

分布式并行网络数据存储技术

Distributed Parallel Network Data Storage Technology

杨 峰 刘心松 罗朝劲

(电子科技大学8010研究室 成都610054)

Abstract The bases of distributed parallel network data storage technology(DPNDS) are high speed network technology, database technology, artificial intelligence and multimedia technology. It will become the foundation of modern information society because of the excellent management efficiency, reliability, availability, expansibility and low cost. In this paper, we discuss the key technology of DPNDS which includes data distributing, parallel data processing, data network storage, object-oriented data storage and then give some suggestions about the new research direction.

Keywords Data distributing, Parallel data processing, Data network storage, Object-oriented data storage

1 引言

到目前为止的信息系统中,都是以服务器和客户端为中心,存贮系统被定位于周边设备。不过,近年来随着社会信息化进程的加快,信息量急剧增加,例如在因特网上存在着数以亿计的 WWW 网页,而著名的门户网站每个星期要制造出数以 T(Tera)Byte 计算的信息。此时因特网的心脏部已经不再是“服务器”,实际上已经成为“存贮系统”。与此同时,高速网络技术、数据库技术、人工智能、多媒体技术等的发展和彼此渗透结合,不断扩展着数据存储新的研究和应用领域,为新的存储技术出现打下了坚实的基础。在这种情况下,研究和开发分布式并行网络数据存储系统(DPNDS)已经成为可能。

所谓的分布式并行网络数据存储,就是对网络数据信息进行合理的组织,利用分布式结构进行存储,以保证高可靠性和可用性。同时在此基础上提供给用户并行的数据查询和知识提取能力,最终实现网络数据系统的主动服务,为信息处理和知识的获取提供一条便捷之路。分布式并行网络数据存储是一个新概念,但其所采用的关键技术,如数据分布技术、并行数据处理、网络数据存储、面向对象数据存储技术等都有了一定的理论基础。对这些技术进行融合和进一步发展,就能构成一种新的存储技术——分布式并行网络数据存储技术。

分布式并行网络数据存储具有诸多优点,如高的数据管理效率、高可靠性和高可用性、低费用、易于扩展处理能力和系统规模等,因此将逐渐成为现代信息社会的基础。下面对其采用的关键技术进行讨论。

2 数据分布

数据分布是指将数据分散到计算机网络中的不同计算机节点上,数据分布策略的好坏对整个应用系统的可用性、可靠性以及数据的存取效率有很大的影响,所以显得特别重要。数据分布问题包括两个方面的内容:一个方面是数据片段的分割问题,即在逻辑上将全局概念模式划分成若干逻辑片段;另一个方面是数据片段的分布,即按一定的策略将这些数据片段分配到网络中的各个结点上,这时逻辑片段就成为具体的物理片段。数据分布的基本原则是:数据应分布在尽可能靠近要使用它的节点,并用负载平衡法使系统性能全局优化。

常用的数据分布方法^[1,2]有最佳适应法(Best-fit

method)、节点全受益法(all beneficial sites)和附加副本法(additional replication)等,其中最佳适应法用于非冗余分布,节点全受益法和附加副本法用于冗余分布。

•最佳适应法 设待分配的数据片段 S,对其每种可能分配方案所花费的通信费用都进行计算,然后选择一个费用最小的方案,把片段 S 分配到相应的节点上去。通常对 S 进行访问的总费用由总的查询费用和更新费用来表示。

•节点全受益法 设待分配的数据片段 S,在全部节点内选择一组节点,当片段 S 的一个副本分配到这一组节点中任何一个节点上时,其收益高于所花费的代价,则决定把片段 S 分配到这一组节点中的所有节点上。这种分配方法是对片段的初始分配费用和片段复制给某些节点后的费用计算得到的,如果复制后的费用低于复制前的费用,则进行复制。

