

基于多属性度量的数据分级访问模型研究

施光源¹ 张宇^{1,2}

(浪潮集团有限公司 济南 250101)¹ (高效能服务器和存储技术国家重点实验室 济南 250101)²

摘要 随着云计算的迅速发展,云存储已成为企业关键信息服务的重要途径。但是,受限于存储资源性能以及大数据等影响,用户往往需要忍受较长的访问延时。为了缓解这种情况,人们提出了智能数据管理技术,用于有效管理大量数据以及降低用户的访问延迟,提高云计算的服务质量。提出一种基于多属性分析的存储端数据分级访问模型。模型通过对被管理数据对象的动态、静态属性进行统计分析来提取属性中的关键信息,并依此进行数据管理决策,将冷/热数据迁移至对应层级,以便能够在合理规划存储资源的同时提高存储系统的访问性能。性能测试实验的结果表明,该模型具有较好的整体性能。

关键词 智能数据管理,存储访问模型,分级存储,数据迁移

中图分类号 TP391 文献标识码 A

Hierarchical Storage Access Model Based on Multi-Attributes Measurement

SHI Guang-yuan¹ ZHANG Yu^{1,2}

(Inspur, Jinan 250101, China)¹

(State Key Laboratory of High-end Server & Storage Technology, Jinan 250101, China)²

Abstract With the rapid development of cloud computing, cloud storage has become an important way of business-critical information services. However, limited by storage resource performance and impact data, users often need to endure a long delay of access. In order to alleviate the situation, intelligent data management technologies have been proposed, to effectively manage large amounts of data, as well as reduce access latency, improving the quality of cloud computing services. Proposing an Hierarchical Storage Access Model based on Multi-Attributes Measurement. Model managed through the statistical analysis of the static and dynamic properties of the data object, extracting key information from the properties, data management and in accordance with this decision, cold/hot data migration to the corresponding levels so that you can reasonable while planning your storage resources to improve storage system performance. Performance test results of the experiment showed that the model has good overall performance.

Keywords Intelligent data magement, Storage access model, Hierarchical storage, Data migration

1 介绍

云计算服务的迅速发展使得数据呈现爆炸式地增长,因此,海量数据的高效管理已经成为业界研究的热点领域^[1]。智能动态数据管理技术^[2]可根据数据访问特征在存储虚拟层对存储设备组成的存储资源进行合理组织,形成多级的存储层次,然后根据智能策略将不经常访问的数据自动迁移到存储成本层次中较低的设备,释放出较高成本的存储空间给更频繁访问或更高优先级的数据,从而大大减少非重要性数据在一级本地磁盘所占用的空间,加快整个系统的存储性能,降低整个存储系统的拥有成本,进而获得更好的性价比。

2 相关工作

现有工作中,对于数据特征模型的研究比较多^[3-5],但是对于数据访问模型的研究还不深入,因此,智能数据管理中的

数据访问模型将是本文主要的研究内容。在智能数据管理系统中,分级存储访问模型与所管理的数据对象相关。Shah 等人在文献[6]中根据数据的业务价值和访问的需求进行数据分类,并同时根据价格性能和可靠性等对存储设备分类,然后依据指定的策略实现数据类型到存储设备类型的映射。文献[7]主要是从能耗的角度来考虑数据管理问题。该文从响应时间的改变来统计能耗的损失,建立了能耗损失模型,该损耗模型分别针对响应时间给出了学习阶段、迁移阶段以及优化阶段对响应时间的统计,从而确保在数据迁移管理中将能耗的损失降低到最低水平。该文从能耗角度建立模型对智能数据管理系统的性能优化有所帮助,但是模型当中缺乏对数据管理操作的具体描述,这对于整个智能系统的管理行为定义不清晰。文献[8]中根据由性能和可靠性不同的两级磁盘阵列组成的二级海量存储系统所建立的数据存储访问模型,在对数据价值进行判定的基础上,实现数据在两级设备间的迁

