



中标软件

中标麒麟高级服务器操作系统 V6.0

GFS2 用户使用手册

中标软件有限公司

2014 年 4 月 20 日

目录

1 GFS2 概述	1
1.1 新的和更改的功能	2
1.1.1 中标麒麟高级服务器操作系统新的和更改的功能	2
1.2 设置 GFS2 前	3
1.3 GFS 和 GFS2 之间的差别	3
1.3.1 GFS2 命令名称	4
1.3.2 GFS 和 GFS2 之间的其它不同之处	5
1.3.3 GFS2 性能改进	6
1.4 GFS2 节点锁定	8
1.4.1 使用 GFS2 执行调节	9
1.4.2 使用 GFS2 锁定转储排除 GFS2 性能故障	10
2 开始	13
2.1 前提任务	13
2.2 初始设定任务	13
2.3 部署 GFS2 集群	14
3 管理 GFS2	15
3.1 生成文件系统	15
3.2 挂载文件系统	18
3.3 卸载文件系统	21
3.4 3.4. 挂载 GFS2 文件系统时的具体注意事项	21
3.5 GFS2 配额管理	22
3.5.1 配置磁盘配额	22
3.5.1.1 将配额设定为强制或者计数模式	22
3.5.1.2 创建配额数据库文件	23
3.5.1.3 为每个用户分配配额	24
3.5.1.4 为每个组群分配配额	25
3.5.2 管理磁盘配额	25
3.5.3 保持配额准确	26
3.5.4 使用 quotasync 命令同步配额	26
3.5.5 参考	27
3.6 增大的文件系统	28
3.7 在文件系统中添加日志	29
3.8 数据日志	31
3.9 配置 atime 更新	32

3.9.1 使用 relatime 挂载	32
3.9.2 使用 noatime 挂载.....	33
3.10 在文件系统中挂起一个动作.....	33
3.11 修复文件系统.....	34
3.12 绑定挂载以及上下文关联路径名.....	36
3.13 绑定挂载和文件系统挂载顺序.....	37
3.14 GFS2 撤回功能	39
4 诊断并修正 GFS2 文件系统的问题	42
4.1 GFS2 文件系统出现性能缓慢	42
4.2 通过 GFS2 设定 NFS.....	42
4.3 GFS2 文件系统挂起并需要在一个节点中重启	43
4.4 GFS2 文件系统挂起并需要重启所有节点	43
4.5 GFS2 文件系统不挂载新添加的集群节点	44
4.6 空格代表在空文件系统中使用.....	44

1 GFS2 概述

中标麒麟高级服务器操作系统 GFS2 文件系统包含在弹性存储附加组件中，是固有文件系统，直接和 Linux 内核文件系统（VFS，Virtual File System）接口相关联。当作为集群文件系统使用时，GFS2 采用分布式元数据和多日志（multiple journal）。中标麒麟高级服务器操作系统只支持将 GFS2 文件系统作为高可用性附加组件部署使用。

 注意：虽然 GFS2 文件系统可部署为独立的系统或者作为集群配置的一部分，但在中标麒麟高级服务器操作系统中不支持将 GFS2 作为单节点文件系统使用。中标麒麟高级服务器操作系统支持很多高性能单节点文件系统，这些系统是为单节点优化的，因此一般比集群文件系统的消耗要小。建议在只需要将单一节点挂载到文件系统的情况下，首先使用这些文件系统，而不是 GFS2。

中标麒麟高级服务器操作系统将继续支持单一节点 GFS2 文件系统挂载集群文件系统快照（例如：用于备份）。

 注意：中标麒麟高级服务器操作系统不支持使用 GFS2 部署超过 16 个节点的集群文件系统。

GFS2 是基于 64 位构架，理论上可提供 8EB 文件系统。但是，目前 64 位硬件支持的最大 GFS2 文件系统为 100TB，32 位硬件支持的最大 GFS2 文件系统为 16TB。如果您的系统要求更大的 GFS2 文件系统，请联络您的产品服务负责人。

当决定您的文件系统大小时，您应该考虑您的恢复需求。在一个很大的文件系统中运行 `fsck.gfs2` 命令需要很长时间且消耗大量内存。另外，在磁盘或者磁盘子系统失败事件中，恢复时间受您的备份介质速度的限制。有关 `fsck.gfs2` 所需内存量的详情请参考“3.11 修复文件系统”。

当在集群中进行配置时，可使用高可用性附加组件配置和管理工具对中标麒麟高级服务器操作系统 GFS2 节点进行配置和管理。然后中标麒麟高级服务器操作系统 GFS2 会在集群的 GFS2 节点间提供数据共享，即在 GFS2 节点间单一、一致的文件系统名称查看。这样就允许不同节点中的进程可以共享 GFS2 文件，其方法与同一节点中的进程可共享本地文件系统中的文件相同，没有明显的差

别。

当在 LVM 之上使用 GFS2 文件系统时，中标麒麟高级服务器操作系统只支持在 CLVM 逻辑卷中创建的 GFS2 文件系统。CLVM 包含在弹性存储附加组件中，这是在集群范围内部署 LVM，由在集群中管理 LVM 逻辑卷的 CLVM 守护进程 clvmd 启用。该守护进程可使 LVM2 跨集群管理逻辑卷，允许集群中的所有节点共享该逻辑卷。有关 LVM 卷管理器的详情请参考《LVM 管理员指南》。

gfs2.ko 内核模块用于部署 GFS2 文件系统，并在 GFS2 集群节点中载入。



注意：当您将 GFS2 文件系统配置为集群文件系统时，您必须确定该集群中的所有节点都可访问共享存储。在不对称集群配置中，不支持这样的情况，即有些节点可访问共享存储而另一些不可以。这并不需要所有节点都确实挂载到 GFS2 文件系统中。

1.1 新的和更改的功能

本小节列出了中标麒麟高级服务器操作系统初始发行版本以及后续版本中所包含的 GFS2 文件系统以及 GFS2 文档的新功能和更改的功能。

1.1.1 中标麒麟高级服务器操作系统新的和更改的功能

中标麒麟高级服务器操作系统包括以下文档和功能更新和更改。

- 1) 在中标麒麟高级服务器操作系统 V6.4 中，中标麒麟高级服务器操作系统不支持将 GFS2 作为单节点文件系统使用。
- 2) 在中标麒麟高级服务器操作系统 V6.4 中，GFS2 文件系统中的 gfs2_convert 命令较之在 GFS 中有所改进。
- 3) 中标麒麟高级服务器操作系统支持 discard、nodiscard、barrier、nobarrier、quota_quantum、statfs_quantum 和 statfs_percent 挂载选项。有关挂载 GFS2 文件系统的详情请参考“3.2 挂载文件系统”。
- 4) 中标麒麟高级服务器操作系统的这个文档包含一个新的部分“1.4GFS2 节点锁定”。这部分描述了一些 GFS2 文件系统的内置功能。
- 5) 从中标麒麟高级服务器操作系统 V5 开始，GFS2 支持标准 Linux 配额工具。GFS2 配额管理文档请参考“3.5GFS2 配额管理”。
- 6) 在中标麒麟高级服务器操作系统之前的发行本中，GFS2 需要使用

gfs2_quota 命令管理配额。使用 gfs2_quota 命令执行 GFS2 配额管理中记录了 gfs2_quota 命令。

- 7) 这个文档现在包含一个新章节,“4 诊断并修正 GFS2 文件系统的问题”。
- 8) 对整篇文档进行了技术修正和澄清。

1.2 设置 GFS2 前

在您安装和设定 GFS2 前,请记录您的 GFS2 文件系统的以下重要功能:

1) GFS2 节点

决定集群中的哪个节点将挂载 GFS2 文件系统。

2) 文件系统数

决定最初创建多少个 GFS2 文件系统(以后可以添加更多的文件系统)。

3) 文件系统名称

为每个文件系统取一个唯一名称。该名称必须与集群中的所有 lock_dlm 文件系统不同。每个文件系统名称都要求使用参数变量格式。例如:在本手册中的一些示例过程中使用 mydata1 和 mydata2。

4) 日志

决定 GFS2 文件系统的日志数。每个挂载到 GFS2 文件系统的节点都需要一个日志。GFS2 允许您以后动态添加日志作为挂载文件系统的附加服务器。有关在 GFS2 文件系统中添加日志的详情请参考“3.7 在文件系统中添加日志”。

5) 存储设备和分区

决定在文件系统中用来创建逻辑卷(使用 CLVM)的存储设备和分区。



注意:当同时在同一个目录中进行多个生成和删除操作时,您可能会看到 GFS2 的性能下降。如果这导致您系统性能出现问题,您应该尽量将某个节点中的文件生成和删除动作定位到具体的目录。

1.3 GFS 和 GFS2 之间的差别

这部分列出了相比 GFS, GFS2 提供的改进和更改。

从 GFS 迁移到 GFS2,您需要使用 gfs2_convert 程序将 GFS 文件系统转换为 GFS2。

1.3.1 GFS2 命令名称

通常 GFS2 和 GFS 的功能是一致的,只是在文件系统命令的名称中要将 GFS 改为 GFS2。表 1-1 中显示了功能等同的 GFS 和 GFS2 命令。

表 1-1 GFS 和 GFS2 命令

GFS 命令	GFS2 命令	描述
mount	mount	挂载文件系统。系统可以确定文件系统的类型是 GFS 还是 GFS2。有关 GFS2 挂载选项的详情请参考 gfs2_mount(8)man page。
umount	umount	卸载文件系统
fsck gfs_fsck	fsck fsck.gfs2	检查并修复卸载的文件系统。
gfs_grow	gfs2_grow	增大挂载的文件系统。
gfs_jadd	gfs2_jadd	在某个挂载的文件系统中添加日志
gfs_mkfs mkfs -t gfs	mkfs.gfs2 mkfs -t gfs2	在存储设备中创建文件系统。
gfs_quota	gfs2_quota	在挂载的文件系统中管理配额。
gfs_tool	gfs2_tool	配置、调试文件系统或者收集有关文件系统的信息。
gfs_edit	gfs2_edit	显示、输出或者编辑文件系统内部结构。gfs2_edit 命令可用于 GFS 文件系统,也可用于 GFS2 文件系统。
gfs_tool setflag jdata/ inherit_jdata	chattr +j (preferred)	在文件或者目录中启用日志功能。
setfacl/getfacl	setfacl/getfacl	为文件或者目录设置或者获得文件访问控制。

setfattr/ getfattr	setfattr/ getfattr	设置或者获得文件的扩展属性。
-----------------------	-----------------------	----------------

