第32卷 第21期 计 算 机 工 程 2006年11月
Vol.32 No.21 Computer Engineering November 2006
·软件技术与数据库· 文章编号：1000—3428(2006)21—0061—03 文献标识码：A 中图分类号：TP334.4

基于透明计算模式的I/O Server的设计

夏 楠[1]，张尧学[2]，杨善林[1]，王晓辉[2]
（1. 合肥工业大学计算机网络系统所，合肥 230009；2. 清华大学计算机科学与技术系，北京 100084）

摘 要：设计并实现了一个基于透明计算模式的I/O Server系统，I/O Server和I/O Client是一个在透明计算环境下，支持多操作系统远程启动和运行的网络存储访问服务I/O Manager的2个软件模块，I/O Server工作在服务器端，I/O Client工作在客户端。在透明计算模式中，各客户机硬件与操作系统分离，用户需要的操作系统的应用程序存储在服务器端。在客户机启动时，I/O Server和启动协议将I/O Client下载到端系统上运行，然后I/O Client向I/O Server发出I/O请求，I/O Server对收到的I/O请求加以分析，进行优先级分类，在优先级分时轮转调度I/O请求、操作服务器上的虚拟硬盘文件，并通过预取和缓存策略减少磁盘I/O操作，将处理结果返回给客户端，支持操作系统的远程启动，并为系统运行时的各种请求提供服务。

关键词：计算机网络；操作系统；客户/服务器；远程启动；缓存

Design of I/O Server Based on Transparent Calculation Model

XIA Nan[1], ZHANG Yaoxue[2], YANG Shanlin[1], WANG Xiaohui[2]
(1. Institute of Computer Network Systems, Hefei University of Technology, Hefei 230009;
2. Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

【Abstract】This paper presents the design and implementation of an I/O server based on transparent calculation model. I/O server and I/O client are two software modules of an I/O manager which is a network storage access service supporting multi-OS remote start in transparent compution environment. I/O server works on the server, and I/O client runs on the client. In transparent calculation model, the operation system of the clients is separated from the hardware, the operation system and applications which users want to use are all stored on the server. When the clients boot, I/O server and the boot protocol works together, downloads I/O client to the clients' memory, and then I/O client sends I/O requests to I/O server. I/O server analyzes the requests received and classifies the priorities, then schedules these classified requests based on the priorities to operate virtual disk files on the server, sends the results to the clients. To reduce the disk I/O, I/O server also applies prefetching and cache policy. I/O server supports the remote boot of the operation systems, and provides services for all kinds of requests as the operation system is running.

【Key words】Computer network; Operation system; Client/Server; Remote boot; Cache

传统的计算机系统将硬件系统和操作系统绑定，用户在执行过程中不能选择运行其他的操作系统，这使得软件系统越来越复杂，维护成本也越来越高。文献[1]提出了一种新的网络计算模式——透明计算，透明计算模式中的客户机上不用预置任何操作系统，用户根据需求，使用各种终端设备(包括固定、移动以及家庭的各种设备)通过网络选择可以在客户机上运行的操作系统和应用，从而显著地降低计算机的维护和升级成本，节省客户机存储空间，同时客户机也充分利用计算资源，减轻了服务器负担。文献[2]提出了一个基于透明计算模式实现的实例，在该系统中，根据用户需求，客户端以文件形式下载需要的操作系统内核，在客户机上加载、运行操作系统，实现终端设备的可定制远程启动。文献[1,2]已经实现了局域网环境下Windows98和Linux的远程启动。

这种以文件形式下载数据的方法，下载速度慢、安全性低、不易扩展到对更多操作系统的支持。本文提出了一种按扇区下载数据的方式，来实现操作系统远程启动的网络存储访问服务。基于C/S架构，I/O Manager包括I/O Server和I/O Client2个软件模块，其中I/O Server工作在服务器端，I/O Client工作在客户端，本文描述I/O Server的设计与实现。

1 运行环境

I/O Manager工作在局域网环境下，能够在多操作系统远程启动协议(Multi-OS Remote Boot Protocol, MRBP)[3]的支持下，实现多台客户机远程加载操作系统。I/O Server是I/O Manager设置的服务器端应用层服务程序，当I/O Client在端系统启动时被动态地加载到客户端内存后，客户端系统访问本地磁盘的I/O请求就被I/O Client重定向到服务器端，I/O Server就是负责响应这些I/O请求。图1给出了I/O Server运行时的系统环境，其中MRBP协议工作在客户机/服务器环境下，MRBP Client是MRBP协议的启动代理，存储在客户端网卡只读存储器中；MRBP Service是MRBP协议的启动服务程序[3]；客户机启动和运行时需要的操作系统、应用程序和数据被保存在服务器上一个或几个二进制文件中，把这些文件称为虚拟硬盘文件(Virtual Disk File, VDF)，并将虚拟硬盘文件分为两类：

