



Red Hat Enterprise Linux 5

Global File System 2

Red Hat Global File System 2

エディション 3

Red Hat Enterprise Linux 5 Global File System 2

Red Hat Global File System 2

エディション 3

Landmann

rlandmann@redhat.com

法律上の通知

Copyright © 2009 Red Hat Inc..

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

このマニュアルは Red Hat Enterprise Linux 5 対応の Red Hat GFS2 (Red Hat Global File System 2) のインストール、設定、及び維持についての情報を提供しています。

目次

はじめに	3
1. 対象者	3
2. 関連したドキュメント	3
3. フィードバック	4
第1章 GFS2 の概要	5
1.1. GFS2 を設定する前に	5
1.2. GFS と GFS2 の相違	6
1.2.1. GFS2 のコマンド名	6
1.2.2. GFS と GFS2 のその他の差異	7
コンテキスト依存のパス名	7
gfs2.ko モジュール	7
GFS2 での Quota 強制を有効にする	8
データジャーナリング	8
ジャーナルを動的に追加	8
atime_quantum パラメータの排除	8
マウントコマンドの data= オプション	8
gfs2_tool コマンド	8
gfs2_edit コマンド	9
1.2.3. GFS2 のパフォーマンス向上	9
第2章 操作の開始	11
2.1. 事前要求タスク	11
2.2. 初期セットアップのタスク	11
第3章 GFS2 の管理	13
3.1. ファイルシステムの作成	13
使用法	13
例	14
オプション全種	15
3.2. ファイルシステムのマウント	16
使用法	17
例	17
完全な使用法	17
3.3. ファイルシステムのアンマウント	19
使用法	19
3.4. GFS2 QUOTA の管理	19
3.4.1. Quota の設定	19
使用法	20
例	20
3.4.2. Quota 制限の表示と使用法	21
使用法	21
コマンドの出力	21
コメント	22
例	22
3.4.3. Quota の同期化	22
使用法	23
例	23
3.4.4. Quota 強制の有効化/無効化	23
使用法	23
例	24
3.4.5. Quota アカウントの有効化	24

使用法	24
例	24
3.5. ファイルシステムの拡張	24
使用法	25
コメント	25
例	25
完全な使用法	25
3.6. ファイルシステムヘジャーナルの追加	26
使用法	26
例	27
完全な使用法	27
3.7. データジャーナリング	27
3.7.1. chattr コマンドを使用したデータジャーナリングの有効化/無効化	28
3.7.2. gfs2_tool コマンドでデータジャーナリングを有効/無効にする	29
3.8. ATIME 更新の設定	29
3.8.1. relatime でマウント	30
使用法	30
例	30
3.8.2. noatime でマウント	30
使用法	30
例	31
3.9. ファイルシステム上の動作の休止	31
使用法	31
例	31
3.10. ファイルシステムの修復	31
使用法	32
例	32
3.11. BIND マウントとコンテキスト依存のパス名	33
付録A ファイルシステムを GFS から GFS2 に変換	35
付録B 改訂履歴	36
索引	37

はじめに

『Global File System の設定と管理』ドキュメントへようこそ。この文書は Red Hat GFS2 (Red Hat Global File System 2) の設定と維持に関する情報を提供します。Red Hat GFS2 は RHEL 5.3 及びそれ以降対応の単独ノード内で、又は Red Hat Cluster Suite 内のクラスター設定の一部として実行することができます。Red Hat Cluster Suite に関する情報には、『Red Hat Cluster Suite の概要』、及び『Red Hat Cluster の設定と管理』を参照して下さい。

全ての正式な Red Hat Enterprise Linux マニュアルとリリースノートの HTML と PDF バージョンはオンライン：<http://www.redhat.com/docs/> で入手することができます。

1. 対象者

この文書は主として、以下の活動に熟知している Linux システム管理者を対象としています：

- カーネル設定を含む Linux システムの管理手続き
- Fibre Channel SAN などの共有ストレージネットワークのインストールと設定

2. 関連したドキュメント

Red Hat Enterprise Linux に関する詳細情報には、以下の書籍を参照して下さい：

- 『Red Hat Enterprise Linux インストールガイド』 — Red Hat Enterprise Linux のインストールに関する情報を提供します。
- 『Red Hat Enterprise Linux 導入ガイド』 — Red Hat Enterprise Linux 5 の導入、設定、及び管理に関する情報を提供します。

Red Hat Cluster Suite に関する詳細情報には、以下の書籍を参照して下さい：

- 『Red Hat Cluster Suite の概要』 — Red Hat Cluster Suite の 高度な概要を提供します。
- 『Red Hat Cluster の設定と管理』 — Red Hat Cluster コンポーネントのインストール、設定、及び管理に関する情報を提供します。
- 『LVM Administrator's Guide: Configuration and Administration』 — Provides a description of the Logical Volume Manager (LVM), including information on running LVM in a clustered environment.
- 『Global File System: 設定と管理』 — Red Hat GFS (Red Hat Global File System) のインストール、設定、及び管理に関する情報を提供します。
- 『Device-Mapper Multipath の使用』 — Red Hat Enterprise Linux. の Device-Mapper Multipath 機能の使用に関する情報を提供します。
- 『Global File System で GNBD の使用』 — Red Hat GFS で Global Network Block Device (GNBD) の使用に関する概要を提供します。
- 『Linux Virtual Server Administration』 — Linux Virtual Server (LVS) に於けるハイパフォーマンスシステムとサービスの設定に関する情報を提供します。

- 『Red Hat Cluster Suite リリースノート』 — Red Hat Cluster Suite の現在のリリースに関する情報を提供します。

Red Hat Cluster Suite ドキュメントとその他の Red Hat ドキュメントは、HTML、PDF、及び RPM バージョンで、Red Hat Enterprise Linux ドキュメント CD 中、および、サイト：<http://www.redhat.com/docs/> から入手できます。

3. フィードバック

誤字・脱字を発見された場合や、このマニュアルを改善する方法を考えつかれた場合は、弊社までご連絡下さい。そしてコンポーネント **rh-cs** に対して Bugzilla (<http://bugzilla.redhat.com/bugzilla/>) にレポートを提出して下さい。

Be sure to mention the manual's identifier:

```
rh-gfs2(EN)-5 (2009-01-05T15:15)
```

By mentioning this manual's identifier, we know exactly which version of the guide you have.

このドキュメントの改善案をお持ちの場合は、出来るだけ詳しく記入して下さい。間違いを発見された場合は、そのセクション番号と周囲の文章も含めて頂くと弊社で素早くその場所を発見できます。

第1章 GFS2 の概要

Red Hat GFS2 ファイルシステムは、Linux カーネルファイルシステムインターフェイス (VFS レイヤー) に直接的に対応するネイティブファイルシステムです。GFS2 ファイルシステムは単独システム内で、あるいはクラスター設定の一部として実装することができます。クラスターファイルシステムの一部として実装された場合、GFS2 は分散型メタデータと複数ジャーナルを運用します。

A GFS2 file system can be created on an LVM logical volume. A logical volume is an aggregation of underlying block devices that appears as a single logical device. For information on the LVM volume manager, see the 『LVM Administrator's Guide』.

GFS2 は 64-bit アーキテクチャを土台としており、これは理論的に 8 EB の ファイルシステムを収容することができます。しかし、現在の GFS2 ファイルシステムの最大サポート可能サイズは 25 TB です。ご使用の GFS2 ファイルシステムが 25 TB 以上を必要とする場合は、Red Hat サービスの担当者にご連絡ください。

ご使用のファイルシステムのサイズを判定する時には、復元時のニーズを考慮する必要があります。非常に大きなファイルシステムでの **fsck** コマンドの実行は長時間とメモリーの大領域を消費します。更には、ディスク/ディスク サブシステムで障害が発生すると、その復元時間は使用するバックアップメディアの速度によって限定されます。

Red Hat Cluster Suite 内で設定されている場合、Red Hat GFS2 ノードは Red Hat Cluster Suite の設定と管理ツールでその構成と管理ができます。Red Hat GFS2 はそれから GFS2 ノード全域に渡って1つの一貫性のあるファイルシステム名前スペース表示で Red Hat クラスター内での GFS2 ノード間のデータ共有を提供します。これにより、異なるノード上のプロセスは、同じノード上のプロセスがローカルファイルシステム上でファイルを共有するのと同じ方法で、認識できる差を見せないまま GFS2 ファイルを共有できるようになります。Red Hat Cluster Suite に関する詳細情報には、『Red Hat Cluster の設定と管理』を参照して下さい。

LVM logical volumes in a Red Hat Cluster suite are managed with CLVM, which is a cluster-wide implementation of LVM, enabled by the CLVM daemon, **clvmd** running in a Red Hat Cluster Suite cluster. The daemon makes it possible to use LVM2 to manage logical volumes across a cluster, allowing all nodes in the cluster to share the logical volumes. to any directory on your system. For information on the LVM volume manager, see the 『LVM Administrator's Guide』.

gfs2.ko カーネルモジュールは GFS2 ファイルシステムを実装して、GFS2 クラスターノードにロードされています。

この章では、ユーザーの GFS2 の理解をお手伝いするための土台として一部基本的な、省略した情報を提供しています。ここには以下のようなセクションがあります：

- [「GFS2 を設定する前に」](#)
- [「GFS と GFS2 の相違」](#)

1.1. GFS2 を設定する前に

GFS2 をインストールしてセットアップする前に、GFS2 ファイルシステムの以下の基幹特性に注意を払って下さい：

GFS2 ノード

GFS2 ファイルシステムをマウントする Red Hat Cluster Suite 内のノードを決定します。

ファイルシステムの数量

作成する初期の GFS2 ファイルシステムの数量を決定します。（後で より多くのファイルシステムを追加することができます。）

ファイルシステム名

各ファイルシステム用に独自の名前を決定します。その名前はクラスター上の全ての **lock_dlm** ファイルシステムと、各ローカルノード上の全ての ファイルシステム (**lock_dlm**及び **lock_nolock**) に対して独自でなければなりません。各ファイルシステム名はパラメータ変数の形式になる 必要があります。例えば、このマニュアルは手順例の中で、ファイルシステム名、**mydata1** と **mydata2** を使用しています。

ジャーナル

Determine the number of journals for your GFS2 file systems. One journal is required for each node that mounts a GFS2 file system. GFS2 allows you to add journals dynamically at a later point as additional servers mount a filesystem. For information on adding journals to a GFS2 file system, see [「ファイルシステムへジャーナルの追加」](#).

