RAID 系统扩容方案研究综述

元 铸,谢 平,耿生玲
(青海师范大学计算机学院,青海西宁 810008)

摘 要: 随着云计算、物联网和人工智能等 IT 技术驱动数字经济产业的繁荣发展,现代企业信息化需求对数据中心存储能力提出了更高的要求和挑战. RAID 系统因具备良好的数据存储可靠性和磁盘阵列可扩展性而得到广泛应用. 为了满足海量数据对存储容量日益增长的需求,业界普遍采用扩容现有 RAID 系统以应对海量数据的存储问题. 电子商务、Web 服务和金融等行业对数据的实时访问,使得数据中心必须为用户提供 7 * 24 的高质服务响应,然而数据迁移量,负载均衡和扩容开销等因素都会影响扩容的效率,因此如何设计出一种高效的扩容方案越来越受到科研人员的关注. 本文根据研究对象的不同将 RAID 扩容方案分为:基于块存储、对象存储、文件系统存储的扩容方案,同时根据 RAID 扩容方案研究历程和优化策略的不同,又可分为优化数据迁移过程和减少数据迁移量的扩容方案. 文章结合不同的应用场景介绍了典型和常见的扩容方案,并从评价扩容方案性能各项重要指标的角度详细分析了现有的 RAID 扩容方案,并指出各种扩容方案的不足以及未来可能的发展方向. 如何权衡影响扩容方案性能的各项因素,设计出数据迁移少,负载均衡好,扩容开销低的高效扩容方案,将会是今后相当长一段时间海量数据存储研究的热点问题.

关键词: 海量数据存储; RAID 系统扩容; 扩容方案; 扩容方案性能指标; 高效扩容
中图分类号: TP361.4 **文献标识码:** A **文章编号:** 0372-2112 (2019)11-2420-12
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2019.11.026

Summary of Research for RAID System Scaling Schemes

YUAN Zhu, XIE Ping, GENG Sheng-ling
(*The Computer College of Qinghai Normal University, Xining, Qinghai* 810008, *China*)

Abstract: With the development of digital economy industry driven by IT technologies such as cloud computing, internet of things and artificial intelligence, modern enterprise information demands put forward higher requirements and challenges for data center storage capacity. Due to data storage reliability and redundant arrays of independent disks scalability, RAID system is widely used. In order to fulfill the requirements of massive data for the increasing storage capacity, the industry generally adopts a method of scaling RAID system to solve the problem of storing massive data. E-commerce, web service and finance access data on real-time, which make the data center must provide high-quality service response for users with 7 * 24, but the factors of data migration, load balane and scaling cost will affect the efficiency of RAID scaling. Therefore, how to design a fast and efficient scaling scheme is getting more and more attention from researchers. According to the different investigative object, this paper classifies RAID scaling approaches into basing on block, object and file system. Meanwhile, according to the investigative developing process and different optimization strategies of RAID scaling, the approaches also can divide into optimizing data migration process and reducing the number of data to be moved. From the perspective of evaluating the performance of the scaling schemes, this article introduces the typical and common scaling schemes by different application scenarios that points out the shortcomings of various scaling schemes and possible improvements. How to balance the factors that affect the performance of the scaling schemes and design a high-performance scaling scheme with less data migration, good load balance, and low scaling cost, which will be a hot issue for massive data storage research for a long time in the future.

Key words: massive data storage; RAID scaling; scaling schemes; index of scaling performance; efficient scaling

---

收稿日期:2019-02-01;修回日期:2019-07-25;责任编辑:马兰英
基金项目:国家自然科学基金(No. 61762075, No. 61862055);青海省自然科学基金(面向大数据环境的磁盘阵列高效扩容架构研究);赛尔网络下一代互联网技术创新项目(No. NGII20180116);青海省物联网重点实验室建设专项(No. 2017-ZJ-Y21)

1　引言

随着大数据时代的深入,用户数据量的指数增长[1,2]和存储管理中成本低廉的存储设备,导致数据中心组件出错愈加频繁. 根据 Sankar 等人[3]对大规模数据中心超过 1000000 台服务器的损坏情况进行为期超过两年的研究和分析,数据显示磁盘硬件故障占数据中心所有设备故障比例的 71.1%. 与传统的三副本策略相比,纠删码利用更少的冗余实现了与副本技术相同的数据存储可靠性需求. 因此,纠删码在 NetApp[4], Cleversafe[5], Windows Azure[6] 和 Oceanstore[7] 等现代存储系统中得到了应用. 同时,根据中国互联网络信息中心[8]发布的数据显示,截至 2017 年 12 月,我国网民规模达 7.72 亿,普及率达到 55.8%,超过全球平均水平(51.7%)4.1 个百分点;CISCO 公司 2019 年 2 月发布的移动可视化网络指数 VNI(2017 - 2022)[9]指出,到 2022 年,全球数据中心流量将增长 3 倍,全球数据中心流量每年将增长 20.6ZB. 用户数据量的增长对数据中心的存储能力和 I/O 带宽提出了更高的要求,所以在现有存储系统的基础上进行扩容就成为一种行之有效的解决方案.

RAID(Redundant Array of Independent or Inexpensive Disks)[10,11]系统具有良好的数据存储可靠性和磁盘可扩展性,同时并行 I/O 为存储系统提供了更大的 I/O 带宽. RAID 系统从诞生至今,根据不同的磁盘组成方式,已经形成多种 RAID 层级,如 RAID0、RAID1、RAID4、RAID5、RAID6 等. 针对大数据环境下,数据体量大、增长快的特点,结合 RAID 存储系统低成本、高可靠、易扩展的存储特性,采用对 RAID 系统扩容的方法将能有效解决海量数据的存储问题.

为满足用户对数据增长带来的大存储能力和高 I/O 性能的需求[12,13],我们采用向 RAID 系统中添加新磁盘的方式. RAID 系统添加新磁盘后,需要迁移一定的数据到新磁盘使得新 RAID 系统满足负载均衡,其中,数据迁移量的多少,校验开销的大小,能否保证数据一致性等问题,都会影响 RAID 系统的扩容效率. 由于海量存储系统负载特征和应用环境的复杂性,现有 RAID 系统扩容需要考虑以下 5 个方面的问题.

(1)RAID6 系统允许双盘失效,具有较高的数据存储可靠性,但由于编码的复杂性与布局的特殊性,会造成扩容开销大,扩容时间长的问题. 特别是,在单节点数据存储采用 RAID6 系统时,计算开销大的问题更加显著.

(2)现有电子商务、Web 服务、金融等行业对数据的实时访问有很高的要求,数据中心能否为用户提供实时在线的高质服务将是一个极大地挑战.

(3)存储数据由于热度的不同,冷热数据访问频率存在显著差异[14]. 冷数据占用大量空间,将会造成存储资源的浪费. 同时,磁盘的高能耗存储也将影响存储系统的运行成本和数据存储可靠性[15,16].

(4)随着分布式存储和面向对象存储的应用[17~19],异构环境下的存储系统扩容,数据分布策略将成为高效扩容的关键因素.

(5)现有扩容方案主要针对 RAID 系统进行扩容,当将其应用于集群环境下的数据存储时[20,21],额外的网络通信开销将对整个扩容过程造成影响.

文章通过结合不同的应用场景介绍了典型和常见的扩容方案来展示 RAID 系统扩容研究的现状,从评价扩容方案性能各项重要指标的角度对比了现有的扩容方案,并分析了各种方案的不足与可能的改进之处,指出了未来扩容方案可能的研究方向.

2　RAID 系统扩容的相关概念

RAID 存储系统是由多个磁盘组合而成的磁盘组,具有低成本、高可靠、易扩展的特点,目前为了保证存储可靠性和数据可用性,数据中心普遍采用 RAID 系统对数据进行条带化管理. 为了便于理解,给出 RAID 系统扩容的相关概念的说明和定义.

(1)数据(Data):原始的用于存储真实用户信息的一块字符串.

(2)块(Block):计算存储最基本的单位. 根据存储信息的不同,将其分为:数据块和校验块. 数据块存储用户数据的真实信息,校验块存储用于恢复用户数据的冗余信息.

(3)条带(Stripe):把连续的数据分割成相同大小的数据块,将每段数据存放在不同的磁盘. 条带可以是一行数据块和校验块的集合或由多列数据块和校验块构成的矩阵. 如图 1 所示.