(1)公有文件(Common VDF, CVDF)：存放的是操作系统和应用程序以及公有数据，所有客户端共享公有文件。

(2)私有文件(Private VDF, PVDF)：每个客户端都有一个

基金项目：国家“863”计划基金资助项目(2004AA111020，2004AA114062)

作者简介：夏 楠(1980－)，男，博士生，主研方向：透明计算，计算机网络；张尧学、杨善林，教授、博导；王晓辉，博士生

收稿日期：2005-09-16 **E-mail：**xinnan1980@163.com

私有文件存放用户的私有应用程序和数据。

客户机加电开机，BIOS 完成自检后跳转到 MRBP Client 处，MRBP Client 与 MRBP Service 通过 MRBP 协议交互，获取客户机 IP 地址等网络参数，并下载服务器提供的操作系统列表文件，在用户选定要加载的操作系统后，MRBP Client 将 I/O Client 下载到客户端内存。I/O Client 接管系统控制权后，向 I/O Server 请求下载存放在 CVDF 中的主引导扇区 MBR，并定义新的磁盘访问中断处理程序，将实模式下对本地磁盘的访问重新定向到服务器上的虚拟硬盘文件，I/O Server 收到请求后将 MBR 以扇区形式下载到客户端内存。MBR 得到控制权后，加载引导扇区、引导扇区要求加载内核加载器，此操作经过网络被透明地发送给 I/O Server，I/O Server 收到请求后，读取 CVDF 中的引导扇区和内核加载器程序，并发送到客户端内存。内核加载器运行后将系统由实模式切换到保护模式，发出下载操作系统内核的请求，I/O Server 收到请求后又将操作系统内核传送到客户端内存。在系统切换到保护模式后，I/O Client 创建本地虚拟硬盘，将保护模式下对磁盘的访问重新定向到服务器端，此后操作系统内核装载操作系统程序时产生的本地虚拟硬盘的 I/O 操作，被封装后发送给服务器端的 I/O Server，I/O Server 对收到的请求进行处理后再将结果返回给本地虚拟硬盘，从而实现操作系统的远程启动。系统启动后运行应用程序或者访问私有数据的请求也同样被本地虚拟硬盘发送给 I/O Server，由 I/O Server 为其提供服务。I/O Server 的运行环境如图 1 所示。

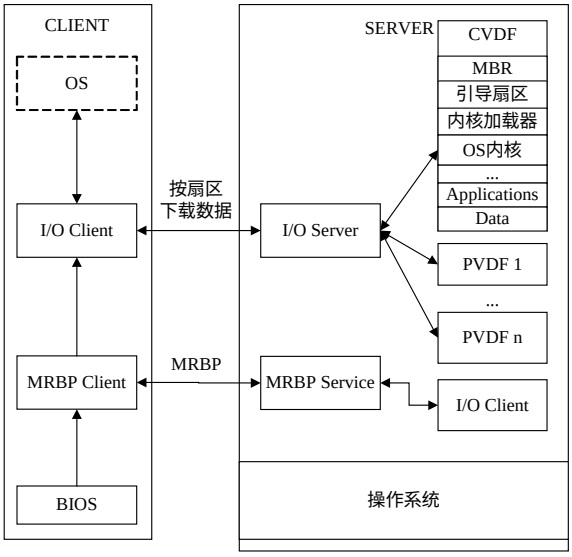


图 1 I/O Server 的运行环境

2 模型结构与实现

I/O Server 是一个工作在服务器端应用层的服务程序，其作用就是响应 I/O Client 的请求，将客户端请求解析成访问服务器上的虚拟硬盘文件的参数：虚拟硬盘文件类型(公有、私有)，操作类型(读、写)，数据长度和起始地址等。然后定位到虚拟硬盘文件的相应位置进行读写操作，并将处理结果以扇区形式发送给 I/O Client。

I/O Server 为多台客户机服务，在服务器 I/O 和局域网网络带宽的限制下，为保证多台客户机的 I/O 请求能得到及时、正确的处理，必须建立合适的模型在保证公平性的前提下，提高 I/O Server 的效率。图 2 是 I/O Server 的模型结构，将 I/O Server 处理网络 I/O 请求的过程分为 4 个模块：分类，调度，数据操作和发布。

通过对 I/O 请求进行优先级分类、输入排队后再进行调度，操作缓存中的虚拟硬盘文件内容，把 I/O 处理串行操作变成并行操作，以提高 I/O Server 的效率。