GNBD サーバーノード

GNBD を使用している場合、必要となる GNBD サーバーノードの数量を決定します。それぞれの GNBD サーバーノードのホスト名と IP アドレスは、後日の GNBD クライアント セットアップの為にメモして置きます。GFS2 を持つ GNBD の使用に関する情報は **Global File System** を持つ **GNBD の使用** ドキュメントでご覧下さい。

ストレージデバイスとパーティション

ファイルシステム内に（CLVM 経由で）論理ボリュームを作成する為に使用する ストレージデバイスとパーティションを決定します。



注記

同じディレクトリ内にある複数のノードから同時に多数の作成と削除の操作が実行されると、パフォーマンスの問題が発生する可能性があります。このパフォーマンス問題がご使用のシステムで発生した場合は、可能な限りノード毎にファイルの作成と削除をそのノード特定のディレクトリを介してローカル化すべきです。

1.2. GFS と GFS2 の相違

このセクションは GFS に比較して GFS2 が提供できる改善と変更を一覧表示します。

Migrating from GFS to GFS2 requires that you convert your GFS file systems to GFS2 with the **gfs2_convert** utility. For information on the **gfs2_convert** utility, see [付録A ファイルシステムを GFS から GFS2 に変換](#).

1.2.1. GFS2 のコマンド名

In general, the functionality of GFS2 is identical to GFS. The names of the file system commands, however, specify GFS2 instead of GFS. [表1.1 「GFS と GFS2 のコマンド」](#) shows the equivalent GFS and GFS2 commands.

表1.1 GFS と GFS2 のコマンド

GFS コマンド	GFS2 コマンド	説明				
<code>mount</code>	<code>mount</code>	ファイルシステムをマウントします。システムはファイルシステムが GFS 又は GFS2 のファイルシステムかどうかを判定します。GFS2 マウントオプションに関する情報には、 <code>gfs2_mount(8) man</code> ページをご覧ください。				
<code>umount</code>	<code>umount</code>	ファイルシステムをアンマウント				
<table border="1"> <tr> <td><code>fsck</code></td> <td><code>fsck</code></td> </tr> <tr> <td><code>gfs_fsck</code></td> <td><code>fsck.gfs2</code></td> </tr> </table>	<code>fsck</code>	<code>fsck</code>	<code>gfs_fsck</code>	<code>fsck.gfs2</code>		アンマウントしたファイルシステムのチェックと修理
<code>fsck</code>	<code>fsck</code>					
<code>gfs_fsck</code>	<code>fsck.gfs2</code>					
<code>gfs_grow</code>	<code>gfs2_grow</code>	マウントしたファイルシステムの拡張				
<code>gfs_jadd</code>	<code>gfs2_jadd</code>	マウントしたファイルシステムにジャーナルを追加				
<table border="1"> <tr> <td><code>gfs_mkfs</code></td> <td><code>mkfs.gfs2</code></td> </tr> <tr> <td><code>mkfs -t gfs</code></td> <td><code>mkfs -t gfs2</code></td> </tr> </table>	<code>gfs_mkfs</code>	<code>mkfs.gfs2</code>	<code>mkfs -t gfs</code>	<code>mkfs -t gfs2</code>		ストレージデバイス上にファイルシステムを作成
<code>gfs_mkfs</code>	<code>mkfs.gfs2</code>					
<code>mkfs -t gfs</code>	<code>mkfs -t gfs2</code>					
<code>gfs_quota</code>	<code>gfs2_quota</code>	マウント済のファイルシステム上で <code>quota</code> を管理				
<code>gfs_tool</code>	<code>gfs2_tool</code>	ファイルシステムを設定、チューン、又は情報収集				
<code>gfs_edit</code>	<code>gfs2_edit</code>	ファイルシステムの内部構造の表示、出力、編集。 <code>gfs2_edit</code> コマンドは GFS ファイルシステムと GFS2 ファイルシステム用に使用できます。				

GFS2 ファイルシステムコマンド用にサポートされているコマンドの一覧を見るには、これらのコマンドの `man` ページをご覧ください。

1.2.2. GFS と GFS2 のその他の差異

This section summarizes the additional differences in GFS and GFS2 administration that are not described in [「GFS2 のコマンド名」](#).

コンテキスト依存のパス名

GFS2 file systems do not provide support for context-dependent path names, which allow you to create symbolic links that point to variable destination files or directories. For this functionality in GFS2, you can use the `bind` option of the `mount` command. For information on managing pathnames in GFS2, see [「Bind マウントとコンテキスト依存のパス名」](#).

`gfs2.ko` モジュール

GFS ファイルシステムを実装するカーネルモジュールは **gfs.ko** です。GFS2 ファイルシステムを実装するカーネルモジュールは **gfs2.ko** です。

GFS2 での Quota 強制を有効にする

In GFS2 file systems, quota enforcement is disabled by default and must be explicitly enabled. To enable and disable quotas for GFS2 file systems, you use the **quota=on|off|account** option for the **mount** command. For information on enabling and disabling quota enforcement, see [「Quota 強制の有効化/無効化」](#).

データジャーナリング

GFS2 file systems support the use of the **chattr** command to set and clear the **j** flag on a file or directory. Setting the **+j** flag on a file enables data journaling on that file. Setting the **+j** flag on a directory means "inherit jdata", which indicates that all files and directories subsequently created in that directory are journaled. Using the **chattr** command is the preferred way to enable and disable data journaling on a file.

ジャーナルを動的に追加

In GFS2 file systems, journals are plain (though hidden) files instead of embedded metadata. This means that journals can be dynamically added as additional servers mount a filesystem. For information on adding journals to a GFS2 file system, see [「ファイルシステムへジャーナルの追加」](#).

atime_quantum パラメータの排除

GFS2 ファイルシステムは **atime_quantum** チューン可能パラメータをサポートしていません。このパラメータは **atime** 更新が発生する頻度を指定するのに GFS ファイルシステムで使用可能でした。その交替として GFS2 では、**relatime** と **noatime** のマウントオプションをサポートします。**relatime** マウントオプションを使用することで、GFS にあった **atime_quantum** パラメータのセッティングに似た動作を達成することが推奨されます。

マウントコマンドの data= オプション

GFS2 ファイルシステムをマウントする場合、**mount** コマンドの **data=ordered**、又は **data=writeback** オプションを指定することができます。**data=ordered** がセットしてあると、トランザクションによって変更されたユーザーデータは、トランザクションがディスクにコミットされる前にディスクに流されます。これは、ファイル内の初期化していないブロックをクラッシュの後でユーザーが見れないようにするはずです。**data=writeback** がセットしてあると、ユーザーデータはそれが改変された後でも、いつでもディスクに書き込まれます。これは **ordered** モードと同様の一貫性保証を提供しませんが、一部の作業負荷では少々速くなるはずです。デフォルトは **ordered** モードです。

gfs2_tool コマンド

gfs2_tool コマンドは、GFS 用の **gfs_tool** コマンドがサポートするものとは異なるオプションセットを GFS2 用にサポートします。以下を含んでいます:

- **gfs2_tool** コマンドは、ファイルシステムが収納するジャーナルの数量を含む、現在設定済のジャーナルに関する情報を表示する **journals** パラメータをサポートします。
- **gfs2_tool** コマンドは **counters** フラグをサポートしません。このフラグは **gfs_tool** コマンドで GFS の統計を表示するために使用されます。

- The **gfs2_tool** command does not support the **inherit_jdata** flag. To flag a directory as "inherit jdata", you can set the **jdata** flag on the directory or you can use the **chattr** command to set the **+j** flag on the directory. Using the **chattr** command is the preferred way to enable and disable data journaling on a file.

gfs2_edit コマンド

gfs2_edit コマンドは、**gfs_edit** コマンドが GFS 用にサポートするのとは異なる GFS2 用のオプションセットをサポートします。

1.2.3. GFS2 のパフォーマンス向上

GFS2 ファイルシステムの多くの機能が、GFS ファイルシステムからのユーザーインターフェイス内でその差を見せませんが、ファイルシステムパフォーマンスは向上します。

