第 32 卷　第 7 期　　计　算　机　工　程　　2006 年 4 月
Vol.32　№ 7　　Computer Engineering　　April 2006
·软件技术与数据库·　文章编号：1000—3428(2006)07—0069—03　文献标识码：A　中图分类号：TP316

机群文件系统 DCFS2 的高可用性研究与实现

李　晖，范志华，熊　劲，马　捷
（中国科学院计算技术研究所国家智能中心，中国科学院研究生院，北京 100080）

摘　要：对 DCFS2 采用的高可用技术进行了介绍，主要对 DCFS2 的客户端高可用的问题进行了分析，解决了在 DCFS2 的结构下的客户端失效的恢复问题。对 DCFS2 的性能测试表明，DCFS2 可提供具有竞争性的带宽和元数据性能和高可用性。
关键词：机群文件系统；DCFS2；高可用技术；日志技术；IP-SAN

Realization of a High-available Cluster File System

LI Hui, FAN Zhihua, XIONG Jin, MA Jie
(National Intellectual Center of Computing Technology Institute, CAS, Graduate School of CAS, Beijing 100080)

【Abstract】 This paper introduces the design and implementation of the high-availability of DCFS2. It emphasizes on the design and implementation of high-avaliability of the DCFS2 client-side file system. Performance tests on DCFS2 show that DCFS2 can provide competitive bandwidth and metadata performance and high-avaliability.
【Key words】 Cluster file system; DCFS2; High-availability technology; Journaling technology; IP-SAN

目前，机群系统已经成为构建高性能计算的主要平台，大量的机群文件系统不断涌现，典型的有美国 Redhat 公司的 GFS，美国 Cluster Filesystem 公司的 Lustre 等。由于机群文件系统本身结构较为复杂，实现复杂而且整个系统规模较大。这些因素决定了机群文件系统对高可用技术的依赖。现有的机群文件系统几乎都在协议或是实现上对高可用进行支持。针对曙光 4000A 超级服务器设计的机群文件系统 DCFS2 不仅具有高性能、支持超大规模文件系统、易管理等特点，而且它的一个重要的特点就是其采用的高可用技术。该技术能够保证整个系统在服务器端故障或是客户端故障的情况下均能快速恢复至一个一致的状态。本文将对 DCFS2 机群文件系统采用的高可用技术进行介绍，主要对 DCFS2 机群文件系统的客户端故障进行了分析，提出了在 DCFS2 结构下的客户端高可用的关键问题以及对应的解决方案，解决了在 DCFS2 的结构下的客户端失效时的恢复问题。

1 背景

1.1 相关研究

目前比较成功的分布式/机群文件系统均无一例外地在高可用方面做了大量的工作，大致可分类如下：

（1）采用设计良好的协议来保证高可用。典型系统是美国 SUN 公司的研制并于 1985 年推出的 NFS 文件系统。NFS 由于采用了无状态协议，因此整个系统的容错能力较强，服务端或是客户端失效只需要简单的重启即可。

（2）采用冗余存储的方法。典型系统是 xFS 文件系统。xFS 文件系统是 Berkley 大学提出的无服务器网络文件系统的原型系统。xFS 采用了无服务器结构，控制和数据可以分布在任何一个用户服务器，每个用户在从其他用户服务器取得服务的同时也可以为别的用户服务器提供服务。由于采用了冗余存储，因此系统不存在单一的故障点，具有较强的可用性。

（3）采用服务器端日志或是在客户端端记录日志的方式来保证系统的一致性的方式。典型系统有 GFS，Lustre 等。

1）GFS 是由美国明尼苏达大学研制的，是基于 SAN 的共享存储文件系统。GFS 采用无集中服务器结构来避免系统的单一失效点。GFS 采用客户端日志的方法，日志存储在共享存储，每个客户端有自己的日志空间。在出现失效情况时，恢复线程根据记录在共享存储上的日志进行恢复，使得文件系统恢复至一个一致的状态。

2）Lustre 是由美国 Cluster Filesystem 公司研制的，是一个基于 OBD 设备的，使用面向对象访问技术的机群文件系统。Lustre 所使用的 OBD 设备本身建立在 ext3 等日志文件系统之上，保证了 OBD 设备本身的一致性。在元数据服务器与 OBD 设备之间的不一致问题则采用日志技术来解决，每个存储设备都有不同的日志空间。