GFS2 文件系统命令支持选项的完整列表请参考那些命令的 man page。

1.3.2 GFS 和 GFS2 之间的其它不同之处

这部分总结了在“1.3.1GFS2 命令名称”中没有论述的 GFS 和 GFS2 管理中的其它不同之处。

1) 上下文关联路径名

GFS2 文件系统不提供对上下文关联路径名的支持，上下文关联路径名允许您生成指向各种目的地文件或者目录的符号链接。在 GFS2 中，您可以使用 mount 命令的 bind 选项来实现这个功能。有关在 GFS2 中管理路径名的详情请参考“3.12 绑定挂载以及上下文关联路径名”。

2) gfs2.ko 模块

使用 GFS 文件系统的内核模块是 gfs.ko。使用 GFS2 文件系统的内核模块是 gfs2.ko。

3) 在 GFS2 中启用配额强制

在 GFS2 文件系统中，默认情况是禁用配额强制，且必须单独启用。有关启用和禁用配额强制的详情请参考“3.5GFS2 配额管理”。

4) 数据日志

GFS2 文件系统支持使用 chattr 命令设定和清除某个文件或者目录的 j 标签。在某个文件中设定+j 标签会启用该文件的数据日志记录。在某个目录中设定+j 标签意味着“继承 jdata”，即对之后所有在那个目录中生成的文件或者目录进行日志记录操作。使用 chattr 命令是在某个文件中启用和禁用数据日志记录的首选方法。

5) 动态添加日志

在 GFS2 文件系统中，日志是存在于文件系统之外的内嵌元数据，这就需要在添加日志前扩展包含该文件系统的逻辑卷大小。在 GFS2 文件系统中，日志是纯文本文件（虽然是隐藏的）。这意味着对于 GFS2 文件系统来说，只要在该文件系统中可放置附加日志的空间，即可动态添加日志将其作为附加服务器挂载到文件系统中。有关在 GFS2 文件系统中添加日志的详情请参考“3.7 在文件系

统中添加日志”。

6) 删除 atime_quantum 参数

GFS2 文件系统不支持 atime_quantum 可调节参数，GFS 文件系统可使用该参数指定 atime 更新的频率。在 GFS2 中支持 relatime 和 noatime 挂载选项。建议您使用 relatime 挂载选项获得与在 GFS 中使用 atime_quantum 参数时得到的类似行为。

7) mount 命令的 data 选项

在挂载 GFS2 文件系统时，您可以指定 mount 命令的 data=ordered 或者 data=writeback 选项。当设定 data=ordered 时，传输修改的用户数据会在传输被提交到磁盘前被冲入磁盘。这样可以让用户无法在崩溃后的文件中看到未初始化的块。设定 data=writeback 时，用户数据会在磁盘有数据后的任何时候被写入磁盘。这不会提供 ordered 模式可提供的一致性保障，但对某些工作负载来说可稍微加快一些速度。默认设置是 ordered 模式。

8) gfs2_tool 命令

gfs2_tool 为 GFS2 支持的选项组与 gfs_tool 命令为 GFS 支持的选项组不同：

- a) gfs2_tool 命令支持 journals 参数，它可输出当前配置日志有关信息，其中包括文件系统包含的日志数。
- b) gfs2_tool 命令不支持 counters 标签，而 gfs_tool 命令可使用该标签显示 GFS 统计。
- c) gfs2_tool 命令不支持 inherit_jdata 标签。要将某个命令标记为“inherit_jdata”，您可以在目录中设定 jdata 标签，或者使用 chattr 命令在目录中设定+j 标签。使用 chattr 命令是在文件中启用和禁用数据日志记录的首选方法。

9) gfs2_edit 命令

gfs2_edit 命令为 GFS2 支持的选项组与 gfs_edit 命令为 GFS 支持的选项组不同。有关该命令每个版本具体支持的选项，请参考 gfs2_edit 和 gfs_edit man page。

1.3.3 GFS2 性能改进

GFS2 文件系统的很多功能与 GFS 文件系统在界面上没有什么不同，但文件

系统性能会有提高。GFS2 文件系统在以下方面提供改进的文件系统性能：

- 1) 在大量使用单一目录时有较好的性能。
- 2) 更快的同步 I/O 操作。
- 3) 更快的缓存读取（无锁定消耗）。
- 4) 对预先分配的文件有更快的直接 I/O（提供合理的较大 I/O 值，比如 4M 大的块）。
- 5) 普遍更快的 I/O 操作。
- 6) 执行 df 命令的速度更快，因为 statfs 调用的速度更快。
- 7) 与 GFS 相比，我们改进了 atime 模式以减少 atime 生成的写入 I/O 操作数量。

GFS2 文件系统在以下方面提供更广泛和主流的支持：

- 1) GFS2 是上游内核（整合到 2.6.19）的一部分。
- 2) GFS2 支持以下特性：
 - a) SELinux 扩展的属性。
 - b) lsattr()和 chattr()属性是通过标准 ioctl()调用设置的。
 - c) 纳秒时间戳。
- 3) GFS2 为文件系统的内在效率提供以下改进。
 - a) GFS2 使用更少的内核内存。
 - b) GFS2 需要非元数据生成数。

分配 GFS2 元数据不需要读取。多日志中的元数据块副本是由从锁定释放前的日志中调用的块管理的。

- 1) GFS2 的日志管理程序更为简单，它对未链接的内节点或者配额修改一无所知。
- 2) gfs2_grow 和 gfs2_jadd 命令使用锁定来防止多个事件同时运行。
- 3) 为类似 creat()和 mkdir()的调用简化了 ACL 编码。
- 4) 在没有重新挂载日志的情况下恢复未连接的内节点以及配额和 statfs 的更改。

1.4 GFS2 节点锁定

要获得最佳 GFS2 文件系统性能，则需要理解其操作的基本原理。单节点文件系统与缓存一同使用，其目的是在频繁使用请求的数据时可消除磁盘访问延迟。Linux 页面缓存（以及缓冲缓存）提供这个缓存功能。

使用 GFS2，每个节点都可有其自身的页面缓存，该缓存中包含 on -disk 数据的一部分。GFS2 使用 glocks（发音为 gee-locks）锁定机制维护节点间缓存的完整性。glock 子系统提供缓存管理功能，该功能使用分布式锁管理器（DLM）部署作为基础沟通层。

glocks 在每个内节点中为缓存提供保护，因此在每个内节点中都有一个锁定用来控制缓冲层。如果为那个 glock 赋予共享模式（DLM 锁定模式：PR），那么那个 glock 保护下的数据可同时被一个或者多个节点缓存，这样多有节点就都有到该数据的本地访问。

如果为 glock 赋予专用模式（DLM 锁定模式：EX），那么只有一个节点可缓存那个 glock 保护的数据。所有修改数据的操作（例如 write 系统调用）都使用这个模式。

如果另一个节点请求 glock，但无法立刻获得，那么 DLM 会向该节点发送一条信息，或者向目前使用 glock 并妨碍新的请求的节点发送信息，要求它们释放其锁定。释放 glock（大多数文件系统操作标准）需要很长时间。释放共享 glock 只需要使该缓存无效，相对缓冲的数据来说速度较快。释放专用 glock 需要 log flush，并向磁盘写回所有更改的数据，之后要使每个共享的 glock 失效。

单一节点文件系统与 GFS2 之间的区别在于单一节点文件系统只有一个缓存，而 GFS2 在每个节点中都有独立的缓存。在这两种情况下，对缓冲数据访问的延迟程度类似，但如果另一个节点之前缓冲了同样的数据，GFS2 对非缓冲数据访问的延迟要大得多。



注意：根据采用 GFS2 缓冲的方法，在出现以下任意情况之一时都会获得最佳性能：

- 1) 在所有节点中都使用只读方式使用内节点
- 2) 只在单一节点中写入或者修改内节点

请注意：在创建和删除文件的过程中插入和删除目录条目也算在内，因为要

写入到目录内节点。

也可能不遵守这个规则，但并不常发生。过度忽略这个规则会对性能有严重影响。

如果您在 GFS2 中使用 read/write 映射 mmap() 某个文件，但只读取它，那么这只计为读取。而在 GFS 中会将其计为写入，因此 GFS2 使用 mmap() I/O 时更灵活。

如果您没有设定 noatime mount 参数，那么读取也会导致写入来更新文件时间戳。文件建议所有 GFS2 用户应该使用 noatime 挂载，除非对 atime 有具体要求。

1.4.1 使用 GFS2 执行调节

通常可以更改出问题的程序保存其数据的方法，这样可获得可观的性能优势。

典型的问题程序例子有电子邮件服务器。通常会制定包含每个用户的文件的 spool 目录 (mbox)，或者为每个用户创建一个目录，其中有包含所有信息的文件 (maildir)。当有来自 IMAP 的请求时，理想的情况是为每个用户赋予一个对特定节点的亲和性。那么将会使用那个节点中的缓存处理他们查看和删除电子邮件信息的请求。显然，如果那个节点失败，那么可在不同的节点中重启该会话。

