck は指定されたファイルに対して NEC Scalable Technology File System(ScaTeFS)のチェックと修復を行います。infile には scatefs_fsck の実行結果ファイルを指定します。

オプション

-n　　ファイルシステムのチェックのみを行い、修復を行いません。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_fsck

4.2.1.10 scatefs_fsck

名前

scatefs_fsck - ScaTeFS のチェックと修復

書式

scatefs_fsck [-n] [-s backtime[m|h|d]] fsid

説明

scatefs_fsck は NEC Scalable Technology File System(ScaTeFS)のチェックと修復を行います。fsid には ScaTeFS のファイルシステム ID を指定します。

オプション

-n　　ファイルシステムのチェックのみを行い、修復を行いません。

-s backtime　　チェック対象として、現在時刻から遡る時間を指定します。
指定した時間より以前に更新があったファイルはチェックの対象から外します。
指定できる単位は d(日), h(時間), m(分) です。単位を省略した場合は m(分) となります。
本オプションを省略した場合は、全ファイルをチェックの対象とします。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_detail , scatefs_f2fsck

4.2.1.11 scatefs_ifstat

名前

scatefs_ifstat - IO サーバのインターフェース状態の表示

書式

scatefs_ifstat {-n node[,node...]|-a} [-if]

説明

scatefs_ifstat は NEC Scalable Technology File System(ScaTeFS)を構築する IO サーバのインターフェースの状態を表示します。

デフォルトではすべてのインターフェース状態を表示します。

IO サーバデーモンが 1 週間以内に core ファイルを出力していた場合、警告を表示します。

オプション

-n node[,node...]　　指定した IO サーバ ID の情報を表示します。
カンマ区切りで複数の IO サーバ ID を指定することが可能です。

-a　　すべての IO サーバを対象とします。

-i　　フローティング IP アドレスの状態を表示します。

-f　　FC パスの状態を表示します。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

返り値

scatefs_ifstat は次のいずれかの状態を返します。

0　IO サーバのインターフェースは正常である。

1　IO サーバのインターフェースに異常がある。

2　何らかのエラーが発生した。

関連項目

scatefs_statcollect , scatefs_logcollect

4.2.1.12 scatefs_lockrelease

名前

scatefs_lockrelease - ScaTeFS のレコードロックの解除

書式

scatefs_lockrelease -l

scatefs_lockrelease -r @clntid [fsid [ino]]

説明

scatefs_lockrelease は NEC Scalable Technology File System(ScaTeFS)のレコードロックの解除を行います。

オプション

-l　　ロックしているレコードロック情報を表示します。

-r @clntid　　クライアント ID(clntid), ファイルシステム ID(fsid), inode 番号(ino) に該当するレコードロック情報を強制解除します。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

4.2.1.13 scatefs_logcollect

名前

scatefs_logcollect - IO サーバのログ表示

書式

scatefs_logcollect {-n node[,node...]|-a} [-m]

説明

scatefs_logcollect は NEC Scalable Technology File System(ScaTeFS)を構築する

IO サーバのログを表示します。

オプション

-n node[,node...]	指定した IO サーバ ID のログを表示します。 カンマ区切りで複数の IO サーバ ID を指定することが可能です。
-a	すべての IO サーバを対象とします。
-m	すべてのログを表示します。

ファイル

/etc/scatefs/system.info	ScaTeFS の情報ファイル

関連項目

scatefs_statcollect , scatefs_ifstat

4.2.1.14 scatefs_migrate

名前

scatefs_migrate - マイグレーション情報のクリア

書式

scatefs_migrate --clear [--force]

説明

scatefs_migrate は NEC Scalable Technology File System(ScaTeFS)のマイグレーション情報をクリアします。

マイグレーション情報のクリアについては「3.2.7 リバランス」の「(4) マイグレーション情報のクリア（メンテナンス時に実施）」を参照してください。

オプション

--clear	マイグレーション情報をクリアします。
--force	強制的に実行します。

ファイル

/etc/scatefs/system.info	ScaTeFS の情報ファイル

返り値

scatefs_migrate は次のいずれかの状態を返します。

0	正常終了
1	異常終了

関連項目

scatefs_rebalance

4.2.1.15 scatefs_mkfs

名前

scatefs_mkfs - ScaTeFS のファイルシステム作成

書式

scatefs_mkfs -f datafile [--force]

