一种新颖的 RAID系统在线扩容方案*

冯 丹, 彭 丽
(华中科技大学 计算机科学与技术学院 信息存储系统教育部重点实验室, 湖北 武汉 430074)

摘 要: 在考虑到算法通用性, 实现简易性, 以及在扩容过程中能够改变阵列分块大小等特性, 以带区为基本导向给出了一种 RAID系统在线扩容方案。它的新颖之处在于: 在整个在线扩容过程中只用维护新旧两份不同的磁盘阵列配置信息来实现不同 RAID级别系统中的在线数据迁移, 以及负载平衡等功能。

关键词: 廉价冗余磁盘阵列; 在线扩容; 逻辑卷管理器; 负载均衡
中图法分类号: TP311 文献标识码: A 文章编号: 1001-3695(2006)12-0244-03

Novel Scheme for On-line Capacity Expansion in RAID System

FENG Dan, PENG Li
(National Lab. of Storage System, College of Computer Sci. & Tech. Huazhong University of Sci. & Tech., Wuhan Hubei 430074, China)

Abstract To be a perfect RAID system, the on-line capacity expansion mechanism is necessary. Capacity expansion algorithms is not only disk-oriented, but also stripe-oriented. In view of generality, easiness and dynamic striping size, present a novel stripe-oriented on-line capacity expansion algorithm, which the key point lies in using two deferent configuration tables of RAID to implement the on-line data migration and load balancing in deferent RAID system.

Key words RAID; On-line Capacity Expansion; LVM; Load Balancing

1 引言

随着计算机和网络在社会中的普及, 人们对不同应用的需求越来越多, 原有的老应用设备已经不能满足需求, 必须提升设备的性能。作为系统中重要的组成部分, 存储系统也需要提升其容量和性能。而当前主要联机大容量存储采用磁盘阵列, 所以随着容量增长的要求, 就必须增加磁盘阵列系统的容量。传统的增加廉价冗余磁盘阵列系统的容量的方式是将 RAID[4]系统中所有数据备份出来, 重新配置, 建立 RAID存储空间, 然后将数据恢复到 RAID系统中。这种方式显然影响了用户的正常使用, 所以考虑到在线扩容, 如 1.1节所介绍。传统在线扩容方法大多是从磁盘的数据布局以及磁盘带宽的使用角度来考虑。从磁盘的角度作考虑虽然对用户响应影响会相对小一些, 但是由于涉及到磁盘数据布局, 所以实现上相当复杂, 适用范围上有磁盘阵列级别和分块大小上的局限性, 且不能在扩容过程中动态改变阵列分块大小, 故不能在扩容的同时进行负载的均衡[2,3]。为了解决这些问题, 并且提高方案的通用性, 本文提出了一种新颖的、通用性很强的磁盘阵列系统在线扩容方案。它无须局限于 RAID级别, 也无须局限于 RAID分块大小, 并且在扩容过程中可以实现同构系统[1]中的负载均衡。

1.1 RAID系统在线扩容的基本概念

不使用在线扩容技术时, 要想增加原有 RAID系统的容量就意味着将 RAID系统中的所有数据备份出来, 重新配置, 建立 RAID存储空间, 然后将数据恢复到 RAID系统中。显然在数据转移过程中读写不能进行。而 RAID扩展技术让用户可以通过在线添加新磁盘、或用大容量磁盘拷贝并替换原磁盘的方法来扩展 RAID空间, 无须关闭计算机、重新启动系统, 也无须备份数据、暂停应用。本文只讨论了用户在线添加新盘的情况, 如图 1所示。在线扩容保证了在将数据重新条带化分配于成员盘中时, 原来的数据完好, 容量保持不变, 在重新条带化及重新分配过程中, 数据仍然可以读写, 不影响应用。