GFS2 ファイルシステムは以下の分野で向上したファイルシステムパフォーマンスを提供します：

- 単独ディレクトリ内での大量使用の為に向上したパフォーマンス
- より高速な同期 I/O 演算
- より速いキャッシュ化読み込み（オーバーヘッドのロックなし）
- 事前割り当てファイルでのより高速なダイレクト I/O（I/O が 4M など十分に大きいことが前提）
- 全般的により高速な I/O 演算
- **df** コマンドの実行は、より高速な **statfs** コールの影響でより速くなります。
- GFS で比較した場合の **atime** により生成された書き込み I/O 操作の頻度を低減するために **atime** モードは改良されています。

GFS2 ファイルシステムは 以下のような方法でより広くより主流となるサポートを提供します。

- GFS2 はアップストリームカーネル（2.6.19 内に統合）の一部です。
- GFS2 は 以下のような機能に対応しています：
 - SELinux の拡張属性
 - 標準 **ioctl()** コールを経由した **lsattr()** と **chattr()** の属性セッティング
 - ナノ秒タイムスタンプ

GFS2 ファイルシステムはファイルシステムの内部効率に対し、以下のような改善を提供します。

- GFS2 のカーネルメモリーの使用は少なくなります。
- GFS2 メタデータ生成の番号を必要としません。

GFS2 メタデータの割り当ては読み込みを必要としません。複数ジャーナル内のメタデータブロックのコピーは、ロックが開放される前にジャーナルからブロックを呼び起こすことにより管理されます。

- GFS2 は、リンクのない inode や quota 変更について認識のないより簡単な ログマネージャを含んでいます。
- **gfs2_grow** と **gfs2_jadd** のコマンドは ロックを使用して複数インスタンスが同時に稼働するのを防止します。
- ACL code on は **creat()** や **mkdir()** のようなコールの為に簡素化されました。
- inode と異なり、quota の変更と **statfs** の変更は ジャーナルを再マウントしないで復元されます。

第2章 操作の開始

この章では、GFS2 の初期セットアップの手順を説明しており、以下のような セクションを含んでいます:

- 「[事前要求タスク](#)」
- 「[初期セットアップのタスク](#)」

2.1. 事前要求タスク

Before setting up Red Hat GFS2, make sure that you have noted the key characteristics of the GFS2 nodes (refer to 「[GFS2 を設定する前に](#)」). Also, make sure that the clocks on the GFS2 nodes are synchronized. It is recommended that you use the Network Time Protocol (NTP) software provided with your Red Hat Enterprise Linux distribution.



注記

GFS2 ノード内のシステムクロックは相互誤差が数分以内になるようにして不要な inode time-stamp 更新を防止するようにしなければなりません。不要な inode time-stamp 更新はクラスタのパフォーマンスにかなりの害を与えます。

2.2. 初期セットアップのタスク

初期の GFS2 セットアップは以下のタスクで構成されます:

1. 論理ボリュームのセットアップ
2. GFS2 ファイルシステムの作成
3. ファイルシステムのマウント

これらの手順に従って GFS2 の初期セットアップをします。

1. LVM を使用して、各 Red Hat GFS2ファイルシステム用に論理ボリュームを作成します。



注記

Red Hat Cluster Suite に含まれている **init.d** スクリプトを使用して、論理ボリュームのアクティベートとディアクティベートをすることが出来ます。**init.d** スクリプトの詳細情報については **Red Hat Cluster の設定と管理** を参照して下さい。

2. Create GFS2 file systems on logical volumes created in Step 1. Choose a unique name for each file system. For more information about creating a GFS2 file system, refer to 「[ファイルシステムの作成](#)」.

以下のいずれかの形式を使用してクラスター化した GFS2 ファイルシステムを作成することができます:

```
mkfs.gfs2 -p lock_dlm -t ClusterName:FSName -j NumberJournals
BlockDevice
```

```
mkfs -t gfs2 -p lock_dlm -t LockTableName -j NumberJournals
BlockDevice
```

以下のいずれかの形式を使用してローカルの GFS2 ファイルシステムを作成できます:

```
mkfs.gfs2 -p lock_nolock -j NumberJournals BlockDevice
```

```
mkfs -t gfs2 -p lock_nolock -j NumberJournals BlockDevice
```

For more information on creating a GFS2 file system, see [「ファイルシステムの作成」](#).

3. At each node, mount the GFS2 file systems. For more information about mounting a GFS2 file system, see [「ファイルシステムのマウント」](#).

コマンドの使用法:

```
mount BlockDevice MountPoint
```

```
mount -o acl BlockDevice MountPoint
```

-o acl mount オプションは ファイル ACL の操作を可能にします。1つのファイルシステムが **-o acl** マウントオプションなしでマウントされている場合、ユーザーは (**getfacl** で) ACL の表示はできますが、(**setfacl** で) それらをセットすることができません。



注記

Red Hat Cluster Suite に収納されている **init.d** スクリプトを使用して、GFS2 ファイルシステムのマウントとアンマウントを自動化することができます。**init.d** スクリプトに関する詳細情報は **Red Hat Cluster の設定と管理** をご覧ください。

第3章 GFS2 の管理

この章では、GFS2 を管理するためのタスクとコマンドを説明しており、以下のような セクションで構成されます:

- 「ファイルシステムの作成」
- 「ファイルシステムのマウント」
- 「ファイルシステムのアンマウント」
- 「GFS2 Quota の管理」
- 「ファイルシステムの拡張」
- 「ファイルシステムヘジャーナルの追加」
- 「データジャーナリング」
- 「**atime** 更新の設定」
- 「ファイルシステム上の動作の休止」
- 「ファイルシステムの修復」
- 「Bind マウントとコンテキスト依存のパス名」

3.1. ファイルシステムの作成

mkfs.gfs2 コマンドを使用して、GFS2 ファイルシステムを作成します。又は **mkfs** コマンドに **-t gfs2** オプションを指定して使用することもできます。ファイルシステムは起動中の LVM ボリューム上で作成されます。**mkfs.gfs2** コマンドを実行するには 以下の情報が必要になります:

- プロトコル/モジュールのロック名 (クラスター用の lock protocol は **lock_dlm**)
- クラスター名 (クラスター設定の一部として実行している場合)
- ジャーナルの数量 (ファイルシステムをマウントするそれぞれのノード毎にジャーナル1つ必要)

GFS ファイルシステムを作成している場合、直接 **mkfs.gfs2** を使用するか、又は **mkfs** コマンドに **-t** パラメータを付けてタイプ **gfs2** の ファイルシステムを指定して、**gfs2** ファイルシステムのオプションを続ける ことが出来ます。

使用法

クラスター化した GFS2 ファイルシステムを作成する場合、以下の形式の いずれかを使用できます:

```
mkfs.gfs2 -p LockProtoName -t LockTableName -j NumberJournals BlockDevice
```

```
mkfs -t gfs2 -p LockProtoName -t LockTableName -j NumberJournals
BlockDevice
```

ローカルの GFS2 ファイルシステムを作成している場合、以下の形式のいずれかを 使用できます:

-

```
mkfs.gfs2 -p LockProtoName -j NumberJournals BlockDevice
```

```
mkfs -t gfs2 -p LockProtoName -j NumberJournals BlockDevice
```



警告

LockProtoName と **LockTableName** のパラメータの使用に熟知していることを確認して下さい。**LockProtoName** と **LockTableName** のパラメータの間違った使用をすると、ファイルシステムか ロックスペースが破損する可能性があります。

LockProtoName

使用するロックプロトコルの名前を指定します。クラスター用の lock protocol は **lock_dlm** です。

LockTableName

このパラメータはクラスター設定の GFS2 ファイルシステム用に指定されます。これには、次の様にコロンで区切った2つの部分（空白なし）で構成されます: **ClusterName:FSName**

- **ClusterName**, Red Hat クラスター用の名前です。このクラスター用に GFS2 ファイルシステムが作成されます。
- **FSName**, ファイルシステムの名前で、これは1文字から16文字までの長さが可能であり、クラスター上の全ての **lock_dlm** ファイルシステムと各ローカルノード上の全てのファイルシステム (**lock_dlm** 及び **lock_nolock**) に渡って独自の名前でなければなりません。

Number

Specifies the number of journals to be created by the **mkfs.gfs2** command. One journal is required for each node that mounts the file system. For GFS2 file systems, more journals can be added later without growing the filesystem, as described in [「ファイルシステムヘジャーナルの追加」](#).

BlockDevice

論理ボリュームか物理ボリュームかを指定します。

例

この例では、これはクラスター化したファイルシステムですので、**lock_dlm** はファイルシステムが使用するロックプロトコルです。クラスター名は **alpha** で、ファイルシステム名は **mydata1** です。このファイルシステムは8つのジャーナルを含み、**/dev/vg01/lvol0** 上に作成されています。

```
mkfs.gfs2 -p lock_dlm -t alpha:mydata1 -j 8 /dev/vg01/lvol0
```

```
mkfs -t gfs2 -p lock_dlm -t alpha:mydata1 -j 8 /dev/vg01/lvol0
```

この例では、2つ目の **lock_dlm** ファイルシステムが作成されて、それがクラスター **alpha** 内で使用できます。ファイルシステム名は **mydata2** です。このファイルシステムには 8つのジャーナルが含まれており、**/dev/vg01/lvol1** 上に作成されています。

```
mkfs.gfs2 -p lock_dlm -t alpha:mydata2 -j 8 /dev/vg01/lvol1
```

```
mkfs -t gfs2 -p lock_dlm -t alpha:mydata2 -j 8 /dev/vg01/lvol1
```

オプション全種

表3.1「コマンドオプション: **mkfs.gfs2**」 describes the **mkfs.gfs2** command options (flags and parameters).