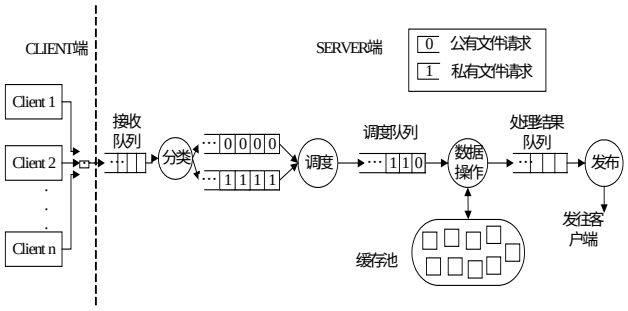


图 2 I/O Server 模型结构

2.1 分类模块

客户端发来的请求首先排队到服务器的网卡缓存队列里，因为多台客户机在同一时刻的状态不一定相同，有的客户机可能正在启动、有的客户机可能在运行应用程序、有的客户机可能在访问自己的私有数据等，而处在不同状态的客户机发出的请求所要求的服务质量也有较大差异，所以 I/O Server 设置分类模块，对网卡缓冲区中的请求进行分析、分类、赋予不同的优先级。

分类模块以FIFO方式从网卡缓冲区中读取请求，分析请求内容，获得虚拟硬盘文件类型号，然后根据虚拟硬盘文件类型，将请求分为公有文件请求和私有文件请求 2 类，排队到不同的缓存队列中，分别用R_0、R_1表示，优先级按R_0、R_1次序依次降低。

2.2 调度模块

客户端发出的 I/O 请求经分类模块分类后被加入到不同优先级的请求队列中，高优先级队列中的 I/O 请求要求响应时间短，应当优先处理，但若严格地遵循高优先级优先处理的策略，在有大量的高优先级 I/O 请求的情况下，低优先级的 I/O 请求可能长时间得不到处理。

为了保证各优先级的I/O请求都能得到公平调度处理，调度模块基于优先级分时轮转的规则，调度不同优先级的I/O请求。调度模块选用一个时间段T作为轮转调度周期，然后将周期T分为公有文件请求时间片T_0、私有文件请求时间片T_1 2 个时间片，在周期T内，调度模块基于分时轮转规则，调度这 2 个优先级的I/O请求。但T_0和T_1的值并不是固定不变的，调度模块为每个队列设置了一个请求计数器，记录在周期T内排队到这 2 个队列中的请求个数，然后根据请求个数的变化，动态调整下一个周期T内分配给 2 个队列的时间片T_0和T_1的值，请求个数多的队列所获得的时间就多些，以提高请求的处理速度。

2.3 数据操作模块

文献[3]中的研究结果表明，大部分的文件读请求是顺序读，而且所有的客户机共享公有文件，一台客户机访问过的数据，也可能被其他的客户机访问或者自身再次访问，为了有效地利用读取到内存中的数据，减少磁盘I/O操作，预取和缓存策略是提高系统性能的有效方法[4]。

数据操作模块建立了一个线程池来并行处理调度队列中的I/O请求。数据操作模块还建立了一个缓存池，并为公有文件和私有文件分别维护一个缓存区索引表，索引表包括缓存块号、客户机IP地址、缓存区数据起始扇区位置、访问次数、

状态以及指向存储数据的内存块的指针等信息。在索引表中存放客户机IP地址的目的，是将用于机群系统的合作缓存[5]概念用到缓存管理中。数据操作模块为每个客户机至少分配一个缓存区，将请求要求访问的数据及其后续的部分数据存放到缓存区中，考虑到文件访问的顺序性，从而便于处理客户端的下个请求时，能够不需要访问磁盘就可以返回结果。

数据操作模块的具体工作流程如下：

(1)数据操作模块从调度队列头部取出一个请求，并从线程池中分配一个线程负责处理该请求；

(2)线程解析请求的内容，判断该请求所要操作的虚拟硬盘文件类型(公有、私有)，然后查询相应的虚拟硬盘文件缓存区索引表；

(3)判断分配给其缓存区中的数据是否合适，如果合适则进行操作并转(7)，如果不合适则再查询索引表，判断分配给其他客户机的缓存区内容是否符合请求，如果符合则转(4)，如果不符合则转(5)；

(4)判断该缓存区是否为写锁定状态。如果不是则进行操作；如果是则等到解除写锁定，然后转(7)；

(5)从硬盘读取数据作为处理结果，转(7)，并查询缓存池中是否有剩余空闲缓存区。如果有则将从硬盘读取的数据放入该缓存区并更新索引表；如果没有则转(6)；

(6)查询索引表，采用 LRU 策略替换缓存区数据，被修改过的缓存区数据由数据操作模块采用延迟写机制保存到对应的虚拟硬盘文件中，更新索引表；

(7)将处理结果存放到处理结果队列，线程工作结束。

2.4 发布模块

发布模块按 FIFO 方式从处理结果队列中取出数据，然后添加数据包头信息，封装后发送给客户端，其中数据包头信息包括虚拟硬盘文件类型、起始扇区位置和扇区数，以便于客户端收到请求后能对处理结果加以验证。