説明

scatefs_mkfs は、NEC Scalable Technology File System(ScaTeFS)の作成を行います。

scatefs_mkfs を実行する前提として、scatefs_addios による IO サーバの登録、scatefs_addiot による IO ターゲットの登録が必要となります。

オプション

-f datafile　作成するファイルシステムの情報を記載した datafile を指定します。

--force　強制実行します。

データファイル

datafile には以下のエントリを記載します。

name　ファイルシステム名を指定します。
datafile に指定できるファイルシステム名は 1 つです。

ファイルシステム名には以下の制限があります。

- ファイルシステム名に "/" または ":" を含めることはできません。
- 数字だけのファイルシステム名を指定することはできません。
- ファイルシステム名は 31 文字まで指定可能です。

name はエントリの始めに記載する必要があります。

iotid	作成するファイルシステムに使用する IO ターゲット ID を指定します。 IO ターゲット ID は、scatefs_detail -t で確認できます。
mode	ルートディレクトリのパーミッションを 3 桁の 8 進数で指定します。 mode は省略可能であり、省略した場合はデフォルト値 755 を設定します。
chunksize	チャンクサイズを指定します。 省略した場合はデフォルト値を設定します。 デフォルト値は、ノンストライプフォーマット場合は 256 メガバイト、ストライプフォーマットの場合は 1 ギガバイトです。 チャンクサイズは 4K(4096) の倍数の値で指定する必要があります。 サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト) です。 たとえば、chunksize 268435456 と chunksize 256m と chunksize 256M はいずれも同じ指定となります。 ストライプフォーマットの場合、chunksize と stripesize の両方を指定します。 ノンストライプフォーマットの場合、chunksize のみを指定します。 ノンストライプフォーマットの場合、ストライプサイズとチャンクサイズは同じ値になります。

stripesize　　ストライプサイズを指定します。

ストライプサイズは 4K(4096) の倍数の値で指定する必要があります。
サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト) です。
たとえば、stripesize 4194304 と stripesize 4m と stripesize 4M はいずれも同じ指定となります。

ストライプフォーマットの場合、チャンクサイズはストライプサイズの倍数であり、かつストライプサイズより大きい値である必要があります。

data_fstype　　データ領域のファイルシステムタイプを指定します。
ファイルシステムタイプには、ext4 または xfs が指定可能です。
data_fstype は省略可能であり、省略した場合はデフォルト値(ext4)を設定します。

以下に、datafile の例を記載します。

[例 1] chunksize, stripesize を省略した場合(デフォルト：ノンストライプフォーマット)

```
name        scatefs00
iotid       0 1 2 3
```

[例 2] chunksize を指定した場合(ノンストライプフォーマット)

```
name        scatefs01
iotid       4 5
chunksize   512M
```

[例 3] chunksize, stripesize を指定した場合(ストライプフォーマット)

name	scatefs02
iotid	6 7
chunksize	1G
stripesize	256M

[例 4] mode を指定した場合

name	scatefs03
iotid	8 9
mode	777

[例 5] data_fstype を指定した場合

name	scatefs04
iotid	10 11
10 11	xfs

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_addios , scatefs_addiot , scatefs_mkfs , scatefs_admin

注意事項

- scatefs_mkfs は IO サーバデーモンを停止してから実行して下さい。
- scatefs_mkfs は fsadmin アカウントで実行して下さい。
- scatefs_mfks は 1 台の IO サーバで実行して下さい。
  1 回の実行によりすべての IO サーバに情報が設定されます。
- scatefs_detail の -f オプションで、作成したファイルシステムを確認できます。

4.2.1.16 scatefs_mkqdir

名前

scatefs_mkqdir - ScaTeFS の QUOTA 対応ディレクトリの作成

書式

scatefs_mkqdir [--dirid id] [--mode nnn] [--uid id] [--gid id] [--chunksize size] [--stripesize size] fs dirname

scatefs_mkqdir -c fs dirname

説明

scatefs_mkqdir は NEC Scalable Technology File System(ScaTeFS)の QUOTA に対

応するディレクトリを作成します。QUOTA 情報は作成したディレクトリ毎に管理し、ディレクトリおよび配下の使用量のカウントと、ハードリミット/ソフトリミット/猶予時間の設定に対応します。このコマンドで作成したディレクトリを削除する場合は、scatefs_rmqdir を使用する必要があります。