表3.1 コマンドオプション: **mkfs.gfs2**

フラグ	パラメータ	説明
-c	<i>Megabytes</i>	Sets the initial size of each journal's quota change file to Megabytes .
-D		出力デバッグを有効にします。
-h		ヘルプ。使用可能なオプションを表示します。
-J	<i>MegaBytes</i>	ジャーナルのサイズをメガバイトで指定します。デフォルトのジャーナルサイズは 128 メガバイトです。最低サイズは 8 メガバイトです。長いジャーナルはパフォーマンスを向上しますが、短いジャーナルよりもメモリーを多く消費します。
-j	<i>Number</i>	mkfs.gfs2 コマンドで作成されるジャーナルの数を指定します。ファイルシステムをマウントするそれぞれのノード毎に1つのジャーナルが必要になります。このオプションが指定されていない場合、1つのジャーナルが作成されます。GFS2 ファイルシステムではファイルシステムを拡張することなく、余分のジャーナルを後で追加することができます。
-O		mkfs.gfs2 コマンドが、ファイルシステムへの書き込み前に確認をしてくることを阻止します。
-p	<i>LockProtoName</i>	<div style="border: 1px solid black; padding: 5px;"> <p>使用するロックングプロトコルの名前を指定します。承認されているロックングプロトコルには 以下があります:</p> <p>lock_dlm — 標準のロックングモジュール。クラスター化したファイルシステムに必要です。</p> <p>lock_nolock — GFS2 がローカルファイルシステム (1 ノードのみ) として機能している時に使用します。</p> </div>

フラグ	パラメータ	説明
-q		静寂モード。何も表示しません。
-r	<i>MegaBytes</i>	リソースグループのサイズをメガバイトで指定します。リソースグループの最低サイズは 32 MB です。リソースグループの最大サイズは 2048 MB となります。大容量のファイルシステムでは、サイズの大きなリソースグループがパフォーマンスを向上する可能性が高くなります。このサイズが指定されていないと、mkfs.gfs2 はファイルシステムのサイズを基にしてリソースグループのサイズを選択します。平均的なサイズのファイルシステムは 256 MB のリソースグループを持ち、それより大きなファイルシステムはより良いパフォーマンスの為により大きなリソースグループサイズを選択します。
-t	<i>LockTableName</i>	<p>lock_dlm プロトコルを使用している時に、ロック表のフィールドを指定する特有の識別子。lock_nolock プロトコルはこのパラメータを使用しません。</p> <p>このパラメータはコロンで分離された（空白なし）次の2つの部分を持ちます: ClusterName:FSName.</p> <p>ClusterName は GFS2 ファイルシステムを作成している Red Hat クラスターの名前です。このクラスターのメンバーのみがこのファイルシステムを使用することができます。クラスター名は クラスター設定ツール を介して /etc/cluster/cluster.conf ファイル内でセットされて、Red Hat Cluster Suite クラスター管理 GUI 内の クラスターステータスツール で表示されます。</p> <p>FSName, ファイルシステム名です。1文字から 16文字までの長さを持ち、名前はクラスター内の全てのファイルシステムで特有のものでなければなりません。</p>
-u	<i>MegaBytes</i>	Specifies the initial size of each journal's unlinked tag file.
-V		コマンドのバージョン情報を表示します。

3.2. ファイルシステムのマウント

Before you can mount a GFS2 file system, the file system must exist (refer to [「ファイルシステムの作成」](#)), the volume where the file system exists must be activated, and the supporting clustering and locking systems must be started (refer to **Configuring and Managing a Red Hat Cluster**). After those requirements have been met, you can mount the GFS2 file system as you would any Linux file system.

ファイル ACL を操作するには、ファイルシステムを **-o acl** マウントオプションでマウントする必要

があります。ファイルシステムが **-o acl** マウントオプションなしでマウントされている場合、ユーザーは (**getfacl** で) ACL を表示できますが、 (**setfacl** で) それらをセットすることができません。

使用法

Mounting Without ACL Manipulation

```
mount BlockDevice MountPoint
```

Mounting With ACL Manipulation

```
mount -o acl BlockDevice MountPoint
```

-o acl

ファイル ACL の操作を可能にする GFS2 特有のオプション。

BlockDevice

GFS2 ファイルシステムが存在するブロックデバイスを指定します。

MountPoint

GFS2 ファイルシステムがマウントされるディレクトリを指定します。

例

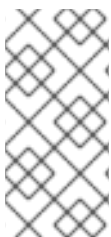
この例では、**/dev/vg01/lvol0** の GFS2 ファイルシステムは **/mygfs2** ディレクトリにマウントされています。

```
mount /dev/vg01/lvol0 /mygfs2
```

完全な使用法

```
mount BlockDevice MountPoint -o option
```

The **-o option** argument consists of GFS2-specific options (refer to [表3.2 「GFS2 特有のマウントオプション」](#)) or acceptable standard Linux **mount -o** options, or a combination of both. Multiple **option** parameters are separated by a comma and no spaces.



注記

mount コマンドは Linux のシステムコマンドです。このセクションに説明してある GFS2 特有のオプションの他にも、標準の **mount** コマンドのオプション（例えば、**-r**）を使用することができます。Linux **mount** コマンドのオプションに関する情報は、Linux **mount** man ページでご覧下さい。

[表3.2 「GFS2 特有のマウントオプション」](#) describes the available GFS2-specific **-o option** values that can be passed to GFS2 at mount time.

表3.2 GFS2 特有のマウントオプション

オプション	説明
acl	ファイル ACL の操作を可能にします。ファイルシステムが acl マウント オプションなしでマウントされている場合、ユーザーは (getfacl)で ACL を表示することはできませんが、 (setfacl)で それをセットすることはできません。
data=[ordered writeback]	data=ordered がセットされている場合、トランザクションで変更されたユーザーデータは、トランザクションがディスクにコミットされる前にディスクに流されます。これによりクラッシュの後にファイル内の初期化していないブロックをユーザーが見れないようにするはずで data=writeback モードがセットされていると、ユーザーデータは、それが変更された後でもいつでもディスクに書き込まれます。これは、 ordered モードと同等の一貫性保証は提供しませんが、一部の作業負荷では、少々速くなります。デフォルト値は ordered モードです。
ignore_local_fs 注意: このオプションは、GFS2 ファイルシステムが共有してある場合には 使用すべきではありません 。	GFS2 がファイルシステムをマルチホストファイルシステムとして扱うように強制します。デフォルトでは、 lock_nolock の使用が自動的に localcaching と localflocks のフラグをオンにします。
localcaching 注意: このオプションは、GFS2 ファイルシステムが共有してある場合には 使用すべきではありません 。	GFS2 に対してそれがローカルファイルシステムとして実行していることを伝えます。そうすると GFS2 はクラスターモードで実行している時には 利用できなかった最適化機能の選択をオンにします。 localcaching フラグは lock_nolock によって自動的にオンにされます。
localflocks 注意: このオプションは、GFS2 ファイルシステムが共有されている場合には 使用すべきではありません 。	GFS2 に対して VFS (virtual file system) レイヤーに全ての flock と fcntl を させるように告げます。 localflocks フラグは lock_nolock によって自動的にオンにされます。
lockproto=LockModuleName	ユーザーが、ファイルシステムで使用するロックングプロトコルを指定できるようにします。 LockModuleName が指定されていない場合、ファイルシステムのスーパーブロックからロックングプロトコル名が読み込まれます。
locktable=LockTableName	ユーザーがファイルシステムで使用するロックング表を指定できるようにします。

オプション	説明
quota=[off/account/on]	ファイルシステムの quota でオン/オフを切り替えます。 account の状態の中に quota をセットすると UID/GID 毎の使用統計がファイルシステムで正しく維持され、limit (限界) と warn (警告) の値は無視されます。デフォルト値は off です。
upgrade	ファイルシステムのオンディスク形式をアップグレードして、GFS2 の新バージョンで使用できるようにします。

3.3. ファイルシステムのアンマウント

GFS2 ファイルシステムは他の Linux ファイルシステムと同じ方法でアンマウントできます。 — **umount** コマンドを使用します。



注記

umount コマンドは Linux のシステムコマンドです。このコマンドに関する情報には、Linux **umount** コマンドの man ページをご覧ください。

使用法

```
umount MountPoint
```

MountPoint

GFS2 ファイルシステムがマウントされるディレクトリを指定します。

3.4. GFS2 QUOTA の管理

ファイルシステム quota (割り当て) はユーザーやグループが使用できるファイルシステムの領域量を制限するのに使われます。ユーザーやグループは quota がセットされるまで、その使用限度を持ちません。GFS2 は制限が設定されていない時でも、各ユーザーとグループで使用される領域を記録します。GFS2 はトランザクションの方法で quota 情報を更新して、システムクラッシュの後でも quota 設定が再構成を必要としないようにします。

To prevent a performance slowdown, a GFS2 node synchronizes updates to the quota file only periodically. The "fuzzy" quota accounting can allow users or groups to slightly exceed the set limit. To minimize this, GFS2 dynamically reduces the synchronization period as a "hard" quota limit is approached.