3 性能分析

在 Windows2003 服务器平台上进行了性能测试，把 I/O Server 置于 Windows2003 服务器平台上，把 I/O Client 动态地加载到透明计算终端中，远程启动 Windows2000 操作系统。

kk 测试的服务器、客户端和网络设备分别为：

(1)服务器端采用 IntelP4、CPU2.8GHz、双通道、内存 2×256MB DDR400、希捷 SATA 80GＢ硬盘；

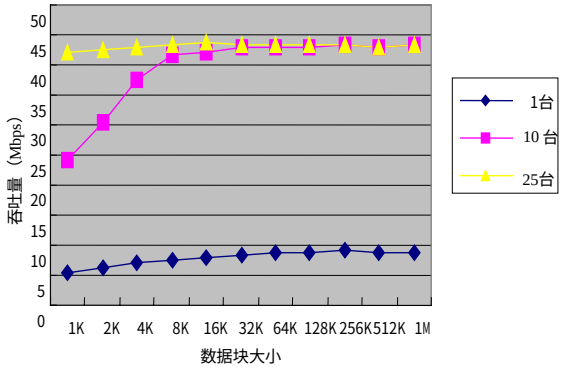
(2)硬盘读速度为 52MBps、写速度为 43MBps、威盛 CPU C3 800 客户端、内存 128MB；

(3)网络为国产千兆位交换机、使用千兆位网卡的服务器；

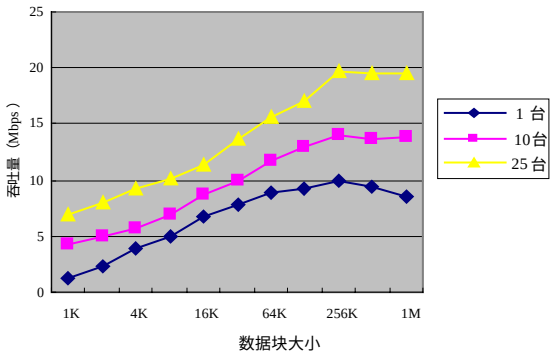
(4)客户机端 10MBps/100MBps 自适应的网卡。

本文测试了远程启动 Windows2000 系统的时间，其中单台透明计算终端的启动时间为 76.73s，而同配置的 PC 机的启动时间为 71.13s，当客户端的数量较少时，其启动和运行应用的性能与同配置的普通 PC 的性能相近。

本文使用 Intel 公司专门开发的用于测试系统 I/O 性能的测试程序 IOMeter，对 I/O Server 进行性能测试。测试了数据块大小对 I/O Server 的总吞吐量以及 CPU 利用率的影响，33%为写操作，且所有的读写分为完全随机读写和完全顺序读写操作，这种设置代表了典型的数据库应用负载特性。在此条件下，测试结果如图 3 所示。结果表明，请求数据块较小时，服务器的响应时间较短，但系统的总吞吐量较小，系统 CPU 利用率不高，当请求的数据块大于 256KB 时，系统可以达到最佳性能。



(a)顺序读写时的吞吐量



(b)随机读写时的吞吐量

图 3 I/O Server 性能测试结果

4 结论

由实验结果可知，基于透明计算模式、能支持操作系统远程启动的 I/O Server，通过将客户机发送的大量网络 I/O 请求进行分类调度，并加上预取缓存机制，能够快速响应网络 I/O 请求，而且在 CPU 利用率不高的情况下能充分利用网络带宽，具有较好的性能，可以很好地支持多种操作系统远程启动，支持透明计算模式。

参考文献

1 张尧学. 透明计算：概念、结构和示例[J]. 电子学报, 2004, 32(12): 169-174.

2 张尧学，彭玉坤，周悦芝等.可管理多媒体网络计算机[J]. 电子学报, 2003, 31(12A): 2054-2058.

3 Shriver E, Small C, Keith A S. Why Does File System Prefetching Work[C]. Proc. of USENIX Annual Technical Conference, Monterey, CA, 1999: 71-84.

4 Cao P, Felten E W, Li K. Implementation and Performance of Integrated Application-controlled File Caching, Prefetching and Disk Scheduling[J]. ACM Transactions on Computer Systems, 1996, 14(4): 311-343.

5 何　军，田范江，王鼎兴. 一种机群网络文件系统的合作高速缓存技术[J]. 计算机学报, 1997, 20(10): 899-907.