(4)条块(Strip):处于同一磁盘的属于同一条带的数据集合. 一个条块可以是一个块,也可以是几个块的集合;可以是只含有数据块或校验块,也可以是同时含有数据块和校验块.

(5)校验链(Parity Chain):一条校验链包括校验块和生成该校验块的数据块. 根据校验链在条带中的布局不同,可分为:行校验链、斜校验链和反斜校验链.

(6)编码(Encoding):根据纠删码规则[22]对数据块进行计算,生成冗余数据的过程.

(7)扩容(Scaling up):向 RAID 系统中添加新磁盘,增加存储能力,提升存储性能的过程.

(8)元数据(Metadata):包含数据的地址、属性等相

关信息,存放在磁盘的起始位置,对数据的任何操作都将导致相应元数据的更新.

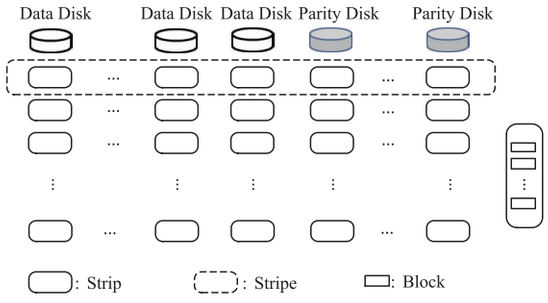

图1 块、条块和条带之间的关系

3 扩容方案的性能评价标准

为了评价扩容方案的性能,本文主要从以下几个方面进行分析.

(1)均匀数据分布:假设原始 RAID 系统有 m 个磁盘,系统中总数据量为 B,扩容时向 RAID 系统添加 n 个新磁盘,为实现均匀数据分布,则每个磁盘应含有 $\frac{B}{m+n}$ 个数据块.(以下如无特殊说明,m、n、B 均表示此含义)

(2)最小数据迁移:添加新磁盘后,RAID 系统由 $m+n$ 个磁盘组成,为实现均匀数据分布,只需从旧磁盘迁移原始总数据量的 $\frac{n}{m+n}$ 到新磁盘. 缩容时,RAID5 最小数据迁移量应满足原始总数据量的 $\frac{1}{n}$,RAID6 最小数据迁移量应满足原始总数据量的 $\frac{n}{m}$.

(3)最小扩容开销:扩容过程存在数据迁移和数据校验,即不仅要保证数据迁移 I/O 开销最小,存在校验的情况下,要同时保证校验 I/O 开销和 XOR 计算开销最小.

$$Cost_{扩容}=IO_{迁移}+IO_{校验}+XOR_{计算}$$

(4)快速数据寻址:利用低空间和时间复杂度的算法计算出数据块在 RAID 系统中的相应位置.

(5)双向扩容:RAID 系统不仅可以添加新磁盘实现扩容,而且可以删除无效和低性能磁盘或考虑节能关闭磁盘实现缩容.

4 基于块存储和对象存储的扩容方案

根据研究对象的不同,本文将扩容方案分为:基于块存储,基于对象存储和基于文件系统存储的方案. 同时,根据优化策略的不同,又可将扩容方案分为优化数据迁移过程和减少数据迁移量的方案.

4.1 基于块存储的扩容方案

4.1.1 优化数据迁移过程的 RAID 系统扩容

针对扩容过程中面临的问题,部分科研人员从数据迁移的角度,对整个 RAID 系统扩容过程进行优化,其中,MDM、SLAS、ALV 等方案都表现了良好的性能,但也显露出数据迁移量大,无法实现均匀数据分布等不足.

RR(Round-Robin)方案又称为轮询方案,此方案几乎需要迁移所有数据块,迁移数据量巨大,但多次扩容后均能保持均匀数据分布. RR 方案第 i 次扩容的 $f_i(x)$ 形式化如下:

$$f_i(x)=\begin{cases}d=x\bmod N_i\\ b=x/N_i\end{cases} \tag{1}$$

其中,x 代表数据块的逻辑编号,N_i 表示第 i 次扩容后的磁盘总数,d 表示数据块所处的磁盘编号,b 表示数据块所处的物理块编号.

Gonzales 和 Cortes[23] 提出 GA 算法,加速了整个 RAID5 系统的扩容过程. 但整个过程几乎迁移了所有数据,存在 XOR 计算开销大,数据重分布时间长的问题.

MD-Reshape[24] 是 Brown 在 Linux 内核中设计的一个重构工具包,它使用固定大小的数据窗口写入映射元数据,用户对数据窗口的访问请求必须排队,直到窗口中所有数据全部迁移完毕. 此方案由于需要频繁的更新元数据,容易形成 I/O 瓶颈,而且并没有解决数据迁移量大的问题.

专利#6000010[25] 提出一种 RAID5 的扩容方案,该方案不需在原始磁盘重写数据块和校验块,扩容过程分为三步:(1)将原始 RAID5 系统中特定行的检验块转换为数据块,但不改变块中所存储的内容;(2)添加新磁盘后,使得每个条带为一个校验块和 $N-1$ 个数据块(N 代表扩容后总磁盘个数);(3)初始化新的校验数据和新的数据盘,使得异或结果为 0. 图 2 中,如步骤(1)所述,将 $P3$,$P4$ 转换为数据块,但块的内容并不改变. 该方案虽然不需在原始磁盘重写校验块,但扩容后不均衡的校验块布局,会导致严重的写惩罚.

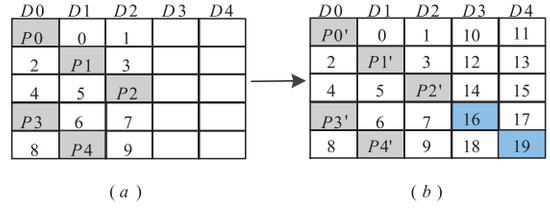

图2 RAID5(3+2)利用专利#6000010进行扩容

MDM[26] 向原始 RAID5 系统添加新磁盘后,将新磁盘中数据与原始 RAID5 系统中的部分数据进行交换,如图 3 所示,经过重组后的 RAID5 系统使数据寻址更复杂,而且数据块和校验块并不能实现均匀分布. 经实验表明,扩容后 RAID5 系统的存储能力得到增加,但存储效率并未得到提升,仅仅保持了原始 RAID5 系统的存储效率.

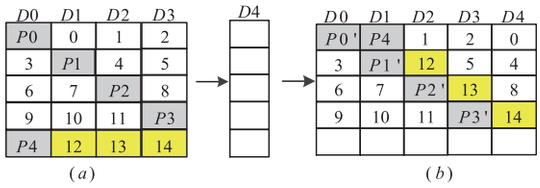

图3　RAID5（4+1）利用MDM方案进行扩容

专利#7111117[27]提出的方案需要在 RAID5 系统中预留备用磁盘,扩容时将原始 RAID5 系统中的数据重分布于新的 RAID5 系统,同时将数据重分布过程中产生的新数据映射到备用磁盘,待数据重分布完成后,将新数据迁移到新的 RAID5 系统中,备用磁盘恢复为初始空闲状态. 该方案确保在线扩容过程中对前台用户 I/O 访问造成的影响最小,但也造成存储空间不能得到充分利用的问题.

SLAS[28]是一种在线扩容方案,该方案提出三种优化策略:滑动窗口技术、懒惰的元数据更新和聚合 I/O. 该方案对元数据的管理采用映射函数与映射表相结合的方法,即对不需要移动的数据块,采用映射函数进行管理,对需要移动的数据块(即滑动窗口内的数据)采用映射表的方式进行管理. 这种映射函数与映射表相结合的方式,减少了占用存储空间的大小,保证了数据存储的可靠性和一致性.

ALV[29]充分利用重排序窗口的特性结合滑动窗口机制,使得单一 I/O 能访问多个连续块,同时该方案根据应用程序工作负载,自适应地调整数据迁移速率的思想,提升了用户访问 I/O 的响应速度,缩短了响应时间. 但该方案同样具有数据迁移量大的不足,与 MD-Reshape 相比,ALV 降低了 53.31% ~73.91% 的用户响应时间和 24.07% ~29.27% 的数据重分布时间.