本文受国家“九七三”重点基础研究计划基金项目(2012CB724101)资助。

施光源(1983—),男,博士,主要研究方向为海量存储、可信计算, E-mail: 118sgy@163.com; 张宇(1983—),男,硕士,主要研究方向为并行计算。

移和用户的透明访问。但是该文中所提出的访问模型太过抽象,并未对智能数据管理进行更进一步的细化,不具有实际的可操作性。文献[9]提出基于扩展块的动态分级方案 EDT,并且通过配置顾问与动态分级管理器两个关键组件来解决性能与损耗如何平衡的问题。但文中只是对损耗的计算模型给出了形式化的描述,并未对智能数据管理重要的组成部分之一的访问模型进行定义。

现有工作对于智能数据管理模型之中的数据特征模型进行的研究相对较多,数据特征模型可以借鉴数据挖掘与统计学等方法,因此具有较好的理论基础。但是,数据访问特征模型的研究还不充分,主要表现为如何对模型中的关键变量进行定义,以及如何定义变量的关系。针对上述问题,本文基于数据的多属性特征建立访问存储模型。在细化模型构成变量的基础上,对变量关系进行形式化定义,然后利用该访问模型构建了智能数据管理原型系统,最后,对模型的可行性进行了实验性验证。

3 数据分级存储访问模型

3.1 模型变量定义

首先,给出数据分级访问模型(Hierarchical Storage Access Model, HSAM)中的变量定义,并对变量含义进行描述。

定义 1 数据分级存储访问模型 HSAM 是一个七元组,形式化表示为:

$$HSAM = \{S, P, OP, DC, SD, MP, F\},$$

其中,模型变量定义如下:

1) $S = \{s_1, s_2, \dots, s_n\}$ 表示智能数据管理过程中所包括的主体集合。该集合中的元素 s_i 表示智能数据管理系统中需要被管理的对象,根据管理粒度不同可以是各种类型文件或者底层的扩展数据块;

2) $P = \{p_1, p_2, \dots, p_n\}$ 为 S 中主体 s_i 的属性集合,该集合中的元素 $p_i = \{sp_{i1}, sp_{i2}, \dots, sp_{in}\}$ 表示与主体相关的属性集合。其中, sp_{in} 表示主体 s_i 所具备的某类属性。这些属性包括主体类型、大小以及主体的访问频率等^[10];

3) $OP = \{mig, alloc, copy, del, get\}$ 表示对集合 S 中的主体进行的智能管理操作。其中, mig 表示数据迁移管理操作, $alloc$ 表示数据分配管理操作, $copy$ 表示数据拷贝管理操作, del 表示数据删除管理操作, get 表示获取 s_i 属性的操作;

4) $DC = \{l_dev, m_dev, h_dev\}$ 为存储设备性能等级集合。其中,按照存储设备性能的高低分为 $l_dev < m_dev < h_dev$, l_dev 级设备主要处于离线状态,常用于数据备份; m_dev 级设备可以处于近线状态,用于存储访问频率不高或者用于临时高速缓冲设备; h_dev 级设备处于高速状态, h_dev 级设备存储速度快,主要存储热点数据,但其容量大小受到一定限制;

5) $SD = \{static, dynamic\}$ 为 p_i 的属性类型集合, $static(p_i)$ 表示 p_i 的静态属性,包括规格、类型以及所属 UUID 等; $dynamic(p_i)$ 表示 p_i 的动态属性,包括平均访问率、平均修改率等。

6) $MP = \{mp_1, mp_2, \dots, mp_n\}$ 表示智能数据管理策略的集合。其中, $mp_k \in MP, 1 \leq k \leq n$ 表示策略集合 MP 中针对特定数据管理操作 OP 所定义的策略元素。