GFS2 はその **gfs2_quota** コマンドを使用して、quota を管理します。他の Linux quota 機能は GFS2 では使用できません。

3.4.1. Quota の設定

2つの quota 設定がそれぞれのユーザーID (UID) 又は、グループ ID (GID) 用に使用できます。*hard limit* (ハードリミット) と *warn limit* (警告リミット) です。

hard limit は 使用できる領域の厳格な量を示します。ファイルシステムはこのディスク領域量 以上をユーザーやグループに使用させません。hard limit の値が **zero** の場合、制限がないことを意味します。

warn limit は通常、hard limit よりも低い値です。ファイルシステムは、warn limit に到達した時点でユーザーやグループに使用している量についての警告を出します。warn limit の値が **zero** の場合、制限がないことを意味します。

これらの使用限度は **gfs2_quota** コマンドを使用してセットされます。コマンドは、GFS2 がマウントされている単独ノード上で実行するだけで十分です。

By default, quota enforcement is not set on GFS2 file systems. To enable quota accounting, use the **quota=** of the **mount** command when mounting the GFS2 file system, as described in [「Quota 強制の有効化/無効化」](#).

使用法

Setting Quotas, Hard Limit

```
gfs2_quota limit -u User -l Size -f MountPoint
```

```
gfs2_quota limit -g Group -l Size -f MountPoint
```

Setting Quotas, Warn Limit

```
gfs2_quota warn -u User -l Size -f MountPoint
```

```
gfs2_quota warn -g Group -l Size -f MountPoint
```

User

制限、又は警告する為のユーザー ID です。これはパスワードファイルからのユーザー名か、あるいは UID 番号です。

Group

制限、又は警告する為のグループ ID です。これはグループファイルからのグループ名か、あるいは GID 番号です。

Size

制限、又は警告する為の新規の値を指定します。デフォルトでこの値は、メガバイトの単位で示します。追加として **-k** や **-s** や **-b** のフラグを使用すると、この単位をそれぞれキロバイト、セクター、ファイルシステムブロックに変更します。

MountPoint

設定が適用される GFS2 ファイルシステムを指定します。

例

この例では、ユーザー *Bert* 用の hard limit は ファイルシステム **/mygfs2** で 1024 メガバイト（1 ギガバイト）にセットしています。

```
gfs2_quota limit -u Bert -l 1024 -f /mygfs2
```


この例では、グループ ID 21 用にファイルシステム `/mygfs2` で `warn limit` を 50 キロバイトにセットしています。

```
gfs2_quota warn -g 21 -l 50 -k -f /mygfs2
```

3.4.2. Quota 制限の表示と使用法

Quota 限度と現在の使用量は、`gfs2_quota get` コマンドを使用して特定のユーザーやグループ用に表示することができます。この quota ファイルの全内容は `gfs2_quota list` を使用して表示可能で、その場合、ゼロ以外の `hard limit`、`warn limit`、又はその値を持つ全ての ID が一覧表示されます。

使用法

Displaying Quota Limits for a User

```
gfs2_quota get -u User -f MountPoint
```

Displaying Quota Limits for a Group

```
gfs2_quota get -g Group -f MountPoint
```

Displaying Entire Quota File

```
gfs2_quota list -f MountPoint
```

User

特定のユーザーに関する情報を表示するユーザー ID です。これはパスワードファイルからの、又は、UID 番号からのユーザー名となります。

Group

特定のグループに関する情報を表示する為のグループ ID です。これはグループファイルからの、又は、GID 番号からのグループ名となります。

MountPoint

設定が適用される GFS2 ファイルシステムを指定します。

コマンドの出力

`gfs2_quota` コマンドからの GFS2 quota 情報は 以下のように表示されます:

```
user User: limit:LimitSize warn:WarnSize value:Value
group Group: limit:LimitSize warn:WarnSize value:Value
```

LimitSize、**WarnSize**、及び **Value** の数 (値) はデフォルトでメガバイトの単位です。 `-k` や、`-s` や、`-b` のフラグをコマンド行に付けると、その単位をそれぞれキロバイト、セクター、ファイルシステム ブロックに変換します。

User

データが関連するユーザー名、又は ID

Group

データが関連するグループ名、又は ID

LimitSize

ハードリミットはユーザー、又はグループ用にセットできます。リミットがセットされていないと、この値は 0 です。

Value

ユーザー、又はグループで使用されるディスク容量の実際のサイズ

コメント

quota 情報を表示している時、**-n** オプションがコマンドラインに追加されていると、**gfs2_quota** コマンドは、UID と GID を名前に解決しません。

Space allocated to GFS2's hidden files can be left out of displayed values for the root UID and GID by adding the **-d** option to the command line. This is useful when trying to match the numbers from **gfs2_quota** with the results of a **du** command.

例

この例は、ファイルシステム **/mygfs2** 上で制限がセットしてある、又は ディスクスペースを使用している全てのユーザーやグループの quota 情報を表示しています。

```
gfs2_quota list -f /mygfs2
```

この例は、ファイルシステム **/mygfs2** でグループ **users** 用の セクター内の quota 情報を表示しています。

```
gfs2_quota get -g users -f /mygfs2 -s
```

3.4.3. Quota の同期化

GFS2 はディスク上のそれ自身の内部ファイルにある全ての quota を保存します。GFS2 ノードは全てのファイルシステムの書き込みでこの quota ファイルを更新するのではなく、60 秒に一度だけ quota ファイルを更新します。これは、ノード群による quota ファイルへの書き込みの競合を避ける為に必要であり、そのような競合はパフォーマンス低下の原因になります。

ユーザーかグループが彼らの quota 限度に近づくと、GFS2 は動的にその quota ファイルの更新の間隔時間を短縮して限度が超過されることを防止します。quota 同期の通常の間隔はチューン可能なパラメータ **quota_quantum** で、これは **gfs2_tool** コマンドを使用して変更できます。デフォルトでは、この期間は 60 秒です。また、**quota_quantum** パラメータは各ノード上でファイルシステムがマウントされる度にセットする必要があります。(**quota_quantum** パラメータへの変更は unmount の後には固執しません)

gfs2_quota sync コマンドを使用して、GFS2 によって実行される 自動更新の時間外にノードからディスク上 quota ファイルへの quota 情報を同期化することができます。

使用法

Synchronizing Quota Information

```
gfs2_quota sync -f MountPoint
```

MountPoint

設定が適用される GFS2 ファイルシステムを指定します。

Tuning the Time Between Synchronizations

```
gfs2_tool settune MountPoint quota_quantum Seconds
```

MountPoint

設定が適用される GFS2 ファイルシステムを指定します。

Seconds

GFS2 による通常の quota ファイル同期化の間隔の新しい時間を指定します。小さな値は競合を増加するのでパフォーマンスを低下することになります。

例

この例は、実行しているノードからの quota 情報を ファイルシステム **/mygfs2** へ同期化します。

```
gfs2_quota sync -f /mygfs2
```

この例は、通常の quota ファイル更新の間隔時間を、単独ノード上のファイルシステム **/mygfs2** 用に 1 時間 (3600 秒) に変更します。

```
gfs2_tool settune /mygfs2 quota_quantum 3600
```

3.4.4. Quota 強制の有効化/無効化

GFS2 ファイルシステムでは、デフォルトで quota 強制は無効になっています。ファイルシステム用に quota 強制を有効にするには、**quota=on** オプションを指定してファイルシステムをマウントします。

使用法

```
mount -o quota=on BlockDevice MountPoint
```

quota 強制を無効にした状態でファイルシステムをマウントするには、**quota=off** オプションを指定してファイルシステムをマウントします。これはデフォルトの設定です。

```
mount -o quota=off BlockDevice MountPoint
```

-o quota={on|off}

ファイルシステムがマウントされた時に Quota 強制が有効か無効かを指定します。

BlockDevice

GFS2 ファイルシステムが存在するブロックデバイスを指定します。

MountPoint

GFS2 ファイルシステムがマウントされるディレクトリを指定します。

例

この例では、**/dev/vg01/lvol0** の GFS2 ファイルシステムは quota 強制を有効にして **/mygfs2** ディレクトリにマウントされています。

```
mount -o quota=on /dev/vg01/lvol0 /mygfs2
```

3.4.5. Quota アカウントの有効化

使用限度の強制や警告値を使用しないで各ユーザーやグループ用にディスク使用の管理と quota アカウントの維持が可能です。これを行うには、**quota=account** オプションを指定してファイルシステムをマウントします。

使用法

```
mount -o quota=account BlockDevice MountPoint
```

-o quota=account

quota limit が強制されていない場合でもファイルシステムでユーザーとグループのディスク使用統計が維持されることを指定します。

BlockDevice

GFS2 ファイルシステムが存在するブロックデバイスを指定します。

MountPoint

GFS2 ファイルシステムがマウントされるディレクトリを指定します。

例

この例では、**/dev/vg01/lvol0** 上の GFS2 ファイルシステムが quota アカウントを有効にした状態で **/mygfs2** ディレクトリにマウントされています。

```
mount -o quota=account /dev/vg01/lvol0 /mygfs2
```