当邮件通过 SMTP 送达时，那么还是将独立节点设定为默认将某个用户的邮件传递到特定节点。如果没有启动默认节点，那么会将该信息由接收节点直接保存到用户的邮件、spool 中。同样，这个设计的目的在于在通常情况下只在一个节点中保留缓冲文件的具体设置，但在节点失败的情况下允许直接访问。

这个设置可使用最佳 GFS2 页面缓存，还可让程序 (imap 或者 smtp) 清楚地了解失败情况。备份通常是另一个令人纠结的领域。如果可能最好直接从节点备份每个节点的工作组件，这样可缓存具体的内节点组。如果您有可定期运行的备份脚本，且与在 GFS2 中运行的应用程序反应时间完全一致，

那么很有可能集群无法最有效地使用页面缓存。

显然，如果您（幸运地）可以停止该程序以便执行备份，那么这就不是个问题。否则，如果只在一个节点中执行备份，那么当它完成后，文件系统的大部分

将被缓存在该节点中，这样就对以后从其它节点访问有一定的性能损失。使用下面的命令在备份结束后放弃一些备份节点中的 VFS 页面缓存可将此影响缓解到一定程度：

```
echo -n 3 >/proc/sys/vm/drop_caches
```

但这并不是一个好的解决方案，如同要保证在每个节点中的工作组要么是共享的，大多数为集群内只读，或者是从单一节点的大量访问。

1.4.2 使用 GFS2 锁定转储排除 GFS2 性能故障

如果由于浪费 GFS2 缓冲而造成集群性能问题，您会看到大量和增长的 I/O 等待时间。您可以使用 GFS2 的锁定转储信息确定造成此问题的原因。

GFS2 锁定转储信息可从 `debugfs` 文件中获得，您可以根据路径名找到该文件，假设 `debugfs` 是挂载在 `/sys/kernel/debug/` 中：

```
/sys/kernel/debug/gfs2/fsname/glocks
```

文件的内容是一些行。每行使用 `G:` 开始，代表一个 `glock`。接下来的行使用一个空格缩进，代表文件中最新与 `glock` 关联的信息项目。

当程序出现问题时，使用 `debugfs` 文件的最佳方法是使用 `cat` 命令获得该文件内容副本（如果您的 RAM 较大，且有很多缓冲的内节点，则需要较长时间），以后再查看得到的数据。



窍门：复制两份 `debugfs` 文件会很有帮助，两个副本的间隔可在几秒甚至一分钟左右。比较与同一 `glock` 号关联的两个追踪信息，您可以了解负载是否在增长（就是说只是速度慢了）或者它在哪里卡住了（这通常是由 `bug` 造成，您应该立即向中标麒麟高级服务器操作系统支持团队提交报告）。

`debugfs` 文件中以 `H:`（拥有者）开始的行代表赋予锁定请求或者正在等待赋予锁定请求。`Holders` 行中的标签字段：`‘F’` 表示是哪一个，`‘W’` 代表等待的请求，`‘H’` 代表授予的请求。有大量等待的 `glocks` 可能正处于争论阶段。

表 1-2 “Glock 标签” 显示不同 `glock` 标签的含义，“Glock 拥有者标签” 显示不同 `glock` 拥有者标签的含义。

表 1-2 Glock 标签

标签	名称	含义
d	等待降级	递延（远程）降级请求
D	降级	降级请求（本地或者远程）
f	清除日志	释放这个 glock 前需要提交该日志
F	冻结	回复忽略的远程节点——恢复进行中。
i	使进程无效	这个 glock 下无效页面的进程中
I	启动	设定何时将 DLM 锁定与这个 glock 关联
l	锁定的	这个 glock 处于更改状态中
p	降级中	该 glock 正在响应降级请求
r	回复等待	从远程节点中接收的回复正在等待过程中
y	脏数据	释放这个 glock 前要刷新到磁盘中的数据

表 1-3 Glock 拥有者标签

标签	名称	含义
a	Async	不等待 glock 结果（以后轮询结果）
A	任意	接受任意兼容锁定模式
c	没有缓存	取消锁定时立即降级 DLM 锁定
e	没有过期	忽略随后的锁定取消请求
E	准确	必须有准确的锁定模式
F	第一	设定赋予这个锁定的第一个拥有者
H	拥有者	表示赋予请求的锁定
p	优先权	在队列头入队的拥有者

t	尝试	“try” 锁定
T	Try 1CB	发送回呼的 “try” 锁定
W	等待	等待请求完成的设置

确定造成问题的 **glock** 后，下一步是要找到关联的内节点。**glock** 号（G:行中的 n:）指的就是这个，其格式为 **type/number**，如果 **type** 是 2，那么 **glock** 就是一个内节点 **glock**，且 **number** 就是内节点号。要追踪内节点，您可以运行 **find -inum number**，其中 **number** 是将 **glock** 文件中的十六进制格式转换为十进制格式的内节点号。

 **警告：**如果您在有锁定冲突的文件系统中运行 **find**，事情可能会变得更糟糕。当您查找冲突的内节点时，最好在运行 **find** 前停止该程序。

表 1-4 Glock 类型

类型号	锁定类型	使用
1	Trans	事务锁定
2	Inode	内节点元数据和数据
3	Rgrp	员组群元数据
4	Meta	超级块
5	Iopen	内节点的最近探测
6	Flock flock(2)	系统调用
8	Quota	配额操作
9	Journal	日志互斥

如果识别的 **glock** 是不同的类型，那么最可能是类型 3:（资源组）。如果您在常规负载时看到大量进程正在等待其它 **glock** 类型，请向中标麒麟高级服务器操作系统支持团队提交报告。

如果您看到在资源组锁定中有大量等待的请求，那么可能有很多原因。其中之一是在文件系统中相对于资源组有大量的内节点。另一个原因就是该文件系统可能接近饱和（按平均计算，需要较长的搜索）。

在这两种情况下可通过添加更多存储以及使用 **gfs2_grow** 命令扩大该文件系统进行改善。

2 开始

2.1 前提任务

在设置中标麒麟高级服务器操作系统 GFS2 前，请确定您记录了 GFS2 节点的关键特征（请参考“1.2 设置 GFS2 前”）。同时还请确定 GFS2 节点的时钟是同步的。建议您使用中标麒麟高级服务器操作系统提供的网络时间协议（NTP）软件。



注意：GFS2 节点中的系统时钟间的差别必须在几分钟之内以便防止内节点时间戳更新。不必要的内节点时间戳更新会严重影响集群的性能。

2.2 初始设定任务

初始 GFS2 设定包含以下任务：

- 1) 设定逻辑卷。
- 2) 生成 GFS2 文件系统。
- 3) 挂载文件系统。

开始按照以下步骤设定 GFS2。

- 1) 使用 LVM 为每个中标麒麟高级服务器操作系统 GFS2 文件系统生成逻辑卷。



注意：您可以使用中标麒麟高级服务器操作系统集群套件中的 `init.d` 脚本自动激活和失活逻辑卷。有关 `init.d` 脚本的详情请参考《配置和管理中标麒麟高级服务器操作系统集群》。

- 2) 在第一步生成的逻辑卷中创建 GFS2 文件系统。

为每个文件系统选择唯一的名称。有关创建 GFS2 文件系统的详情请参考“3.1 生成文件系统”。

您可以使用以下格式之一创建一个集群 GFS2 文件系统：

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

有关创建 GFS2 文件系统的详情请参考“3.1 生成文件系统”。

- 3) 在每个节点中挂载 GFS2 文件系统。

有关挂载 GFS2 文件系统的详情请参考“3.2 挂载文件系统”。

用法:

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

-o acl 选项允许操作文件 ACL。如果挂载某个文件系统是没有使用 -o acl 挂载选项, 用户可以查看 ACL (使用 getfacl 命令), 但不可以设定它们 (使用 setfacl 命令)。



注意: 您可以使用中标麒麟高级服务器操作系统高可用性附加组件中的 init.d 脚本自动挂载和卸载 GFS2 文件系统。

2.3 部署 GFS2 集群

部署集群文件系统不是简单的替换单节点部署。我们建议您预留约 8-12 周的时间测试新的安装, 以便测试该系统并保证其在要求的性能水平正常工作。在此期间可解决所有性能和功能问题, 且所有疑问都可提交中标麒麟高级服务器操作系统支持团队。我们还建议考虑部署集群的客户在部署前请中标麒麟高级服务器操作系统支持团队审核其配置, 这样可以避免之后可能存在的支持问题。

3 管理 GFS2

3.1 生成文件系统

您可使用 `mkfs.gfs2` 命令创建 GFS2 文件系统。您还可以使用指定了 `-t gfs2` 选项的 `mkfs` 命令。文件系统是在活跃的 LVM 卷中创建的。运行 `mkfs.gfs2` 命令时需要以下信息：

- 1) 锁定协议/模块名称（集群的锁定协议为 `lock_dlm`）
- 2) 集群名称（当作为集群配置的一部分运行时）
- 3) 日志数目（每个可能挂载文件系统的节点都需要一个日志）

当创建 GFS2 文件系统时，您可以直接使用 `mkfs.gfs2`，或者使用带 `-t` 参数的 `mkfs` 命令，并使用 `gfs2` 文件系统选项将文件系统类型指定为 `gfs2`。