•附加副本法 设待分配的数据片段 S,首先用最佳适应法确定一个非冗余的最佳分配方案。然后从效益最佳的副本出发,分别计算在剩余节点中的一个节点上增加片段 S 副本后整个系统的总费用,找出其中的最小费用,如果该费用大于增加 S 副本前的最小费用,则停止计算,否则在相应的节点上增加片段 S 副本。这样一直计算下去,直到找出最小费用为止。附加副本法是一种典型的启发式方法,它不但考虑到副本之间的相互影响,还考虑了随着副本的增加而带来的费用上升问题。

除了利用上述的静态数据分布方法外,在系统运行期间还要根据运行的情况进行动态数据重分布。Wang, J 等人提出一种在线(on-line)数据重分布方法^[3],其对数据、索引和日志的操作进行改进,通过增加并发性和减少时间请求来减小数据重分布的开销,使得在线数据重分布成为可能。此外,对超媒体文档数据分布也要进行特别处理^[4-6]。Ahmad, I 等人在文[4]中研究了如何利用启发式方法来获得近似优化的超媒体文档数据分布,其使用两种算法来查找 MDO(Multimedia Data Object)的近似优化分配方法:一种基于 Hill-climbing 启发式方法,另一种基于 Neighborhood search 方法。他们的结论表明通常要对算法的执行时间和分配效果进行折衷。

对于具体应用,究竟采用哪种分配方法由应用的实际情况决定。通常的根据是更新/检索比(U/Q),对于 $U/Q \ll 1$ 的数据片段,应尽量设置副本,以减少远程查询来提高数据的存取速度;对于 $U/Q \gg 1$ 的数据片段,尽量集中管理,以减少为

杨 峰 博士研究生,研究方向为分布式并行数据存储和宽带综合网络。刘心松 教授,博士生导师,研究方向为计算机网络、CATV 和电话宽带综合网络、分布式计算机、并行处理、计算机网络与通信。罗朝劲 硕士研究生,研究方向为分布式并行数据库。

保持多个副本之间一致性而付出的巨大更新开销。

3 并行处理技术

并行处理技术是提高网络数据存储任务快速响应能力的十分有效的方法。在硬件方面,并行处理是设置若干个能同时工作的部件和设备,如多个处理机并行、中央处理部件与外部设备并行、多个外部设备并行等。在软件方面,并行处理是设置若干个可以同时运行的单位(进程或者线程),通过这些单

位的运行,可以完成相同或不同的预定功能,其可以在单处理机上交替运行,也可以在多处理机组成的系统中同时在多个处理机上运行。

3.1 硬件并行

硬件并行处理主要有4种体系结构:传统的向量并行机VPP、对称式多处理机SMP、工作站机群WC及大规模并行处理机MPP。而根据处理机之间的依赖程度,则可划分为如图1所示的三种结构模式^[7,8]。

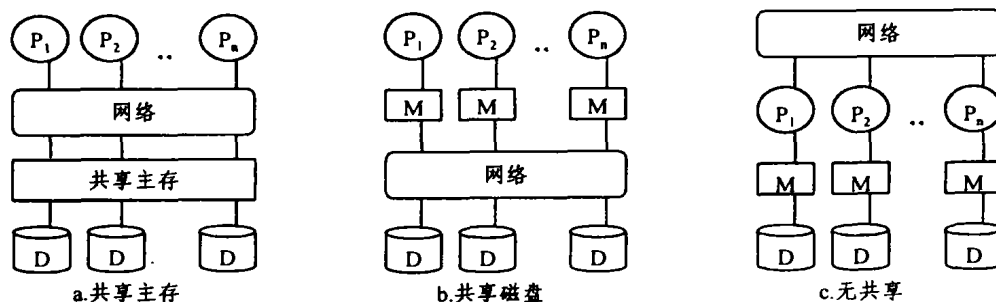


图1 并行结构的三种模式

•共享主存 SM(Share Memory):系统中的所有处理机都直接连接到一个全局存储器共享磁盘系统;

•共享磁盘 SD(Share Disks):系统中的所有处理机都有一个私有存储器,但都直接连接到一个共享磁盘系统;

