第35卷 第11期 计 算 机 工 程 2009年6月
Vol.35 No.11 Computer Engineering June 2009
·软件技术与数据库· 文章编号：1000—3428(2009)11—0040—03 文献标识码：A 中图分类号：N945

并行文件系统的框架设计和性能研究

谢桂园[1]，魏文国[2]
(1. 广东技术师范学院培训中心，广州 510665；2. 广东技术师范学院电子与信息学院，广州 510665)

摘 要：利用InfiniBand技术特征实现高效的并行文件系统(EPFS)，设计一个高性能的透明传输层，对数据流的缓存管理、动态和公平的缓存共享，以及有效的内存注册和注销进行研究。实验表明，当I/O节点足够多时，随着计算节点的增加，基于InfiniBand技术的EPFS比基于TCP/IP的EPFS的读写性能增长更快。并且，两级别的内存注册和注销方法AFMRD比受约束的缓存技术更好地改进I/O性能。
关键词：并行文件系统；缓存管理；InfiniBand技术

Frame Design and Performance Research of Parallel File System

XIE Gui-yuan[1], WEI Wen-guo[2]
(1. Training Center, Guangdong Polytechnic Normal University, Guangzhou 510665;
2. College of Electronics and Information, Guangdong Polytechnic Normal University, Guangzhou 510665)

【Abstract】This paper examines the feasibility of InfiniBand technology to design and implement an Effective Parallel File System(EPFS). It designs a transport layer customized for EPFS by trading transparency and generality for performance, buffer management for flow control, dynamic and fair buffer sharing, and efficient memory registration and deregistration. Compared with an EPFS implementation over TCP/IP on the InfiniBand network, the implementation offers better I/O performance if the number of I/O nodes is not bottleneck. Further, two-level memory registration and deregistration technology(AFMRD) works much better than pin-down cache technology.
【Key words】parallel file system; buffer management; InfiniBand technology

1 概述

网络存储系统的性能受限于内存复制、网络访问开销、协议实现开销等问题。因为InfiniBand技术采用了用户级别网络和远程内存直接访问(RDMA)技术，使得在不改变操作系统的前提下能够解决上述问题。很多用户级别的通信协议使用到网络存储中，例如文献[1]对数据库存储的VIA网络进行了实验，文献[2]对DAFA在VIA上的性能特征进行了研究，本文的工作是基于InfiniBand体系结构(IBA)的。文献[3]研究了InfiniBand中的传输层技术，本文研究的侧重点主要在于通信机制、以及缓存管理器和通信管理器处理I/O密集的应用程序时如何协作。文献[4]提出了减少内存注册和注销开销的不同方法，例如受约束的缓存和批处理内存注销，本文的方法组合了上述2种方法的优点。

2 EPFS体系结构

EPFS是Linux集群下的并行文件系统。在EPFS中，部分节点被当作I/O服务器，部分节点被配置成元数据服务器，EPFS将文件条带化分布到多个I/O节点上来取得并行访问的汇集性能。在每个I/O节点上有一个I/O的后台程序来响应计算节点的I/O请求，因此，数据直接在I/O服务器和计算节点之间传输。

EPFS最初设计成固化在TCP/IP协议上，sockets被用于传输消息。采用TCP/IP的流语义来避免缓冲区管理。因为sockets和InfiniBand的用户层接口在语义和功能上都有显著的不同，所以采用模块化体系结构以便进行有效的设计和实现。

EPFS中共有6个模块，如图1所示：缓存管理器，通信管理器，EPFS传输层(包括客户端和服务器端)，客户端产生请求的EPFS库，服务器端的请求管理器和文件访问管理器。传输层使用用户级别的InfiniBand原语来传输数据，缓存管理器使用文件访问管理器提供的信息来为传输层和文件访问提供缓冲区，通信管理器选择通信机制和调度数据传输。

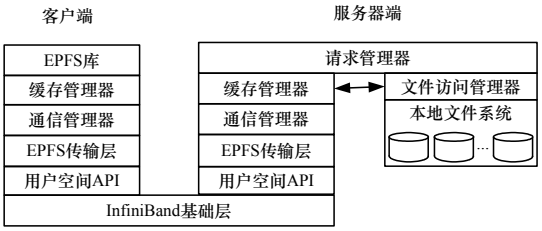

图1 EPFS体系结构

InfiniBand与其他网络相比对EPFS提供更加弹性的设计空间，通信管理器为每个消息选择通信机制，通过调度来减少网络阻塞和延迟，同时它也能为每个消息指定优先级。本文的研究专注于传输层和缓冲区管理。