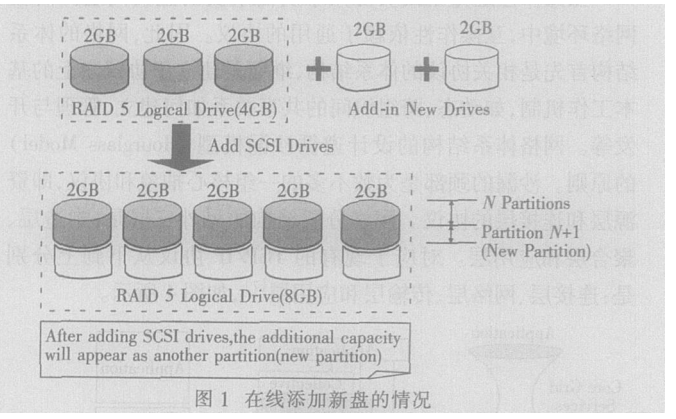

图 1 在线添加新盘的情况

1.2 逻辑卷管理器

逻辑卷管理器[5]用于维护一个逻辑卷管理配置表, 表中包括所有的磁盘配置表、逻辑卷配置表、逻辑卷组配置表以及逻辑单元号映射表。物理盘与逻辑卷之间的归属关系、逻辑卷和逻辑卷组之间的归属关系以及逻辑卷组和逻辑单元号的映射关系都通过查询各表中相应字段即可确定。

阵列程序与主机端的存储管理控制台之间均需维护逻辑卷管理配置表, 用户配置阵列或者修改阵列参数, 必须将修改

收稿日期: 2005-09-28; 修返日期: 2005-11-01
基金项目: 国家“973”计划资助项目(2004CB318201)

过的配置表传送给阵列的程序; 当阵列程序查询到发现新加入的物理磁盘时, 必须将其报告给主机端的存储管理控制台, 而传递双方信息的桥梁就是逻辑卷管理配置表。本系统中采用了 In-Band的管理方式, 即直接通过 SCSI总线传递配置表信息。其中自定义了两个 SCSI命令, 即 READ CONFIG (F0h)和 WRITE CONFIG (F1h)。主机端发出 READ CONFIG 命令后, 阵列的逻辑卷管理器识别出该命令, 即将配置表从 DOM (Disk On Module)盘上读出并通过光纤通道传递给主机; 主机端发出 WRITE CONFIG命令后, 阵列的逻辑卷管理器识别出该命令, 即通过光纤通道接收到配置表, 并将其写入到 DOM盘上。

2　RAID系统在线扩容算法

通常, 在阵列内存中只用维护一份配置表[5]信息。为了扩容的需要, 作如下的代码实现: ①需要两套配置表数据结构来保持新旧两份配置信息; ②通过未使用的扩展 SCSI命令作为扩容命令来触发扩容任务, 也就是通过主机端的存储管理控制台发送扩容命令给阵列。情况 ②为当阵列监测模块检测到用户在线加盘操作, 填充一份新的阵列配置表信息, 从而触发扩容任务。

扩容过程: 当主机端发送扩容命令给阵列端或阵列端检测到有加盘操作后产生了新的配置表信息时, 便触发了扩容任务。阵列将新的配置信息放在先前分配好的配置表结构中, 此时内存中维护两份配置表信息。此时扩容任务将两份配置表信息进行比较, 当检测到阵列中某个卷的盘数增加时, 进入真正的扩容流程, 然后开始数据的动态迁移操作。在线扩容算法如下: 当前使用的 RAID系统是由多卷按 JBOD方式组成, 每个卷又为任意的组成方式。算法中选取在其中的某个卷中加盘, M 代表该卷中原先的磁盘个数, $N-M$ 代表在线添加的盘数。

Do

(1)确定这次要转移的起始地址和长度;

(2)设置锁, 禁止用户对该区域数据的访问;

(3)使得内存中的配置表信息为旧的 M 盘阵列配置信息;