•无共享 SN (Share Nothing):系统中的所有处理机都有自己的私有存储器和磁盘系统,系统中没有共享硬件资源。

SM 和 SD 结构的缺点是系统可扩充性不好,系统干扰较大。SM 中网络的带宽必须是磁盘带宽和处理器带宽之和,因而增加了网络的压力。为减少网络的压力,SM 中每台处理器都配有私有 Cache,但有研究表明,在数据存储应用中,这些私有 Cache 的装载和倒空会严重影响处理机的性能^[9]。SD 结构对于数据的并发读、写操作不十分有效,因为若某台处理机想要更新一个数据,必须先获得该数据的副本,同时必须向其它处理机宣布它的更新意图。只有当它的宣布得到所有处理机认同后,它才能开始读取并更新,这一过程都要通过网络信息交换来完成。

相比之下,SN 结构中的共享资源最少,因而干扰也最小。SN 结构具有对网络带宽要求低和易于扩充规模等优点,此外,SN 结构还特别适合 Client/Server 计算模式。虽然 SD 和 SM 结构中都采用了很多优化措施,但这些措施的结果都是倾向于数据划分,实际上就是倾向于 SN 结构,然而它们却没有 SN 的网络简单和代价低的优点。

3.2 软件并行

多线程(Multithread)并行技术可以大大提高数据存储访问系统的性能。在多线程结构的系统中,服务器作为一个单一进程运行,调度、任务转换、盘缓冲、锁和事务处理均由服务器管理而不要花费任何 OS 开销,每增加一个用户只需增加一个线程,可以有更多的存储空间用于磁盘缓冲和过程缓冲,从而减少 I/O 开销。此外,线程的切换只是简单的运行状态改变,而不会引起进程空间的切换,从而减少了系统方面的开销。综合这些因素,使得单进程多线程数据访问系统的响应时间和吞吐能力比在每用户进程结构中有十分明显的改进。

提高数据存储访问系统性能的另一个重要手段是并行查询优化,特别是并行连接(parallel join)和并行排序(parallel sort),由于它们的运算代价高且使用频繁而受到特别的重视。并行查询优化被看作常规查询优化的扩展,其执行计划的

产生通常包括两种方式:两阶段(two-phase)方式^[10~12]和一阶段(one-phase)方式^[13]。两阶段方式包含两个不同的顺序阶段:执行计划产生阶段和执行计划的资源分配阶段。而在一阶段方式中,执行计划的产生和资源分配被合并处理。无论采用哪种方式产生并行执行计划,都存在四种不同的优化方法:静态、动态、混合和准混合(quasi-hybrid)。准混合优化方法^[14]首先应用一个静态方法,然后使用一个可能会重新调用的动态方法,和纯混合法不同,其执行期间的查询优化器主要用于修正目的。

4 数据网络存储

数据网络存储是分布式并行网络数据存储体系结构的关键问题之一。早期的存储系统是计算机系统的一部分,大多以存储设备(如磁盘)的形式出现,计算机系统可以通过总线连接到磁盘,或者通过输入/输出系统与磁盘系统相连。随着网络的发展,数据的存储逐渐由单机向多机方式和专用机发展,数据的共享与传递也逐渐从依赖主机系统向依赖网络系统发展。当前网络数据存储大致可以分为三种类型:附服务器存储 DAS(Server Attached Storage)、附网存储 NAS(Network Attached Storage)和存储区域网络 SAN(Storage Area Network)。

4.1 附服务器存储

DAS 是以服务器为中心的存储体系,其特征为存储设备是通用服务器的一部分,存储器与服务器之间通过传统的 I/O 总线通讯,数据的输入/输出由服务器负责。该服务器同时提供应用程序的运行,例如视频流、数据库等服务,数据访问与操作系统、文件系统和服务器程序紧密相关。当用户数量增加或服务器正在提供服务时,响应会变慢,在网络带宽足够的情况下,服务器本身成为数据输入/输出的瓶颈。当前绝大多数存储系统都属于这种类型,随着网络的迅速扩展,这种附于通用服务器的存储方案已不再适合大型网络系统中稳固吞吐率、可扩展性、可靠性和易管理的需求。