3.5. ファイルシステムの拡張

gfs2_grow コマンドを使用すると、ファイルシステムが存在するデバイスが拡大された後に、GFS2 ファイルシステムを拡張することができます。既存の GFS2 ファイルシステム上で **gfs2_grow** コマンドを実行することで、ファイルシステムの現在の限界とデバイスの限界との間の余分領域を新規に

初期化された GFS2 ファイルシステム拡張で埋めることができます。この埋め込み操作が終了すると、ファイルシステムのリソースインデックスが更新されます。クラスター内の全てのノードはその後、追加されたストレージ領域を使用できます。

gfs2_grow コマンドはマウント済のファイルシステムで実行する必要がありますが、クラスター内の 1 つのノードで実行するだけで十分です。他の全てのノードは、拡張が達成されたことを自動的に感知して新規の領域を使い始めます。

使用法

```
gfs2_grow MountPoint
```

MountPoint

設定が適用される GFS2 ファイルシステムを指定します。

コメント

gfs2_grow コマンドを実行する前に:

- ファイルシステム上の重要なデータをバックアップします。
- **df MountPoint** コマンドの実行によって拡張されることになるファイルシステムで使用するボリュームを決定します。
- Expand the underlying cluster volume with LVM. For information on administering LVM volumes, see the **LVM Administrator's Guide**

gfs2_grow コマンドを実行した後は、**df** コマンドを使用して、ファイルシステム内で利用できる新規領域をチェックします。

例

この例では、**/mygfs2fs** ディレクトリ上のファイルシステムが拡張されています。

```
[root@dash-01 ~]# gfs2_grow /mygfs2fs
FS: Mount Point: /mygfs2fs
FS: Device:      /dev/mapper/gfs2testvg-gfs2testlv
FS: Size:       524288 (0x80000)
FS: RG size:    65533 (0xffffd)
DEV: Size:      655360 (0xa0000)
The file system grew by 512MB.
gfs2_grow complete.
```

完全な使用法

```
gfs2_grow [Options] {MountPoint | Device} [MountPoint | Device]
```

MountPoint

GFS2 ファイルシステムがマウントされているディレクトリを指定しています。

Device

ファイルシステムのデバイスノードを指定します。

表3.3「ファイルシステムを拡張している間に利用できる GFS2 特有のオプション」 describes the GFS2-specific options that can be used while expanding a GFS2 file system.

表3.3 ファイルシステムを拡張している間に利用できる GFS2 特有のオプション

オプション	説明
-h	ヘルプ。使用法について短いメッセージを表示
-q	静寂 (Quiet) 。冗長レベルを下げます。
-r MegaBytes	新規のリソースグループを指定します。デフォルトサイズは 256MB です。
-T	テスト。全ての計算をしますが、ディスクへの書き込みとファイルシステムの拡張はしません。
-V	コマンドのバージョン情報を表示します。

3.6. ファイルシステムヘジャーナルの追加

gfs2_jadd コマンドを使用してジャーナルを GFS2 ファイルシステムに追加することができます。背後にある論理ボリュームを拡大することなく、いつでも動的に GFS2 ファイルシステムにジャーナルを追加することができます。**gfs2_jadd** コマンドはマウント済のファイルシステム上で実行する必要がありますが、クラスター内の1つのノードで実行するだけで十分です。他の全てのノードは拡張が達成されたことを感知します。

ジャーナルを GFS ファイルシステムに追加する前に、**gfs2_tool** コマンドで **journals** オプションを使用して、GFS2 ファイルシステムが現在保有しているジャーナルの数量を確認できます。以下の例では、**/mnt/gfs2** にマウントしてあるファイルシステム内のジャーナルの数量とサイズを表示しています。

```
[root@roth-01 ../cluster/gfs2]# gfs2_tool journals /mnt/gfs2
journal2 - 128MB
journal1 - 128MB
journal0 - 128MB
3 journal(s) found.
```

使用法

```
gfs2_jadd -j Number MountPoint
```

Number

新規に追加されるジャーナルの数を指定します。

MountPoint

GFS2 ファイルシステムがマウントされているディレクトリを指定しています。

例

この例では、1つのジャーナルが `/mygfs2` ディレクトリのファイルシステムに追加されます。

```
gfs2_jadd -j1 /mygfs2
```

この例では、2つのジャーナルが `/mygfs2` ディレクトリのファイルシステムに追加されます。

```
gfs2_jadd -j2 /mygfs2
```

完全な使用法

```
gfs2_jadd [Options] {MountPoint | Device} [MountPoint | Device]
```

MountPoint

GFS2 ファイルシステムがマウントされているディレクトリを指定しています。

Device

ファイルシステムのデバイスノードを指定します。

表3.4「ジャーナル追加時に利用できる GFS2 特有のオプション」 describes the GFS2-specific options that can be used when adding journals to a GFS2 file system.

表3.4 ジャーナル追加時に利用できる GFS2 特有のオプション

フラグ	パラメータ	説明
-h		ヘルプ。使用法について短いメッセージ表示
-J	MegaBytes	新規ジャーナルのサイズをメガバイトで指定します。デフォルトのジャーナルサイズは、128 メガバイトです。最低限サイズは 32 メガバイトです。異なるサイズのジャーナルをファイルシステムに追加するには、各サイズのジャーナル毎に gfs2_jadd コマンドを実行する必要があります。指定するサイズは切り捨て値を使用して、ファイルシステムが作成された時に指定してあるジャーナルセグメントサイズの数倍値になるようにします。
-j	Number	gfs2_jadd コマンドにより追加される新規ジャーナルの数を指定します。デフォルト値は 1 です。
-q		静寂 (Quiet)。冗長レベルを下げます。
-v		コマンドのバージョン情報を表示します。

3.7. データジャーナリング

Ordinarily, GFS2 writes only metadata to its journal. File contents are subsequently written to disk by the kernel's periodic sync that flushes file system buffers. An **fsync()** call on a file causes the file's data to be written to disk immediately. The call returns when the disk reports that all data is safely written.

データジャーナリングでは、メタデータに加えて更にファイルデータがジャーナルに書き込まれるため、特に小さなファイルでは **fsync()** 時間を低減する結果となります。**fsync()** はデータがジャーナルに書き込まれるとすぐに戻ります。これによりファイルデータをメインのファイルシステムに書き込む時間よりかなり速くなります。

fsync() に依存してファイルデータを同期するアプリケーションはデータジャーナリングによってパフォーマンスの向上を受けます。データジャーナリングはフラグ付きディレクトリ（及び、そのサブディレクトリ）内に作成された GFS2 ファイル用に自動的に有効にされます。0 の長さの既存のファイルもデータジャーナリングをオン/オフに切り替えることができます。

Enabling data journaling on a directory sets the directory to "inherit jdata", which indicates that all files and directories subsequently created in that directory are journaled. You can enable and disable data journaling on a file or a directory with either of the following methods:

- **chattr +j** あるいは、**chattr -j** コマンドをファイルかディレクトリで実行する
- ファイル又はディレクトリにある **jdata** フラグを **gfs2_tool setflag** あるいは、**gfs2_tool clearflag** コマンドでセットするか抹消する

chattr コマンドの使用が、ファイル、又はディレクトリでデータジャーナリングを有効/無効にする推奨手法です。

3.7.1. chattr コマンドを使用したデータジャーナリングの有効化/無効化

chattr コマンドを使用すると、ファイル上のデータジャーナリングを有効にしたり無効にしたりすることができます。以下のコマンド群は **/mnt/gfs2/gfs2_dir/newfile** ファイルのデータジャーナリングを有効にして、それからフラグが正しくセットされていることをチェックします。

```
[root@roth-01 ~]# chattr +j /mnt/gfs2/gfs2_dir/newfile
[root@roth-01 ~]# lsattr /mnt/gfs2/gfs2_dir
-----j--- /mnt/gfs2/gfs2_dir/newfile
```

以下のコマンド群は **/mnt/gfs2/gfs2_dir/newfile** ファイル上のデータジャーナリングを無効にして、それからフラグが正しくセットされていることをチェックします。

```
[root@roth-01 ~]# chattr -j /mnt/gfs2/gfs2_dir/newfile
[root@roth-01 ~]# lsattr /mnt/gfs2/gfs2_dir
----- /mnt/gfs2/gfs2_dir/newfile
```

また、**chattr** を使用して、ディレクトリ上に **j** フラグをセットすることもできます。このフラグがディレクトリ用にセットされると、このディレクトリに以後に作成される全てのファイルとディレクトリがジャーナル化されます。以下のコマンドのセットは **gfs2_dir** ディレクトリに **j** フラグをセットして、そのフラグが正しくセットしてあることをチェックします。その後、コマンドのセットは **/mnt/gfs2/gfs2_dir** ディレクトリに **newfile** と言う新しいファイルを作成して、そのファイル用に **j** フラグが正しくセットしてあることをチェックします。**j** フラグはそのディレクトリ用にセットしてあるため、**newfile** もジャーナリングが有効になっているはずで

```
[root@roth-01 ~]# chattr -j /mnt/gfs2/gfs2_dir
[root@roth-01 ~]# lsattr /mnt/gfs2
-----j--- /mnt/gfs2/gfs2_dir
```



```
[root@roth-01 ~]# touch /mnt/gfs2/gfs2_dir/newfile
[root@roth-01 ~]# lsattr /mnt/gfs2/gfs2_dir
-----j--- /mnt/gfs2/gfs2_dir/newfile
```