4.1.2　减少数据迁移量的 RAID 系统扩容

另一部分研究人员从减少数据迁移量的角度对 RAID 系统扩容进行优化,其中 FastScale、McPod、GSR、MiPiL、ISM 等方案对 RAID0、RAID4、RAID5 系统扩容,减少了数据迁移量,缩短了扩容时间. 同时针对 RAID6 系统,根据不同编码的布局特点,提出了 RS6、Xscale、HCS 等扩容方案.

AutoRAID[30]是一种综合多种 RAID 层级优势且支持在线扩容的技术,扩容时只需将新磁盘安装好,系统马上就可以利用新磁盘的空间,不需要迁移数据到新磁盘. AutoRAID 技术另一个优势在于对添加磁盘的容量没有统一要求,可以管理由不同容量磁盘组成的 RAID 系统.

Semi-RR[31]方案利用伪随机函数只需从原始磁盘迁移一定比例的数据到新磁盘,数据迁移量减少,如图 4 所示,但多次扩容后不能保证均匀数据分布. Semi-RR 方案第 i 次扩容过程的 $g_i(x)$ 形式化如下:

$$g_i(x)=\begin{cases} g_{i-1}(x), & \text{if}(x \bmod N_i) < N_{i-1} \\ f_i(x), & \text{otherwise} \end{cases} \tag{2}$$

其中,N_i 表示第 i 次扩容后的磁盘总数,N_{i-1}表示第 i 次扩容前的磁盘总数,$f_i(x)$由上述式(1)可得.

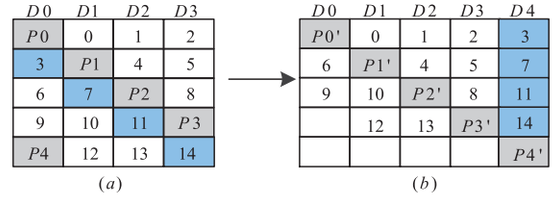

图4　RAID5 (4+1) 利用Semi-RR方案进行扩容

GSR[32]对 RAID5 扩容过程中与 RR 相比 I/O 操作数减少了 81.5%,数据迁移量降低了 68%,同时 GSR 不仅可以实现扩容,而且可以实现缩容,这是很多方案都不具备的特点. 如图 5 所示,GSR 将磁盘存储的数据分为 Retained OUS、Remapped OUS、Destructed OUS 三个区域,根据划分的三个区域分别进行处理,只需迁移原始总数据量的$\frac{n}{m+n}$,即可实现均匀数据分布. 但 GSR 这种分区域的方案,也造成只有原始磁盘可以访问保持不变区域的数据只有新磁盘才能访问迁移区域的数据,在工作负载表现出较强的局部性访问时,将会造成很大的性能损失.

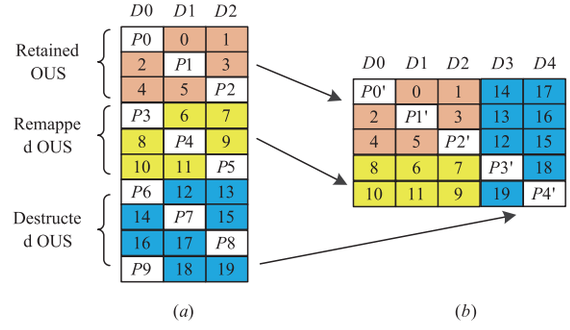

图5　RAID5（3+2）利用GSR方案进行扩容

CRAID[33]扩容时统计出 RAID5 系统中访问热度较高的数据(称为 Hot Data),将 Hot Data 存放在专门的磁盘,改善了相关数据访问的顺序,分摊了重组成本,提升了用户访问性能. CRAID 虽然只需迁移少量数据,但检测热数据变化的开销,将对存储性能造成影响. 并且,RAID 系统扩容属于偶然事件,但对热数据信息的统计却一直被执行.

Fast Scale[34]提出划定平行四边形迁移区域的策略,其中将一个磁盘上连续的 $m+n$ 个数据称为一个 Segment,将 $m+n$ 个磁盘中处于同一物理地址的 Segment 称为一个 Region. 该方案将处于迁移区域的数据迁移到新磁盘,迁移区域的宽为 m,高为 n,假设 d_0:表示数据所处的原始磁盘编号,b_1:表示数据在 Region 中的原始位置,d:表示数据迁移后新的磁盘编号,b:表示数据迁移后新

的物理块编号,b_0:表示数据的原始物理块编号. 迁移时,分为两种情况:(1)如果 $m \geq n$ 且 $b_1 \leq n-1$,则 $d = d_0 + m$;如果 $m \geq n$ 且 $b_1 \geq m-1$,则 $d = d_0 + n$;反之 $d = m + n - 1 - (b_1 - d_0)$. (2)如果 $m < n$ 且 $b_1 \leq m-1$,则 $d = d_0 + m$;如果 $m < n$ 且 $b_1 \geq n-1$,则 $d = d_0 + n$;反之 $d = d_0 + b_1 + 1$. 物理块编号两种情况下均不发生改变 $b = b_0$. 如图6和图7所示,该方案采用聚合I/O和检查点延迟优化了数据迁移过程,保证了数据一致性. Fast Scale与SLAS相比,减少了86.06%的数据重分布时间.

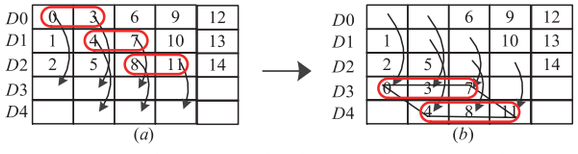

图6 利用聚合I/O,并行读取数据块,使得原来6个读I/O变为3个;并行写入数据块,使得原来6个I/O,变为2个

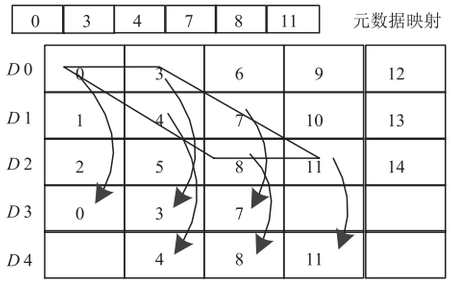

图7 检查点延迟,使得元数据更新I/O次数降低,确保数据一致性

McPod[35]方案是对有独立校验盘的RAID4进行扩容. 其中,"***M***":表示只从旧磁盘迁移少量数据到新磁盘. "***c***":表示聚合I/O,将小而多的I/O转变为大而少的I/O. "***P***":表示Piggyback和平行校验更新,数据只在同一校验链内迁移,使得校验数据在数据迁移过程中无需更新. "***o***":表示外包校验更新,将更新的校验数据存放在代理磁盘,如图8所示,代理磁盘最终代替原来的校验磁盘成为新的校验磁盘. "***d***":表示懒惰的元数据更新,减少元数据更新次数. 该方案与RR相比缩短了67.78%~79.64%的数据重分布时间和14.24%~27.16%的用户响应时间.

PBPC[36]算法利用平行四边形策略完成数据迁移后,为了适应RAID5布局,对新RAID系统中校验数据的位置进行判断:(1)新的校验块位置为空;(2)新的校验块位置为旧校验数据;(3)新的校验块位置为数据块. 当为前两种情况时,新的校验块位置按照该块为空的情况,直接填写新校验块数据. 当为第三种情况时,需要找到旧校验块的位置与数据块进行交换,使得新的校验位置为旧校验数据,之后,则按第二种情况处理. 该方案同样满足了均匀数据分布和最小数据迁移的要求.