3 EPFS的传输层设计

EPFS的传输层在计算节点、I/O服务器和元数据管理器之间提供数据、元数据和控制指令的传输。本文分析EPFS的各种消息的特征，并描述相关的通信策略选择，包括通信选择、消息传输机制和事件处理，然后提出优化的小数据传输和管道化(pipelined)的批数据传输来优化EPFS的传输层。

基金项目：广东省自然科学基金资助项目(06025383)
作者简介：谢桂园(1977－)，女，讲师，主研方向：计算机网络，分布式计算，高性能计算；魏文国，副教授、博士
收稿日期：2009-01-26 **E-mail：**gsxgy@gdin.edu.cn

3.1 EPFS中的消息和缓冲区

EPFS 中的消息分为请求消息、应答消息、数据消息和控制消息 4 种。请求消息从计算节点发给 I/O 节点或者元数据服务器，表示读、写和查询数据，管理节点也使用请求消息来管理 I/O 服务器节点的元数据；应答消息由服务器反馈给请求发起者；数据消息用来传递文件的读写；控制消息是 EPFS 的内部消息，进行流控制等功能。

EPFS 中有 2 类缓冲区：内部缓冲区和 RDMA 缓冲区。内部缓冲区由 EPFS 系统分配，当连接建立时被分配并保留相当长的一段时间，在服务器端用于服务多个客户请求；RDMA 缓冲区在计算节点和 I/O 节点之间实现零拷贝的数据传输，客户端的 RDMA 缓冲区由应用程序在读写操作时提供，服务器端的 RDMA 缓冲区用于数据传递到磁盘或者网络之前缓存到内存中。

3.2 传输层优化

使用 2 种方案来优化小数据的传输：内嵌(inline)和快速 RDMA 写。对批量的数据传输，使用管道化(pipeline)通信。

内嵌(inline)数据传输：零拷贝数据传输需要在传输之前注册应用程序的缓冲区，在传输之后注销缓冲区。对小数据来说，使用零拷贝数据传输可能得不偿失。在内嵌(inline)数据传输方案中，数据首先被复制到被预注册的内部缓冲区，并通过发送/接收机制传输。如果数据在写请求或者读反馈时能装配到内部缓冲区中，则它们通过一次信息就可发送。该技术在文献[2]中也被使用。

快速 RDMA 写：当传输的数据不大时，RDMA 读和 RDMA 写有显著的性能差异。这意味着对小数据传输，使用 RDMA 写更好，快速 RDMA 写主要用于优化 EPFS 的写操作。

管道化批量数据传输：每次 I/O 有 2 个阶段，即数据在客户缓冲区和服务器缓冲区直接传输的通信阶段和数据从服务器缓冲区写到磁盘的 I/O 阶段。为了提高性能，对大数据的读写将这两阶段交替进行是必须的。将两阶段交替进行的一种方法是将大的读写划分成多个小数据的读写。管道化通信也减少了 I/O 服务器端的内存压力，I/O 服务器可以使用双缓冲区来响应并发的请求。

4 缓冲区管理器的设计

缓冲区管理器为 EPFS 传输层提供缓冲区和文件访问管理。缓冲区可以是内部缓冲区或者 RDMA 缓冲区。缓冲区管理器有 2 个主要任务：(1)为 RDMA 缓冲区提供有效的内存注册和注销；(2)提供公平和动态的缓冲区共享。

4.1 服务器端RDMA缓冲区管理

服务器端 RDMA 缓冲区用于从客户端接收数据和从文件中读数据。这些缓冲区被有效地用于平衡网络和磁盘的性能差异。因为高并发的请求和可能的大请求，所以必须在专用的服务器上分配部分内存作为 RDMA 缓冲区使用，很明显：服务器对不同的请求可以重用这些缓冲区。因此，这些内存区域在开始时可以预注册。从动态性考虑，希望 I/O 服务器端内存注册和注销的频率低一些。

在基于 InfiniBand 的 EPFS 传输层，数据是作为一个整体而不是字节流传输，缓冲区被明确地分配。另一个问题是请求的大小不同，需要缓冲区管理器能提供不同大小的相邻缓冲区，尽量避免内存碎片。

服务器的缓冲区管理设计如下：所有的 RDMA 缓冲区按照区域(zone)分配，每个区域有相同大小的一些缓冲区。给定一个特定的传输大小，查找对应的区域列表来得到一个连续的缓冲区，若没有可用的缓冲区，则可从更大的区域列表中选一个；若没有可用的更大的缓冲区，将要传输的数据分块以便能使用小的 RDMA 缓冲区。通过这种方法，RDMA 缓冲区不会动态地产生更多的碎片。

4.2 客户端RDMA缓冲区管理

客户端缓冲区起初用于 EPFS 内存的有效注册和注销，内存注册和注销是开销大的操作，管理不好会影响系统性能。另一方面，EPFS 的 I/O 应用程序需要大量的 I/O 缓冲区在请求提交时分配，因此不可能预注册所有的 I/O 缓冲区，动态的缓冲区注册和注销很难避免。