4.2 附网存储

这种存储方式多采用专用数据服务器,它去掉了通用服务器原有的不适用的大多数应用服务功能,而仅提供文件系统功能,用于存储服务,称之为“瘦服务器(Thin Server)”。

这种存储设备在功能上完全独立于网络上的主服务器,客户机与存储设备之间数据访问已不再需要文件服务器的干预,允许客户机与存储设备之间直接的数据访问。

一个 NAS 里面包括核心处理器、文件服务管理工具、一个或者多个硬盘驱动器用于数据的存储,并且专门优化了系统的软硬件体系结构,以方便存储到网络之间的有效数据发送,其网络带宽可达到10MB~100MB。NAS 数据服务器通过局域网的接口与应用服务器连接,由于采用局域网上通用数据传输协议,如 SMB、NFS、CIFS 等,因此能够在异构的服务器间共享数据,这一点在 Windows 和 Unix 混合环境下是十分重要的。

4.3 存储区域网络

SAN 是一种新的网络存储体系结构,其较好地解决了大数据流量的存储问题,向“网络就是存储”的目标迈进了一大步^[15]。SAN 采用高速数据连接通道—光纤通道 FC (Fiber Channel) 连接服务器和存储系统,由 SAN 取代服务器实施对整个存储过程的控制和管理,而服务器只承担监督工作。这样减少了对服务器处理时间的占用,服务器可以腾出更多的 CPU 时间去处理客户的服务请求,提高了系统的吞吐能力。

从结构上看,SAN 中的服务器和数据存储系统是相互独立的,存储系统直接连接到 FC 交换机或集线器上,以便于扩展系统规模。同时所有的存储设备和存储数据均可采用中心化管理,使得整个存储系统具有可伸缩性,并且还可以通过存储设备的集群方式而达到高可用性。由于 SAN 是一种新的网络存储技术,尚处于成长期,有关 SAN 的国际标准尚未出台,还有很多技术难题,如跨服务器平台的数据共享和互操作性、高效的存储资源分配和管理机制等有待于进一步的研究与开发^[16,17]。

从软件角度看,上述三种类型存储系统的体系结构是不一样的。在 DAS 中,应用与存储是一体化的,通过系统总线访问存储设备;NAS 是应用与存储分离的系统,应用服务器通过局域网访问文件存储系统,通常 NAS 以标准化访问协议(如 NFS)提供服务;在 SAN 中,文件系统与存储系统完全分离,存储系统实际上成为运行应用程序的服务器设备,二者以高速 FC 连接。

5 面向对象数据存储

数据存储信息通常被分为两类:一类是具有严格数据模式的格式化数据,这类数据可以表示传统的数字、字符等,它们易于组织、管理和处理;另一类是非格式化数据,这类数据可以表示文本、图像、视频、声音以及非数字形式表示的图形等,它们的管理和处理比较困难。如果对非结构化数据采用结构化数据的存储管理方式,则需要进行复杂的分割和组织,这会大大降低系统的性能,因此需要采用更为有效的方法,而目前最可行的方法就是面向对象技术。

面向对象数据存储技术将数据与操作方法一体化为对象的概念,数据和过程一起封装。现已出现了一些借鉴面向对象程序设计的思想和成果的原型和产品。在数据模型方面,对象、封装、对象识别符、类层次、子类、继承概念和功能已初步形成;在数据管理方面,提出了持久性对象、长的事务处理、版本管理、方案进化、一致性维护和分散环境的适应性问题;在数据访问界面上,提出了消息扫描、持久性程序设计语言、计算完备性等概念。总之,面向对象数据存储系统的形象正逐步明朗起来。

但是,到目前为止所提出的对象存储模式^[18,19]都还不能令人满意。通常采用的聚簇方法和基于记录类型的方法实际

上都是把静态对象和动态对象封装存储在一起,这既造成时间上的浪费,也增加了空间管理的难度。而基于集合的存储方法也因对象的行为特征与集合的数学特征之间的矛盾而很难成为一种理想的存储模式。此外,在面向对象数据存储的实现技术方面也还有许多问题需要解决,如模拟非格式化数据的内容和表示,反映多媒体对象的时空关系,允许有类型不确定对象的存在等。这些都有待于进一步的深入研究。