MiPiL[37]方案是一种适用于RAID5系统的扩容方案,同样采用Segment和Region的区域划分方式. 如图9

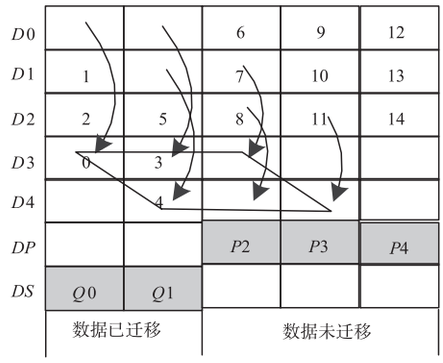

图8 外包校验更新,DP为检验磁盘,Ds为代理磁盘

所示,扩容时,首先对RAID5系统进行标准化,标准化过程分为两步:(1)利用 $r = (r_0 + \delta) \bmod N_0$ 变换行号,其中 r_0 表示原始行号,N_0 表示扩容前的磁盘个数,δ 表示 $N_0 - 1$ 与最后一列中校验数据所处行号之间的差值;(2)将Region中第一个 N_0 列中的校验数据块,依次分布在对角线上. 采用Piggyback校验更新,实现了最小校验开销,同时,懒惰的元数据更新,减少了元数据写操作次数,保证了数据的可靠性和一致性. MiPiL方案在线扩容过程中,与RR相比降低了74.07%~77.57%的数据重分布时间和25.78%~70.50%的用户响应时间.

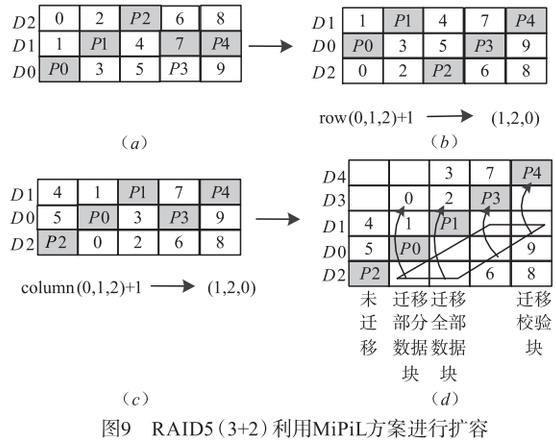

图9 RAID5(3+2)利用MiPiL方案进行扩容

PBM[38]采用Zone和Group的策略对原始RAID5系统进行区域划分,假设 z:表示每个Zone在Group中的编号,s:表示每个条带在Zone中的条带号,d:表示每个数据所处的磁盘号,(z,s,d):表示一个数据块的Zone编号、条带编号和磁盘编号. 扩容时,当一个area区域中存在两个校验块时,需要将处于较大条带号的校验块和通过预定义与其关联的数据块一同迁移到新磁盘,其中 p 表示校验块,用 (z_p, s_p, d_p) 对其进行预定义,当 s_p 满足 $s_p \in [m, m+n-1]$ 时,利用等式 $A(P) = \{z_p, s_p - i, (d_p - 2i) \bmod m) \mid 0 \leq i \leq m-1\}$ 找出与校验块相关联的数据块. PBM方案能否达到数据块和校验块的均匀分布与 m 有关,如图10所示,当 m 为奇数时,每

个 Zone 中能达到均匀数据分布;如图 11 所示,当 m 为偶数时,每个 Group 中能达到均匀数据分布. PBM 满足了在线场景下的扩容要求,与 RR 扩容方案相比缩短了 73.6% 的扩容时间.

ISM[39] 同样采用 Segment 和 Region 的定义对 RAID5 系统进行区域划分,如图 12 所示,利用迁移窗口 Window 对数据进行迁移,迁移窗口 Window 宽为 m,高为 n. 其中 Segment、迁移窗口 Window 和磁盘的关系为:Segment i $(0 \leq i \leq m+n-1)$,迁移窗口所处的条带编号为 $S_{i*m+j}(0 \leq j \leq m-1)$,磁盘号为 $D_{m-i+k}(0 \leq k \leq n-1)$. ISM 支持在线扩容,多次连续扩容均能保持均匀数据分布,实现了最小数据迁移,数据迁移过程中避免了校验数据的更新. 在采用真实负载模拟在线扩容场景时,ISM 与 GSR 相比减少了 64.15% ~87.30% 的扩容时间,与 ALV 相比降低了 92.38% ~95.98% 的扩容时间.

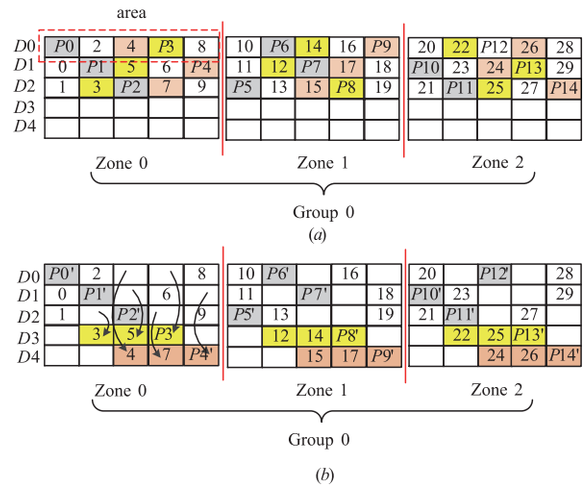

图10　RAID5利用PBM方案进行扩容，为奇数，每个Zone都能达到数据均匀

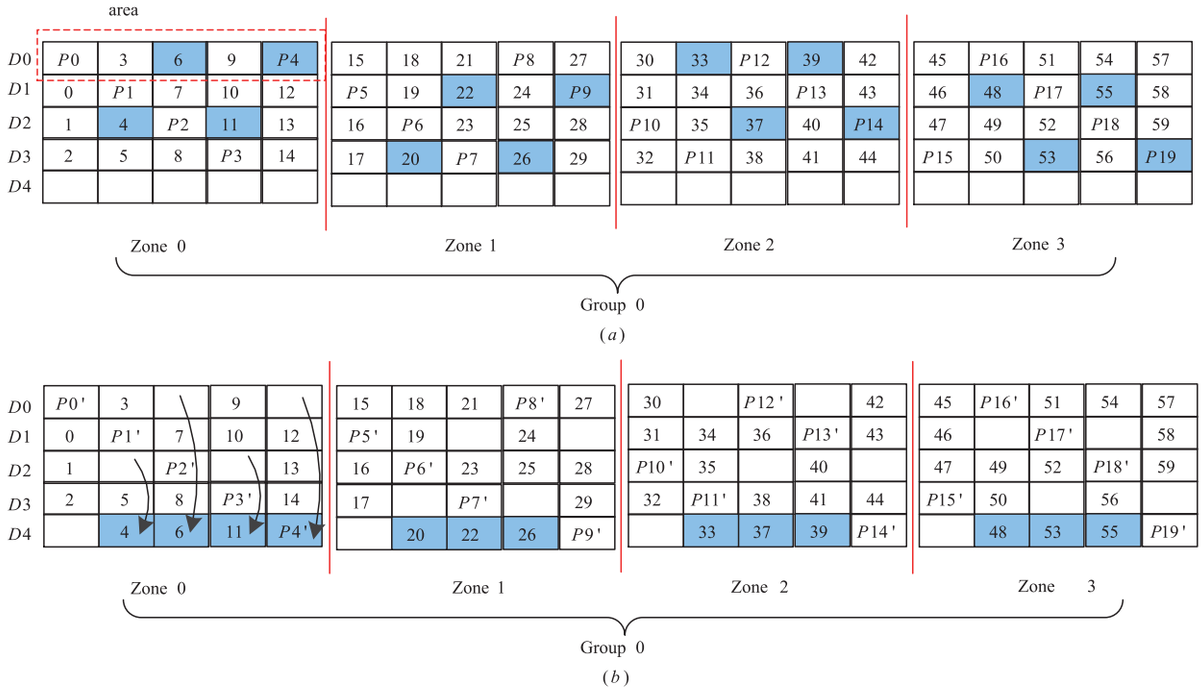

图11　RAID5利用PBM方案进行扩容，m为偶数,每个Group都能达到均匀数据分布

SDM[40] 是为适应 RAID6 独特的编码布局而提出的扩容方案,扩容过程分为四步:(1)SDM 对要迁移块的优先级进行定义;(2)对数据布局进行比较;(3)负载均衡检验,通过数据在条带中的布局,判断是否达到负载均衡;(4)将数据迁移到相应位置. SDM 通过对迁移数据优先级进行判断,使得扩容过程中只需迁移最少的数据,即可实现均匀数据分布,与 RR 和 Semi-RR 相比,减少了 72.7% 的迁移 I/O 操作,缩短了 96.9% 的迁移时间.