3.7.2. gfs2_tool コマンドでデータジャーナリングを有効/無効にする

以下のコマンドは GFS2 ファイル `/mnt/gfs2/gfs2file` でデータジャーナリングを有効にします。

```
[root@roth-01 ~]# gfs2_tool setflag jdata /mnt/gfs2/gfs2file
```

`lsattr` コマンドを使用すると、`jdata` フラグがセットされたことを確認できます。

```
[root@roth-01 ~]# lsattr /mnt/gfs2
-----j--- /mnt/gfs2/gfs2file
```

ディレクトリに `jdata` フラグがセットしてあると、そのディレクトリ内で それ以後に作成される全てのファイルとディレクトリはジャーナル化されます。

以下のコマンドセットは `/mnt/gfs2` ディレクトリ内に `gfs2_dir` と呼ばれる新規のサブディレクトリを作成して、そのディレクトリに `jdata` フラグをセットし、その後、フラグが正しくセットされたことをチェックします。それからこのコマンドセットはそのディレクトリ内にファイル `gfs2file` を作成して、フラグがこのファイル用に正しくセットしてあることを確認します。

```
[root@roth-01 ~]# mkdir /mnt/gfs2/gfs2_dir
[root@roth-01 ~]# gfs2_tool setflag jdata /mnt/gfs2/gfs2_dir
[root@roth-01 ~]# lsattr /mnt/gfs2
-----j--- /mnt/gfs2/gfs2_dir
[root@roth-01 ~]# touch /mnt/gfs2/gfs2_dir/newfile
[root@roth-01 ~]# lsattr /mnt/gfs2/gfs2_dir
-----j--- /mnt/gfs2/gfs2_dir/newfile
```

ファイルやディレクトリから `jdata` フラグを抹消するには、以下の例のようにして、`gfs2_tool clearflag jdata` コマンドを使用します。

```
[root@roth-01 ~]# gfs2_tool clearflag jdata /mnt/gfs2/gfs2file
[root@roth-01 ~]# lsattr /mnt/gfs2
----- /mnt/gfs2/gfs2file
```

3.8. ATIME 更新の設定

それぞれのファイル inode とディレクトリ inode はそれに関連する 3 つのタイムスタンプを持ちます：

- `ctime` — inode が最後に変更になった時刻
- `mtime` — ファイル（又はディレクトリ）データが最後に修正された時刻
- `atime` — ファイル（又はディレクトリ）データが最後にアクセスされた時刻

GFS2 と他の Linux ファイルシステム上でデフォルト設定になっているように `atime` 更新が有効な場合は、ファイルが読み込まれる度に inode は更新される必要があります。

`atime` によって提供される情報を使用するアプリケーションが少ししかないため、これらの更新はか

なりの量の不要な書き込みトラフィックとファイルロッキングトラフィックを浪費する可能性があります。そのようなトラフィックはパフォーマンスを低下するため、**atime** 更新はオフにするか、低減するほうが良いでしょう。

atime 更新の影響を低減するのに2つの方法が使用できます:

- **relatime** (相対的 **atime**) でマウントすると、以前の **atime** 更新が **mtime** 又は **ctime** の更新より古い場合に、**atime** を更新します。
- **noatime** でのマウントは、ファイルシステム上の **atime** 更新を無効にします。

3.8.1. relatime でマウント

relatime (相対的 **atime**) Linux マウントオプションは、ファイルシステムがマウントされる時点に指定できます。これは以前の **atime** 更新が **mtime** あるいは **ctime** の更新より古い場合に **atime** を更新するように指定します。

使用法

```
mount BlockDevice MountPoint -o relatime
```

BlockDevice

GFS2 ファイルシステムが存在するブロックデバイスを指定します。

MountPoint

GFS2 ファイルシステムがマウントされるディレクトリを指定します。

例

この例では、GFS2 ファイルシステムは **/dev/vg01/lvol0** に存在し、ディレクトリ **/mygfs2** にマウントしてあります。**atime** 更新は以前の **atime** 更新が **mtime** 又は **ctime** の更新より古い場合にのみ実行されます。

```
mount /dev/vg01/lvol0 /mygfs2 -o relatime
```

3.8.2. noatime でマウント

Linux マウントオプション **noatime** は、ファイルシステムのマウント時に指定できます。これはファイルシステムでの **atime** 更新を無効にします。

使用法

```
mount BlockDevice MountPoint -o noatime
```

BlockDevice

GFS2 ファイルシステムが存在するブロックデバイスを指定します。

MountPoint

GFS2 ファイルシステムがマウントされるディレクトリを指定します。

例

この例では、GFS2 ファイルシステムは `/dev/vg01/lvol0` に存在し、ディレクトリ `/mygfs2` に `atime` 更新を オフ にしてマウントしてあります。

```
mount /dev/vg01/lvol0 /mygfs2 -o noatime
```

3.9. ファイルシステム上の動作の休止

`gfs2_tool freeze` コマンドを使用すると、ファイルシステムへの書き込み動作を休止することができます。書き込み動作の休止により、ハードウェアベースデバイスのスナップショットを使用してファイルシステムを一貫した状態でキャプチャーできるようになります。`gfs2_tool unfreeze` コマンドで休止を解除します。

使用法

Start Suspension

```
gfs2_tool freeze MountPoint
```

End Suspension

```
gfs2_tool unfreeze MountPoint
```

MountPoint

ファイルシステムを指定します。

例

この例では、ファイルシステム `/mygfs2` への書き込みを休止します。

```
gfs2_tool freeze /mygfs2
```

この例はファイルシステム `/mygfs2` への書き込み休止を終了します。

```
gfs2_tool unfreeze /mygfs2
```

3.10. ファイルシステムの修復

ファイルシステムをマウントしたまま、ノードが故障すると、ファイルシステムジャーナリングにより、迅速な復元が可能になります。しかし、ストレージデバイスが電源を無くしたり、物理的に切断されると、ファイルシステムの破損が発生する可能性があります。（ジャーナリングはストレージサブシステムの故障からの復元には使用できません）この種の破損が発生した場合、GFS2 ファイルシステムは `fsck.gfs2` コマンドの使用で復元することができます。



警告

fsck.gfs2 コマンドは、全てのノードからアンマウントされた ファイルシステム上でのみ実行される必要があります。

注記

以前に GFS ファイルシステムで **gfs_fsck** コマンドを使用した経験がある場合、**fsck.gfs2** コマンドは以下の部分に於いて **gfs_fsck** の以前のリリースから異なることに注意して下さい:

- **Ctrl+C** でのインタラクティブモードはセットできません。**Ctrl+C** を押すと、**fsck.gfs2** コマンドをキャンセルします。キャンセルしたくない場合は、**Ctrl+C** を押さないでください。
- **-v** フラグを使用して冗長レベルを高めることができます。2つめの **-v** フラグを追加するとそのレベルを更に高めます。
- **-q** フラグを使用して冗長レベルを低下することができます。2つめの **-q** フラグを追加すると更にそのレベルを低下できます。
- **-n** オプションの使用でファイルシステムを読み込み専用として開き、クエリに対して自動的に **no** と答えます。このオプションは **fsck.gfs2** コマンドの効力を発揮しない状態でエラーを見つけるためのコマンドの試行法を提供します。

その他のコマンドオプションについての情報には、**gfs2.fsck** の man ページを参照して下さい。

使用法

```
fsck.gfs2 -y BlockDevice
```

-y

-y フラグは、全ての質問に対して **yes** で答えるようにします。**-y** フラグが指定してあると、**fsck.gfs2** コマンドは 変更するまではプロンプトで答えを要求しません。

BlockDevice

GFS2 ファイルシステムが存在するブロックデバイスを指定します。

例

この例では、ブロックデバイス **/dev/testvol/testlv** に存在する GFS2 ファイルシステムが修復されています。修復への全てのクエリは自動的に **yes** で回答されます。

```
[root@dash-01 ~]# fsck.gfs2 -y /dev/testvg/testlv
Initializing fsck
Validating Resource Group index.
```

```

Level 1 RG check.
(level 1 passed)
Clearing journals (this may take a while)...
Journals cleared.
Starting pass1
Pass1 complete
Starting pass1b
Pass1b complete
Starting pass1c
Pass1c complete
Starting pass2
Pass2 complete
Starting pass3
Pass3 complete
Starting pass4
Pass4 complete
Starting pass5
Pass5 complete
Writing changes to disk
fsck.gfs2 complete

```

3.11. BIND マウントとコンテキスト依存のパス名

GFS2 ファイルシステムはコンテキスト依存のパス名 (CDPN) をサポートしません。CDPN はユーザーが使用できる目的地ファイルやディレクトリへポイントするシンボリックリンクを作成できるようにするものです。GFS2 内でのこの機能には、`mount` コマンドで `bind` オプションを使います。

