

提高数据库系统性能的关键技术

陈莉君 (西安邮电学院计算机系 710061)

摘要:数据库应用系统性能的高低影响着最终用户对系统的满意程度。本文分析了影响数据库性能的主要因素,给出了内存缓冲管理这一关键技术的实现机制,提出了改善数据库系统的两种途径。

关键词:数据库系统 缓冲区 C/S LRU

1. 引言

随着数据库技术的不断发展,人们对数据库系统的要求越来越高,这体现在两个方面,一是数据库体系结构的先进性,二是实现技术的先进性,因此,各大数据库厂商如 IBM、SYBASE、ORACLE 等都推出其先进的数据库系统。但是,从用户的角度来看,追求先进性并不是最终目的,而达到良好的性能/价格比才是他们所需要的。目前,数据库应用系统的体系结构还是以 CLIENT/SERVER(以下简称 C/S)结构为主,因此,要达到良好的性能/价格比,应从两方面着手:即 C/S 具体结构的确定及客户端和服务端实现技术的选择。关于 C/S 结构,以单服务器情况下的功能划分,有三种方案,它们是:CS (Client/Server) 结构、RU (RAD - Unify) 结构、EWS (Extended Workstation Server) 结构。CS 结构中,客户没有查询能力,没有数据分布;RU 结构中,客户有查询能力,没有数据分布;EWS 结构中,客户有很强的查询能力,有数据分布。数据库系统的开发者可以根据用户的具体需求,选择合适的结构。关于客户端的实现技术,一般的开发者考虑较多,尤其是众多前端开发工具的流行,为客户端实现良好的人机界面提供了较好的环境。关于服务器端的实现技术,一般的开发者认为是数据库厂商应考虑的,因此考虑较少。实际上,影响数据库系统性能的主要因素在于数据库服务器端技术的充分应用,这些技术包括:内存缓冲器管理技术、大块 I/O 技术、表的分割技术等。本文主要讨论通过内存缓冲管理技术的使用提高数据库系统的性能。

2. 内存缓冲管理技术

由于 64 位编址的实现和对称多处理大组装盒的生产,计算机硬件具有越来越大的内存和日趋增长的强大计算处理能力,这为高性能数据库系统提供了令人鼓舞的机遇。

一个典型的数据库系统由存储在磁盘上的永久性数据构成,什么时候要处理这些数据,必须首先将它读到内存,也就通常所说的数据缓冲区,通常是调用底层操作系统的块 I/O 来实现的,数据库管理系统需要组织好这些读到内存区的数据页面,以备将来使用,以后,假如服务器在内存能找到所需页面,就不必做物理 I/O,只是进行

逻辑 I/O,我们称它为命中缓冲区。数据缓冲区就是用于服务器为用户进行数据库的各种操作,我们知道,这两种 I/O 在访问速度方面有很大的差距,所以,内存缓冲区的性能是影响数据库应用系统的关键因素之一。数据库系统的内存管理由缓冲区管理程序实现,对于一个用户查询,它通常需要访问一定数量的逻辑页面,页面多少和使用的索引有关,而优化器的主要任务是选择涉及逻辑页面数量最少的访问路径。缓冲器管理程序的目标是为实现所需的逻辑 I/O 尽可能地减少物理 I/O,有些访问路径可能有很多的逻辑 I/O,但是物理 I/O 很少。

缓冲区管理程序的效率是用“缓冲区命中率”来衡量,它是申请访问的页面在内存缓冲区中的比例。另一个衡量手段是访问竞争,因为缓冲区管理程序是共享资源,对它的并发访问是由“锁”机制来控制的。在对称多处理器结构中,要取得最优的性能,那么减少多处理器为了访问这些资源而带来的竞争就特别重要。

3. 缓冲管理程序的实现机制

(1)缓冲区数据结构。一个缓冲区由两部分组成:

①缓冲数组:含有磁盘上的数据的存储器数组。

②缓冲首部:描述缓冲区特性的数据结构。

缓冲首部到缓冲数组之间有一对一的影射关系,所以下面的讨论把这两部分统称为“缓冲区”。

一个缓冲区的数据与磁盘存储器上的一个磁盘块中的数据相对应。缓冲区是磁盘块在内存的拷贝,磁盘块的内容影射到缓冲区中。该影射是临时的,且在同一时间内,绝不能将一个磁盘块影射到多个缓冲区中。

缓冲区首部的结构如图 1 所示

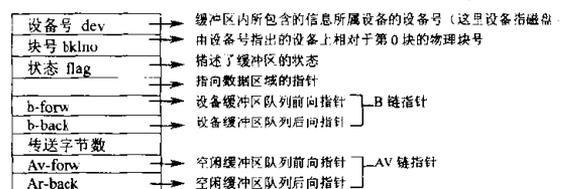


图 1 缓冲区首部结构

其中,一个缓冲区的状态是由以下内容组成:

忙标志 BUSY:缓冲区当前正“忙”,或者说是“上锁”状态;

有效位 AVE:缓冲区包含的数据有效;

写标志 WRITE:操作系统(OS)当前正把缓冲区的内容写到磁盘上;

读标志 READ:OS 当前正从磁盘往缓冲区写信息;

等待位 WAIT:一个进程当前正在等候缓冲区变为闲。

(2)队列结构。系统通过 B 链和 AV 链对所有缓冲区进行管理。

AV 链形成的空闲缓冲区队列如图 2。在此队列中的所有缓冲区的状态标志 BUSY = 0。当数据库系统启动时,每个缓冲区都放到该队列中。该队列的特点是:保存被最近使用的次序,即一个使用过的缓冲区释放时放在队尾。

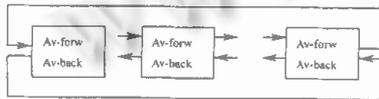


图 2 空闲缓冲区队列

B 链形成的队列结构如图 3。处于该队列的缓冲区的 FLAG 中 BASY = 1。

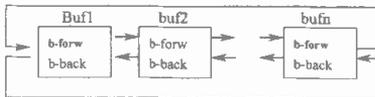


图 3 磁盘缓冲区队列

(3)缓冲区管理算法。一个缓冲区被分配用于读写磁盘上的数据库系统的数据时,它进入该磁盘的缓冲区队列(B链),该缓冲区 flag 的 BUSY 位置 1。

当缓冲区的信息读到用户内存区后,或用户信息写到缓冲区后,这样的缓冲区可以释放。此时,flag 中的 BUSY = 0,且送到空闲缓冲区队尾,这样做是为了使缓冲区能充分得到使用。因为,如果有用户需要缓冲区时,它可以从空闲缓冲区队列中去找一个,而找到的那一个缓冲区一定是这众多缓冲区中最应淘汰的一个。

当一个缓冲区被送入空闲缓冲区队尾时,它仍留在

该磁盘的缓冲区队列上。这样设计的优点是:

①在空闲 buf 队列中的缓存,只要还没有重新分配就保持其内容不变,因此,如果需要,只要简单地将相应 buf 从空闲 buf 队列中抽出,就可以按原状继续使用它。这样,对读、写操作而言,都避免了重复而又耗费的物理 I/O 操作过程,可大大提高数据库系统的性能。

②如果要将一个缓存重新分配移作它用,则只需将它从空闲 buf 队列和原磁盘 buf 队列中同时抽出,送入新的 buf 队列。这样就实现了进程对有限缓存的共享。

当从空闲队列上不断地摘下缓冲区时,一个装有有效数据的缓冲区会越来越靠近空闲队列的头部。因此,离队列头近的 buf 与离队列头远的 buf 相比,前者是最近最少使用的。这就保证了在所有空闲缓冲区中,淘汰最后一次使用时间离现在时刻最远的一个缓冲区的内容,即 LRU(最近最少使用)算法。

从上面的算法可以看出,缓冲区的使用可减少磁盘访问次数,从而提高整个系统的吞吐量,减少响应时间。通过使用 LRU 算法,避免不必要的磁盘写,使得缓冲区有较高的命中率。

4. 结束语

我们在大型数据库应用系统的开发过程中,曾遇到数据库系统性能不佳给用户带来的麻烦,如当众多用户通过拨号入网连接到远程数据库时,由于线路本身的速度限制及数据库系统性能不高,造成用户对系统响应时间的无法忍受,解决问题的途径有两条:

(1) 如果数据库系统本身提供了改善其性能的策略,就要充分利用。例如 SYBASE 数据库系统提供的命名缓冲区、大块 I/O、表的分割等技术,为改善数据库应用系统的性能创造了良好的条件。

(2) 通过自己的努力改善数据库系统的性能。例如,通过文中所介绍的内存缓冲区管理技术,以提高系统性能。

本文给出的提高数据库系统性能的技术具有通用性,不限于某一个具体的数据库系统,但要求开发者对操作系统、数据结构等计算机软件知识有扎实的基础。

参考文献

[1] 李昭原等编,数据库技术新进展,北京:清华大学出版社,1997
 [2] 邵佩英、杨孝如编,SYABSE 系统管理,北京:中国水利水电出版社,1997
 [3] 庞丽萍编,操作系统原理,武汉:华中理工大学出版社,1994

(来稿时间:1999年1月)