RS6[41] 是基于横式编码 RDP 提出的一种扩容方案,RDP 编码的 RAID6 存在行检验和斜校验两个校验盘,存储区域满足 $(p-1)*(p+1)$ 的布局(p 必须为素数). RS6 扩容过程中,如图 13 和图 14 所示,首先,选择最佳的迁移参数,确定新磁盘位置;然后确保数据只在同一条行校验链移动,保证行校验数据无需更新;同时数据迁移到新位置后,原始斜校验数据得到充分利用,保证斜校验开销最小. Piggyback 校验更新,保证了在线扩容过程中数据存储的一致性和可靠性. RS6 与 RR 相比,减少了 60.0% ~88.9% 的数据迁移量,降低了 40.27% ~69.88% 的迁移时间.

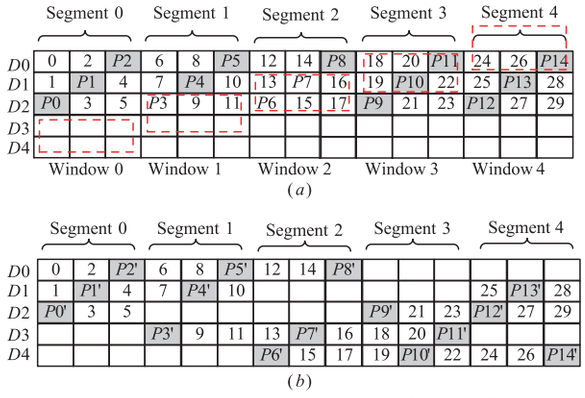

图12 RAID5（3+2）利用ISM方案进行扩容

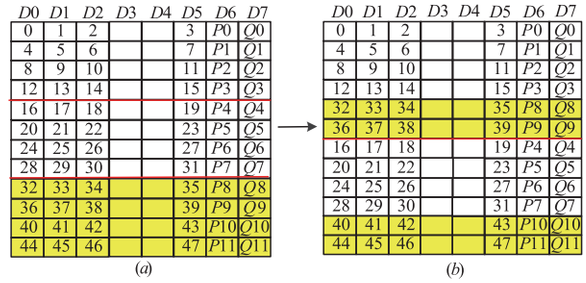

图13 RS6方案的标准化，由p=5添加2个新磁盘扩容到p=7

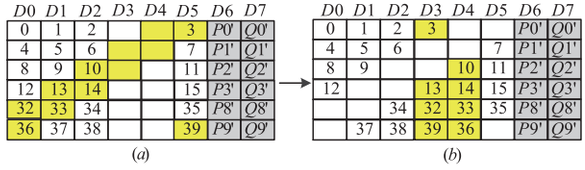

图14 划定迁移区域，将数据迁移到新磁盘

HCS[42]是基于纵式编码 H-Code 提出的一种扩容方案. H-Code 编码的 RAID6 存在一个独立的行校验盘，斜校验数据分布于数据盘，其中存储条带满足$(p-1)*(p+1)$布局(p必须为素数). HCS 扩容时，主要采用两种策略：(1)反对角数据选择；(2)水平迁移数据. HCS 与 SDM 相比减少了 3.6% 的扩容时间，提升了 4.62% 的存储性能，但利用 HCS 扩容后并不能保证均匀数据分布.

Xscale[43]是一种基于 X-Code 编码的在线扩容方案，X-Code 编码的 RAID6 没有单独的校验磁盘，数据块与校验块分布于不同的条带，且存储条带需为 $N*N$(N必须为素数)的布局. Xscale 主要采用最小数据迁移和清除元数据更新两种技术实现 RAID6 扩容，迁移时，从逻辑视角分析，将数据块与逻辑块分开，根据 m 和 n 的值确定迁移区域，其中宽为 n，高为 $m+n$，确定迁移区域后，将区域中的数据迁移到新磁盘. 如图 15 所示，根据转换规则需要将最后两行 13、14 由校验行转换为数据行. Xscale 与 RR 相比，数据迁移量减少了 63.6% ~ 89.5%，数据重分布时间减少了 35.62% ~ 37.26%，I/O 延迟降低了 23.29% ~ 37.74%.

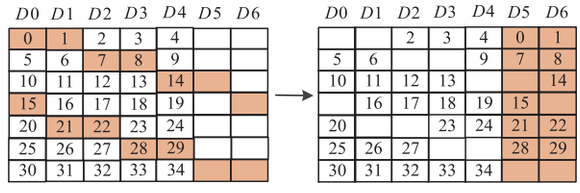

图15 利用Xscale方案对基于X-code编码的RAID6进行数据迁移

H-Scale[44]方案具备四个优点：(1)动态地减少数据迁移开销，加速整个扩容过程；(2)保证原始存储数据的一致性；(3)扩容后达到均匀数据分布；(4)为不同 RAID 层级提供普遍的扩容方案. H-Scale 确保数据只从原始磁盘迁移到新磁盘，避免了访问瓶颈的产生，如图 16 和图 17 所示，为不同 RAID 层级扩容，提供了一种普遍的在线扩容方案，实现了最小数据迁移，在采用真实负载测试的情况下，加速了 60% 的扩容过程.

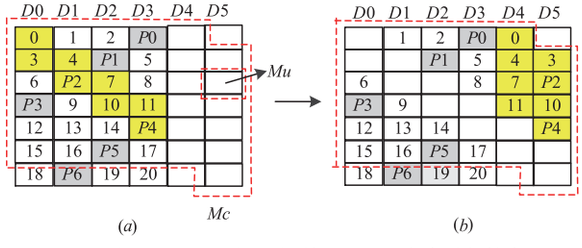

图16 RAID5（4+2）利用H-Scale方案进行扩容

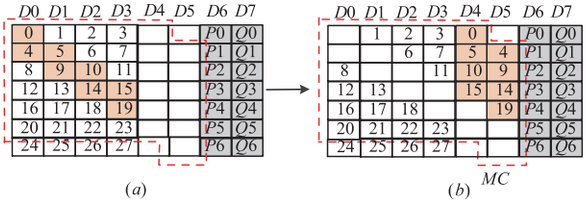

图17 利用H-Scale方案对RDP编码的RAID6进行扩容

MDS-Frame[45]方案根据不同纠删编码的布局，在两种不同编码之间相互转换，实现 RAID6 系统的双向扩容. MDS-Frame 根据编码规则的不同，分为 $p-1$，p，$p+1$ 和 $p+2$ 四种类型编码(p 为素数)，当扩容或缩容时，将两种编码转换迁移的实际数据量与平均数据迁移量进行比较，如小于平均迁移量则为高效扩容，否则为低效扩容. 经实验证明，MDS-Frame 与其他扩容方案相比，减少了 44.1% 的迁移 I/O，缩短了 95.2% 的迁移时间.

ADR[46]能将 RAID 系统中低性能或损坏的磁盘剔除，降低数据中心能耗，确保 RAID 系统能够提供更高的存储性能. 如图 18 所示，ADR 实现了 RAID6 系统的缩容，减少了数据寻址开销，降低了 XOR 计算开销，满

足了缩容方案最小数据迁移量的要求,与 RR 相比,减少了 52.1% 的 I/O 操作,降低了 63.5% 的迁移时间.

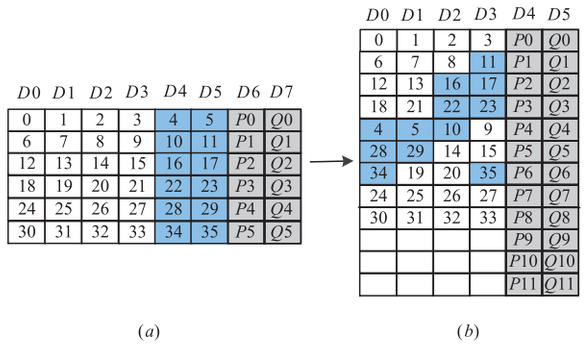

图18　利用ADR对RDP编码的RAID系统进行缩容，移除两个磁盘

4.2　基于对象存储的扩容方案

随着对象存储的发展,文献[31,47,48]提出的随机 RAID 算法,成为数据布局研究的重点. 文献[48]提出一种随机数据布局策略,主要依据磁盘的容量从两方面进行分析:(1)统一磁盘;(2)非统一磁盘. 当为统一磁盘扩容时,从原始 RAID 系统“剪切”每个磁盘$\left[\frac{1}{n+1},\frac{1}{n}\right]$范围中的数据“复制”到新磁盘$\left[0,\frac{1}{n+1}\right]$的范围(将每个磁盘的范围看成[0,1]),使得阵列能够达到均匀数据分布. 当为非统一磁盘时,引入级别概念,在每个级别中尝试在磁盘之间均匀分布所有数据块,将因磁盘饱和无法存储的数据块存放在下一级别. 当缩容时,即采用相反的策略,将数据还原到其他磁盘. 该方案实现双向扩容的同时,对同构与异构阵列问题进行了分析. 文献[49]对磁盘布局(DRP)问题,针对磁盘卡槽受限制和不受限制两种情况,提出了磁盘增加和删除序列的算法. 文献主要从空间开销和时间开销两方面进行了分析,提出的算法使得扩容过程中数据迁移开销达到最小. 文献[48]和文献[49]中提出的方案是基于高级哈希函数假设提出的,目的都是为了减少数据迁移的开销,但两种方案都没有能够提出一种有效的哈希函数.