 注意：当您使用 `mkfs.gfs2` 命令创建 GFS2 文件系统时，您不能缩小该文件系统。但您可以使用 `gfs2_grow` 命令增大现有文件系统的大小，如“3.6 增大的文件系统”所述。

用法：

当创建集群的 GFS2 文件系统时，您可以使用以下任意格式之一：

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

您可以使用以下任意格式之一创建本地 GFS2 文件系统：

 注意：在中标麒麟高级服务器操作系统中，中标麒麟高级服务器操作系统不支持将 GFS2 作为单节点文件系统使用。

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

 警告：请确定您非常熟悉 `LockProtoName` 和 `LockTableName` 参数的使用。不正确的 `LockProtoName` 和 `LockTableName` 参数使用可能导致文件系统或者锁定空间崩溃。

LockProtoName

指定要使用的锁定协议名称，集群的锁定协议为 `lock_dlm`。

LockTableName

这个参数是用来指定集群配置中的 GFS2 文件系统。它有两部分，用冒号隔开（没有空格）如下：`ClusterName:FSName`

1) ClusterName

用来创建 GFS2 文件系统的集群名称。

2) FSName

文件系统名称，长度可在 1-16 个字符之间。该名称必须与集群中所有 lock_dlm 文件系统以及每个本地节点中的所有文件系统（lock_dlm 和 lock_nolock）不同。

NumberJournals

指定由 mkfs.gfs2 命令生成的日志数目。每个要挂载文件系统的节点都需要一个日志。对于 GFS2 文件系统来说，以后可以添加更多的日志而不会增大文件系统，如“3.7 在文件系统中添加日志”所述。

BlockDevice

指定逻辑卷或者物理卷。

示例：

在下述示例中，lock_dlm 是文件系统使用的锁定协议，因为这是一个集群的文件系统。集群名称为 alpha，文件系统名为 mydata1。文件系统包含八个日志，日志是在/dev/vg01/lvol0 中生成的。

```
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
```

在下述示例中，生成了第二个 lock_dlm 文件系统，它可用于集群 alpha。文件系统名为 mydata2。文件系统包含八个日志，日志是在/dev/vg01/lvol1 中生成的。

```
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
```

完整选项：

表 3-1 “命令选项：mkfs.gfs2” 描述 mkfs.gfs2 命令选项（标签和参数）。

表 3-1 命令选项:mkfs.gfs2

标签	参数	描述
-c	Megabytes	以 Megabytes 为单位设定每个日志配额更改文件的初始大小。

标签	参数	描述
-D		启用调试输出。
-h		帮助，显示可用选项。
-J	MegaBytes	以 MB 为单位指定日志大小。默认日志大小为 128MB，最小为 8MB。较大的日志可提高性能，但会比较小的日志占用更多的内存。
-j	Number	指定由 mkfs.gfs2 命令生成的日志数目。挂载文件系统的每个节点都需要一个日志。如果没有指定该选项，则会生成一个日志。对于 GFS2 文件系统，您可以后添加附加日志而不会增大文件系统。
-O		防止 mkfs.gfs2 命令在写入文件系统前进行确认。
-p	LockProtoName	指定要使用的锁定协议名称，可识别的锁定协议包括：lock_dlm—标准锁定模式，用于集群的文件系统。lock_nolock—当 GFS2 作为本地文件系统作用时使用（只有一个节点）。
-q		静默，什么都不显示。
-r	MegaBytes	以 MB 为单位指定源组群大小，最小源组群值为 32MB，最大源组群值为 2048MB。在大型文件系统中源组群越大性能越高。如果没有指定这个信息，mkfs.gfs2 会根据文件系统大小选择源组群大小：中等大小的文件系统的源组群为 256MB，大一点的文件系统会有较大的源组群以获得更好的性能。
-t	LockTableName	在您使用 lock_dlm 协议时用来指定锁定表格字段的唯一识别程序，lock_nolock 协议不使用这个参数。这个参数有两个部分，用冒号隔开（没有空格）如下：ClusterName:FSName。ClusterName 是用来创建 GFS2 文件系统的集群名称，只有集群成员有使用此文件系统的权限。集群名称可使用 Cluster Configuration Tool 在文件/etc/cluster/cluster.conf 中设定，并在中标麒麟高级服务器操作系统集群套件集群管理 GUI 的 Cluster Status Tool 中显示。

标签	参数	描述
		FSName, 文件系统名称, 长度可在 1-16 个字符之间, 且必须不同于集群中的其它文件系统名。
-u	MegaBytes	指定每个日志未链接标签文件的初始大小。
-V		显示命令版本信息。

3.2 挂载文件系统

在您挂载 GFS2 文件系统前, 该文件系统必须存在 (请参考 “3.1 生成文件系统”), 文件系统所在的卷必须是活跃的, 且必须启动了集群和锁定系统支持。达到这些要求后, 您就可以将这个 GFS2 文件系统挂载到任意 Linux 文件系统中。



注意: 通常计算机在开机时会自动启动集群管理器 (cman), 如果没有启动 cman, 请安装 cman 软件包后启动该服务。如果 cman 没有启动, 尝试挂载 GFS2 文件系统会产生以下出错信息:

```
[root@gfs-a24c-01 ~]# mount -t gfs2 -o noatime /dev/mapper/mpathap1 /mnt
gfs_controld join connect error: Connection refused
error mounting lockproto lock_dlm
```

要控制文件 ACL, 您必须使用 -o acl 挂载选项挂载文件系统。如果挂载文件系统时没有使用 -o acl 选项, 用户可以查看 ACL (使用 getfacl), 但不能对其进行设置 (使用 setfacl)。

用法:

不使用 ACL 控制挂载

```
mount BlockDevice MountPoint
```

使用 ACL 控制挂载

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

-o acl

允许控制文件 ACL 的具体 GFS2 选项。

BlockDevice

指定 GFS2 文件系统所在的块设备。

MountPoint

指定要挂载 GFS2 文件系统的目录。

示例：

在这个示例中，位于/dev/vg01/lvol0 的 GFS2 文件系统被挂载到/mygfs2 目录中。

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

完整用法：

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

-o option 参数包含 GFS2 具体选项（请参考表 3-2 “GFS2 特定挂载选项”）或者可接受的标准

Linux mount -o 选项，或者两者之和。多个 option 参数可使用逗号分开，没有空格。

 **注意：** mount 命令是 Linux 系统命令。除了使用这部分论述的 GFS2 具体选项，您还可以使用其它标准 mount 命令选项（例如：-r）。有关其它 Linux mount 命令选项请参考 mount man page。

 **注意：** 这个表格包含了只用于本地文件系统选项的描述。但请注意：在中标麒麟高级服务器操作系统发行本中中标麒麟高级服务器操作系统不支持将 GFS2 作为单节点文件系统使用。中标麒麟高级服务器操作系统将继续在挂载集群文件系统快照时支持单节点 GFS2 文件系统（例如：用于备份）。

表 3-2 “GFS2 特定挂载选项”描述在挂载时可传递给 GFS2 的 GFS2 特定-o option 选项值。

表 3-2 GFS2 特定挂载选项

选项	描述
acl	允许控制文件 ACL。如果挂载文件系统时没有使用 acl 挂载选项，那么用户可以查看 ACL（使用 getfacl），但不能对其进行设置（使用 setfacl）。

data=[ordered writeback]	当设置 data=ordered 时，事务中修改的用户数据会将该事物递交到磁盘前被冲入磁盘。这样可以让用户无法在崩溃后的文件中看到未初始化的块。设定 data=writeback 时，用户数据会在磁盘有数据后的任何时候被写入磁盘。这样无法提供 ordered 模式可提供的一致性保障，但可稍微提高一些工作负载的速度。默认设置为 ordered 模式。
ignore_local_fs 警告：在共享 GFS2 文件系统时不应该使用这个选项。	强制 GFS2 将文件系统视为多主机文件系统。默认情况是使用 lock_nolock 自动打开 localflocks 标签。
localflocks 警告：在共享 GFS2 文件系统时不应该使用这个选项。	告知 GFS2 让 VFS（虚拟文件系统）层完成所有 flock 和 fcntl 操作。lock_nolock 可自动打开 localflocks 标签。
lockproto=LockModuleName	允许用户指定文件系统要使用的锁定协议。如果没有指定 LockModuleName，则会从文件系统的超级块中读取锁定协议。
locktable=LockTableName	允许用户指定文件系统要使用的锁定表。
quota=[off/account/on]	为文件系统打开或者关闭配额。在 account 状态中配置配额可让文件系统正确维护对每个 UID/GID 使用统计，忽略限制和警告值。默认值为 off。
errors=panic withdraw	当指定 errors=panic 时，文件系统错误将导致内核崩溃。默认的行为与指定 errors=withdraw 一致，即将该系统从文件系统中退出，且直到重启前都无法访问。在有些情况下，该系统仍保持运行。有关 GFS2 退出功能请参考“3.14GFS2 撤回功能”。
discard/nodiscard	导致 GFS2 为释放的块生成“discard” I/O 请求。合适的硬件可使用这些请求部署自动精简配置以及类似的方案。
barrier/nobarrier	当清洗日志时 GFS2 会发送 I/O 屏障。默认值为 on。如果基础设备不支持 I/O 屏障则会自动将这个选项改为 off。强烈建议您随时在 GFS2 中使用 I/O 屏障，除非块设备设计为不使用，这样它就不会丢失其写入缓存内容（例如：如果它在 UPS 中或者没有写入缓存）。

quota_quantum=secs	在将更改的配额信息写入配额文件前将其保存在某个节点的秒数。这是设定此参数的首选方法。该数值是一个大于 0 的整数。默认为 60 秒。设定为较短的间隔会让配额信息更快地更新，且更不可能让某些人超过其配额。较长的间隔可让文件系统操作更迅速有效地包括配额。
statfs_quantum=secs	设置 statfs 慢速版本的首选方法是将 statfs_quantum 设定为 0。默认值为 30 秒，该值设定了将 statfs 与主 statfs 文件同步前的最大时间段。可将该值调整为更迅速但不准确的 statfs 值，也可将其设定为更慢但更准确的值。当将该选项设定为 0 时，statfs 将总是报告真实值。
statfs_percent=value	提供在没有超时前，将 statfs 信息与主 statfs 文件同步前该信息更改的最大比例值。如果将 statfs_quantum 设定为 0，那么会忽略这个设置。