为了减少动态的缓冲区注册和注销开销，另外增加一个受约束(pin-down)的缓存[5]，受约束的缓存延迟注销已经注册的缓存信息，并将它们保存起来，当缓冲区被重用时，这些信息又从受约束的缓存中被恢复。当缓冲区重用率较高时该方法比较有效。

无论如何，EPFS 关注的 I/O 密集的应用程序使用大量不同的 I/O 缓冲区，这些缓冲区的重用率可能较低。因此上述受约束的缓存方法行不通，下面提出一个两级结构来有效地支持 I/O 密集的应用程序的内存注册和注销。

4.3 两级结构的内存注册和注销

若应用程序一直使用不同的缓冲区，则动态缓冲区注册不可避免。为了减少开销，InfiniBand 提供有效的注册操作，有关缓冲区注销的优化技术可以参考文献[1]研究的方法：“批处理注销技术”。

本文提出一个两级结构：受约束的缓存加上快速内存注册/注销(AFMRD)，该结构具有受约束的缓存和批处理注销两者的好处。当缓冲区被注册时，首先检查它的注册信息是否在受约束的缓存中。若是，信息立即被取出来，否则，使用 FMR(快速内存注册)来注册用户缓冲区，并将注册信息插入到受约束的缓存中。若受约束的缓存没有空间，则从受约束的缓存删除一项并放到注销列表中，当注销列表的长度达到一个阈值时，使用 FMD(快速内存注销)来注销其中所有的缓冲区。当缓冲区被注销时，只是在受约束的缓存中将该缓冲区的引用计数修改成 0，其后真实发生的注销以批处理方式完成。

5 性能结果

笔者已经在 InfiniBand 的 VAPI 接口[6]上实现了 EPFS 有关的设计，实验在 6 台联想万全 R510G6 服务器组成的集群上进行，每个节点配置了 Intel 5320 双核 CPU、2 GB FBD 内存、2×73 GB SCSI 硬盘、一块 InfiniBand 网卡、一块千兆以太网卡，操作系统是 RedHat 9。使用 EPFS 的测试程序和典型的应用程序来对比测试，然后量化不同的缓冲区管理策略对性能的影响。

5.1 基于InfiniBand的EPFS的并发写带宽

使用基于 MPI 的 EPFS-test 程序测试并发的写带宽，每个计算节点产生一个进程都作如下相同的读写操作：所有进程打开同一个新的 EPFS 文件，并发、组合地写该文件的不同区域，关闭该文件，重新打开它，同时从该文件中读取前面写的数据块，然后关闭文件。记录每个节点读写的时间开销，取所有进程中花费最多的时间来计算读写带宽。如图 2 所示，标记“TCP/IP(2*I/O, 4 MB)”表示 2 个 I/O 节点写 4 MB 大小的文件。可以看到当 I/O 节点足够多时，每增加一个计算节点，基于 TCP/IP 的 EPFS 的写带宽增加大约 160 MB/s，而基于 InfiniBand 的 EPFS 大约增加 360 MB/s，即 EPFS 在

InfiniBand 网络比在 TCP/IP 网络的读写性能增长更快。

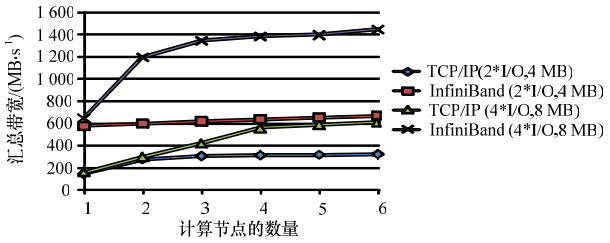

图 2　基于 TCP/IP vs. InfiniBand 的 EPFS 写性能比较

5.2　两级结构的内存注册和注销AFMRD

使用 EPFS-test 评估 3 种不同的内存注册和注销方法，结果如图 3 所示。

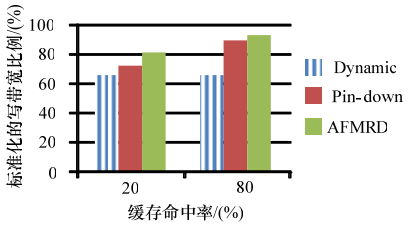

图 3　3 种内存注册/注销方法的写带宽比较

第 1 种方法是对每个 I/O 操作都进行动态的内存注册和注销(简称为 Dynamic)，第 2 种方法是使用受约束的缓存(简称为 Pin-down)，第 3 种是 AFMRD。测试程序执行 500 次 I/O 操作，使得可能有 500 个不同的缓冲区大小。明确地控制受约束缓存的缓存命中率为 20%(表示低的缓存命中率)和 80%(表示高的缓存命中率)，缓存大小为 100 KB。图 3 表示 3 种不同方法的 EPFS 写带宽。这些数据都是与没有缓冲区注册和注销的方法标准化后的结果。得到如下结论：3 种方法中 AFMRD 能显著地改进 I/O 性能。