RUSH[50,51]和 CRUSH[52]是两种基于对象的分布式存储系统的数据在线放置和重组策略,两者都支持双向扩容. RUSH 在分布式存储系统中采用 Hash 的方式对数据进行管理,根据每个磁盘不同的权重,解决了异构存储系统中数据布局问题,为分布式系统提供了数据副本和纠删码两种数据保护机制. CRUSH 具有两个优点:(1)任何组件都可以独立计算出每个 Object 所在的位置;(2)需要很少的元数据,只有当增加或删除设备时,元数据才发生改变. CRUSH 实现了异构存储系统中的数据管理,采用映射函数的数据管理机制,减少了对中心服务器的访问开销,同样提供了副本和纠删码两种数据保护机制. RUSH 和 CRUSH 在均匀数据分布和最小数据迁移方面都实现了概率性优化.

文献[53]提出一种克服随机数据布局不足的切片策略. 该策略保留一个小表,其中包含有关事前插入和删除操作的信息,在提供均匀数据分布的同时显著降低了所需的随机性,减少了所需的主存容量.

随着存储集群化的深入,面向集群环境的扩容方案近年来得到广泛研究. Scale-RS[54]是基于 Reed-Solomon(RS)编码面向集群环境的一种扩容方案,采用分散式节点实现扩容. 首先,Scale-RS 通过在新旧 chunk 之间平均放置 data block,实现均匀数据分布. 其次,通过从旧 chunk 向新 chunk 迁移必要的 data block 不仅满足数据迁移流量的下限,而且还通过从存储单个数据块中生成校验块来减少校验更新的流量. 同时,Scale-RS 从读取并行性和写入吞吐量方面提高了存储集群的 I/O 性能. 文献[55]对 Cauchy Reed-Solomon(CRS)编码的集群环境提出一种 Post-Scaling 编码矩阵(也叫校验矩阵)和一种数据迁移策略优化模型. Post-Scaling 编码矩阵优化了 I/O 开销和网络通信开销,提出的搜索算法实现了高效且低计算复杂度的数据迁移,进一步减少了 CRS 扩容的 I/O 开销. 扩容过程采用集中式的节点调度,实现了最小数据迁移和均匀数据分布,降低了扩容开销. NCScale[56]是一种基于网络编码 Vandermonde-based Reed-Solomon code 集群存储系统的扩容方案,同样采用分散式节点实现扩容. 该方案充分利用各存储节点的计算资源获得了近似最小的扩容带宽,同时确保了容错、均衡纠删编码数据布局和分散化的扩容属性,与 Scale-RS 相比减少了 50% 的扩容时间.

5　基于文件系统的扩容方案

随着大数据时代的发展,基于文件系统的存储依然是一种非常便捷的数据管理方式.

基于 ZFS 文件系统的扩容过程中,为解决 RAID5 存在的“write hole”问题,提出一种更高层的解决方案 RAID-Z[57]. RAID-Z 采用动态的存储条带宽度,使得每个 RAID-Z 写操作都是一个全存储条带写操作. RAID-Z 这种更高层次的方案,提供了更好的数据一致性,动态条带宽度使得扩容过程更高效.

HDFS 是基于 Hadoop 的分布式文件系统,为解决 3 副本策略存储空间利用率的问题,提出了 HDFS RAID[58]. HDFS RAID 以文件为单位计算校验,将文件分成多个条带,条带越小,计算出的校验数据越小,存储空间成本越低,数据恢复成本越高,条带越大,则刚好相反. HDFS 采用 XOR 和 RS(Reed-Solomon)两种编码方式,将同一条带的块分散放置在不同的数据节点上,在保证数据一致性的前提下,有效节省了冗余数据

所占空间的大小.

6 扩容方案的分析与讨论

表1对现有典型的扩容方案主要从数据布局,数据迁移量,扩容开销,数据寻址和双向扩容五个方面进行了对比分析和总结.

RR是公认数据布局最优的方案,但却具有最大的数据迁移量和扩容开销. GA算法同样具有数据迁移量大的问题,而且在RAID5系统中,校验开销巨大. MD-Reshape在RR基础上进行了改进,但数据迁移量和扩容开销比较大的问题并没有得到解决,而且存在元数据更新频繁的问题. Semi-RR与RR相比数据迁移量减少,性能显著提升,但多次扩容后,并不能保证均匀数据分布. SLAS和ALV基于滑动窗口的思想,采用聚合I/O技术减少了迁移I/O开销,懒惰的元数据更新保证了数据存储的一致性,但数据迁移量和扩容开销依然很大. FastScale、McPod和PBPC三种扩容方案分别对RAID0、RAID4、RAID5进行扩容,与SLAS和ALV相比,数据迁移量显著减少,满足了最小数据迁移,实现了均匀数据分布,缩短了扩容时间. AutoRAID技术只需将新磁盘添加进RAID系统,系统马上利用新磁盘空间,不再需要将数据迁移到新磁盘,而且能够兼容不同容量的磁盘,解决了异构RAID系统存储问题. MDM、GSR、MiPiL、PBM、CRAID和ISM都适用于应用性最广的RAID5,这些扩容方案都具备最小数据迁移量、最小扩容开销和快速数据寻址的特性,除MDM,PBM外,其他方案都满足了均匀数据分布,其中,PBM能否满足数据均匀分布与原始RAID系统中磁盘数目m有关. 专利#6000010和#7111117提出的两种扩容方案,减少了数据迁移量和扩容开销,但在均匀数据分布方面,表现较差. SDM、RS6、HCS、Xscale、H-Scale、和MDS-Frame都对RAID6进行扩容,除HCS均匀数据分布较差,其余方案都表现出良好的性能. GSR和MDS-Frame两种方案对缩容问题进行了分析,ADR是专为缩容问题设计的方案. 这三种方案都为缩容方案的研究提供了新的思路.

文献[48]、文献[49]提出的策略都实现了双向扩容,减少了迁移开销. RUSH和CRUSH都支持在线扩容,对异构分布式存储系统扩容进行了分析,减少了数据迁移,降低了校验开销,满足了均匀数据分布的要求. 文献[53]提出的方案实现了双向扩容和均匀数据分布.

Scale-RS和Post-Scaling虽然实现了集群环境下的扩容,然而现有RAID扩容方案和Scale-RS在集群环境下并未实现最小扩容带宽的目标. NCScale获得了近似的最小扩容带宽,缩短了扩容时间. RAID-Z和HDFS RAID保证了数据存储的一致性和可靠性,降低了扩容开销.