7) $F = f_p \times f_o$ 为一个函数集合。其中, $f_p(S) = \{f_p | f_p:$

$P \xrightarrow{static(P) \vee dynamic(P)} MP\}$, f_p 表示对于主体 S 的属性 P 进行度量的函数,通过对主体的静态属性 $Static(P)$ 或者动态属性 $Dynamic(P)$ 进行度量得到主体的管理策略 MP ; $f_o(S) = \{f_o | f_o: S \xrightarrow{MP \wedge DC} OP\}$, f_o 表示对于主体的管理操作决策函数,根据管理策略 MP 以及设备等级 DC 来做出对于主体 S 的管理操作决策 OP 。 F 的元素为 $f = (f_p, f_o)$, 其中度量函数与决策函数因主体属性类型的不同而各异,例如如果需要对于主体的访问热度进行度量,那么 f_p 将对主体的访问频率进行度量, f_o 将根据度量结果得到 mig 决策,将数据从 m_dev 级设备迁移到 h_dev 级设备。

在 HSAM 模型中,假设对于主体 S 的属性 P 可以完全获得,并且存储设备的 DC 仅考虑设备的存储性能,并且假设 h_dev 级设备存储容量有限,并且存在存储空间不足的情况。

3.2 HSAM 操作 OP

定义 2 智能数据管理分级存储访问模型中对于主体的操作 OP 的定义如下:

1) 分配操作 $alloc$: 当对主体 s 执行分配操作 $alloc$ 时,将主体 s 添加到智能数据管理主体集合 S 中,并且将主体 s 所具备的属性 p 加入到属性集合 P 中,为主体 s 指定默认的设备 dc , 其中 $dc \in DC$ 。 $alloc$ 操作形式化表示为:

$$alloc(s, p, dc) = (S', P', default), \forall dc \in DC$$

$$S' = S \cup \{s\}, P' = P \cup \{p\}, dc = default$$

2) 获取操作 get : 当对主体进行获取操作 get 时,如果主体 s 未在集合 S 中,则需要执行 $alloc$ 操作,并且对主体 s 的主体集合以及属性集合进行更新;否则需要对属性集合进行更新。 get 操作形式化表示如下:

$$get(s, p, dc) = (S', P', dc)$$

$$\forall dc \in DC$$

$$S' = \begin{cases} S \cup \{s\}, & \text{if } s \notin S \\ S, & \text{otherwise} \end{cases}$$

$$P' = P \cup \{p\}$$

3) 拷贝操作 $copy$: 当对主体 s 执行 $copy$ 操作时,如果主体 s 在集合 S 中,并且主体的属性 p 在集合 P 中,那么将得到新的主体 s' 以及 p' , 并将其加入到集合 S 与 P 中,为 s' 指定存放的层级 $dc = specific$ 。 $copy$ 操作形式化表示为:

$$copy(s, p, dc) = (S', P', specific),$$

$$\forall s \in S, \forall p \in P, \forall dc \in DC$$

$$S' = S \cup \{s'\}, P' = P \cup \{p'\}, dc = specific$$

4) 删除操作 del : 当主体 s 结束某项任务后将导致主体从主体集合 S 中被移除,但如果经过检验函数 E 验证后,主体 s 还被其他进程所占有,那么主体 s 将不能被移除。其中 $E(s)$ 表示对主体 s 的占有者进行判断。同样,对于主体 s 的属性 p , 如果 s 被移除,那么属性 p 也将从属性集合 P 中被移除,主体所在的层级变为 $none$ 或者不发生改变。 del 操作的形式化表示如下:

$$del(s, p, dc) = (S', P', none | dc)$$

$$S' = S \ominus \{s\} = \begin{cases} S, & |E(\{s\})| \geq 2 \\ S - \{s\}, & \text{otherwise} \end{cases}$$

$$P' = P \ominus \{p\} = \begin{cases} P, & |E(\{s\})| \geq 2 \\ P - \{p\}, & \text{otherwise} \end{cases}$$