オプション

fs	ScaTeFS のファイルシステム名もしくはファイルシステム ID を指定します。
dirname	作成するディレクトリ名を ScaTeFS のトップディレクトリから始まるパス名で指定します。 -c オプションを指定してパス情報を修正する場合は、現状のパス名を指定します。
--dirid id	作成するディレクトリに対応するディレクトリ ID を指定します。 id には 1 から 4,294,967,295 の整数を指定します。 --dirid の指定が無い場合は自動でユニークな値を設定します。
--mode nnn	パーミッションを 3 桁の 8 進数で指定します。
--uid id	クライアントのユーザ ID を指定します。
--gid id	クライアントのグループ ID を指定します。

--chunksize size　　チャンクサイズを指定します。
省略した場合はデフォルト値を設定します。
デフォルト値は、ノンストライプフォーマット場合は 256 メガバイト、ストライプフォーマットの場合は 1 ギガバイトです。

チャンクサイズは 4K(4096) の倍数の値で指定する必要があります。
サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト) です。
たとえば、--chunksize 268435456 と --chunksize 256m と --chunksize 256M はいずれも同じ指定となります。

--stripesize size　　ストライプサイズを指定します。

ストライプサイズは 4K(4096) の倍数の値で指定する必要があります。
サイズに指定可能な単位は k/K(キロバイト), m/M(メガバイト), g/G(ギガバイト) です。
たとえば、--stripesize 4194304 と --stripesize 4m と --stripesize 4M はいずれも同じ指定となります。

ストライプフォーマットの場合、チャンクサイズはストライプサイズの倍数であり、かつストライプサイズより大きい値である必要があります。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル
/etc/scatefs/diridtab　　ディレクトリ QUOTA の情報ファイル

関連項目

scatefs_quota , scatefs_repquota , scatefs_edquota , scatefs_rmqdir

4.2.1.17 scatefs_quotacheck

名前

scatefs_quotacheck - ScaTeFS quota ファイルの修復

書式

scatefs_quotacheck [-cugd] fsname...

scatefs_quotacheck [-cugd] -a

説明

scatefs_quotacheck は NEC Scalable Technology File System(ScaTeFS)のファイルシステムのディスク使用量をチェックし、quota ファイルが破損していれば修復します。

オプション

-u	ユーザの使用状況をチェックします。 -u,-g,-d が指定されない場合はすべての QUOTA 情報をチェックします。
-g	グループの使用状況をチェックします。
-a	/etc/scatefs/system.info に記載されている、すべてのファイルシステムの使用状況をチェックします。 -a オプションが指定された場合、fsname の指定は無視します。
-c	既存の quota ファイルを読み込まずに使用量をチェックします。
fsname	指定したファイルシステムをチェックします。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_quota , scatefs_edquota , scatefs_repquota

注意事項

- scatefs_quotacheck はすべてのクライアントからファイルシステムをアンマウントした状態で実行して下さい。
- scatefs_quotacheck は fsadmin アカウントで実行して下さい。
- scatefs_quotacheck は 1 台の IO サーバで実行して下さい。
  1 回の実行によりすべての IO サーバに情報が設定されます。

4.2.1.18 scatefs_rcliadm

名前

scatefs_rcliadm - ScaTeFS コマンドのリモート実行管理

書式

scatefs_rcliadm add host user [-n]

scatefs_rcliadm delete {host [user]|--all} [-n]

scatefs_rcliadm info [host] [user]

scatefs_rcliadm trans

scatefs_rcliadm check

説明

scatefs_rcliadm は NEC Scalable Technology File System(ScaTeFS)を使用するクライアントのうち、IO サーバの ScaTeFS コマンド実行を許可するクライアント/ユーザの登録/削除/確認を行います。