表1 典型扩容方案对比(1、均匀数据分布;2、最小数据迁移;3、最小扩容开销;4、快速数据寻址;5 双向扩容)

扩容方案	1	2	3	4	5	是否适用于RAID6
RR	√	×	×	√	√	有条件的适用
GA	√	×	×	√	√	有条件的适用
MD-Reshape	√	×	×	√	×	有条件的适用
专利#6000010	×	√	√	√	×	×
MDM	×	√	√	√	×	×
专利#7111117	×	√	√	√	×	×
SLAS	√	×	×	√	×	×
ALV	√	×	×	√	×	×
Semi-RR	×	×	×	√	×	有条件的适用
GSR	√	√	√	√	√	×
CRAID	√	√	√	√	×	×
FastScale	√	√	√	√	×	×
McPod	√	√	√	√	×	×
PBPC	√	√	√	√	×	×
MiPiL	√	√	√	√	×	×
PBM	×	√	√	√	×	×
ISM	√	√	√	√	×	×
SDM	√	√	√	√	×	√
RS6	√	√	√	√	×	√
HCS	×	√	√	√	×	√
Xscale	√	√	√	√	×	√
H-scale	√	√	√	√	×	√
MDS-Frame	√	√	√	√	√	√
ADR	√	√	√	√	√	√
Scale-RS	√	√	×	√	√	√
Post-Scaling	√	√	√	×	×	√
NCScale	√	√	√	×	√	√

7 总结与展望

随着数据量的不断增加,现有存储系统容量远远跟不上数据增长的速度,业界普遍采用扩容的方法实现存储系统容量的增加. 本文结合不同的应用场景介绍了目前典型的扩容方案,并从评价扩容方案的各项性能指标的角度进行了分析和对比,为以后扩容方案可能的研究方向提供了理论基础.

经过科研人员的努力,对RAID0、RAID4、RAID5扩容方案的研究已经有了重大突破,特别是对适用性最广的RAID5,提出了多种有效的扩容方案. RAID6由于

纠删码布局的特殊性,使得数据存储可靠性、一致性得到保证,但同时导致扩容方案设计的复杂性,这就使得对 RAID6 扩容方案的研究将会成为一个热点问题. RAID 存储系统扩容未来可能的研究方向展望如下.

(1)大数据环境下,由于当下大多数行业,要求存储中心保证每周 7 * 24 运转,这使得扩容过程必须保证对前台用户访问造成的影响最小. 因此如何确保设备不宕机情况下,保证用户访问质量,缩短扩容时间,其读写优化技术有待进一步研究.

(2)针对 RAID 系统存在磁盘损坏或者磁盘性能下降,以及节能降耗的问题,如何设计一种扩容方案既能实现扩容增加存储能力,又能实现缩容将性能较差的磁盘删除,降低存储能耗,也会成为今后研究的一个重点.

(3)随着云存储及面向对象存储的广泛应用,分布式存储系统扩容研究受到人们的极大关注,其异构存储扩容取决于高质量的随机函数,因此异构存储环境下如何提出高质量的随机函数以及数据分布策略是未来一个重要研究方向.

(4)现有并行 RAID 环境实现了最小 I/O 开销,加速了整个扩容过程. 而在集群环境下的 RAID 扩容,需要考虑如何充分利用节点的计算资源,提升扩容性能. 同时扩容过程中不仅要考虑 I/O 开销最小而且要实现最优网络通信开销. 因此,如何在集群环境下优化扩容过程及减少网络通信开销将成为未来集群化存储研究的关注点.

(5)固态磁盘阵列具有高 I/O 性能,充分利用固态盘高 I/O 能力优化调度扩容过程中数据读写流程;同时兼顾新型介质的忍耐性及生命周期,结合大数据内存存储应用场景,面向集群式内存的容错机制及可扩展性也将是未来一个重要的研究方向.

随着大数据时代的到来,RAID 系统本身良好的灵活性和可扩展性对于海量数据的存储,具有天然的优势. 因此,对于 RAID 扩容方案的研究,仍然会是未来研究的一个重大课题.

参考文献

[1] 孟小峰,慈祥. 大数据管理:概念、技术与挑战[J]. 计算机研究与发展,2013,50(1):146 - 169.
Meng Xiaofeng,Ci Xiang. Big data management:concepts, techniques and challenges[J]. Journal of Computer Research and Development,2013,50(1):146 - 169. (in Chinese)

[2] 曹媛. 到 2021 年全球智能手机流量预计将翻番[J]. 计算机与网络,2017,(12):16 - 16.

[3] Sankar S,Shaw M,Vaid K,et al. Datacenter scale evaluation of the impact of temperature on hard disk drive failures [J]. Acm Transactions on Storage,2013,9(2):1 - 24.

[4] Corbett P,English B,Goel A,et al. Row-diagonal parity fordouble disk failure correction[A]. Usenix Conference on File & Storage Technologies[C]. California:Publishing USENIX Association,2004. 1 - 14.

[5] Ofpaper P. AONT-RS:Blending security and performance in dispersed storage systems[A]. Usenix Conference on File & Storage Technologies[C]. California: Publishing USENIX Association,2011. 191 - 202.

[6] Huang C,Simitci H,Xu Y,et al. Erasure coding in windows azure storage[A]. Usenix Annual Technical Conference[C]. Massachusetts Publishing USENIX Association, 2012. 15 - 26.

[7] Rhea S,Wells C,Eaton P,et al. Maintenance-free global data storage[J]. IEEE Internet Computing,2001,5(5):40 - 49.

[8] 李静. 第 41 次《中国互联网络发展状况统计报告》发布[J]. 中国广播,2018,(03):96 - 96.

[9] Cisco. Mobile Visual Network Index (VNI)[J/OL]. https://www.cisco.com/c/zh_cn/solutions/service-provider/visual-networking-index-vni.

[10] Patterson D A. A case for redundant arrays of inexpensive disks (RAID)[A]. Proc ACM SIGMOD Conference [C]. Illinois:Publishing ACM,1988. 109 - 116.

[11] Chen P M,Lee E K,Gibson G A,et al. RAID:high-performance,reliable secondary storage[J]. ACM Computing Surveys,1994,26(2):145 - 185.

[12] Yu X,Gum B,Chen Y,et al. Trading capacity for performance in a disk array[A]. Conference on Symposium on Operating System Design & Implementation[C]. California:Publishing USENIX Association,2000. 243 - 258.

[13] S Kim D. On-line Reorganization of data in scalable continuous media servers[A]. International Conference on Database and Expert Systems Applications[C]. Zurich: Publishing IEEE Computer Society,1996. 751 - 768.

[14] 魏学才,宫庆媛,沈佳杰,周扬帆,王新. 适应冷热数据存储的多编码架构的设计与实证[J]. 计算机应用与软件,2017,34(2):35 - 41.

[15] 田磊,冯丹,岳银亮,等. 磁盘存储系统节能技术研究综述[J]. 计算机科学,2010,37(9):1 - 5.
TIAN Lei, FEND Dan, YUE Yin-liang, WU Su-zhen, MAO Bo. Survey on power-saving technologies for disk-based storage systems[J]. Computer Science, 2010, 37 (9):1 - 5. (in Chinese)

[16] 杨丽鸿. 浅析计算机磁盘存储系统节能技术[J]. 科学技术创新,2018,(03):88 - 89.

[17] 景蛲娴,李易. 基于海量数据的分布式存储与共享方案[J]. 科技创新导报,2018,15(24):125 - 127.

[18] 王意洁,许方亮,裴晓强. 分布式存储中的纠删码容错技术研究[J]. 计算机学报,2017,(01):236 – 255.

[19] 赵瑞峰,汤晓安,干哲. 基于集群技术的海量数据存储技术研究[J]. 微计算机信息,2010,26(16):196 – 198.

[20] 张峰豪. 纠删码集群存储的数据访问优化技术研究[D]. 湖北:华中科技大学,2013.

[21] 黄建忠,梁先海,曹强,等. 面向纠删码存储集群的弹性I/O调度机制研究[J]. 计算机研究与发展,2014,(S1):195 – 203.
Huang Jianzhong, Liang Xianhai, Cao Qiang, et al. Research on elastic I/O scheduling for erasure-coded storage clusters[J]. Journal of Computer Research and Development,2014,(S1):195 – 203. (in Chinese)

[22] 罗象宏,舒继武. 存储系统中的纠删码研究综述[J]. 计算机研究与发展,2012,49(1):1 – 11.
Luo Xianghong, Shu Jiwu. Summary of research for erasure code in storage system[J]. Journal of Computer Research and Development, 2012, 49(1):1 – 11. (in Chinese)

[23] Gonzalez J L, Cortes T. Increasing the capacity of RAID5 by online gradual assimilation[A]. International Workshop on Storage Network Architecture and Parallel I/Os[C]. Antibes: Publishing ACM, 2004. 17 – 24.