3.3 卸载文件系统

可使用与卸载 Linux 文件系统相同的方法卸载 GFS2 文件系统 — 即使用 `umount` 命令。



注意：umount 是 Linux 系统命令。有关此命令的详情请参考 Linux umount 命令 man page。

用法：

```
umount MountPoint
```

MountPoint 指定当前挂载 GFS2 文件系统的目录。

3.4 挂载 GFS2 文件系统时的具体注意事项

该系统不会了解那些在系统关闭的过程中卸载的，通过手动挂载而不是使用 `fstab` 文件中的条目自动挂载的 GFS2 文件系统。因此，GFS2 脚本不会卸载 GFS2 文件系统。在运行 GFS2 关闭脚本后，标准关闭进程会杀死所有保留的用户进程，包括集群基础结构，并尝试卸载该文件系统。没有集群基础构架这个卸载会失败，且该系统会停滞。

要防止卸载 GFS2 文件系统时的系统停滞，您应该进行以下操作之一：

- 1) 总是使用 `fstab` 文件中的条目挂载 GFS2 文件系统。
- 2) 如果已经手动使用 `mount` 命令挂载了 GFS2 文件系统，请确定在重启或

者关闭该系统前手动使用 `umount` 命令卸载该文件系统。

如果在这些情况下关闭系统的过程中，卸载文件系统时该文件系统停滞，请执行硬件重启。这样倒不会丢失数据，因为在关闭进程的初期已经同步了该文件系统。

3.5 GFS2 配额管理

文件系统配额是用来限制某个用户或者组群使用的文件系统空间。在设置前对用户或者组群没有配额限制。当使用 `quota=on` 或者 `quota=account` 选项时，GFS2 会不断跟踪每个用户或者组群使用的空间，即使没有设定限制也是如此。GFS2 以互动的方式更新配额信息，因此系统崩溃并不需要重建配额用量。

为防止性能下降，GFS2 节点只会定时为配额文件更新同步。“fuzzy”配额核算可允许用户或者组群稍微超过设定的限制。为最小化这种情况，GFS2 会在接近“hard”配额限制时动态缩短同步周期。



注意：从中标麒麟高级服务器操作系统 V5 开始，GFS2 支持标准 Linux 配额工具。要使用这个工具，您需要安装 `quota RPM`。这是在 GFS2 中管理配额的首选方法，且应该在所有使用配额新部署的 GFS2 中使用。本小节记录了如何使用这些工具管理 GFS2 配额。

在中标麒麟高级服务器操作系统 V5 之前的版本中，GFS2 文件系统中的 `gfs2_quota` 命令执行 GFS2 配额管理。

3.5.1 配置磁盘配额

请使用以下步骤实施磁盘配额：

- 1) 设置配额的强制或者计数模式。
- 2) 使用当前块使用信息初始化配额数据库文件。
- 3) 分配配额策略。（在计数模式中，不强制分配配额策略，此步骤可以略过。）

在以下小节中会详细讨论这些步骤的具体内容。

3.5.1.1 将配额设定为强制或者计数模式

在 GFS2 文件系统中，默认禁用配额。要为文件系统启用配额，请在挂载文件系统时指定 `quota=on` 选项。

有可能在没有强制限制和警告值的情况下为每个用户和组群跟踪磁盘用量并维护配额核算。要做到这一点，请使用 `quota=account` 选项挂载文件系统。

用法:

要挂载启用配额的文件系统, 请在挂载文件系统时使用 `quota=on` 选项。

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

要在挂载文件系统时即使没有强制配额限制也要使用配额计数维护, 则请在挂载文件系统时指定 `quota=account` 选项。

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

要在挂载文件系统时禁用配额, 请使用 `quota=off` 选项挂载文件系统。这是默认设置。

```
mount -o quota=off BlockDevice MountPoint
quota={on|off|account}
```

`on` ——指定挂载文件系统时启用配额。

`off` ——指定挂载文件系统时禁用配额。

`account` ——即使在没有强制配额限制的情况下, 也指定根据文件系统维护用户和组群的用量统计。

BlockDevice

指定 GFS2 文件系统所在的块设备。

MountPoint

指定要挂载 GFS2 文件系统的目录。

示例:

在这个示例中, `/dev/vg01/lvol0` 中的 GFS2 文件系统被挂载到 `/mygfs2` 目录中并启用了配额。

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

在这个示例中, `/dev/vg01/lvol0` 中的 GFS2 文件系统被挂载到 `/mygfs2` 目录中并启用了配额计数, 但没有强制。

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

3.5.1.2 创建配额数据库文件

挂载了每个启用了配额的文件系统后, 该系统就可以使用磁盘配额。但是该系统本身还不支持配额。下一步就是要运行 `quotacheck` 命令。

`quotacheck` 命令检查启用了配额的文件系统, 并在每个文件系统中构建当前磁盘用量表。该表格可用于更新磁盘用量的操作系统副本。另外还会更新该文件

系统的磁盘配额文件。

要在该文件系统中创建配额文件，请使用 `quotacheck` 命令的 `-u` 和 `-g` 选项。必须为用户和组群指定这两个选项方可进行初始化。如果为 `/home` 文件系统启用配额，则请在 `/home` 目录中生成该文件：

```
quotacheck -ug /home
```

3.5.1.3 为每个用户分配配额

最后一步是使用 `edquota` 命令分配磁盘配额。请注意：如果您使用计数的模式挂载文件系统（即指定 `quota=account` 选项），则不强制使用配额。

请在 `shell` 提示符后成为 `root` 用户，并执行以下命令为用户配置配额：

```
edquota username
```

为每个需要配额的用户执行这个步骤。例如：如果在 `/etc/fstab` 中为 `/home` 分区（在下面的示例中为 `/dev/VolGroup00/LogVol02`）启用配额，并执行 `edquota testuser` 命令，则在系统默认的编辑器中会显示以下内容：

```
Disk quotas for user testuser (uid 501):
Filesystem blocks soft hard inodes soft hard
/dev/VolGroup00/LogVol02 440436 0 0
```



注意：`edquota` 使用由 `EDITOR` 环境变量定义的文本编辑器。要更改编辑器，请在 `~/.bash_profile` 文件中将 `EDITOR` 环境变量设定为到您选择的编辑器的完整路径。

第一列是启用了配额的文件系统的名称。第二列显示目前该用户使用的块数。下两列是为该用户在该文件系统中设定的软限制和硬限制。

软限制定义可使用的最大磁盘空间量。

硬限制是用户或者组群可以使用的绝对最大磁盘空间量。达到这个上限后就再没有可以使用的磁盘空间了。

GFS2 文件系统不为内节点维护配额，因此这些列不适用于 GFS2 文件系统，为空白。

如果有任何值为 0，就是没有设定那个限制。您可以使用文本编辑器更改所需限制。例如：

```
Disk quotas for user testuser (uid 501):
Filesystem blocks soft hard inodes soft hard
/dev/VolGroup00/LogVol02 440436 500000 550000
```

请使用这个命令确认是否为该用户设定了配额：

```
quota testuser
```

3.5.1.4 为每个组群分配配额

还可以根据组群分配配额。请注意：如果您使用计数模式挂载您的文件系统（指定 `account=on` 选项），则不强制配额。

请使用以下命令为 `devel` 组群设定组群配额（设定组群配额前就存在该组群）：

```
edquota -g devel
```

这个命令在文本编辑器中显示该组群的现有配额：

```
Disk quotas for group devel (gid 505):
Filesystem blocks soft hard inodes soft hard
/dev/VolGroup00/LogVol02 440400 0 0
```

GFS2 文件系统不为内节点维护配额，因此这些列不适用于 GFS2 文件系统，为空白。您可以修改这些限制，然后保存文件。

请使用以下命令确认是否设定了组群配额：

```
quota -g devel
```

3.5.2 管理磁盘配额

如果使用配额，则需要对其进行维护 — 大多数是查看是否超过了配额，并确定配额是准确的。

当然，如果用户不断超过其配额，或者持续达到其软限制，系统管理员可根据用户类型以及影响其工作的磁盘空间选择不同的处理方法。管理员可帮助用户决定如何使用较少的磁盘空间，或者增加该用户的磁盘配额。

您可以运行 `repquota` 程序创建磁盘用量报告。例如：命令 `repquota /home` 可有这样的输出：

```
*** Report for user quotas on device /dev/mapper/VolGroup00-LogVol02
Block grace time: 7days; Inode grace time: 7days
Block limits File limits
User used soft hard grace used soft hard grace
-----
root -- 36 0 0 4 0 0
kristin -- 540 0 0 125 0 0
```

```
testuser -- 440400 500000 550000 37418 0 0
```

请使用以下命令查看所有启用了配额的文件系统的磁盘用量报告（选项-a）：

```
repquota -a
```

虽然这个报告很好理解，但有几点需要注意。每个用户后显示的——可让您迅速确定是否超过块限制。如果超过块软限制，在输出结果的第一个-位置会出现+。第二个-表示内节点限制，但 GFS2 文件系统不支持内节点限制，因此那个字符会保留为-。GFS2 文件系统不支持宽限期，因此 `grace` 一列将为空白。