5) 迁移操作 mig : 迁移操作 mig 是指当主体 s 的属性 p 满足函数 f 。以及 MP 的要求时,将主体 s 放置到不同的层级

dc。由于迁移操作对于上层是透明的,因此,主体集合 S 和属性集合 P 未发生改变。

$$mig(s, p, dc) = (S, P, DC), \forall s \in S, \forall p \in P, \forall dc \in DC$$

3.3 HSAM 模型触发条件

在 HSAM 中,对于数据对象的管理是通过对其所具备的多种属性进行度量后,根据度量结果制定数据对象的管理策略,再次,依据所得到的管理策略以及存储设备的层级信息对数据对象进行 3.2 节中所定义的相关操作。HSAM 模型中变量关系描述如图 1 所示,其中不同形式的线条表示了不同的操作行为。

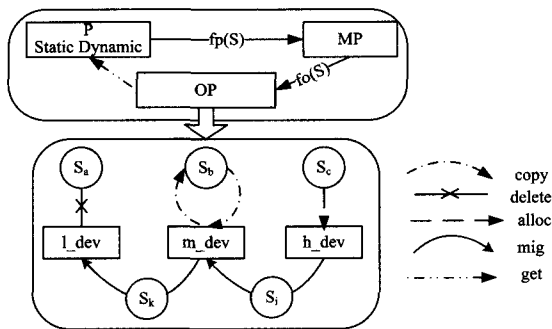


图 1 HSAM 模型

在 HSAM 模型中,当主体 $\forall s_i \in S$ 以及主体的属性 $\forall p_i \in P$, 并且执行 $get(s_i, p_i, dc)$ 操作后得到 $p_i = (sp_{i1}, sp_{i2}, \dots, sp_{in})$, 那么属性度量函数 $f_{p_i}(s_i) = f_{p_i}(sp_{i1}, sp_{i2}, \dots, sp_{in})$ 。根据 sp_{in} 的度量结果 $result(sp_{in})$, 以及前期统计或者由系统管理员所设定的各个属性特征的阈值 $threshold(sp_{in})$ 指定对于主体 s_i 的管理策略 mp_i , 并且将 mp_i 加入到策略集合 MP 中。

基于 s_i 不同的属性特征 sp_{in} , 当获得不同的度量结果 $result(sp_{in})$ 时, 将进行不同的操作 op 。例如, 假设 $result(sp_{in})$ 为平均访问热度, $threshold(sp_{in})$ 为平均访问热度的门限值:

1) 当 $result(sp_{in}) \leq threshold(sp_{in})$ 时, $f_o(sp_{in}) \xrightarrow{mp_i, \Delta dc} mig$, 如果 $dc(sp_{in}) = m_dev$, 使得 $dc(sp_{in}) = l_dev$; 如果 $dc(sp_{in}) = h_dev$, 那么 $dc(sp_{in}) = m_dev$ 或者 $dc(sp_{in}) = l_dev$; 如果 $result(sp_{in}) > threshold(sp_{in})$, 那么 $f_o(sp_{in}) \xrightarrow{mp_i, \Delta dc} mig$, 如果 $dc(sp_{in}) = m_dev$, 使得 $dc(sp_{in}) = h_dev$; 如果 $dc(sp_{in}) = l_dev$, 那么 $dc(sp_{in}) = m_dev$ 或者 $dc(sp_{in}) = h_dev$ 。

2) 当 $result(sp_{in})$ 为平均修改热度等属性特征时, 也可以使用上述触发条件。

另外, 根据 mp_i 中定义的属性特征的要求, 可以执行其他操作行为 OP 。

4 模型实现

4.1 模型实例化

将存储访问模型 HSAM 应用到实际的智能数据管理系统中, 首先需要将模型中的变量进行实例化, 将层级 l_dev, m_dev, h_dev 分别对应到不同访问性能的存储设备, 比如 $h_dev = SSD, m_dev = SAS, l_dev = SATA$ 等, 将 $p_i = (sp_{i1}, sp_{i2}, \dots, sp_{in})$ 定义为可以度量的应用属性。另外, 需要将度量函数 f_p 与决策函数 f_o 进行实例化。在对存储访问模型