1.2 DCFS2 的结构

DCFS2 在充分考虑存储技术发展趋势的基础上，采用直接基于 IP 网络存储设备的共享存储体系结构，即 DCFS2 是建筑在多个网络存储设备上的全局共享文件系统。

DCFS2 的总体结构如图 1 所示。

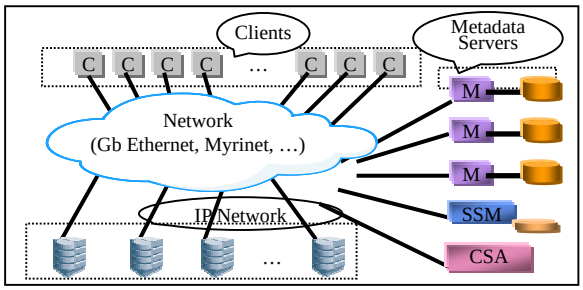

图 1　DCFS2 的结构

基金项目：国家“863”计划基金资助项目(2002AA1Z2102)
作者简介：李　晖（1980－），男，硕士生，主研方向：机群文件系统；范志华，博士生；熊　劲、马　捷，博士、副研究员
收稿日期：2005-04-05　**E-mail：**li_hui@ncic.ac.cn

DCFS2 采用将文件数据与文件系统元数据分开进行处理的策略，采用元数据服务器来存储和处理文件系统的元数据：文件的属性、目录文件和文件系统的属性；文件数据则存储在 IP 网络存储设备中，由客户端直接访问 IP 网络存储设备。DCFS2 采用多元数据服务器结构，这样可以满足系统规模较大时元数据处理性能要求很高的需求。

DCFS2 由 4 部分组成：一个空间管理服务器（SSM），用来负责 IP 存储设备的空间的管理和空间的分配与回收；多个元数据服务器（MGR），负责整个文件系统的元数据的处理；CSA 是 DCFS2 的配置和管理工具；各客户节点上的 DCFS2 是同一文件系统（统一的名字空间），用户和应用直接用 OS 提供的系统调用和命令即可访问 DCFS2 中的文件。

1.3 DCFS2 元数据服务器故障恢复策略

为了取得较好的可扩展性和较高的性能，DCFS2 采用了多元数据服务器结构，文件系统按子树在各个服务器上分布。整个文件系统的元数据分别存储在对应的元数据服务器所在的本地文件系统上。对于元数据服务器故障，DCFS2 主要考虑解决以下两个问题：

（1）由于元数据缓存的引入，当某个元数据服务器失效时，其缓存内的元数据将丢失，从而造成该元数据服务器磁盘上的元数据内部具有不一致性，造成文件系统的错误。

（2）由于 DCFS2 是个多元数据服务器的结构，一个文件系统的操作可能会在两个元数据服务器上进行元数据的修改，这时，如果一个元数据服务器故障造成缓存的被修改，元数据丢失，而另一个元数据服务器上的修改则记录到了物理磁盘上，那么也会出现整个文件系统元数据的不一致性。

DCFS2 采用分布式日志技术来解决以上两个问题。在正常处理时，DCFS2 的各个元数据服务器记录元数据修改日志。在故障恢复时采取失效的元数据服务器在重启后根据记录的日志以及与其他元数据服务器进行协商的结果来决定恢复的策略。

2 DCFS2 客户端高可用关键问题

DCFS2 机群文件系统为了提供高性能的元数据和数据服务采用了许多性能优化技术，比如缓存、服务器端维护状态、客户端申请空间的预取等。但是这些技术的引入对文件系统的高可用带来了新的问题，如果仅仅在元数据服务器端采用日志的方法在客户端故障的情况下，无法保证整个文件系统的一致性。为了保证整个文件系统的高可用就必须解决客户端故障带来的文件系统不一致问题。本节主要分析在 DCFS2 的结构下客户端高可用需要解决哪些关键问题。

2.1 服务器端状态问题

DCFS2 文件系统在客户端失效的情况下，由于 DCFS2 采用有状态的服务器模式，在客户端重启之后，会出现服务器状态与客户端状态不一致的情况，因此一旦出现这种情况会影响到用户对文件系统的使用。此外由于服务器端维护了客户端打开文件的计数，在某客户端出现故障之后，服务器端维护的这个计数的值必然是错误的，将会影响到整个文件系统之后的运行。

2.2 数据一致性问题

DCFS2 文件系统在客户端故障的情况下会存在文件长度、文件的块映射信息以及数据的一致性问题，具体情况为：

（1）如果由于客户端故障导致文件的数据块没有写回，块映射信息没有写回，但是文件长度已经写回，在这种情况下文件的数据内容丢失，并且如果写回的文件的长度比原长度大的话会导致用户会在文件的尾部读到一个空洞。