注意：不考虑基础文件系统，NFS 都不支持 `repquota` 命令。

3.5.3 保持配额准确

如果您已经禁用配额运行文件系统一段时间并要再次启用配额，您应该运行 `quotacheck` 命令创建、检查并修复配额文件。另外，如果您认为配额文件不准确，您可以运行 `quotacheck` 命令，因为有可能在系统崩溃时没有完全卸载文件系统。

有关 `quotacheck` 命令的详情请查看 `quotacheck man page`。



注意：当文件系统在所有节点中相对停滞的时候运行 `quotacheck`，因为磁盘活性会消耗计算的配额值。

3.5.4 使用 `quotasync` 命令同步配额

GFS2 在其磁盘自身内部文件中保存所有配额信息。GFS2 节点不需要在每次文件系统写入时更新这个配额文件，它会每 60 秒更新一次这个配额文件。这在避免节点间在写入配额文件时发生冲突是很有必要的，这种冲突会降低节点性能。

如果某个用户或者组群的配额使用接近了配额限制，GFS2 会动态减小配额文件更新的时间周期以防止超过配额限制。通常配额同步的时间周期为一个可调整的参数 `quota_quantum`，您可以使用 `gfs2_tool` 命令修改它。默认情况下，时间周期为 60 秒。同时，必须在每个节点以及每次挂载文件系统时设定 `quota_quantum` 参数。（卸载后不会保留修改的 `quota_quantum` 参数。）

您可以使用 `quotasync` 命令在 GFS2 执行自动更新时将某个节点的配额信息与磁盘配额文件同步。

用法：

同步配额信息：

```
quotasync [-ug] -a | mntpoint...
```

-u

同步该用户的配额文件。

-g

同步该组群的配额文件

-a

同步所有目前启用配额并支持同步的文件系统。缺少-a时，应指定文件系统挂载点。

mntpoint

指定要执行动作的GFS2文件系统。

调整同步时间间隔

```
gfs2_tool settune MountPoint quota_quantum Seconds
```

MountPoint

指定要执行动作的GFS2文件系统。

Seconds

指定GFS2对常规配额文件进行同步化的新时间周期。数值越小，竞争越激烈，同时还会降低性能。

示例：

这个示例与为文件系统/mnt/mygfs2在磁盘配额文件中运行的节点同步所有缓冲的脏配额。

```
quotasync -ug /mnt/mygfs2
```

这个示例将单一节点中的文件系统/mnt/mygfs2的常规配额文件更新的默认时间周期改为一小时（3600秒）。

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

3.5.5 参考

有关磁盘配额的详情请参考以下命令的man page：

quotacheck

edquota

repquota

quota

3.6 增大的文件系统

`gfs2_grow`是在其所在文件系统设备被扩展后用来扩展GFS2文件系统的。在现有GFS2文件系统中运行`gfs2_grow`命令则会填满目前文件系统终点和新初始化的GFS2文件系统扩展设备终点之间的所有剩余空间。当完成填充工作后，会为文件系统更新源索引。集群中的所有节点则可以使用以添加的额外存储空间。

`gfs2_grow`必须在挂载的文件系统中运行，但只需要在集群的一个节点中运行。其它节点可感觉到扩展的发生，并可自动使用新的空间。

 注意：您使用 `mkfs.gfs2` 命令创建 GFS2 文件系统后，您就无法缩小该文件系统的大小。

用法：

```
gfs2_grow MountPoint
```

MountPoint

指定要执行动作的GFS2文件系统。

 注意：在运行 `gfs2_grow` 命令前请您：

- 1) 备份文件系统中的重要数据。
- 2) 运行 `df MountPoint` 命令确定要进行扩展的文件系统的容量。
- 3) 使用 LVM 扩展基础集群容量。有关管理 LVM 容量的信息请参考 LVM 管理员指南。

运行`gfs2_grow`命令后，请运行`df`命令查看文件系统中新的可用空间。

示例：

在这个示例中扩展了`/mygfs2fs`目录中的文件系统。

```
[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.
```

完整用法：

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

MountPoint

指定要挂载GFS2文件系统的目录。

Device

指定文件系统的设备节点。

表 3-3 “扩展文件系统是可用的GFS2具体选项” 描述在扩展GFS2文件系统时所要使用的GFS2具体选项。

表 3-3 扩展文件系统是可用的 GFS2 具体选项

选项	描述
-h	帮助，显示简短用法信息。
-q	静默，降低详细等级。
-r	MegaBytes 指定新资源组群大小，默认值为256MB。
-T	测试。完成所有计算，但不要向磁盘中写入数据，也不要扩展文件系统。
-V	显示命令版本信息。

3.7 在文件系统中添加日志

gfs2_jadd命令可用来在GFS2文件系统中添加日志。您可以在任意点动态在GFS2文件系统中添加日志，且不需要扩展基础逻辑卷。gfs2_jadd必须在挂载的文件系统中运行，但只需要在集群的一个节点中运行。其它节点可感觉到扩展的发生。



注意：如果某个 GFS2 文件系统已满，则 gfs2_jadd 将会失败，即使将逻辑卷包含的文件系统扩展到超过该文件系统的大小也是如此。这是因为在 GFS2 文件系统中，日志是纯文本文件，而不是嵌入的元数据，因此只是增大基础逻辑卷大小不会为日志提供空间。

在向GFS文件系统中添加日志前，您可以使用gfs2_tool命令的journals选项找出GFS2文件系统目前含有多少日志。以下示例显示挂载在/mnt/gfs2的文件系统中的日志数目和大小。

```
[root@roth-01 ../cluster/gfs2]# gfs2_tool journals /mnt/gfs2
journal2 - 128MB
journal1 - 128MB
```

```
journal0 - 128MB
```

```
3 journal(s) found.
```

用法:

```
gfs2_jadd -j Number MountPoint
```

Number

指定要添加的新日志数目。

MountPoint

指定要挂载GFS2文件系统的目录。

示例:

这个示例中，在/mygfs2目录的文件系统中添加了一个日志。

```
gfs2_jadd -j1 /mygfs2
```

这个示例中，在/mygfs2目录的文件系统中添加了两个日志。

```
gfs2_jadd -j2 /mygfs2
```

完整用法:

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

MountPoint

指定要挂载GFS2文件系统的目录。

Device

指定文件系统的设备节点。

表 3-4 “添加日志时可用的GFS2具体选项”描述用来在GFS2文件系统中添加日志的GFS2具体选项。

表 3-4 添加日志时可用的 GFS2 具体选项

标签	参数	描述
-h		帮助，显示简短用法信息。
-J	MegaBytes	以 MB 为单位指定新日志的大小。默认日志大小为 128MB，最小值为 32MB。要在文件系统添加不同大小的日志，必须为每个不同大小的日志运行 gfs2_jadd 命令。指

		定的大小会不断下降,因此在生成文件系统时会指定多个日志区段。
-j	Number	用 gfs2_jadd 命令指定要添加的新日志数目, 默认值为 1。
-q		静默, 降低详细等级。
-V		显示命令版本信息。

3.8 数据日志

通常GFS2只在其日志中写入元数据, 接下来会由内核的周期性sync将文件内容写入磁盘, 并被冲入系统缓冲。对某个文件的fsync()调用可导致将文件数据立刻写入磁盘, 但磁盘报告所有数据已经安全写入后会返回调用。

数据日志可缩短非常小的文件的fsync()时间, 因为文件数据在写入元数据外还要写入日志。随着文件的增大这个优势会明显降低。写入中等到大文件时打开数据日志会非常慢。

依赖fsync()同步文件数据的应用程序可能因使用数据日志而使性能有所提高。在被标记的目录及其所有子目录中生成的GFS2文件可自动启用数据日志。现有长度为0的文件也可以打开或者关闭其数据日志功能。

在将目录设定为“inherit jdata”的目录中启用数据日志, 这个设定表示以后所有在那个目录中生成的文件和目录都进行日志操作。您可以使用chattr命令在文件中启用和禁用数据日志。

下面的命令在/mnt/gfs2/gfs2_dir/newfile文件中启用了数据日志, 并查看是否正确设定了标签。

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

以下命令禁用了/mnt/gfs2/gfs2_dir/newfile文件中的数据日志, 并查看是否正确设定了标签。

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

您还可以使用chattr命令在目录中设定j标签。当您为某个目录设定此标签时,

以后在那个目录中创建的所有文件和目录也都会进行日志操作。下面的一组命令可在gfs2_dir目录中设定j标签，然后查看是否正确设定了该标签。此后，该命令会在/mnt/gfs2/gfs2_dir目录中生成一个名为newfile新文件，然后查看是否将该文件设定了j标签。因为为该目录设定了j标签，那么应该也为newfile启用了日志操作。

```
[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
```

3.9 配置 atime 更新

每个文件内节点和目录内节点都有三个与之相关的时间戳：

- 1) ctime — 最后一次修改内节点状态的时间
- 2) mtime — 最后一次修改文件（或者目录）数据的时间
- 3) atime — 最后一次访问文件（或者目录）数据的时间

如果启用了atime更新，因为在GFS2和其它Linux文件系统中这是默认设置，那么每次在读取文件时都需要更新其内节点。

因为有些应用程序使用atime提供的信息，那些更新可能会需要大量不必要的写入流量和文件锁定流量。那个流量可能会降低性能，因此请尽量关闭或降低atime更新频率。

有两个可用来降低atime更新效果的方法：

- 1) 使用 relatime (relative atime) 挂载，可在前一个 atime 更新比 mtime 或者 ctime 更新旧时，才更新 atime。
- 2) 使用 noatime 挂载，可在那个文件系统中禁用 atime 更新。