HSAM 的变量进行实例化以后, 为了能够实现 HSAM 中的管理操作 $OP = \{mig, alloc, copy, del, get\}$, 设计了智能数据管理的原型系统。

4.2 软件架构

1) 管理界面接口提供一个图形用户接口或者一个命令行接口来设置分配策略、管理逻辑卷和进行存储设备级别设定。管理界面接口接受用户输入, 并且生成 XML 文件, 通过分析该文件所获得的关键信息将被存储在内核驱动中。XML 提供了描述在指定位置所实施的文件、分级等信息的语言。还有一些 XML 策略文件与用户接口库进行交互。XML 策略文件包括管理数据的策略以及存储级别信息。这些 XML 生成的 xml 文件被传递给 XML 分析器, 然后分析这些信息传输给系统管理模块。

2) 策略分析器分析从系统管理模块接口库接收的 XML 文件, 然后从中提取信息。这些信息然后被传递给系统管理模块接口, 再通过系统管理模块接口库传递给数据管理器。

3) 数据管理器与用户和内核空间函数进行通信。它接受 XML 分析器生成的策略文件, 并且将这些信息传输给内核空间。数据管理器维护内部映射表, 这些映射表为每一级设备维护关于层级以及块范围的所有信息。

4) 存储资源管理器负责提供来自底层的设备信息给数据管理器。资源管理器收集节点和块分布信息给数据管理器使用, 并且维护一个线性表来存储所有的信息, 比如起始位置、长度等。

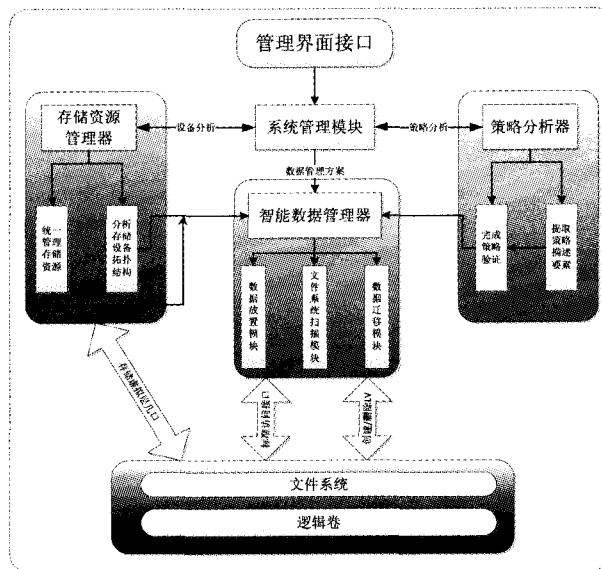


图 2 软件架构

4.3 主要管理流程设计

4.3.1 放置流程

智能数据管理中数据放置机制是对数据进行管理的基础功能。该机制能够根据文件的静态特征, 如文件大小、文件类型以及所属的组别等对文件进行初始化的优化管理, 将符合策略定义的数据放置到对应的存储设备上。具体执行流程如图 3 所示, 步骤如下:

1) 首先, 读取被管理数据的目录结构, 以便能够对整个文件系统中被管理的文件进行统计分析。

2) 获取内核中保存的启动关键信息, 从中读取放置配置

信息,例如数据放置的层级等。

3)根据获取的数据放置信息对遍历的文件重分配存储空间,并且将文件迁移到配置的设备上。

4)循环执行步骤1)~3),直到所有文件都按照放置策略规定的要求完成数据的管理。

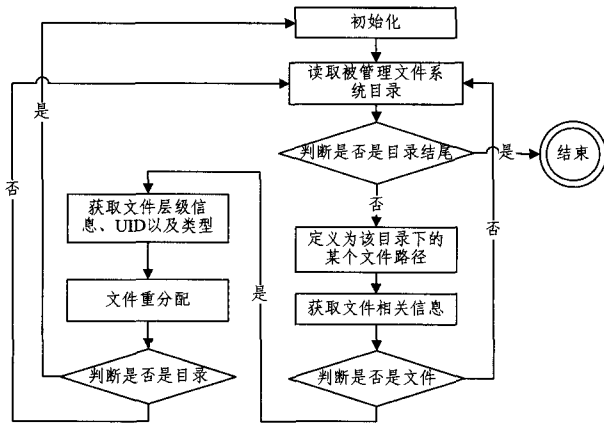


图3 数据放置管理执行流程图

4.3.2 迁移流程

智能数据管理文件迁移机制主要完成数据的再优化工作。在系统运行期间对文件动态属性进行监控,如果出现访问频率或者I/O热度过高或者过低的文件,将利用迁移机制将其放置到高速设备或者低速设备上,从而完成对于数据的优化管理。具体执行流程如图4所示,步骤如下:

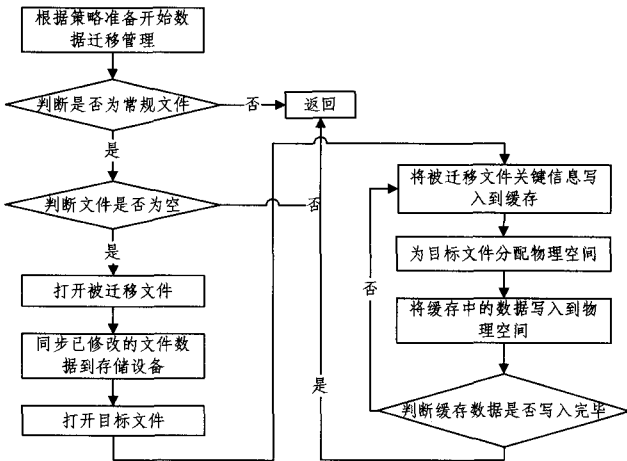


图4 迁移机制执行流程图

1)在迁移之前,首先检测是否为常规文件,并且文件是否为空文件。

2)为了保证磁盘上实际文件系统与缓存中内容的一致性,同步内存中所有已修改的文件数据到储存设备。由于存在延迟写(delayed write)降低了文件内容的更新速度,使得欲写到文件中的数据在一段时间内并没有写到磁盘上。当系统发生故障时,这种延迟可能造成文件更新内容的丢失。

3)打开一个新的目标文件,该文件用于存储被迁移的文件。

4)提取被迁移文件的信息,存储到fiemap结构所定义的缓存中。

5)利用fallocate来分配物理空间,空间大小需要根据被迁移文件的信息来决定,主要是从缓存中获取文件所占的块

大小。

6)将缓存中的数据放置到目标文件所指定的物理空间。

5 实验分析

5.1 实验目标

为了验证数据分级访问模型对存储系统整体性能的提升效果,构建了智能数据管理的原型系统进行测试,实验中主要包括的存储设备如表1所列。实验的目的主要是为了评估用户不同的吞吐量对数据管理操作的影响。

为了对性能有一个准确的评估,首先测量中等和高性能的存储设备在密集IO负载的情况下的一般访问请求。随后,针对这两个存储设备分析了数据管理操作中性能影响最为明显的卷迁移操作对前台应用程序所产生的影响。实验中,运行IOZone基准程序。IOZone是一个文件系统的benchmark工具,可以测试不同的操作系统中文件系统的读写性能。

表1 实验设备配置信息

实验设备	配置说明	层级
浪潮 DS3500	CPU: Dual Core Xeon 1.6G, RAM: 32G, SATA: 500G	m_dev
U盘	USB2.0, 容量 8G, 最大读取速度 10M, 最大写入速度 5M	L_dev