(4)产生一个低优先级的读请求, 按照 M 个盘阵列布局读取扩容前阵列中确定数据区域的数据到我们先前申请的扩容缓冲区中;

(5)等待所有读操作完成;

(6)设置一个全局变量的值, 其中保存当前扩容读操作的起始地址;

(7)释放对这个区域的锁;

(8)设置锁, 禁止用户对要写的数据区域的访问;

(9)使得内存中的配置信息为新的 N 盘阵列配置信息;

(10)产生一个低优先级的写请求, 将读出的数据按照 N 个盘阵列布局方式重新分布到现存的盘上, 其中写操作的起始地址、长度和扩容读操作一样;

(11)等待所有写操作完成;

(12)恢复内存中的配置表信息为旧的 M 盘配置信息;

(13)判断若是最后一次转移, 更新内存中的配置信息为新的 N 盘配置信息, 且将新的配置信息写到 DOM盘上去;

(14)释放对该区域的锁。

Until(所有需要转移的数据都得到了转移)。

在该算法中, 低优先级的读请求一直要到磁盘访问队列中没有其他正常优先级的读请求时, 才开始发出。一旦发出, 低优先级的读请求就不会被高优先级的访问请求中断。用低优先级的读请求的目的是为了减少对用户响应时间的影响, 因为普通商用磁盘并不支持抢先式访问。同样, 用于扩容的写请求也是低优先级的。算法中对正处于扩容过程中的读写数据区域设置保护锁是因为: ①用户写操作会导致数据不一致, 必须保证此时扩容进程不读取数据。②在扩容过程中, 新用户数据的写操作与扩容的写操作存在冲突的可能性。

基于在线的原因, 在阵列主控程序[4]中应对各种读写命令先加一些判断, 然后再对其进行恰当的处理。读写请求可能有下列几种情况: ①扩容过程中构造的读命令; ②扩容过程中构造的写命令; ③没有触发扩容任务之前或是扩容操作全部结束之后传来的正常用户读写命令; ④扩容读操作完成, 但是扩容写操作还没有进行期间传来的正常用户读写命令; ⑤在一次完整的扩容操作完成之后, 但全部扩容操作没有完成之前来的正常用户读写命令。其中前三种情况均应按照正常的读写命令处理流程来进行处理; 后两种应进行特殊处理。在情况 ④下读写命令应该与扩容的缓冲区进行交互, 如图 2所示。图中的命中与否表示当前读写操作的数据区域是否与扩容读写操作的数据区域有重叠。在情况 ⑤下读写命令无须与扩容缓冲区进行交互, 但是仍然应考虑按照新或旧配置来进行读写的问题, 类似于情况 ④的处理。

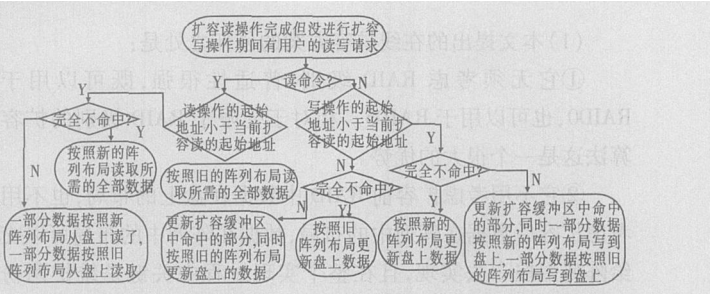

图 2　扩容过程中对正常用户读写请求的处理流程

3　性能测试

阵列配置为 CPU: Intel Pentium 4 Xeon 2.4GHz, 内存: 512MB DDR; FC适配器: QLogic QLA2310F; SCSI适配器: LSI22320双通道适配器; SCSI硬盘: Seagate ST3146807LC (硬盘参数: 平均寻道时间为 4.7ms, 邻道寻道时间为 0.5ms, 转速为 10 000rpm; 峰值数传率为 320Mbps) RAID级别为 RAID5, 阵列分块大小设置为 64KB, 共八个盘, 两个通道, 每个通道四个盘。扩容前阵列内部三个盘组成 RAID5的卷, 该卷构成 JBOD卷组。扩容后五个盘组成 RAID0的卷, 该卷组成 JBOD卷组。阵列控制软件未采用流水 I/O调度策略。软件平台为 VxWorks5.5; 测试工具为 IOMeter(2004.07.30)。