登録されたクライアント/ユーザは、scatefs_rcli コマンドを経由し、IOS サーバの一部の ScaTeFS コマンド(scatefs_df, scatefs_detail, scatefs_logcollect, scatefs_quota, scatefs_edquota, scatefs_repquota)のリモート実行が可能となります。

add　リモート実行を許可するホスト/ユーザを追加します。
ホスト/ユーザのペアで指定する必要があります。
-n オプションを指定した場合はホスト/ユーザの一覧を他 IO サーバへ転送しません。

delete　登録されているホスト/ユーザの削除を行います。ホスト/ユーザを指定した場合は、該当するホスト/ユーザを削除します。ホストのみを指定した場合は、ホストに対応するユーザすべてを削除します。
--all を指定した場合はすべてのエントリを削除します。
-n オプションを指定した場合はホスト/ユーザの一覧を他 IO サーバへ転送しません。

info　リモート実行を許可しているホスト/ユーザの情報を表示します。
ホストまたはユーザを指定した場合、関連するホスト/ユーザの情報を表示します。
ホスト/ユーザ両方を指定した場合は、該当するホスト/ユーザの情報を表示します。

trans　　ホスト/ユーザの一覧を他 IO サーバに転送します。
-n オプションを指定した add/delete の後に使用します。

check　　ホスト/ユーザの一覧のフォーマットチェックと IO サーバ間での差分確認を行います。

オプション

-n　　ホスト/ユーザの一覧を他の IO サーバに転送しません。add/delete に指定可能です。

--all　　すべてのエントリを削除します。delete に指定可能です。

関連項目

scatefs_df , scatefs_detail , scatefs_logcollect , scatefs_quota , scatefs_repquota , scatefs_edquota

注意事項

- scatefs_rcliadm は fsadmin アカウントで実行して下さい。
  scatefs_rcliadm は 1 台の IO サーバで実行して下さい。1 回の実行によりすべての IO サーバに情報が設定されます。
- add/delete/trans 実行中は、クライアントからの scatefs_rcli が失敗する場合があります。
  add/delete/trans が完了してから実行して下さい。
- add/delete を -n オプション無しで連続実行すると、エラーになる場合があります。
  その場合、1 分程度待ってから再実行して下さい。
- add/delete を連続して実行する場合は、-n オプションを指定して実行したすべての add/delete が完了した後で trans を実行することにより、処理全体の高速化が期待できます。
  (例)

```
$ scatefs_rcliadm add hostA user000 -n
      :
$ scatefs_rcliadm add hostA user999 -n
$ scatefs_rcliadm trans
```

4.2.1.19 scatefs_rebalance

名前

scatefs_rebalance - リバランス機能の管理

書式

scatefs_rebalance --start-extraction

scatefs_rebalance --stop-extraction

scatefs_rebalance --start-migration

scatefs_rebalance --stop-migration

scatefs_rebalance --report

scatefs_rebalance --clear

説明

scatefs_rebalance は NEC Scalable Technology File System(ScaTeFS)のリバランス機能を管理します。リバランス機能の詳細は「3.2.7 リバランス」を参照してください。

オプション

--start-extraction	リバランス対象ファイルの抽出を開始します。
--stop-extraction	抽出を停止します。
--start-migration	マイグレーションサービスを開始します。
--stop-migration	マイグレーションサービスを停止します。
--report	リバランス機能実行状態を表示します。
--clear	リバランス対象ファイルの抽出結果をクリアします。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

返り値

scatefs_rebalance は次のいずれかの状態を返します。

0　正常終了

1　異常終了

関連項目

scatefs_migrate

4.2.1.20 scatefs_repquota

名前

scatefs_repquota - ScaTeFS の QUOTA 情報一覧の表示

書式

scatefs_repquota [-sbug] [-d [-n|v] [-t N]] [fsname..]

説明

scatefs_repquota は NEC Scalable Technology File System(ScaTeFS)の QUOTA 情報一覧を表示します。

各ユーザーの現在のファイル数と使用容量(キロバイト(1024 バイト)単位)を、scatefs_edquota で設定した値とともに表示します。

オプション

-s　それぞれのサイズにサイズ文字を付加します。
サイズ文字には、K(キロ)、M(メガ)、G(ギガ)、T(テラ)、P(ペタ)、E(エクサ)があります。
ブロック数は 2 の累乗を用いるので、たとえば M は 1,048,576 バイトを表します。
ファイル数は 10 の累乗を用いるので、たとえば M は 1,000,000 ファイルを表します。
-b が指定された場合はサイズ文字の付加は無視します。