5.2 实验分析

测试设备包括一台浪潮的DS3500存储系统,其中包括500G的SATA盘和一个8G的U盘。其中,将SATA盘划分100G作为m_dev层级,U盘作为L_dev层级。然后利用磁盘操作命令lvcreate等将两层级设备映射为一个逻辑卷HSAM_LV。这样,对于上层应用来说HSAM_LV是一个连续的存储池,因此下层的智能数据管理操作对于上层应用来说是透明的。接下来,分别对L_dev设备上的数据与m_dev设备上的数据进行密集的I/O操作,分别记录数据的吞吐量。然后执行智能数据管理的迁移操作,在执行成功后,再次进行同样的密集I/O操作,然后记录数据的吞吐量。

对于单位时间内I/O数量的测试能够反映出系统的请求响应性能。在本实验中,如图5所示,首先执行对于文件系统中某类文件的读写请求,同时利用Iostat系统I/O监测指令以1s的时间间隔记录I/O数。首先是未执行分级操作时,开始读写请求后I/O数随之增加,当读写测试执行到10s左右,开始执行启动分级存储的分配操作,这时将数据迁移到m_dev层级设备上,由于分配操作要执行数据的迁移,使得文件被加锁,因此读写I/O数明显降低,然后再完成分配操作后,系统的I/O数重新回到起始水平。

利用IOZone对数据分级存储系统的读写性能进行对比实验。在实验中,对文件系统中常用的操作进行文件大小为1GB、记录大小为1MB的密集测试,测试结果如图6所示。测试选择读取操作Write、重读操作Re-write、写入操作Read与重写入操作Re-write。其中,Write测试向一个新文件写入的性能,Re-write测试向一个已存在的文件写入的性能,Read测试读一个已存在的文件的性能,Re-Read测试读一个最近读过的文件的性能。首先,在完成迁移操作后,读写性能都有不同程度的提升,这表明将热点数据迁移到快速设备后能够有效地提供存储系统的整体性能。另外,当一个新文件被写入时,不仅仅是那些文件中的数据需要被存储,还包括那些用

于定位数据存储存储在存储介质的具体位置的元数据,因此, Write 的性能通常会比 Re-write 的性能低。

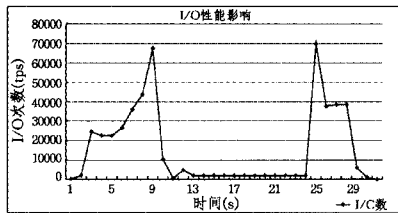


图5 I/O性能影响

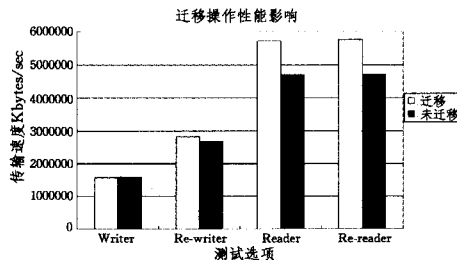


图6 迁移操作性能影响

结束语 本文对智能数据管理中分级存储数据访问模型展开研究,对分级存储的整体执行过程进行抽象,定义了分级存储访问模型 HSAM,并且给出了分级存储的形式化描述以及触发条件的定义。在此基础上,设计了 HSAM 模型的原型系统,给出了系统的软件架构以及迁移、分配等主要操作的执行算法。最后,对 HSAM 的原型系统进行实验,结果表明在启动分级存储功能后,其能够完成数据的迁移以及分配等操作,并且能有效提高系统的读写性能。

参考文献

[1] Massiglia P. Exploiting Multi-tier File Storage Effectively 2009 [R/OL]. <http://www.snia.org/education/tutorials/2009/>

spring/file/PaulMassiglia_Exploiting_Multi-Tier_File_StorageV05.pdf

[2] 赵晓南,李战怀,曾雷杰,等. 分级存储管理技术研究[J]. 计算机研究与发展,2001,48(Suppl.):105-111

[3] Tim L C, Simth C H M, Miller E L. An improved long-term file-usage prediction algorithm[OL]. 1999. <http://users.soe.ucsc.edu/~elm/Papers/cmg99.pdf>