在 IOMeter中将传输请求块设置为 128KB, 在不同的工作负载中, Outstanding I/O个数分别为 1, 2, 4, 8, 16时, 分别测试了在线扩容前后以及扩容过程中阵列对 100%顺序读写请求的数传率和平均响应时间。上述实验条件下的测试结果如表 1和表 2所示。

表 1　100%顺序读 RAID5

Outstanding I/O个数	扩容前		扩容过程中		扩容完成后	
	数传率 (Mbps)	平均响应时间 (ms)	数传率 (Mbps)	平均响应时间 (ms)	数传率 (Mbps)	平均响应时间 (ms)
1	19.50	6.400 3	2.38	72.466 4	29.77	4.247 1
2	24.68	10.124 6	2.48	144.500	38.53	6.360 7
4	25.49	19.679 0	2.50	289.134	43.15	11.597 9
8	25.37	39.486 2	2.51	578.199	46.86	21.506 0
16	25.23	79.174 6	2.52	998.656	46.56	42.798 5

表 2 100% 顺序写 RAID5

Outstanding I/O 个数	扩容前		扩容过程中		扩容完成后	
	数传率(Mbps)	平均响应时间(ms)	数传率(Mbps)	平均响应时间(ms)	数传率(Mbps)	平均响应时间(ms)
1	23.51	5.307 8	3.38	54.460 9	28.28	4.427 6
2	28.29	8.847 2	3.48	108.964	35.21	7.043 8
4	28.31	17.666 5	3.50	206.136	35.47	14.086 8
8	28.23	35.959 7	3.52	413.133	35.55	28.286 8
16	28.12	71.202 6	3.53	827.122	35.53	56.279 5

表 1、表 2表明: 在 RAID5的在线扩容过程中, 虽然阵列系统对用户读写请求的数传率会变小, 且平均响应时间会变长, 但最关键是保证了在增加阵列容量的同时不影响用户的正常使用。由于 RAID0的实现比 RAID5容易, 既然 RAID5的扩容过程没有影响用户的使用, 证明以上提出的在线扩容方案是切实可行的, 并可将其推广于大规模存储系统的在线扩容过程中。

4 算法分析

(1)本文提出的在线扩容方案的创新之处是:

①它无须考虑 RAID 级别, 普适性很强, 既可以用于 RAID0, 也可以用于 RAID5。相对于受限于 RAID 级别的扩容算法这是一个很大的优势。

②它不用考虑扩容前 RAID 中数据在盘上的布局, 也不用考虑扩容后数据在阵列中的布局, 数据在阵列中的布局可以交给阵列主控[4]去实现, 且在整个实现过程中关键是维护两份配置信息, 实现容易。相对于以磁盘为导向的算法这也是一种优势。

③该算法允许在进行负载均衡和在线扩容的同时改变阵列卷一级的分块大小。例如可按照 64KB 的阵列分块大小将数据从阵列中 M 个盘上读出来, 然后按照 16KB的分块大小将数据重新分布到扩容后的阵列中去。显然在此过程中就可以将某个盘上频繁访问的物理段迁移到其他磁盘上, 这样便可提高存储的效率。同理, 可以将阵列的分块大小由 16KB 变为 64KB, 便可以在某些逻辑块不被频繁访问时将它合并后放到某个磁盘上。