6　结束语

本文研究了如何利用 InfiniBand 网络、用户级别的通信和 RDMA 特征改进 EPFS 的吞吐量、读写带宽等。验证了两级别的内存注册和注销在 I/O 密集的环境中比其他的方案有效。下一阶段的工作包括将这些技术完美地集成到新的 EPFS 系统中。

参考文献

[1] Zhou Yuanyuan, Bilas A, Jagannathan S, et al. VI-attached Database Storage[J]. IEEE Transactions on Parallel and Distributed Systems, 2005, 16(1): 35-50.

[2] Magoutis K, Addetia S, Fedorova A, et al. Making the Most out of Direct Access Network-attached Storage[C]//Proc. of the 2nd USENIX Conference on File and Storage Technologies. San Francisco, CA, USA: [s. n.], 2003.

[3] Zahir R. Lustre Storage Networking Transport Layer[EB/OL]. (2008-05-02). http://www.lustre.org/docs.html.

[4] Welsh M, Basu A, Eicken T V. Incorporating Memory Management into User-level Network Interfaces[R]. New York, USA: Cornell University, Tech. Rep.: TR97-1620, 1997.

[5] Tezuka H, Carroll F O, Hori A, et al. Pindown Cache: A Virtual Memory Management Technique for Zero-copy Communication[C]//Proc. of the 12th Int. Parallel Processing Symposium. Orlando, FL, USA: [s. n.], 1998.

[6] Mellanox Technologies, Inc.. Mellanox IB-Verbs API(VAPI), Rev. 0.95[Z]. 2003.

编辑　任吉慧

(上接第 39 页)

TTL=TTL−1
If　TTL>0　then
Add(<SCID1, SCID2,…, SCID*n*>)
/*将自身的服务社区 ID (SCID)加入到已经请求过的服务社区代理列表中*/
For item in RouteTable
/*RouteTable 指要执行此转发策略的领域社区代理上的路由表*/
If Similarity(Request's Goal，item's SCGoal)>ξ　then　/*ξ 为设定的阈值*/
If item's SCID not in <SCID1, …, SCIDn>　then
<<SCID1, SCID2,…, SCIDn>, TTL, < Peer-Req, Request >> /*向该服务社区代理转发该消息*/
End if
End if
End for
End if

(3)服务社区代理接收到 P2P 网络转发的查询请求时，除完成上面所述的转发任务外，还将原始的查询消息〈Peer-ReqID, Request〉从收到的消息中解析出来，在服务社区内以广播方式转发。社区内各 Peer(包括服务社区代理)收到查询消息后，对 Request 与本地 Web 服务本体库中 *WS* 的 *Capability* 进行精确的语义匹配，如果存在相似度 SemSim 大于某个阈值，则直接按〈Peer-Ans ID, URL, SemSim〉的消息格式将结果反馈给 Peer-Req。

4　结束语

本文提出基于类别 Chord 环、领域社区、服务社区的 3 层拓扑服务发现 P2P 网络以及基于该网络的服务发现算法，有效解决了现有语义 Web 服务发现机制存在的问题，兼顾了服务发现效率与服务发现的查全率、查准率。但上述网络和算法仍然需要完善，下一步工作包括建设一个合理的类别本体、优化服务社区构建策略和服务请求信息转发策略。

参考文献

[1] Verma K, Sivashanmugam K, Sheth A. Meteor-S WSDI: A Scalable P2P Infrastructure of Registries for Semantic Publication and Discovery of Web Services[J]. Journal of Information Technology and Management, 2004, 6(1): 17-40.

[2] 陈德伟，许　斌，蔡月茹. 服务部署与发布绑定的基于 P2P 网络的 Web 服务发现机制[J]. 计算机学报, 2005, 28(4): 615-626.

[3] 于守健，夏小玲，乐嘉锦. 基于语义描述的分布式 Web 服务发布与发现[J]. 计算机工程, 2007, 33(7): 117-119.

[4] 刘志忠，王怀民，周　斌. 一种双层 P2P 结构的语义服务发现模型[J]. 软件学报, 2007, 18(8): 1922-1932.

[5] Lai Yusheng, Wu Chung-Hsien. Meaningful Term Extraction and Discriminative Term Selection in Text Categorization via Unknown-word Methodology[J]. ACM Transactions on Asian Language Information Processing, 2002, 1(1): 34-64.

编辑　陈　晖