[4] Mesnier M, Thereska E, Ganger G R, et al. File classification in self-* storage systems[C]//Proc of the 1st Int Conf on Autonomic Computing. Washington, D C; IEEE,2004,44-51

[5] Jin Hai, Xiong Muzhou, Wu Song. Information values evaluation model for ILM[C]//Proc of 9th ACIS Int Conf on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD2008). Washington, D C; IEEE, 2008;543-548

[6] Shah G, Voruganti K, Shivam P, et al. ACE:Classification for information lifecycle management[C]// Proc of NASA Mass Storage Systems and Technologies (MSST 2006). Washington, D C; IEEE, 2006

[7] Gong Zhang, Lawrence Chiu, Clem Dickey, et al. Automated Look ahead Data Migration SSD-enabled Multi-tiered S storage Systems[C]// Proc of 26th IEEE(MSS T2010) Symposium on Massive Storage Systems. Incline Village, Nevada,2010

[8] 聂雪军,秦磊华,周敬利,等. 内容感知存储系统中自动分级存储模型研究[J]. 小型微型计算机系统,2011,32(6)

[9] Guerra J, Pucha H, Glider J, et al. Cost Effective Storage using Extent Based Dynamic Tiering[C]// The Proceeding of USENIX Conference on File and Storage Technologiesp. 2011; 273-286

[10] 宋丽娜,戴华东,任怡. 基于海量数据存储系统多级存储介质的热点数据区分方法[J]. 计算机研究与发展,2012,49(Suppl.): 6-11

(上接第 149 页)

[2] 罗亮,冯象初,张选德,等. 基于非局部双边随机投影低秩逼近图像去噪算法[J]. 电子与信息学报,2013,35(1):99-105

[3] Perona P, Malik J. Scale-space and edge detection using anisotropic diffusion [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,1990,12 (7):629-639

[4] Monteil J, Beghdadi A. A new interpretation of the nonlinear anisotropic diffusion for image enhancement [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, 21 (9):940-946

[5] 白俊奇,陈钱. 基于各向异性扩散的红外图像噪声滤波算法[J]. 光学学报,2008,28(5):866-869

[6] Yu J H, Wang Y Y, Shen Y Z. Noise reduction and edge detection via kernel anisotropic diffusion [J]. Pattern Recognition Letters,2008,29:1496-1503

[7] 王毅,张良培,李平湘. 各向异性扩散平滑滤波的改进算法[J]. 中国图象图形学报,2006,11(2):210-216

[8] 林宙辰,石青云. 一个能去噪和保持真实感的各向异性扩散方程[J]. 计算机学报,1999,11:1133-1137

[9] 赵德,何传江,陈强. 结合局部熵的各向异性扩散模型[J]. 模式识别与人工智能,2012,25(4):642-647

[10] 王相海,李放. 同向平均梯度的各向异性扩散模型[J]. 中国图象图形学报,2011,16(12):2105-2111

[11] Scott T A. Locally monotonic diffusion [J]. IEEE Trans on Image Processing,2000,48(5):1379-1389

[12] 余瑞星,朱冰,吕梅柏. 一种新的水平集停止项函数选取方法研究[J]. 系统仿真学报,2008,20(22):6154-6157

[13] 易子麟,尹东,胡安洲,等. 基于非局部均值滤波的 SAR 图像去噪[J]. 电子与信息学报,2012,34(4):950-955

[14] 何艳敏,甘涛,陈武凡. 基于稀疏表示的两级图象去噪[J]. 电子与信息学报,2012,34(9):2268-2272

[15] Papari G, Petkov N. Edge and line oriented contour detection: State of the art[J]. Image and Vision Computing,2011,29(2): 79-103

[16] Papari G, Petkov N. Adaptive pseudo dilation for Gestalt edge grouping and contour detection [J]. IEEE Trans. Image Process,2008,17(10):1950-1962