结束语 分布式并行网络数据存储技术是高速网络技术、数据库技术、人工智能、多媒体技术等融合与发展,是一项复杂而庞大的系统工程。其具有诸多优点,如高的数据管理效率、高可靠性和高可用性、低费用、易于扩展处理能力和系统规模等,因此将逐渐成为现代信息社会的基础。本文对分布式并行网络数据存储所采用的关键技术,如数据分布技术、并行数据处理、数据网络存储、面向对象数据存储等进行了详细的阐述,并提出了一些需要进一步研究的方向。

参考文献

- Teorey T J. Data Modeling & Design. Morgan Kaufmann Publishers Inc, 1994
- Stefanoceri G P. Distributed Database Principle and System, 1984
- Wang J, Miyazaki M, Kameda H, Li J. Improving performance of parallel transaction processing systems by balancing data load on line. Seventh International Conference on Parallel and Distributed Systems, 2000. 331~338
- Siu-Kai So, Ahmad I, Karlapalem K. Response time driven multimedia data objects allocation for browsing documents in distributed environments. IEEE Transactions on Knowledge and Data Engineering, 1999, 11(3): 386~405
- Siu-Kai So, Ahmad I, Karlapalem K. A data allocation algorithm for distributed hypermedia documents. Seventeenth IEEE Symposium on Reliable Distributed Systems, 1998, 473~478
- 杨峰,刘心松,邱元杰. 分布式并行视频服务器设计技术. 小型微型计算机系统, 2000, 21(11): 1169~1172
- Stonebraker M. The case for shared nothing. IEEE Transactions on Database Engineering, 1986, 9(1)
- 杨利,周兴铭,郑若忠. 并行数据库的体系结构. 计算机工程与科学, 1994, 16(2): 7~17
- Thakkon S, Sweiger M. Performance of an OLTP applications on symmetry multiprocessor system. In: Proc. of the 17th annual Intl. Symposium on Computer Architecture, 1990
- Bonneau S, Hameurlain A. Hybrid Simultaneous Scheduling and Mapping in SQL Multi-Query Parallelization. In: Proc. of the 10th Intl. Conf. on Database and Expert Systems Applications, 1999, 1677: 88~99
- Chckuri C, Hasan W. Scheduling Problem in Parallel Query Optimization. Symposium in Principles of Database Systems, 1995. 255~265
- Garofalakis M N, Ioannidis Y E. Multi-dimensional Resource Scheduling for Parallel Queries. In: Proc. of ACM SIGMOD, 1996, 25: 365~376
- Kosch H. Managing the operator ordering problem in parallel databases. Future Generation Computer Systems, 1999, 14
- Bonneau S, Hameurlain A. Query Mapping onto a Shared-Nothing Multiprocessor Architecture: from Static to Dynamic. In: Proc. of the 2nd National Conf. on Dynamic Mapping and Load Partitioning, 1998. 87~91
- Christensen B. Building a storage-area network. Data Communications, 1998, 27(6): 67~74
- Kim Chang-Soo, Kim Gyoung-Bae, Shin Bum-Joo. Volume management in SAN environment. In: Proc. of Eighth International Conference on Parallel and Distributed Systems, 2001. 500~505
- Milanovic S, Petrovic Z. Building the enterprise-wide storage area network. Intl. Conf. on Trends in Communications, 2001, 1: 136~139
- Ge Yu, Kaneko K, Bai G, Makinouchi A. Transaction management for a distributed object storage system WAKASHI-design, implementation and performance. In: Proc. of the Twelfth Intl. Conf. on Data Engineering, 1996. 460~468
- Nakajima T. An adaptive object storage system for mobile computers. Seventh Intl. Conf. on Parallel and Distributed Systems: Workshops, 2000. 335~342