3.9.1 使用 relatime 挂载

当挂载文件系统时，可指定relatime (relative atime) Linux挂载选项。如果前一个atime更新比mtime或者ctime更新旧，这就可指定更新atime。

用法：

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

BlockDevice

指定GFS2文件系统所在的块设备。

MountPoint

指定要挂载GFS2文件系统的目录。

示例：

在这个示例中，GFS2文件系统位于/dev/vg01/lvol0，并挂载到目录/mygfs2。
只有在前一个atime更新比mtime或者ctime更新旧时才会进行atime更新。

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

3.9.2 使用 noatime 挂载

当挂载文件系统时，可指定Linux挂载选项noatime，它可在那个文件系统中禁用atime更新。

用法：

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

BlockDevice

指定GFS2文件系统所在的块设备。

MountPoint

指定要挂载GFS2文件系统的目录。

示例：

在这个示例中，GFS2文件系统位于/dev/vg01/lvol0，并挂载到关闭了atime更新的/mygfs2目录中。

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

3.10 在文件系统中挂起一个动作

您可以使用gfs2_tool freeze命令挂起对某个文件系统的写入动作。挂起写入动作允许使用基于硬件的设备快照捕获处于一致状态的文件系统。gfs2_tool unfreeze命令可终止挂起。

用法：

开始挂起

```
gfs2_tool freeze MountPoint
```

结束挂起

```
gfs2_tool unfreeze MountPoint
```

MountPoint

指定文件系统。

示例：

这个示例挂起了对文件系统/mygfs2的写入动作。

```
gfs2_tool freeze /mygfs2
```

这个示例终止了对文件系统/mygfs2写入动作的挂起。

```
gfs2_tool unfreeze /mygfs2
```

3.11 修复文件系统

当挂载文件系统节点失败时，文件系统日志允许快速恢复。但如果存储设备断电或者断开物理连接，则会发生文件系统崩溃。（无法使用日志进行存储子系统失败修复。）当这种崩溃发生时，您可以使用fsck.gfs2命令修复GFS2文件系统。



警告： fsck.gfs2 命令必须只能在从所有节点中卸载的文件系统中运行。



注意： 如果您以前有在 GFS 文件系统中使用 gfs_fsck 命令的经验，请注意 fsck.gfs2 命令在以下方面和之前发布的 gfs_fsck 有所不同：

在运行 fsck.gfs2 时按 Ctrl+C 会中断进程并显示提示信息，询问您是否要取消该命令，跳过剩余操作或者继续该进程。

- 1) 您可以使用-v 标签提高详细等级。添加第二个-v 标签会再次提高等级。
 - 2) 您可以使用-q 标签降低详细等级。添加第二个-q 标签会再次降低等级。
- n 会以只读方式打开某个文件系统并自动对所有查询回答 no。该选项提供了在不允许 fsck.gfs2 命令生效的前提下使用命令找出错误的方法。

有关其它命令选项详情请参考 fsck.gfs2 man page。

运行fsck.gfs2命令要求系统内存高于操作系统使用的内存而低于内核使用的内存。GFS2文件系统中的每块内存本身需要大约5比特额外内存，或者5/8字节。因此要估算在您的文件系统中运行fsck.gfs2命令所需内存字节数，您可以确定包含的文件系统块的数量，然后乘以5/8。

例如：要确定在大小为16TB，块大小为4K的GFS2文件系统中运行fsck.gfs2命令大约需要多少内存，首先要确定该文件系统中包含多少内存块，可使用16TB

除以4K:

```
17592186044416 / 4096 = 4294967296
```

这个文件系统中包含4294967296个块，再乘以5/8就可确定需要多少内存:

```
4294967296 * 5/8 = 2684354560
```

该文件系统大约需要2.6GB可用内存方可运行fsck.gfs2命令。请注意：如果块大小为1K，那么运行fsck.gfs2命令将需要四倍于以上值的内存，即大约11GB。

用法:

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

-y

-y标签可使对所有问题的回答都为yes。如果指定-y标签，fsck.gfs2命令则不会在进行修改前提示您回答问题。

BlockDevice

指定GFS2文件系统所在的块设备。

示例:

在这个示例中，修复了位于块设备/dev/testvol/testlv中的GFS2文件系统。所有关于修复查询的回答都自动为yes。

```
[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
    
```

3.12 绑定挂载以及上下文关联路径名

GFS2文件系统不提供对上下文关联路径名（CDPN）的支持，CDPN允许您生成指向不同目的地文件或者目录的符号链接。在GFS2中使用mount命令的bind选项实现此功能。

mount命令的bind选项允许您在不同位置重新挂载部分文件结构，且使其在初始位置仍可用。该命令的格式如下：

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

执行此命令后，olddir目录中的内容在两个位置可用：olddir和newdir。您还可以使用这个选项生成在两个位置可用的独立文件。

例如：在执行以下命令后，/root/tmp中的内容将和之前挂载的/var/log目录内容一致。

```

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

另外，您可以使用/etc/fstab文件中的条目在挂载时得到同样的结果。
/etc/fstab中的以下条目可使/root/tmp的内容和/var/log目录中的内容一致。

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

在您挂载文件系统后，您可以使用mount命令查看该文件系统是否被挂载了，如下示例所示：

```

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

对于支持上下文关联路径名的文件系统，您可以将/bin目录定义为上下文关联路径名，并根据系统构架将其解析为以下路径之一：

```
/usr/i386-bin
```

```
/usr/x86_64-bin
```

```
/usr/ppc64-bin
```

您可以通过生成空/bin目录得到同样的结果。然后使用脚本或者在/etc/fstab文件中的条目，将每个独立构架目录使用mount --bind命令挂载到/bin目录。例如：您可以使用以下命令作为脚本中的一行：

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

另外，您还可以使用以下行作为/etc/fstab文件的条目：

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

绑定挂载可为您提供比上下文关联路径名更大的灵活性，因为您可以使用此特性根据您定义的条件挂载不同的目录（比如文件系统的%fill值）。上下文关联路径名对其可处理的环境有更多的限制。请注意：您将需要根据条件（比如%fill）编写您自己的挂载脚本。



警告：当您使用 bind 选项挂载文件系统，且初始文件系统以 rw 挂载时，新的文件系统也会被以 rw 形式挂载，即使您使用的是 ro 标签，ro 则被静默忽略了。在这种情况下，可能会在/proc/mounts 目录中将新的文件系统标记为 ro 而引起误导。

3.13 绑定挂载和文件系统挂载顺序

当您使用mount命令的bind选项时，您必须确定使用正确的顺序挂载该文件系统。在以下示例中，必须在/tmp目录中执行绑定挂载前挂载/var/log目录：

```
mount --bind /var/log /tmp
```

按以下方法决定文件系统挂载顺序：

- 1) 通常文件系统挂载顺序由 fstab 文件中文件系统出现的顺序决定。例外情况包括使用_netdev 标签挂载的文件系统或者有自身初始化脚本的文件系统。
- 2) 有自身初始化脚本的文件系统在初始化进程后期挂载，即在挂载 fstab 文件中的文件系统之后挂载。
- 3) 使用_netdev 标签挂载的文件系统会在该系统中启用网络时挂载。

如果您的配置需要创建绑定挂载以便挂载GFS2文件系统，您可以命令fstab文件进行如下操作：

- 1) 挂载绑定挂载所需本地文件系统。

2) 绑定挂载要挂载 GFS2 文件系统的目录。

3) 挂载 GFS2 文件系统。

如果您的配置需要您在GFS2文件系统中绑定挂载本地目录或者文件系统，在fstab文件中列出正确的文件系统顺序也不会正确挂载文件系统，因为在GFS2初始化脚本运行前不会挂载该GFS2文件系统。在这种情况下，您应该在初始化脚本中写入命令执行绑定挂载，这样在挂载GFS2文件系统前就不会发生绑定挂载。

以下脚本是自定义初始化脚本示例。这个脚本在GFS2文件系统的两个目录中执行绑定挂载。在这个示例中，在/mnt/gfs2a有一个GFS2挂载点，可在集群启动并运行初始化脚本时执行挂载。在这个示例脚本中chkconfig状态值说明：

- 1) 345 说明启动该脚本的运行等级。
- 2) 29 是启动优先性，在这个示例中表示该脚本会在启动时在 GFS2 初始化脚本之后运行，后者的启动优先性为 26。
- 3) 73 为停止优先性，在这个示例中表示该脚本会在关闭时在 GFS2 脚本之后停止，后者的停止优先性为 74。

启动和停止值表示您可以通过执行service start和service stop命令手动执行所示动作。例如：如果该脚本名为fredwilma，则您可以执行service fredwilma start。

应将这个脚本放到/etc/init.d目录并拥有与该目录中其他脚本相同的权限。您可以执行chkconfig on命令将该脚本链接到所示的运行等级中。例如：如果该脚本名为fredwilma，那么您可以执行chkconfig fredwilma on。

```
#!/bin/bash
#chkconfig: 345 29 73
#description: mount/unmount my custom bind mounts onto a gfs2 subdirectory
#BEGIN INIT INFO
#Provides:
#END INIT INFO

. /etc/init.d/functions
case "$1" in
start)
#In this example, fred and wilma want their home directories
```

```
# bind-mounted over the gfs2 directory /mnt/gfs2a, which has
#been mounted as /mnt/gfs2a

    mkdir -p /mnt/gfs2a/home/fred &> /dev/null
    mkdir -p /mnt/gfs2a/home/wilma &> /dev/null
    /bin/mount --bind /mnt/gfs2a/home/fred /home/fred
    /bin/mount --bind /mnt/gfs2a/home/wilma /home/wilma

    ;;
stop)
    /bin/umount /mnt/gfs2a/home/fred
    /bin/umount /mnt/gfs2a/home/wilma

    ;;
status)
    ;;
restart)
    $0 stop
    $0 start
    ;;
reload)
    $0 start
    ;;
*)
    echo $"Usage: $0 {start|stop|restart|reload|status}"
    exit 1
esac
exit 0
```