`mount` コマンドの `bind` オプションにより、ユーザーはファイル階級の一部を本来の場所でまだ使用できる間に別の場所で再マウントできるようになります。このコマンドの形式は以下ようになります。

```
mount --bind olddir newdir
```

このコマンドを実行した後、`olddir` ディレクトリの内容は、2つの場所、`olddir` と `newdir` で利用できるようになります。また、このオプションを使用して個別ファイルを2つの場所で利用できるようになることが可能です。

例えば、以下のコマンド群を実行した後は、`/root/tmp` の内容は以前にマウントされていた `/var/log` ディレクトリの内容と同一になります。

```

[root@mencryfa ~]# cd ~root
[root@mencryfa ~]# mkdir ./tmp
[root@mencryfa ~]# mount --bind /var/log /tmp

```

別の方法として、`/etc/fstab` ファイル内のエントリを使用して、マウント時に同じ結果を達成できます。以下の `/etc/fstab` エントリは `/root/tmp` の内容が `/var/log` ディレクトリの内容と同一になる結果を得ます。

```

/var/log                /root/tmp              none    bind
0 0

```

ファイルシステムをマウントした後、`mount` コマンドを使用して、以下の例のようにファイルシステムがマウントされていることを確認できます。

■

```
[root@mencryfa ~]# mount | grep /tmp
/var/log on /root/tmp type none (rw,bind)
```

コンテキスト依存のパス名をサポートするファイルシステムでは、**/bin** ディレクトリを コンテキスト依存のパス名として定義しておく、それがシステムアーキテクチャに応じて以下のパスの1つへ 解決されるでしょう。

```
/usr/i386-bin
/usr/x86_64-bin
/usr/ppc64-bin
```

これと同じ機能は空の **/bin** ディレクトリを作成することで 達成できます。その後は、スクリプトか、**/etc/fstab** ファイル内の エントリを使用して、**mount -bind** コマンドで、個別アーキテクチャディレクトリの それぞれを **/bin** ディレクトリにマウントできます。例えば、スクリプト内の1行として 以下のコマンドを使用することができます。

```
mount --bind /usr/i386-bin /bin
```

別の方法として、**/etc/fstab** ファイル内に以下のエントリを使用できます。

```
/usr/i386-bin          /bin          none          bind          0 0
```

bind マウントはコンテキスト依存のパス名よりもより大きな柔軟性を提供します。この機能を使用して、定義する如何なる基準（ファイルシステムの **%fill** の値など）に応じて異なるディレクトリにマウントすることができます。コンテキスト依存のパス名はそれらが担当できるものに限定されます。しかし、**%fill** の値などの基準に応じてマウントの為の自分のスクリプトを書く必要があることに注意してください。



警告

オリジナルのファイルシステムが **rw** でマウントされている場合、**ro** フラグを使用して **bind** オプションでファイルシステムをマウントしても、新規のファイルシステムは **rw** でマウントされます。**ro** フラグは少々無視されます。このケースでは、新規のファイルシステムは **/proc/mounts** ディレクトリ内で **ro** とマークが付くかもしれませんが、これは間違いとなります。

付録A ファイルシステムを GFS から GFS2 に変換

GFS ファイルシステムを GFS2 ファイルシステムに変換するには、**gfs2_convert** コマンドを使用します。

1. クラスタ内の全てのノード群から GFS ファイルシステムをアンマウントします。
2. GFS ファイルシステム上で **gfs_fsck** コマンドを実行して、ファイルシステムに破損がないことを確認します。
3. 既存の GFS ファイルシステムを再マウントしてバックアップを取ります。
4. ファイルシステムをアンマウントして **gfs2_convert gfsfilesystem** を実行します。システムは警告と確認の質問を表示してから、**gfsfilesystem** を GFS2 に変換します。

以下の例では ブロックデバイス **/dev/testvg/testlv** 上の GFS ファイルシステムを GFS2 ファイルシステムに変換しています。

```
[root@dash-01 ~]# gfs2_convert /dev/testvg/testlv
gfs2_convert version 2 (built Sep 25 2007 12:41:29)
Copyright (C) Red Hat, Inc. 2004-2006 All rights reserved.

This program will convert a gfs1 filesystem to a gfs2 filesystem.
WARNING: This can't be undone. It is strongly advised that you:

    1. Back up your entire filesystem first.
    2. Run gfs_fsck first to ensure filesystem integrity.
    3. Make sure the filesystem is NOT mounted from any node.
    4. Make sure you have the latest software versions.
Convert /dev/testvg/testlv from GFS1 to GFS2? (y/n)y
Converting resource groups.
Converting inodes.
6 inodes converted.
Fixing file and directory information.
1 directories, 2 dirents fixed.
Converting journals.
Converting journal space to rg space.
Building system structures.
Removing obsolete gfs1 structures.
Committing changes to disk.
/dev/gfs2testvg/gfs2testlv: filesystem converted successfully to gfs2.
[root@dash-01 ~]#
```

付録B 改訂履歴

改訂 3-10.400 Rebuild with publican 4.0.0	2013-10-31	Rüdiger Landmann
改訂 3-10 Rebuild for Publican 3.0	2012-07-18	Anthony Towns
改訂 1.0-0	Thu Jan 29 2009	

索引

A

adding journals to a file system, [ファイルシステムヘジャーナルの追加](#)

atime, configuring updates, [atime 更新の設定](#)

mounting with noatime , [noatime](#) でマウント

mounting with relatime , [relatime](#) でマウント

audience, [対象者](#)

B

bind mounts, [Bind](#) マウントとコンテキスト依存のパス名

C

configuration, before, [GFS2](#) を設定する前に

configuration, initial, [操作の開始](#)

prerequisite tasks, [事前要求タスク](#)

D

data journaling, [データジャーナリング](#)

F

feedback, [フィードバック](#)

file system

adding journals, [ファイルシステムヘジャーナルの追加](#)

atime, configuring updates, [atime 更新の設定](#)

mounting with noatime , [noatime](#) でマウント

mounting with relatime , [relatime](#) でマウント

bind mounts, [Bind](#) マウントとコンテキスト依存のパス名

context-dependent path names (CDPNs), [Bind](#) マウントとコンテキスト依存のパス名

data journaling, [データジャーナリング](#)

growing, [ファイルシステムの拡張](#)

making, [ファイルシステムの作成](#)

mounting, [ファイルシステムのマウント](#)

quota management, [GFS2 Quota](#) の管理

displaying quota limits, [Quota](#) 制限の表示と使用法

enabling quota accounting, [Quota](#) アカウントの有効化

enabling/disabling quota enforcement, [Quota](#) 強制の有効化/無効化

setting quotas, [Quota](#) の設定

synchronizing quotas, [Quota](#) の同期化

repairing, [ファイルシステムの修復](#)

suspending activity, [ファイルシステム上の動作の休止](#)
unmounting, [ファイルシステムのアンマウント](#)

G

GFS2

atime, configuring updates, [atime 更新の設定](#)
mounting with noatime , [noatime](#) でマウント
mounting with relatime , [relatime](#) でマウント

managing, [GFS2 の管理](#)

quota management, [GFS2 Quota の管理](#)

displaying quota limits, [Quota 制限の表示と使用法](#)

enabling quota accounting, [Quota アカウントの有効化](#)

enabling/disabling quota enforcement, [Quota 強制の有効化/無効化](#)

setting quotas, [Quota の設定](#)

synchronizing quotas, [Quota の同期化](#)

GFS2 file system maximum size, [GFS2 の概要](#)

GFS2-specific options for adding journals table, [完全な使用法](#)

GFS2-specific options for expanding file systems table, [完全な使用法](#)

growing a file system, [ファイルシステムの拡張](#)

I

initial tasks

setup, initial, [初期セットアップのタスク](#)

introduction, [はじめに](#)

audience, [対象者](#)

M

making a file system, [ファイルシステムの作成](#)

managing GFS2, [GFS2 の管理](#)

maximum size, GFS2 file system, [GFS2 の概要](#)

mkfs.gfs2 command options table, [オプション全種](#)

mount table, [完全な使用法](#)

mounting a file system, [ファイルシステムのマウント](#)

O

overview, [GFS2 の概要](#)

configuration, before, [GFS2 を設定する前に](#)

P

path names, context-dependent (CDPNs), [Bind マウントとコンテキスト依存のパス名](#)

preface (参照 [introduction](#))

prerequisite tasks

configuration, initial, [事前要求タスク](#)

Q

quota management, [GFS2 Quota の管理](#)

displaying quota limits, [Quota 制限の表示と使用法](#)

enabling quota accounting, [Quota アカウムの有効化](#)

enabling/disabling quota enforcement, [Quota 強制の有効化/無効化](#)

setting quotas, [Quota の設定](#)

synchronizing quotas, [Quota の同期化](#)

R

repairing a file system, [ファイルシステムの修復](#)

S

setup, initial

initial tasks, [初期セットアップのタスク](#)

suspending activity on a file system, [ファイルシステム上の動作の休止](#)

T

tables

GFS2-specific options for adding journals, [完全な使用法](#)

GFS2-specific options for expanding file systems, [完全な使用法](#)

mkfs.gfs2 command options, [オプション全種](#)

mount options, [完全な使用法](#)

U

unmounting a file system, [ファイルシステムのアンマウント](#)