（2）如果客户端故障导致文件的数据块没有写回，块映射信息已经写回，文件长度已经写回，在这种情况下数据内容变成存储设备上的当前内容，而不是新写入的内容。

2.3 空间一致性问题

DCFS2 文件系统在客户端故障的情况下可能出现空间丢失的情况。具体的问题如下：

（1）客户端在出现故障时如果预申请的空间还没有全部使用完，那么在客户端重启后这些信息将丢失，即该客户端不再占有这些空间，而空间管理服务器则认为这些文件块已经被使用，造成了空间的丢失。

（2）如果客户端出现故障时，预申请的空间在文件的写过程中已经使用，但是文件的块映射信息没能写回到 mgr，在客户端重启后之前的这些块映射信息都已丢失，而空间管理服务器认为这些空间都已经被使用，导致了这部分空间的丢失。

3 DCFS2 客户端高可用的实现

3.1 服务器端状态问题的实现

为保证各个客户节点与服务器端的状态的一致性，DCFS2 文件系统在客户端失效后，必须在元数据服务器端进行与该客户端相关的文件锁的释放以及打开文件的引用计数的恢复。

当某客户端故障时，配置管理工具(CSA)监测到故障或是管理员发现故障后，都可以通过配置管理工具通知整个文件系统该客户端发生故障。当服务器端收到客户端故障通知后就执行相应的清理工作，收回该客户端所申请到的文件锁，释放对应文件的打开计数。

3.2 数据一致性问题的实现

为了解决 DCFS2 客户端故障造成的数据一致性问题，DCFS2 采取了将设置文件长度的消息和写回文件的块映射信息的消息合并在一起，并且确保数据在该消息发送之前已经写回存储服务器的方法。

在写回时机上采用统计用户已使用的文件块映射信息的个数，当达到一定的限度时就启动一次同步操作，将数据写回，然后将设置文件长度和映射信息的消息写回到元数据服务器，保证文件系统达到一个一致的状态。在文件关闭时同样必须进行一次同步操作来保证一致性。这样就决定了对于文件映射信息所使用的核心的 page cache 在内存的使用上不能完全依赖于内核的内存管理机制，而应该由核心模块自己来负责这些页面的创建、删除等问题。

3.3 空间一致性问题的实现

针对空间可能丢失的问题，DCFS2 采取客户端记录日志的方法：每次从空间管理服务器那里获取到的空间，首先在日志上记录下来；每次写回块映射信息之前将这个文件的文件系统编号和 inode 号，以及新分配给这个文件的所有块记录下来；收到写回块映射消息的应答之后，给这些块标记，表示这些块已真正被使用（也就是在客户端失效后不需要确认元数据服务器是否已经收到这些块）。一旦元数据服务器确认已收到这些块映射信息，那么就保证了这些块映射信息在日志中记录的数据项就可以删除了。

日志文件采用固定大小的文件方式，当写到文件末尾时重新回到文件起始位置寻找可用项来记录日志信息。

如果一旦客户端出现失效，客户端在重新启动后 DCFS2 将进入恢复流程。具体恢复的流程如下：

（1）故障通知阶段。使用配置管理工具通知文件系统各个部分该客户端发生失效。元数据服务器在收到故障通知后，需要进行与该客户端相关的文件锁的释放以及打开文件的引用计数的恢复，以保证其他客户端对该文件能继续正常访问。

（2）协商阶段。失效客户端重启，启动后扫描日志，根据日志记录的情况与元数据服务器或是空间管理服务器进行协商，判断对应的空间是否丢失。

（3）恢复阶段。根据协商的结果，如果发生空间丢失的情况，则将该空间信息归还给空间管理服务器。

（4）通知恢复结束阶段。客户端恢复结束，通知文件系统各个部分，各客户端已恢复正常。

4 性能评测

测试平台为 3 节点的 Redhat Linux 服务器，Linux 核心版本为 2.4.18-3，每个节点都是 PIII 1GHz 的双 CPU 的 SMP 节点，1GB 内存，一个 18.4GB 的 SCSI 硬盘。节点间通过百兆网互联。CSA 和 SSM 运行在同一节点，一个节点运行元数据服务器，一个节点作为客户端节点。