(2)该算法由于有很强的通用性, 很可能会在其他性能指标上低于其他的扩容算法, 如用户的响应时间可能相对其他针对磁盘作考虑的算法来说会长一些。但是在用户响应时间稍长的同时, 扩容的时间就会相对短一些。在实际应用中, 可以尽量在存储系统比较空闲的时候进行在线扩容操作。但很多类型的应用还是要求尽可能短的响应时间, 今后应在缩短用户响应时间上再作进一步研究。

5 结论

本文提出并实现了一种新的 RAID 系统在线扩容方案, 由于它无需考虑数据在磁盘上的布局问题, 故与阵列级别无关, 通用性很强。该扩容算法的新颖之处在于只需维护新旧两份不同的磁盘阵列配置信息就可以实现不同 RAID 级别系统中的在线数据迁移, 且该算法能对动态负载平衡以及动态 RAID 级别更改等功能的实现起到一定的导向作用。实验结果也表明这是一种很可取的在线扩容方案。在今后的研究中, 还要针对不同的用户请求优化算法。

参考文献:

[1] Gregory R Ganger. Disk Subsystem Load Balancing Disk Striping vs Conventional Data Placement[C]. Proceedings of the Hawaii International Conference on System Sciences, 1993. 40-49.

[2] Scheuermann P, Weikum G, Zabback P. Data Partitioning and Load Balancing in Parallel Disk Systems[J]. The VLDB Journal, 1998, (2): 48-66

[3] Asit Dan, Dinkar Sitaram. An Online Video Placement Policy Based on Bandwidth to Space Ratio (BSR) [C]. Proceedings of the ACM SIGMOD International Conference on Management of Data, 1995. 376-385.

[4] 张江陵, 冯丹. 海量信息存储[M]. 北京: 科学出版社, 2003. 23-31.

[5] 陈安定, 谢汶. Linux下的逻辑卷管理[J]. 四川电力技术, 2003, 26(1): 47-48

作者简介:

冯丹(1970-), 女, 湖北京山人, 教授, 博导, 主要研究方向为计算机存储系统; 彭丽(1982-), 女, 湖北监利人, 硕士研究生, 主要研究方向为计算机存储系统。

(上接第 243页)下一步的工作重点是网格相关的技术、标准以及实际应用的开发与推广。

参考文献:

[1] 顾翊, 张申生, 朱祥飞. 一种企业应用集成(EAI)方案的研究[J]. 计算机工程与应用, 2003, 1(6): 209-212.

[2] 周航滨, 夏安邦, 张长昊. 基于Web服务的跨企业信息集成框架[J]. 计算机集成制造系统-CIMS, 2003, 9(1): 2-5.

[3] 黄双喜, 范玉顺, 赵大哲, 等. 基于Web服务的企业应用集成[J]. 计算机集成制造系统-CIMS, 2003, 9(10): 864-867.

[4] 孙晋文, 肖建国. 企业应用集成与基于Web Services的构架应用[J]. 计算机工程与应用, 2003, 39(21): 205-208.

[5] 刘英丹, 董传良. 利用Web Service实现企业应用集成[J]. 计算机应用, 2003, 23(7): 124-126.

[6] 黄允中, 顾志松, 张世永. 网格技术框架的探讨和研究[J]. 计算机工程, 2003, 29(13): 133-134.

[7] I Foster, C Kesselman, S Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations[J]. Intl Journal of High Performance Computing Applications, 2001, 15(3): 200-222

[8] I Foster. What is the Grid? A Three Point Checklist[EB/OL]. http://www.gridtoday.com/02/0722/100136.html, 2002-07-22

[9] I Foster, C Kesselman, S Tuecke, *et al*. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration[EB/OL]. http://www.globus.org/research/papers/ogsa.pdf, 2004-11-11.

作者简介:

于锋(1980-), 男, 山东烟台人, 硕士, 研究方向为网格、分布式计算、嵌入式系统等; 曹乾(1962-), 男, 山东荣成人, 教授, 硕士, 研究方向为网格、CIMS、嵌入式系统等。