[24] Brown N. OnlineRAID-5 resizing. drivers/md/RAID5. c in the source code of Linux Kernel 2. 6. 32. 9[J/OL]. http://www. kernel. org/. Feb 2010.

[25] Method of increasing the storage capacity of a level fiveRAID disk array by adding, in a single step, a new parity block and N-1 new data blocks which respectively reside in a new columns, where N is at least two document type and number[P]. United States Patent: 6000010, 1999-12-07.

[26] Hetzler S R. data storage array scaling method and system with minimal data movement[P]. United States Patent: 8239622 B2, 2012-08-07.

[27] Franklin C R, Wong J T. Expansion of RAID subsystems using spare with immediate access to new space[P]. United States Patent: 7111117B2, 2006-09-19.

[28] Zhang G, Shu J, Xue W, et al. SLAS: An efficient approach to scaling round-robin striped volumes[J]. ACM Transactions on Storage, 2007, 3(1):1 – 39.

[29] Zhang G, Zheng W, Shu J. ALV: A new data redistribution approach toRAID-5 scaling[J]. IEEE Transactions on Computers, 2010, 59(3):345 – 357.

[30] Wilkes J, Golding R, Staelin C, et al. The HP AutoRAID hierarchical storage system[A]. Fifteenth ACM Symposium on Operating Systems Principles[C]. Colorado: Publishing ACM, 1995. 96 – 108.

[31] Goel A, Shahabi C, Yao S Y D, et al. SCADDAR: An efficient randomized technique to reorganize continuous media blocks[A]. International Conference on Data Engineering[C]. California: Publishing IEEE Computer Society, 2002. 473 – 482.

[32] Wu C, He X. GSR: A global stripe-based redistribution approach to accelerate RAID-5 scaling[A]. International Conference on Parallel Processing[C]. Pennsylvania: Publishing IEEE Computer Society, 2012. 460 – 469.

[33] Miranda A, Cortes T. CRAID: online RAID upgrades using dynamic hot data reorganization[A]. Usenix Conference on File and Storage Technologies[C]. California: Publishing USENIX Association, 2014. 133 – 146.

[34] Zheng W, Zhang G. FastScale: accelerateRAID scaling by minimizing data migration[A]. Usenix Conference on File and Storage Technologies[C]. California: Publishing USENIX Association, 2011. 149 – 161.

[35] Zhang G, Wang J, Li K, et al. Redistribute data to regain load balance during RAID-4 scaling[J]. IEEE Transactions on Parallel & Distributed Systems, 2015, 26(1):219 – 229.

[36] Daci G, Ndreu M. Increasing RAID-5 performance improving the scaling approaches[A]. Computer and Information Technology[C]. California: Publishing IEEE, 2013.

[37] Zhang G, Zheng W, Li K. Rethinking RAID-5 data layout for better scalability[J]. IEEE Transactions on Computers, 2014, 63(11):2816 – 2828.

[38] Mao Y, Wan J, Zhu Y, et al. A new parity-based migration method to expand RAID-5[J]. IEEE Transactions on Parallel & Distributed Systems, 2014, 25(8): 1945 – 1954.

[39] Liang J, Xu Y, Li Y, et al. ISM-An intra-stripe data migration approach for RAID-5 scaling[A]. International Conference on Networking, Architecture, and Storage[C]. Shenzhen: Publishing IEEE Computer Society, 2017.

[40] Wu C, He X, Han J, et al. SDM: A stripe-based data migration scheme to improve the scalability of RAID-6[A]. IEEE International Conference on CLUSTER Computing[C]. Beijing: Publishing IEEE Computer Society, 2012. 284 – 292.

[41] Zhang G, Li K, Wang J, et al. Accelerate RDP RAID-6 scaling by reducing disk I/Os and XOR operations[J]. IEEE Transactions on Computers, 2014, 64(1):32 – 44.

[42] Xia S, Mao Y, Tan M, et al. HCS: Expanding H-Code rAID 6 without recalculating parity blocks in big data circumstance[A]. International Conference of Young Computer Scientists, Engineers and Educators[C]. Harbin: Publishing Communications in Computer and Information Science, 2015. 65 – 72.

[43] Zhang G, Wu G, Lu Y, et al. Xscale: Online X-Code

RAID-6 scaling using lightweight data reorganization[J]. IEEE Transactions on Parallel & Distributed Systems, 2016,27(12):3687 – 3700.

[44] Wan J, Xu P, He X, et al. H-Scale: A fast approach to scale disk arrays via hybrid stripe deployment[J]. ACM Transactions on Storage,2016,12(3):1 – 30.

[45] Wu C, He X. A flexible framework to enhance RAID-6 scalability via exploiting the similarities among MDS codes[A]. International Conference on Parallel Processing [C]. Lyon:Publishing IEEE Computer Society,2013. 542 – 551.

[46] Du C, Wu C, Li J. An advanced data redistribution approach to accelerate the scale-down process of RAID-6 [A]. International Conference on Algorithms and Architectures for Parallel Processing [C]. Dalian: Publishing Lecture Notes in Computer Science,2014. 286 – 299.

[47] Santos J R, Muntz R R, Ribeiro-Neto B. Comparing random data allocation and data striping in multimedia servers[J]. ACM Sigmetrics Performance Evaluation Review,2000,28(1):44 – 55.

[48] Abstract E, Salzwedel K, Scheideler C, et al. Efficient, distributed data placement strategies for storage area networks [A]. ACM Symposium on Parallel Algorithms & Architectures[C]. Maine:Publishing ACM,2000. 119 – 128.

[49] Seo, Beomjoo, Zimmermann, et al. Efficient disk replacement and data migration algorithms for large disk subsystems[J]. ACM Transactions on Storage,2005,1(3):316 – 345.

[50] Honicky R J, Miller E L. A fast algorithm for online placement and reorganization of replicated data[A]. International Parallel and Distributed Processing Symposium[C]. Nice:Publishing IEEE Computer Society,2003. 57 – 68.

[51] Honicky R J, Miller E L. Replication under scalable hashing:a family of algorithms for scalable decentralized data distribution[A]. International Parallel and Distributed Processing Symposium [C]. New Mexico: Publishing IEEE Computer Society International,2004. 1357 – 1366.

[52] Weil S A, Brandt S A, Miller E L, et al. CRUSH: Controlled, scalable, decentralized placement of replicated data [A]. Proceedings of the ACM/IEEE SC2006 Conterence on High Performance Networking and Computing[C]. Florida:Publishing ACM,2006.

[53] Miranda A, Effert S, Kang Y, et al. Reliable and randomized data distribution strategies for large scale storage systems[A]. International Conference on High PERFORMANCE Computing[C]. Bengaluru:Publishing IEEE Computer Society,2011.

[54] Huang J, Liang X, Xiao Q, et al. Scale-RS: An efficient scaling scheme for RS-coded storage clusters[J]. IEEE Transactions on Parallel & Distributed Systems, 2015, 26 (6):1704 – 1717.

[55] Wu S, Xu Y, Li Y, et al. I/O-efficient scaling schemes for distributed storage systems with CRS codes[J]. IEEE Transactions on Parallel and Distributed Systems,2016,27 (9):2639 – 2652.

[56] Xiaoyang Zhang, Yuchong Hu, Patrick P. C. Lee, Pan Zhou. Toward optimal storage scaling via network coding:from theory to practice[A]. Proceedings of the IEEE International Conference on Computer Communications 2018[C]. Hawaii, USA: Publishing IEEE, 2018. 1808 – 1816.

[57] Bonwick J. RAID-Z[J/OL]. http://blogs. sun. com/bonwick/entry/RAIDz. Nov. 2005.

[58] Facebook. HDFS RAID[J/OL]. http://wiki. apache. org/Hadoop/hdfs-RAID. Nov. 2011.

作者简介

元　铸　男,1993 年出生,硕士研究生. 主要研究方向是海量存储系统、新型存储技术.
E-mail:761443043@qq.com

谢　平(通信作者)　男,1979 年出生,工学博士,副教授. 主要研究方向是并行与分布式文件系统、网络存储、容错存储和新型存储技术等.
E-mail:xieping@qhnu.edu.cn

耿生玲　女,1970 年出生,博士,教授,CCF 会员(E20033713M). 主要研究方向为计算理论、数据挖掘、控制与决策的研究.
E-mail:geng_sl@126.com