为测试客户端日志对文件系统的带宽造成的影响，我们使用 iozone 工具进行带宽测试，所得高可用级别即记录客户端日志与普通级别即不记录客户端日志情况下的性能比较如图 1～图 5。

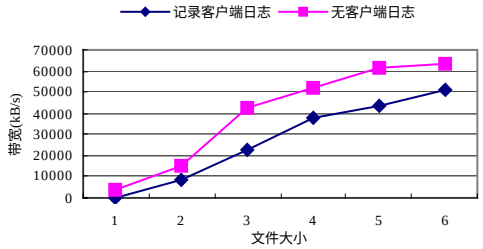

图 2 小文件写性能比较

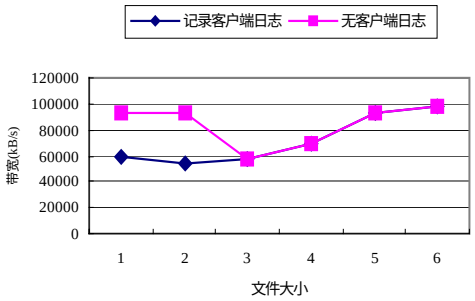

图 3 小文件读性能比较

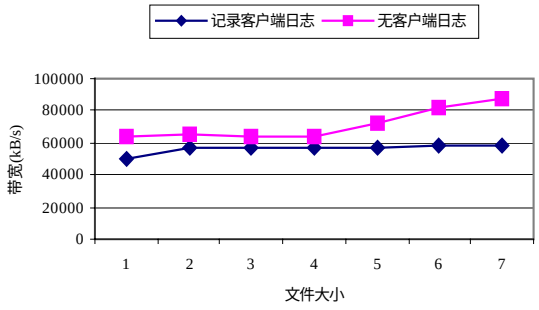

图 4 大文件写性能比较

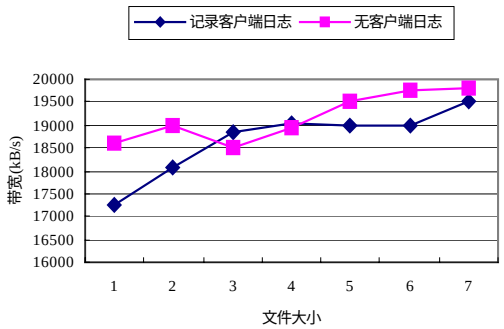

图 5 大文件读性能比较

从以上测试结果可以看出，由于我们对日志的记录进行了优化，客户端日志对性能的影响很小，能够在较高可用性的情况下提供较好的性能。同时从测试结果中可以看出，在文件读写量较少的情况下，DCFS2 采用的缓存对性能的优化作用比较明显，而在读写量较大的情况，如 512MB 的情况，由于内存容量的限制导致缓存的优化效果不再明显。

使用 postmark 工具，测试所得的结果如表 1 所示，其中 2 000/5 000 为测试规模，即指定测试的文件数为 2 000，transactions 的个数为 5 000。

表 1 使用 postmark 工具测试所得结果

2000/5000	无日志	有日志
Transactions per second	58	61
Data read (kB/s)	158.55	116.43
Data written (kB/s)	294.19	216.04
Creation alone	285 per second	44 per second
Deletion alone	940 per second	940 per second

从 postmark 的测试结果中可以看出，记录客户端日志对文件系统的带宽以及吞吐率均有一定的影响，但是对用户的事务处理影响不大。

同时我们还进行了多次高可用恢复测试，测试均能够正确地将文件系统恢复至一个最近的一致状态。

5 结论

本文对 DCFS2 采用的高可用技术进行了介绍，其中详细地分析了解决客户端的高可用性会面临的问题以及 DCFS2 采取的对应策略。在确定使用客户端日志来解决客户端故障之后，我们在曙光 4000A 超级服务器上实现了 DCFS2 机群文件系统。通过性能测试可以看出，优化后的 DCFS2 客户端日志对性能的影响较小，能够在保证较高可用性的情况下向用户提供较高的带宽和吞吐率。

参考文献

1. 熊　劲. DCFS2 文件系统总统设计报告（内部）[R]. 中科院计算所国家智能中心, 2003.
2 Presla N K W. Implementing Journaling in a Linux Shared Disk File System [C]. The Seventeenth IEEE Symposium on Mass Storage Systems, 2000.
3 Fan Zhihua, Xiong Jin, Ma Jie. A Failure Recovery Mechanism for Distributed Metadata Servers in DCFS2 [C]. High Performance Computing Asia2004, Japan, 2004.