-b　設定済み QUOTA 情報のバックアップファイルを作成します。
バックアップファイルは、ハードリミット、ソフトリミット、猶予時間情報を格納します。
バックアップファイル名には fsid と sgid を含みます。

-u　ユーザの QUOTA 情報を表示します。

-g　グループの QUOTA 情報を表示します。

-d　ディレクトリの QUOTA 情報を表示します。

-n　情報としてディレクトリ ID を表示します。

-v　情報としてディレクトリ ID とディレクトリ名を表示します。
-n、-v が指定されない場合はディレクトリ名を表示します。
ディレクトリ名は、scatefs_mkqdir で作成または変更したパス名を、ScaTeFS のルートディレクトリを表す「/」を含まずに表示します。

-t N　ディレクトリ名の表示領域をシングルバイト文字で N 文字数分とします。
省略した場合はシングルバイト文字で 16 文字数分です。
指定内容に関わらず、ディレクトリ名をフルパスで表示します。
-n が指定された場合は無効です。

fsname　ファイルシステムの QUOTA 情報を表示します。
省略した場合は、すべてのファイルシステムの QUOTA 情報を表示します。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

/etc/scatefs/diridtab　　ディレクトリ QUOTA の情報ファイル

scatefs_quota.*.user または scatefs_quota.*.group または scatefs_quota.*.dir

設定済み QUOTA 情報のバックアップファイル

関連項目

scatefs_quota , scatefs_edquota , scatefs_quotacheck ,

scatefs_mkqdir , scatefs_rmqdir

4.2.1.21 scatefs_rmqdir

名前

scatefs_rmqdir - ScaTeFS の QUOTA 対応ディレクトリの削除

書式

scatefs_rmqdir [-f] fs dirname

説明

scatefs_rmqdir は NEC Scalable Technology File System(ScaTeFS)の QUOTA に対応するディレクトリを削除します。

オプション

fs	ScaTeFS のファイルシステム名もしくはファイルシステム ID を指定します。
dirname	削除するディレクトリを ScaTeFS のトップディレクトリから始まるパス名で指定します。
-f	QUOTA に対応するディレクトリとして登録されていない場合でも強制で削除します。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

/etc/scatefs/diridtab　　ディレクトリ QUOTA の情報ファイル

関連項目

scatefs_quota , scatefs_repquota , scatefs_edquota , scatefs_mkqdir

4.2.1.22 scatefs_statcollect

名前

scatefs_statcollect - IO サーバの統計情報の表示

書式

scatefs_statcollect {-n node[,node...] | -a} [-p] [-f] [-t sampling-time]

scatefs_statcollect {-n node[,node...] | -a} -i {ipaddr [-s netmask] | all} [-c]

scatefs_statcollect {-n node[,node...] | -a} -c

説明

scatefs_statcollect は NEC Scalable Technology File System(ScaTeFS)を構築する IO サーバの統計情報を表示します。デフォルトでは RPC と関数両方を表示します。

オプション

-n node[,node...]	指定した IO サーバ ID の情報を表示します。 カンマ区切りで複数の IO サーバ ID を指定することが可能です。
-a	すべての IO サーバを対象とします。
-i	統計情報を表示するクライアントの IP アドレスを IO サーバに設定します。 設定後は、指定したクライアントに関する統計情報のみを表示します。 IO サーバのデフォルトは all (すべてのクライアント)です。 IO サーバの設定をデフォルトに戻す場合は all を指定します。
-s	-i オプションで指定した IP アドレスに対するサブネットマスクを設定します。
-c	すべての統計情報をクリアします。
-p	RPC の統計情報を表示します。
-f	関数の統計情報を表示します。
-t sampling-time	採取する時間(秒)を sampling-time に指定します。 コマンド実行から指定秒間の統計情報を表示します。

ファイル

/etc/scatefs/system.info　　ScaTeFS の情報ファイル

関連項目

scatefs_logcollect , scatefs_ifstat

付録 A　発行履歴

A.1　発行履歴一覧表

2024 年　　1 月　初版

A.2　追加・変更点詳細

- 初版

  新規作成

NEC Scalable Technology File System for AI
(ScaTeFS for AI)
ユーザーズガイド

2024年 1月 初版

日本電気株式会社
東京都港区芝五丁目７番１号
TEL(03)3454-1111 (大代表)