3.14 GFS2 撤回功能

GFS2撤回功能是集群中GFS2文件系统的数据完整功能。如果GFS2内核模块探测到GFS2文件系统中存在不一致性，并伴随I/O操作，则该文件系统对该集群来说就不可用。该I/O操作会停止，同时该系统会等待下一个出错的I/O操作，防止进一步的破坏。当出现这种情况时，您可以停止手动停止任意服务或者程序，然后重启并重新挂载GFS2文件系统以便重新执行日志操作。如果问题仍存在，您可以在集群的所有节点中卸载该文件系统并使用fsck. gfs2命令执行文件系统恢复。GFS2撤回功能没有内核崩溃那么严重，但可造成使用另一个节点fence这

个节点。

如果您使用启用了启动脚本的gfs2配置您的系统，且在/etc/fstab文件中包含GFS2文件系统，则会在重启时重新挂载该GFS2文件系统。如果因为文件系统崩溃造成GFS2文件系统撤回，则建议您在重新挂载该文件系统前运行fsck.gfs2命令。在这种情况下，要防止您的文件系统在引导时重新挂载，您可以执行以下步骤：

- 1) 使用以下命令暂时在受影响的节点中禁用启动脚本：

```
chkconfig gfs2 off
```

- 2) 重启受影响的节点，启动集群软件。此时将不会挂载该 GFS2 文件系统。
- 3) 在集群的所有节点中卸载该文件系统。
- 4) 在文件系统中运行 fsck.gfs2，如果该节点确认没有文件系统崩溃。
- 5) 运行以下命令在受影响的节点中重新启用该启动脚本：

```
chkconfig gfs2 on
```

- 6) 在集群的所有节点中重新挂载 GFS2 文件系统。

可能造成GFS2收回的不一致性示例为错误的块计数。当GFS内核删除文件系统中的某个文件上，它会系统地删除与那个文件关联的所有数据和元数据。完成此操作后，它会检查块计数。如果该块计数不是1（意思是只剩下磁盘内节点自己），则表明文件系统不一致，因为块计数与找到的块列表不匹配。

您可以使用-o errors=panic选项挂载文件系统覆盖GFS2收回功能。当指定这个选项时，所有导致系统收回的错误都将导致系统panic。这样可停止节点的集群沟通，并fence该节点。

在内部，GFS2通过让内核向gfs_controld守护进程发送请求撤回的信息启动撤回功能。

gfs_controld会运行dmsetup程序替换该文件系统中的设备映射器错误目标以防止进一步访问块设备。然后它会告诉内核操作完成。这是GFS2支持要求总是在GFS2使用CLVM设备的理由，否则就不可能插入设备映射器目标。

设备映射器错误目标的目的，是确保将来所有的I/O操作都将有一个I/O出错信息，会让该文件系统以较旧的方式卸载。结果是当出现撤回时，通常会在系统日志中看到来自设备映射器的I/O出错信息。

偶尔在dmsetup程序无法根据要求插入错误目标时，撤回会失败。如果撤回

时内存不足，并且因造成撤回的问题而无法重新使用内存时，会发生这种情况。

撤回并不总是意味着GFS2中有错误。有时撤回功能可由与基础块设备有关的设备I/O错误引发。强烈建议您检查日志来查看发生撤回是否是这样的原因。

4 诊断并修正 GFS2 文件系统的问题

4.1 GFS2 文件系统出现性能缓慢

您会发现 GFS2 文件系统显示出比 EXT3 文件系统缓慢的性能。在某些情况下 GFS2 的性能可能受到一些因素的影响。本文档中有关于处理 GFS2 性能问题的信息。

4.2 通过 GFS2 设定 NFS

由于GFS2锁定子系统增加的复杂性及其集群的本质，通过GFS2设定NFS要求您非常仔细、小心进行配置。本小节论述了您在通过GFS2文件系统配置NFS服务时应该注意的地方。

 **警告：**如果是使用 NFS 导出 GFS2 文件系统，且 NFS 客户端应用程序使用 POSIX 锁定，那么您必须使用 `locallocks` 选项挂载该文件系统。这样做预期的效果是将每个服务器强制 POSIX 锁定为本地：例如，非集群，独立的。（如果 GFS2 尝试使用 NFS 在集群的多个节点中部署 POSIX 锁定会有很多问题。）对于在 NFS 客户端中运行的应用程序，如果在不同的服务器中挂载两个客户端，本地的 POSIX 锁定意味着两个客户端可同时拥有同一个锁定。如果所有客户端都使用一个服务器挂载 NFS，那么不同服务器分别赋予同一锁定的问题就不存在了。

除锁定注意事项外，您应该在通过GFS2文件系统配置NFS时考虑以下问题。

- 1) 中标麒麟高级服务器操作系统只支持使用主动/被动锁定的 NFSv3 进行
 中标麒麟高级服务器操作系统高可用性附加组件配置，并具有以下特征：
 - a) 后端文件系统是在 2-16 个节点集群中运行的 GFS2 文件系统。
 - b) NFSv3 服务器的定义是一次从单一集群节点中导出整个 GFS2 文件系统的服务。
 - c) 该 NFS 服务器可能在从一个集群节点到另一个节点时失败（主动/被动配置）。
 - d) 不允许到 GFS2 文件系统的访问，除非是通过 NFS 服务器。这包括本地 GFS2 文件系统访问以及通过 Samba 或者集群的 Samba 的访问。
 - e) 在该系统中不支持 NFS 配额。

这个配置为该文件系统提供HA，并降低系统当机时间，因为失败的节点不会造成在从一个节点到另一个节点有NFS服务器失败时要执行fsck命令。

- 2) GFS2 的 NFS 导出中 fsid=NFS 选项是必须的。
- 3) 如果您的集群出现问题（例如：集群成为 inquorate，且 fencing 不成功），集群的逻辑卷和 GFS2 文件系统将会停滞，且不可能有访问直到集群成为 quorate。您在决定在这个过程中定义的简单故障切换解决方案是否为您系统的最佳解决方案时，应该考虑这种可能性。

4.3 GFS2 文件系统挂起并需要在一个节点中重启

如果您的GFS2文件系统挂起，且不返回由此运行的命令，但重启后，会返回常态的具体节点，这表示可能有锁定问题。出现这种情况时您应该收集以下数据：

- 1) 在每个节点中为该文件系统执行的 gfs2 锁定转储：

```
cat /sys/kernel/debug/gfs2/fsname/glocks >glocks.fsname.nodename
```

- 2) 每个节点中为该文件系统执行的 DLM 锁定转储：您可以使用以下 dlm_tool 命令收集这个信息：

```
dlm_tool lockdebug -sv lsname
```

在这个命令中，lsname是有问题的附加系统DLM使用的锁定空间名称。您可以在group_tool命令的输出结果中找到这个值。

- 3) sysrq -t 命令的输出结果。
- 4) /var/log/messages 文件的内容。

您收集到数据后，可以在中标麒麟高级服务器操作系统支持团队生成一个ticket，并提供您收集的数据。

4.4 GFS2 文件系统挂起并需要重启所有节点

如果您的GFS2文件系统挂起且不会返回由此运行的命令，则需要您在使用它之前重启该集群中的所有节点，并检查以下问题。

- 1) 您可能会有一个失败的 fence。

GFS2文件系统将停滞以保证在失败的fence事件中数据的完整性。检查信息日志查看挂起时是否有失败的fence。请确定您正确了配置的fencing。

2) GFS2 文件系统可能已经撤回。

检查信息日志查看关键字`withdraw`，看看是否有来自GFS2表示已经撤回的文件系统的信息和`calltrace`。撤回表示可能为系统崩溃、存储失败或者是一个bug。卸载该文件系统，更新`gfs2-utils`软件包并在该文件系统中执行`fsck`命令以便返回该服务。在中标麒麟高级服务器操作系统支持团队生成一个支持ticket。通知它们您有GFS2撤回问题，并提供有日志的`sosreport`。

有关GFS2撤回功能的详情请参考“3.14 GFS2撤回功能”。

3) 这个出错信息表示有锁定问题或者 bug。

在这种情况下之一收集数据并在中标麒麟高级服务器操作系统支持团队生成一个支持ticket，如“4.3 GFS2文件系统挂起并需要在一个节点中重启”所述。

4.5 GFS2 文件系统不挂载新添加的集群节点

如果您在集群中添加一个新节点，且您发现您无法在那个节点中挂载GFS2文件系统，那么相较尝试访问GFS2文件系统的节点，您的GFS2文件系统日志可能较少。您必须在每个要挂载文件系统的GFS2主机中都有一个日志（使用`spectator`挂载选项组挂载的GFS2文件系统除外）。您可以使用`gfs2_jadd`命令在GFS2文件系统中添加日志。

4.6 GFS2 空文件系统

如果您有一个空GFS2文件系统，`df`命令显示使用了空间。这是因为GFS2文件系统日志消耗磁盘空间（日志数乘以日志大小）。如果您创建有大量日志的GFS2文件系统或者指定大日志，那么您将在执行`df`时看到已有日志数乘以日志大小的空间被使用。即使您没有指定较大的日志数或者大日志，小GFS2文件系统（1GB或者更小）将使用默认GFS2日志大小的空间。